



dsPIC30F5015/5016

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。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. 的英文原版本档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，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、dsSPEAK、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、rLAB、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。

dsPIC30F5015/5016 增强型闪存 16 位数字信号控制器

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。关于 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)。关于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：

- 改进型哈佛架构
- 带灵活寻址模式的 C 编译器优化指令集架构
- 83 条基本指令
- 24 位宽指令，16 位宽数据路径
- 66 KB 片上闪存存储空间（指令字）
- 2 KB 片上数据 RAM
- 1 KB 非易失性数据 EEPROM
- 高达 30 MIPS 的工作速度：
 - 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）
- 36 个中断源：
 - 5 个外部中断源
 - 8 个用户可选择的中断源优先级
 - 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
- DSP 操作的累加器回写
- 模寻址和位反转寻址模式
- 两个 40 位宽累加器，具有可选的饱和和逻辑
- 17 位 x 17 位单周期硬件小数 / 整数乘法器
- 所有 DSP 指令都是单周期的
- 单周期 ± 16 位移位

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：可以把 16 位定时器配对形成 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 帧模式）
- I²C™ 模块支持多主器件 / 从器件模式，支持 7 位 / 10 位寻址
- 1 个带 FIFO 缓冲器的 UART 模块
- 1 个 CAN 模块，与 2.0B 标准兼容

电机控制 PWM 模块特性：

- 8 路 PWM 输出通道
 - 互补或独立输出模式
 - 边沿和中心对齐模式
- 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 位向上 / 向下位置计数器
- 计数方向状态
- 位置测量（x2 和 x4）模式
- 输入端具有可编程数字噪声滤波器
- 备用 16 位定时器 / 计数器模式
- 位置计数器发生计满返回 / 下溢时产生中断

dsPIC30F5015/5016

模拟特性:

- 带 4 个 S/H 输入的 10 位模数转换器 (Analog-to-Digital Converter, ADC):
 - 1 Msps 转换速率
 - 16 路输入通道
 - 在休眠和空闲期间进行转换
- 可编程欠压复位

单片机特性:

- 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
- 可在软件控制下自行再编程
- 上电复位 (Power-on Reset, POR)、上电延时

定时器 (Power-up Timer, PWRT) 和振荡器起振定时器 (Oscillator Start-up Timer, OST)

- 灵活的看门狗定时器 (Watchdog Timer, WDT), 带片上低功耗 RC 振荡器以便可靠地工作
- 故障保护时钟监视器操作可检测时钟故障, 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™)
- 可选择的功耗管理模式:
 - 休眠、空闲和备用时钟模式

CMOS 技术:

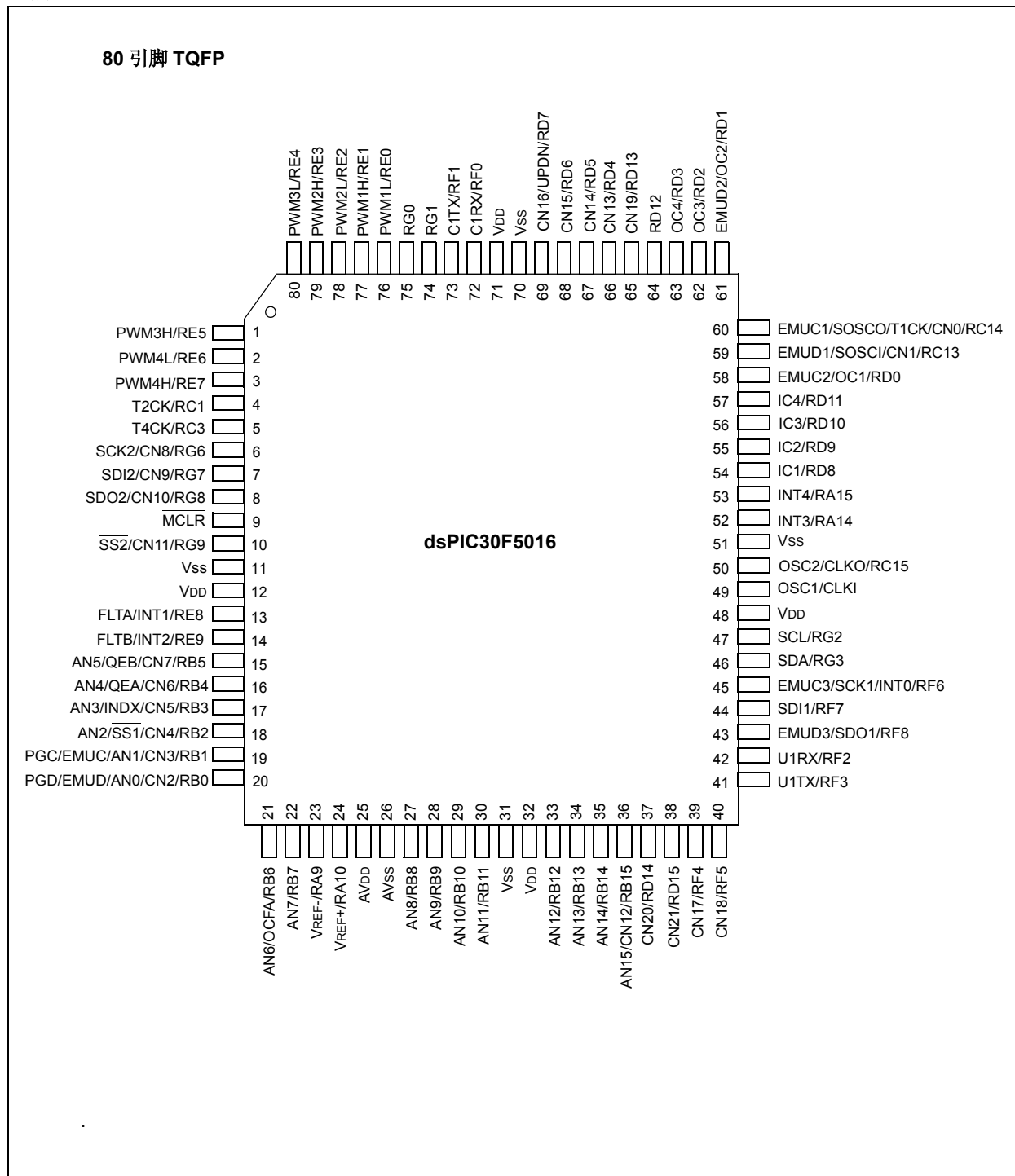
- 低功耗高速闪存技术
- 宽工作电压范围 (2.5V 至 5.5V)
- 工业级和扩展级温度范围
- 低功耗

dsPIC30F 电机控制和电源转换系列 *

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	A/D 10 位 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
dsPIC30F6010	80	144K/48K	8192	4096	5	8	8	8 通道	16 通道	有	2	2	1	2

* 此表提供了 dsPIC30F5015/5016 外设特性的汇总。同时给出了 dsPIC30F 电机控制和功率转换系列中的其他可用器件, 以便进行特性比较。

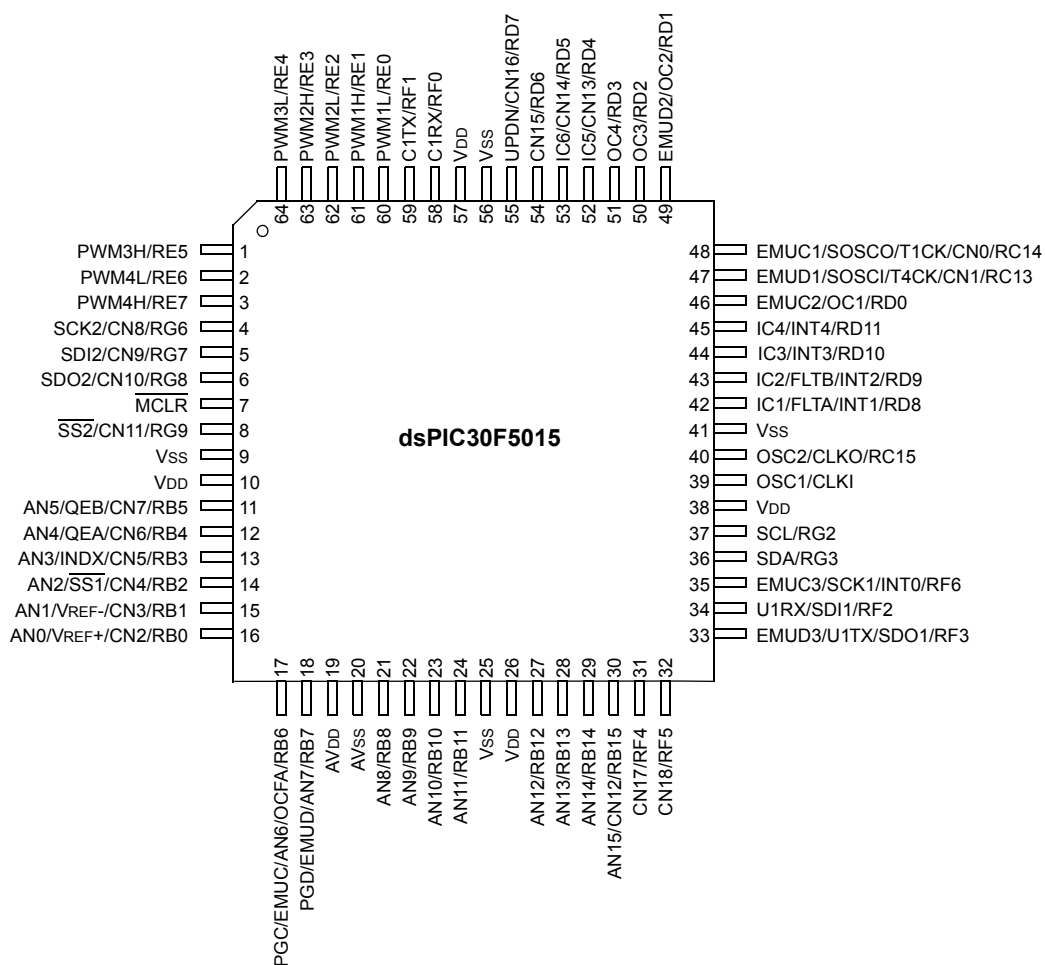
引脚示意图



dsPIC30F5015/5016

引脚示意图

64 引脚 TQFP



目录

1.0	器件概述	7
2.0	CPU 架构概述	15
3.0	存储器构成	23
4.0	地址发生器单元	35
5.0	中断	41
6.0	闪存程序存储器	49
7.0	数据 EEPROM 存储器	55
8.0	I/O 端口	59
9.0	Timer1 模块	65
10.0	Timer2/3 模块	69
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 (QEI) 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	系统集成	133
21.0	12 位高速模数转换器 (ADC) 模块	149
22.0	指令集汇总	161
23.0	开发支持	169
24.0	电气特性	173
25.0	封装信息	213
附录 A:		217
索引		219
Microchip 网站		225
变更通知客户服务		225
客户支持		225
读者反馈表		226
产品标识体系		227

致客户

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

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

最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

<http://www.microchip.com>

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

勘误表

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

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

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

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

客户通知系统

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

dsPIC30F5015/5016

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。关于 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)。关于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)。

本文档给出了 dsPIC30F5015 和 dsPIC30F5016 器件的特定信息。dsPIC30F 器件在其高性能 16 位单片机 (MCU) 架构中，融合了丰富的数字信号处理器 (Digital Signal Processor, DSP) 功能。

图 1-1 给出了 dsPIC30F5015 器件的框图。框图后面的表 1-1 简要说明了器件的 I/O 引脚排列以及 dsPIC30F5015 上端口引脚的复用功能。

图 1-2 给出了 dsPIC30F5016 器件的框图。框图后面的表 1-2 简要说明了器件的 I/O 引脚排列以及 dsPIC30F5016 上端口引脚的复用功能。

dsPIC30F5015/5016

图 1-1: dsPIC30F5015 框图

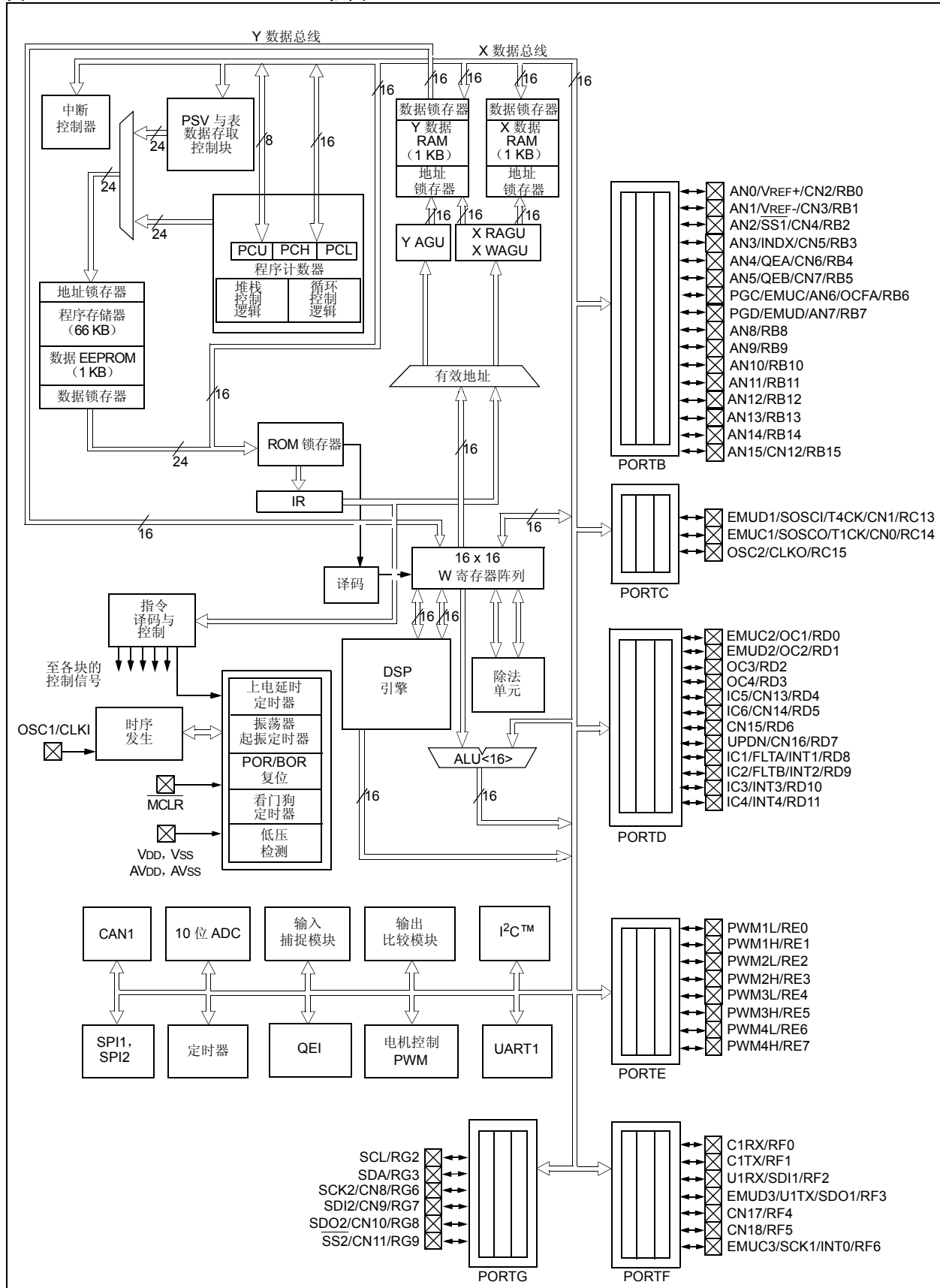


表1-1简要说明了器件的I/O引脚排列以及dsPIC30F5015器件上端口引脚的复用功能。一个端口引脚上可能存在多种功能。出现复用时，外设模块的功能要求可能会强制改写端口引脚的数据方向。

表 1-1: dsPIC30F5015 的 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。 AN6 和 AN7 还分别用于器件编程数据和时钟输入。
AVDD	P	P	模拟模块的正电源。
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 引脚功能关联。 晶振输出。在晶振模式下，连接到晶体或谐振器。也可在 RC 和 EC 模式下用作 CLKO。总是与 OSC2 引脚功能关联。
CN0-CN18	I	ST	输入电平变化通知输入。 在所有输入端均可软件编程为内部弱上拉。
C1RX C1TX	I O	ST —	CAN1 总线接收引脚。 CAN1 总线发送引脚。
EMUD EMUC EMUD1 EMUC1 EMUD2 EMUC2 EMUD3 EMUC3	I/O I/O I/O I/O I/O I/O I/O I/O	ST ST ST ST ST ST ST ST	ICD 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 辅助通信通道数据输入 / 输出引脚。 ICD 辅助通信通道时钟输入 / 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-IC4	I	ST	捕捉输入 1 至 4。
INDX QEA QEB UPDN	I I I O	ST ST ST CMOS	正交编码器索引脉冲输入。 QEI 模式下正交编码器 A 相输入。 定时器模式下辅助定时器外部时钟 / 门控输入。 QEI 模式下正交编码器 B 相输入。 定时器模式下辅助定时器外部时钟 / 门控输入。 位置向上 / 向下计数器方向状态。
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。
FLTA FLTB PWM1L PWM1H PWM2L PWM2H PWM3L PWM3H PWM4L PWM4H	I I O O O O O O O O	ST ST — — — — — — — —	PWM 故障 A 输入。 PWM 故障 B 输入。 PWM 1 低电平输出。 PWM 1 高电平输出。 PWM 2 低电平输出。 PWM 2 高电平输出。 PWM 3 低电平输出。 PWM 3 高电平输出。 PWM 4 低电平输出。 PWM 4 高电平输出。

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

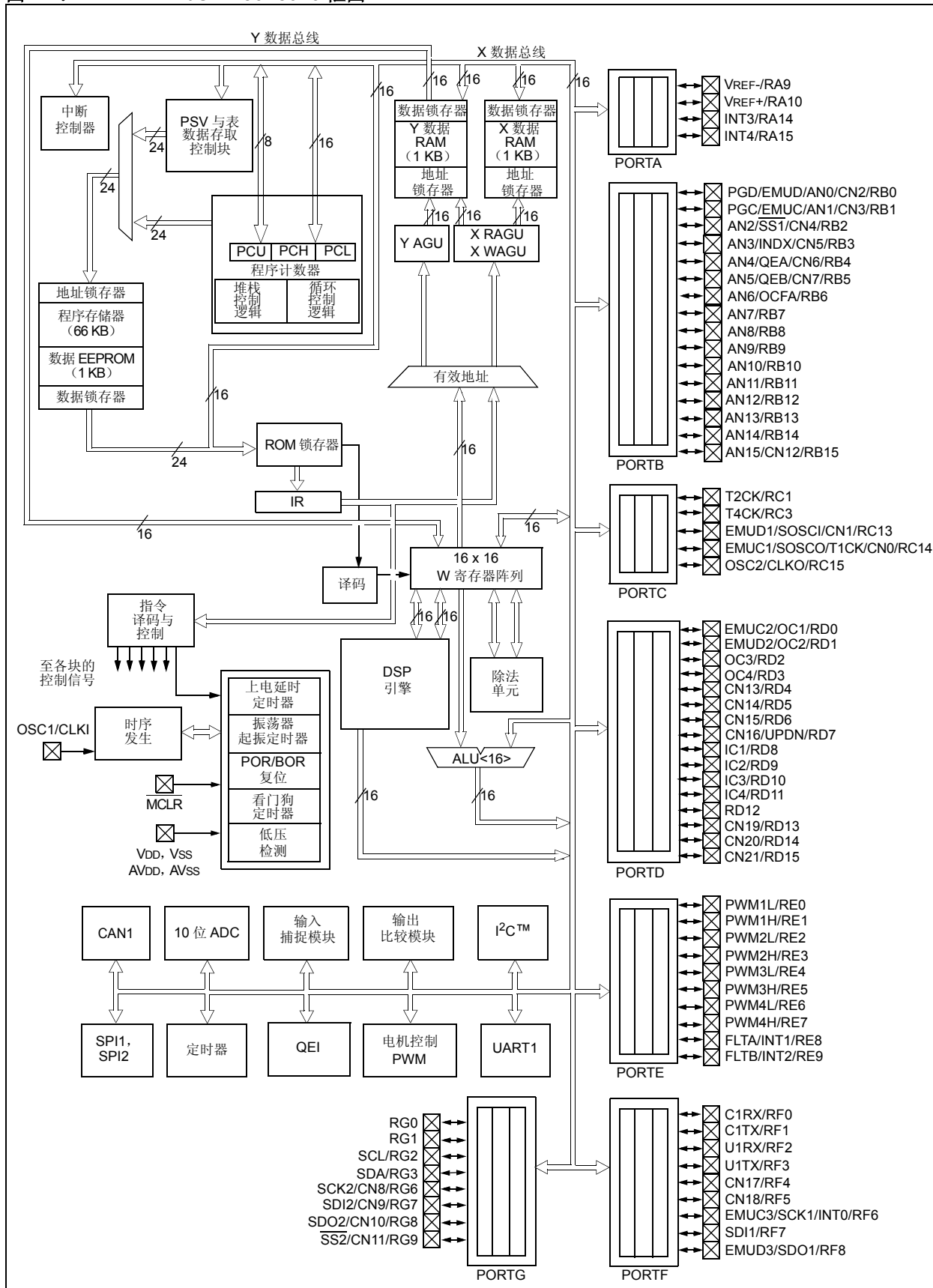
dsPIC30F5015/5016

表 1-1: dsPIC30F5015 的 I/O 引脚说明 (续)

引脚名称	引脚类型	缓冲器类型	说明
MCLR	I/P	ST	主复位输入或编程电压输入。此引脚是器件的复位引脚，低电平有效。
OCFA OC1-OC4	I O	ST —	比较故障 A 输入（对于比较通道 1、2、3 和 4）。 比较输出 1 至 4。
OSC1 OSC2	I I/O	ST/CMOS —	晶振输入。配置在 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。 晶振输出。在晶振模式下，连接到晶体或谐振器。也可在 RC 和 EC 模式下用作 CLK0。
PGD	I/O	ST	在线串行编程数据输入 / 输出引脚。
PGC	I	ST	在线串行编程时钟输入引脚。
RB0-RB15	I/O	ST	PORTB 是双向 I/O 端口。
RC13-RC15	I/O	ST	PORTC 是双向 I/O 端口。
RD0-RD11	I/O	ST	PORTD 是双向 I/O 端口。
RE0-RE7	I/O	ST	PORTE 是双向 I/O 端口。
RF0-RF6	I/O	ST	PORTF 是双向 I/O 端口。
RG2-RG3 RG6-RG9	I/O I/O	ST ST	PORTG 是双向 I/O 端口。
SCK1 SDI1 SDO1 SS1 SCK2 SDI2 SDO2 SS2	I/O I O I I/O I O I	ST ST — ST ST ST — ST	SPI #1 的同步串行时钟输入 / 输出。 SPI #1 数据输入。 SPI #1 数据输出。 SPI #1 同步从动模式。 SPI #2 的同步串行时钟输入 / 输出。 SPI #2 数据输入。 SPI #2 数据输出。 SPI #2 同步从动模式。
SCL SDA	I/O I/O	ST ST	I ² C™ 的同步串行时钟输入 / 输出。 I ² C 的同步串行数据输入 / 输出。
SOSCO SOSCI	O I	— ST/CMOS	32 kHz 低功耗晶振输出。 32 kHz 低功耗晶振输入。配置在 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。
T1CK T4CK	I I	ST ST	Timer1 外部时钟输入。 Timer4 外部时钟输入。
U1RX U1TX	I O	ST —	UART1 接收。 UART1 发送。
VDD	P	—	逻辑和 I/O 引脚的正电源。
VSS	P	—	逻辑和 I/O 引脚的参考地。
VREF+	I	模拟	模拟参考电压（高）输入。
VREF-	I	模拟	模拟参考电压（低）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

图 1-2: dsPIC30F5016 框图



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

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

表 1-2: dsPIC30F5016 的 I/O 引脚说明 (续)

引脚名称	引脚类型	缓冲器类型	说明
MCLR	I/P	ST	主复位输入或编程电压输入。此引脚是器件的复位引脚，低电平有效。
OCFA	I	ST	比较故障 A 输入（对于比较通道 1、2、3 和 4）。
OCFB	I	ST	比较故障 B 输入（对于比较通道 5、6、7 和 8）。
OC1-OC4	O	—	比较输出 1 至 4。
OSC1	I	ST/CMOS	晶振输入。配置在 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。
OSC2	I/O	—	晶振输出。在晶振模式下，连接到晶体或谐振器。也可在 RC 和 EC 模式下用作 CLK0。
PGD	I/O	ST	在线串行编程数据输入 / 输出引脚。
PGC	I	ST	在线串行编程时钟输入引脚。
RA9-RA10	I/O	ST	PORTA 是双向 I/O 端口。
RA14-RA15	I/O	ST	
RB0-RB15	I/O	ST	PORTB 是双向 I/O 端口。
RC1	I/O	ST	PORTC 是双向 I/O 端口。
RC3	I/O	ST	
RC13-RC15	I/O	ST	
RD0-RD15	I/O	ST	PORTD 是双向 I/O 端口。
RE0-RE9	I/O	ST	PORTE 是双向 I/O 端口。
RF0-RF8	I/O	ST	PORTF 是双向 I/O 端口。
RG0-RG3	I/O	ST	PORTG 是双向 I/O 端口。
RG6-RG9	I/O	ST	
SCK1	I/O	ST	SPI #1 的同步串行时钟输入 / 输出。
SDI1	I	ST	SPI #1 数据输入。
SDO1	O	—	SPI #1 数据输出。
SS1	I	ST	SPI #1 同步从动模式。
SCK2	I/O	ST	SPI #2 的同步串行时钟输入 / 输出。
SDI2	I	ST	SPI #2 数据输入。
SDO2	O	—	SPI #2 数据输出。
SS2	I	ST	SPI #2 同步从动模式。
SCL	I/O	ST	I ² C™ 的同步串行时钟输入 / 输出。
SDA	I/O	ST	I ² C 的同步串行数据输入 / 输出。
SOSCO	O	—	32 kHz 低功耗晶振输出。
SOSCI	I	ST/CMOS	32 kHz 低功耗晶振输入。配置在 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。
T1CK	I	ST	Timer1 外部时钟输入。
T2CK	I	ST	Timer2 外部时钟输入。
T4CK	I	ST	Timer4 外部时钟输入。
U1RX	I	ST	UART1 接收。
U1TX	O	—	UART1 发送。
VDD	P	—	逻辑和 I/O 引脚的正电源。
VSS	P	—	逻辑和 I/O 引脚的参考地。
VREF+	I	模拟	模拟参考电压（高）输入。
VREF-	I	模拟	模拟参考电压（低）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

dsPIC30F5015/5016

注:

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。关于 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)。关于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)。

本文档提供了对 dsPIC30F5015/5016 CPU 和外设功能的汇总。关于此功能的完整说明，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

2.1 内核概述

内核的指令字长为 24 位。程序计数器 (Program Counter, PC) 为 23 位宽，其最低有效位 (Least Significant bit, LSb) 始终清零 (见第 3.1 节“程序地址空间”)，最高有效位 (Most Significant bit, MSb) 在正常的程序执行期间被忽略，某些专用指令除外。因此，PC 能够寻址最大为 4M 指令字的用户程序空间。使用指令预取机制来帮助维持吞吐量。使用 DO 和 REPEAT 指令，支持无循环计数管理开销的程序循环结构，并且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，每个寄存器均可作为数据、地址或偏移量寄存器。一个工作寄存器 (W15) 用作中断和调用时的软件堆栈指针。

数据空间为 64 KB (32K 字)，被分为两块，称为 X 和 Y 数据存储器。每个存储块有各自独立的地址发生单元 (Address Generation Unit, AGU)。大多数指令只通过 X 存储空间 AGU 进行操作，这样对外界而言数据空间就是单独而统一的。乘法-累加 (MAC) 类双源操作数 DSP 指令，通过 X 和 Y 的 AGU 进行操作，这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 数据空间边界视具体器件而定，用户不能改变它们。每个数据字由 2 个字节组成，大多数指令可以把数据当作字或字节来进行寻址。

对于存储在程序存储器中的数据，有两种存取方法：

- 数据存储空间的高 32 KB 可以映射到程序空间的下半部分 (用户空间)，映射的边界可以是任何 16K 程序字边界，边界由 8 位程序空间可视性页 (Program Space Visibility Page, PSVPAG) 寄存器定义。这使得任何指令都能像存取数据空间那样存取程序空间，但存取需要一个额外的周期。此外，使用这个方法，只能存取每个指令字的低 16 位。
- 使用任何工作寄存器，通过表读和表写指令，也可以对程序空间内的 32K 字页进行线性间接存取。表读和表写指令能够用来存取一个指令字的全部 24 位。

X 和 Y 地址空间都支持无开销循环缓冲区 (模寻址)。模寻址的主要目的是要消除 DSP 算法的循环开销。

X AGU 还支持对目标有效地址进行位反转寻址，对于基 -2 FFT 算法，这极大地简化了输入或输出数据的重新排序。关于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、寄存器直接寻址、寄存器间接寻址、寄存器偏移量寻址和立即数偏移量寻址模式。指令与预定义的寻址模式相关联，这取决于其功能需求。

对于大多数指令，内核能够在每个指令周期内执行数据 (或程序数据) 存储器读取、工作寄存器 (数据) 读取、数据存储器写入以及程序 (指令) 存储器读取操作。因此，支持 3 操作数指令，允许在单个周期内执行 $C = A + B$ 这样的操作。

内核包含 DSP 引擎，这极大地提高了内核的运算能力和吞吐量。它具有一个高速 17 位 x17 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。累加器或任何工作寄存器中的数据可以在单个周期内右移 16 位或左移 16 位。DSP 指令可以无缝地与所有其他指令一起操作，且设计为能获得最佳实时性能。在把两个 W 寄存器相乘时，MAC 类指令能够从存储器中同时取出两个操作数。为了能够同时取数据操作数，对于 MAC 类指令，数据空间拆分为两块，对所有其他指令数据空间则是线性的。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。但是，为了最大限度地获得可用的执行时间，使用了单级指令预取机制，在指令执行的前一个周期存取并部分译码指令。除了一些特例外，大部分指令都在单个周期内执行。

内核具有用于陷阱和中断的向量异常处理结构，具有 62 个独立的向量。异常包括最多 8 个陷阱（其中有 4 个是保留的）和 54 个中断。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，其范围从 8 到 15。

2.2 编程模型

编程模型如图 2-1 所示，它包括 16x16 位工作寄存器（W0 至 W15）、2x40 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）。工作寄存器可作为数据、地址或偏移量寄存器。所有寄存器都是存储器映射的。W0 用作文件寄存器寻址的 W 寄存器。

这些寄存器中，一些寄存器有与之相关的影子寄存器，如图 2-1 所示。影子寄存器用作临时保持寄存器，事件发生时可在影子寄存器和主机寄存器之间传递内容。所有影子寄存器均不能直接存取。寄存器与其影子寄存器之间的数据传递遵循以下规则：

- 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内容传递。
- 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，循环结束时其内容从各自的影子寄存器中弹出。

对工作寄存器进行字节操作时，只影响到目标寄存器的最低有效字节（LSB）。但是，存储器映射工作寄存器的一个好处是，可以通过字节宽数据存储空间存取来处理最低和最高有效字节。

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 专用于软件堆栈指针（Stack Pointer, SP），异常处理、子程序调用与返回时将自动修改 W15。但是，W15 可以被任何指令所引用，引用方式与引用所有其他 W 寄存器相同。这简化了堆栈指针的读、写和处理（例如，创建堆栈帧）。

注： 为了防止出现不对齐的堆栈访问，W15<0> 始终清零。

复位时 W15 初始化为 0x0800。在初始化过程中，用户可以重新设定 SP，使它指向数据空间内的任何单元。

W14 专用于 LNK 和 ULNK 指令中所定义的堆栈帧指针。但是，W14 可以被任何指令所引用，引用方式与引用所有其他 W 寄存器相同。

2.2.2 状态寄存器

dsPIC DSC 内核有一个 16 位状态寄存器（SR），其低有效字节（LSB）称为 SR 低字节（SRL），其高有效字节（MSB）称为 SR 高字节（SRH）。SR 寄存器的组成见图 2-1。

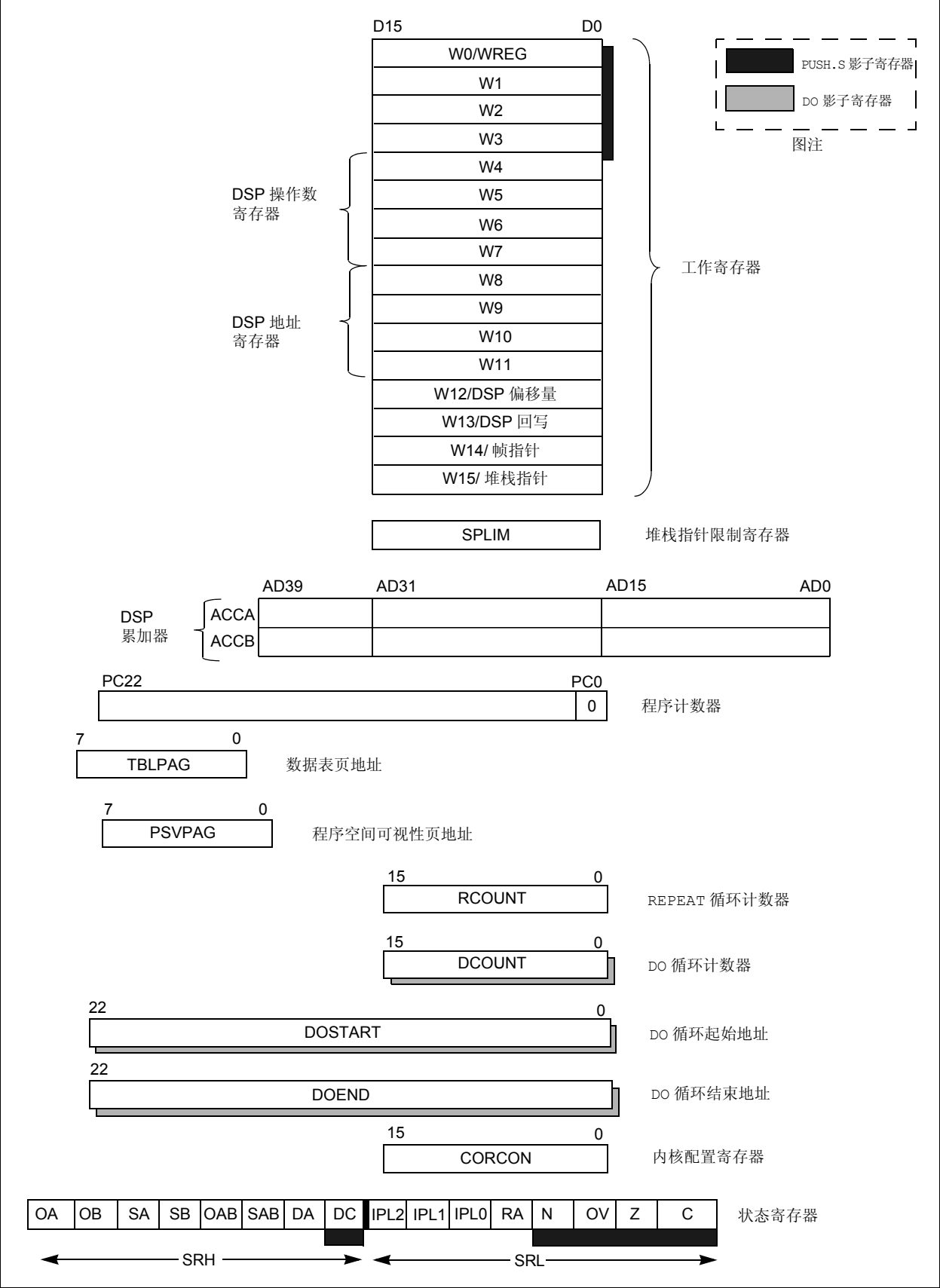
SRL 包含所有的 MCU ALU 运算状态标志位（包括 Z 位），以及 CPU 中断优先级状态位 IPL<2:0> 和重复有效状态位 RA。在异常处理期间，SRL 与 PC 的 MSB 连接起来形成一个完整的字值，然后被压入堆栈。

SR 寄存器的高字节包含 DSP 加法器 / 减法器状态位、DO 循环有效位（DA）和半进位（DC）状态位。

2.2.3 程序计数器

程序计数器为 23 位宽，bit 0 始终清零。因此，PC 能够最多寻址 4M 指令字。

图 2-1: dsPIC30F5015/5016 编程模型



dsPIC30F5015/5016

2.3 除法支持

dsPIC DSC 支持 16/16 位有符号小数除法运算，以及 32/16 位、16/16 位有符号和无符号整数除法运算，除法形式均为单指令迭代除法。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.sw——16/16 有符号除法
5. DIV.uw——16/16 无符号除法

除法指令必须在一个 REPEAT 循环内执行。任何其他执行方式（例如，一系列不连续的除法指令）都不会正常运行，因为指令流取决于 RCOUNT。除法指令不会自动设置 RCOUNT 值，因此 RCOUNT 必须明确 REPEAT 指令中正确指定，如表 2-1 所示（REPEAT 将执行目标指令 { 操作数的值 +1 } 次）。必须为 DIV/DIVF 指令的 18 次迭代设置 REPEAT 循环计数。因此，完整的除法操作需要 19 个周期。

注： 除法流可以被中断。但是，用户需正确保存中断现场。

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.ud	无符号除法: $(Wm+1:Wm)/Wn \rightarrow W0$; $Rem \rightarrow W1$
DIV.sw	有符号除法: $Wm/Wn \rightarrow W0$; $Rem \rightarrow W1$
DIV.uw	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 x17 位乘法器、一个桶形移位寄存器和一个 40 位加法器 / 减法器（两个目标累加器、舍入逻辑和饱和逻辑）组成。

dsPIC30F 器件采用单周期指令流，可以执行 DSP 指令或 MCU 指令。许多硬件资源可以在 DSP 和 MCU 指令之间共用。例如，指令集具有 DSP 和 MCU 乘法指令，它们使用相同的硬件乘法器。

DSP 引擎还能够执行固有的不需要其他数据的累加器至累加器操作。这些指令是 ADD、SUB 和 NEG。

通过 CPU 内核配置寄存器（CORCON）中的各个位，可以有多种 DSP 引擎选择，这些选择如下：

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. 对于写数据存储器的自动饱和和使能 / 禁止（SATDW）。
7. 累加器饱和模式选择（ACCSAT）。

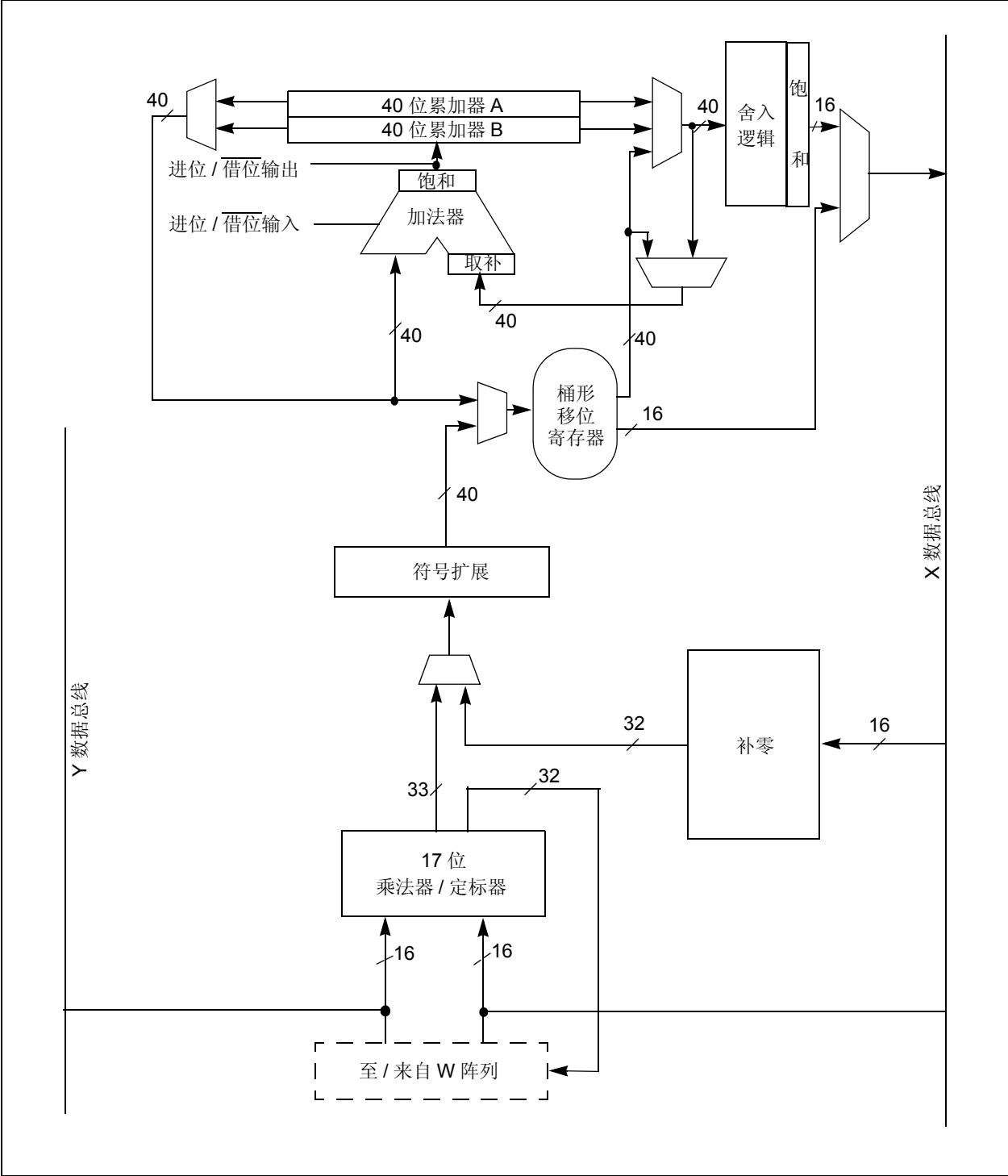
注： CORCON 的组成见表 3-3。

DSP 引擎的框图如图 2-2 所示。

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 不变
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

该 17x17 位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。17x17 位乘法器 / 定标器的输出是 33 位值，它将被符号扩展为 40 位。整数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} 到 $2^{N-1} - 1$ 。对于 16 位整数，数据范围为 -32768 (0x8000) 到 32767 (0x7FFF)，包括 0 在内。对于 32 位整数，数据范围为 -2,147,483,648 (0x8000 0000) 到 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为 -1.0 到 $(1 - 2^{1-N})$ 。对于 16 位小数，Q15 数据范围为 -1.0 (0x8000) 到 0.999969482 (0x7FFF)，包括 0 在内，其精度为 3.01518×10^{-5} 。在小数模式下，16x16 乘法运算将产生 1.31 乘积，其精度为 4.65661×10^{-10} 。

同一个乘法器还用于支持 MCU 乘法指令，包括整数的 16 位有符号、无符号和混合符号乘法。

可以指示 MUL 指令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

数据累加器包含一个 40 位加法器 / 减法器，它带有自动符号扩展逻辑。它可以选择两个累加器 (A 或 B) 之一作为其预累加的源累加器和后累加的目标累加器。对于 ADD 和 LAC 指令，将被累加或装入的数据可选择通过桶形移位寄存器在累加之前进行换算。

2.4.2.1 加法器 / 减法器，溢出和饱和

加法器 / 减法器是一个 40 位加法器，一个输入可以选择为零，而另一个输入可以是原数据或求补后的数据。对于加法，进位 / 借位输入为高电平有效，另一个输入是原数据（没有求补的）；对于减法，进位 / 借位输入为低电平有效，另一个输入是求补后的数据。加法器 / 减法器产生溢出状态位 SA/SB 和 OA/OB，状态位被锁存在状态寄存器中并在其中得到反映。

- 从 bit 39 溢出：这是灾难性溢出，会破坏累加器的符号。
- 溢出到警戒位 (bit 32 到 bit 39)：这是可恢复的溢出。每当警戒位有任何不一致，就将把这个状态位置 1。

加法器有一个额外的饱和模块，如果选取的话，饱和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、上述溢出状态位、SATA/B (CORCON<7:6>) 和 ACCSAT (CORCON<4>) 模式控制位，来确定何时饱和以及达到何值为饱和。

状态寄存器中有 6 个支持饱和和溢出的位，它们是：

1. OA:
ACCA 溢出到警戒位
2. OB:
ACCB 溢出到警戒位
3. SA:
ACCA 已饱和 (bit 31 溢出并饱和)
或
ACCA 溢出到警戒位并饱和 (bit 39 溢出并饱和)
4. SB:
ACCB 已饱和 (bit 31 溢出并饱和)
或
ACCB 溢出到警戒位并饱和 (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 和 OB 位。这两位置 1 时，表明最近的操作已溢出到累加器警戒位 (bit 32 到 bit 39)。如果 OA 和 OB 位置 1 且 INTCON1 寄存器中相应的溢出陷阱标志使能位 (OVATE 和 OVBTE) 也置 1，则还可选择用 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，例如，校正系统增益。

每次数据通加法器/减法器，就会修改 SA 和 SB 位，但用户只能对它们进行清零。这两位置 1 时，表明累加器已溢出其最大范围（对于 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，以此指示产生了灾难性溢出。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。

在状态寄存器（SR）中，对于溢出和饱和状态位，可以将 OA 和 OB 的逻辑或形成 OAB 位，将 SA 和 SB 的逻辑或形成 SAB 位。这样，只需检查状态寄存器中的一个位，程序就能判断是否有累加器溢出；检查状态寄存器中的另一个位，就可以判断是否有累加器饱和。对于通常要使用两个累加器的复数运算而言，这很有用。

器件支持三种饱和和溢出模式：

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）或最大的负 9.31 值（0x80000000）装入目标累加器。SA 或 SB 位置 1 并保持置 1 状态直到被用户清零。这称为“超饱和”（super saturation），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 1.31 值（0x007FFFFFFF）或最大的负 1.31 值（0x0080000000）装入目标累加器。SA 或 SB 位置 1 并保持置 1 直到被用户清零。当这种饱和模式生效时，不使用警戒位（因此 OA、OB 或 OAB 位不会被置 1）。
3. **bit 39 灾难性溢出：**
来自加法器的 bit 39 溢出状态位用来将 SA 或 SB 位置 1；这两位置 1 后，将保持置 1 状态直到被用户清零。此时不发生饱和操作，且允许累加器溢出（破坏其符号）。如果 INTCON1 寄存器中的 COVTE 位置 1，灾难性溢出会导致一个陷阱异常。

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字（bit 16 到 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 总线寻址组合的 X 和 Y 地址空间，执行回写操作。支持以下寻址模式：

1. **W13，寄存器直接寻址：**
非操作目标的累加器的舍入内容以 1.15 小数形式写入 W13。
2. **[W13]+ = 2，执行后递增的寄存器间接寻址：**
非操作目标的累加器的舍入内容以 1.15 小数形式写入 W13 指向的地址。然后 W13 递增 2（对于字写入）。

2.4.2.3 舍入逻辑

舍入逻辑是一个组合模块，在累加器写（存储）过程中执行常规的（有偏）或收敛的（无偏）舍入功能。舍入模式由 CORCON 寄存器中 RND 位的状态决定。它会产生一个 16 位的 1.15 数据值，该值被送到数据空间写饱和和逻辑。如果指令不指明舍入，就会存储一个截取的 1.15 数据值，简单地丢弃最低有效字（lsW）。

常规舍入取累加器的 bit 15，对它进行零扩展并将扩展后的值加到 ACCxH 字（累加器的 bit 16 到 bit 31）。如果 ACCxL 字（累加器的 bit 0 到 bit 15）在 0x8000 和 0xFFFF 之间（包括 0x8000），则 ACCxH 递增 1。如果 ACCxL 在 0x0000 和 0x7FFF 之间，则 ACCxH 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）。

除非 ACCxL 等于 0x8000，否则收敛的（或无偏）舍入操作方式与常规舍入相同。ACCxL 等于 0x8000 时，要对 ACCxH 的 LSb（累加器的 bit 16）进行检测。如果它为 1，ACCxH 递增 1。如果它为 0，ACCxH 不变。假设 bit 16 本身是随机的，这样的机制将消除任何可能累加的舍入偏移。

通过 X 总线，SAC 和 SAC.R 指令将目标累加器内容的截取（SAC）或舍入（SAC.R）形式存入数据存储空间（这受数据饱和的影响，请参见第 2.4.2.4 节“数据空间写饱和”）。注意，对于 MAC 类指令，累加器回写操作将以同样的方式进行，通过 X 总线寻址组合的 MCU（X 和 Y）数据空间。对于此类指令，数据始终要进行舍入。

2.4.2.4 数据空间写饱和

除了加法器/减法器饱和，对数据空间进行写操作也会饱和，但不会影响源累加器的内容。数据空间写饱和和逻辑块接受来自舍入逻辑块的一个 16 位的 1.15 小数值作为输入，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适当的 1.15 小数值作为输出，写入数据存储空间。

如果 CORCON 寄存器中的 SATDW 位置 1，将检测（经过舍入或截取后的）数据是否溢出，并进行相应的调整。如果输入数据大于 0x007FFF，则写入存储器中的数据被强制为最大的正 1.15 值，0x7FFF。如果输入数据小于 0x007FFF，则写入存储器中的数据被强制为最大的负 1.15 值，0x8000。源累加器的 MSb（bit 39）用于确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 桶形移位寄存器

桶形移位寄存器在单个周期内最多可算术或逻辑右移 16 位或左移 16 位。源寄存器可以是两个 DSP 累加器中的任何一个或 X 总线（支持寄存器或存储器数据的多位移位）。

移位寄存器需要一个有符号二进制值，用来确定移位操作的幅度（位数）和方向。正值将操作数右移。负值则将操作数左移。值为 0 则不改变操作数。

桶形移位寄存器为 40 位宽，因此，它为 DSP 移位操作提供 40 位的结果，而为 MCU 移位操作提供了 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 到 bit 31，左移则存放在 bit 0 到 bit 15。

3.0 存储器构成

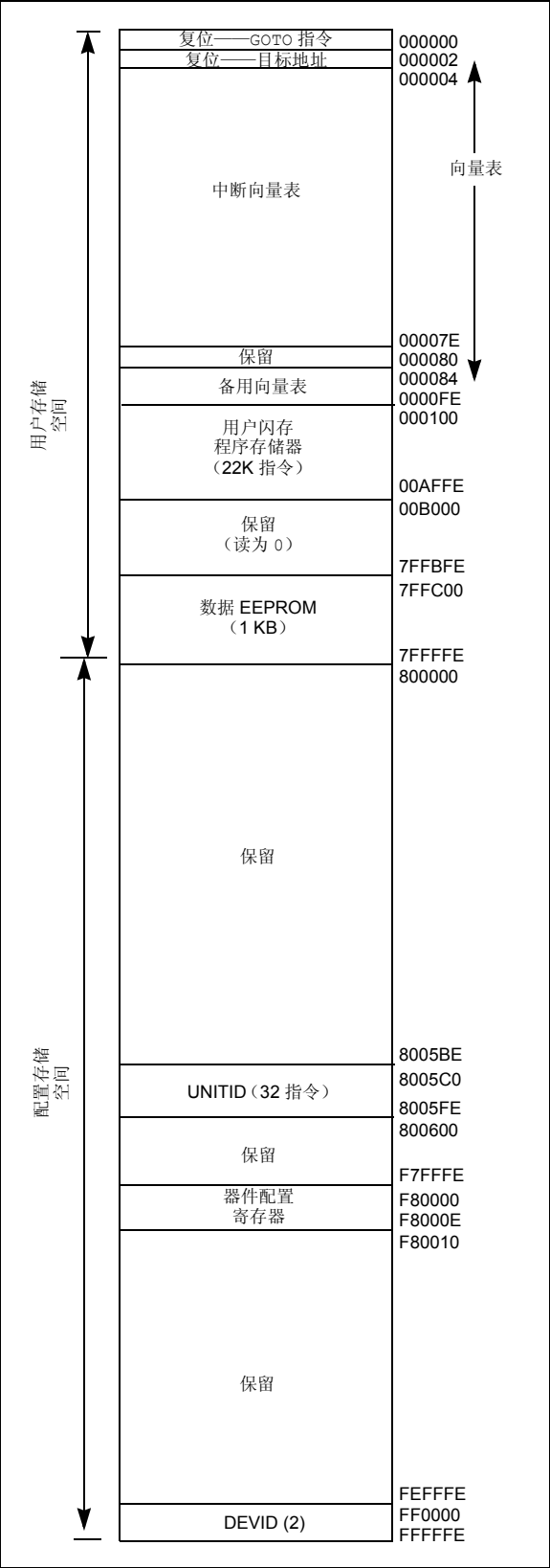
注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。关于 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。关于器件指令集和编程的更多信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

3.1 程序地址空间

程序地址空间为 4M 指令字。当程序空间映射到数据空间（如表 3-1 定义）时，可以通过 23 位 PC、表指令有效地址（Effective Address, EA）或数据空间 EA 来寻址。请注意，为了提供与数据空间寻址的兼容性，程序空间地址在两个连续的程序字之间递增 2。

对于除 TBLRD/TBLWT 外的所有访问，用户程序空间访问限制在低 4M 指令字地址范围（0x000000 至 0x7FFFFFFE）；TBLRD/TBLWT 使用 TBLPAG<7> 来决定用户或配置空间访问。在表 3-1 中，bit 23 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: dsPIC30F5015/5016 的程序存储空间映射

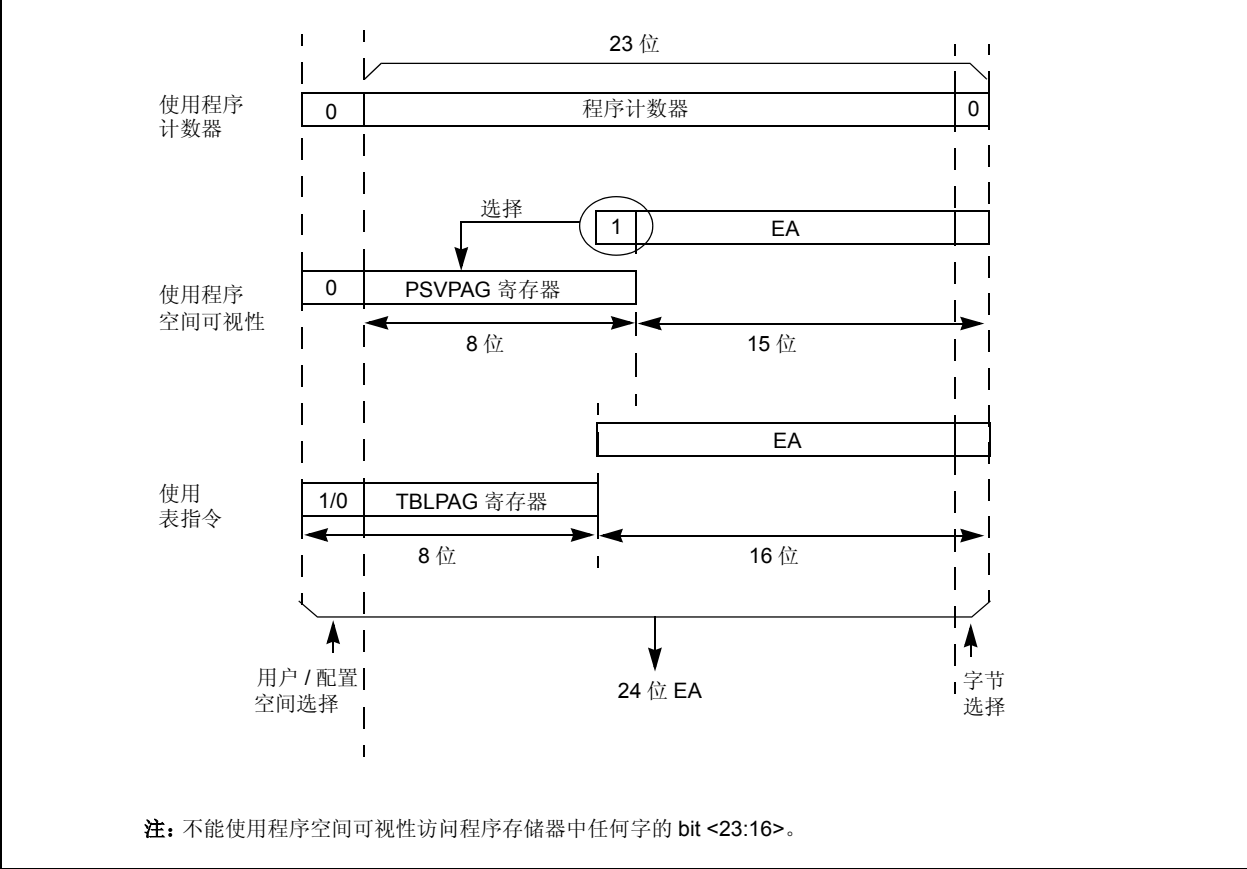


dsPIC30F5015/5016

表 3-1: 程序空间地址构成

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

图 3-2: 访问程序空间中数据的地址生成



3.1.1 使用表指令访问程序存储器中的数据

本器件架构取指令时是取 24 位宽的程序存储器。因此，指令始终是对齐的。但是，由于器件架构是改进型哈佛架构，数据也可以出现在程序空间中。

有两种方法可以访问程序空间：通过特殊的表指令，或通过把 16K 字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 指令提供了读或写程序空间内任何地址的最低有效字的直接方法，无需通过数据空间。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 将递增 2。这使得程序存储器地址能够直接映射到数据空间地址。因此，程序存储器可以看作是两个 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 访问存有最低有效字的空间，而 TBLRDH 和 TBLWTH 则访问存有最高有效字的空间。

图 3-2 显示如何为表操作和数据空间访问（PSV = 1）创建 EA。这里，P<23:0> 指程序空间字；而 D<15:0> 指数据空间字。

器件提供了一组表指令，用来在程序空间和数据空间之间传送字节或字大小的数据。

- 1. **TBLRDL**: 表读低位
字: 读程序地址的最低有效字;
P<15:0> 映射到 D<15:0>。
字节: 读程序地址的 LSB 中的一个; 当字节选择 = 0 时, P<7:0> 映射到目标字节; 当字节选择 = 1 时, P<15:8> 映射到目标字节;
- 2. **TBLWTL**: 表写低位 (关于闪存编程的详细信息, 请参见第 6.0 节“闪存程序存储器”)。
- 3. **TBLRDH**: 表读高位
字: 读程序地址的最高有效字;
P<23:16> 映射到 D<7:0>; D<15:8> 始终 = 0。
字节: 读程序地址的 MSB 中的一个; 当字节选择 = 0 时, P<23:16> 映射到目标字节; 当字节选择 = 1 时, 目标字节始终 = 0。
- 4. **TBLWTH**: 表写高位 (关于闪存编程的详细信息, 请参见第 6.0 节“闪存程序存储器”)。

图 3-3: 表访问程序数据（最低有效字）

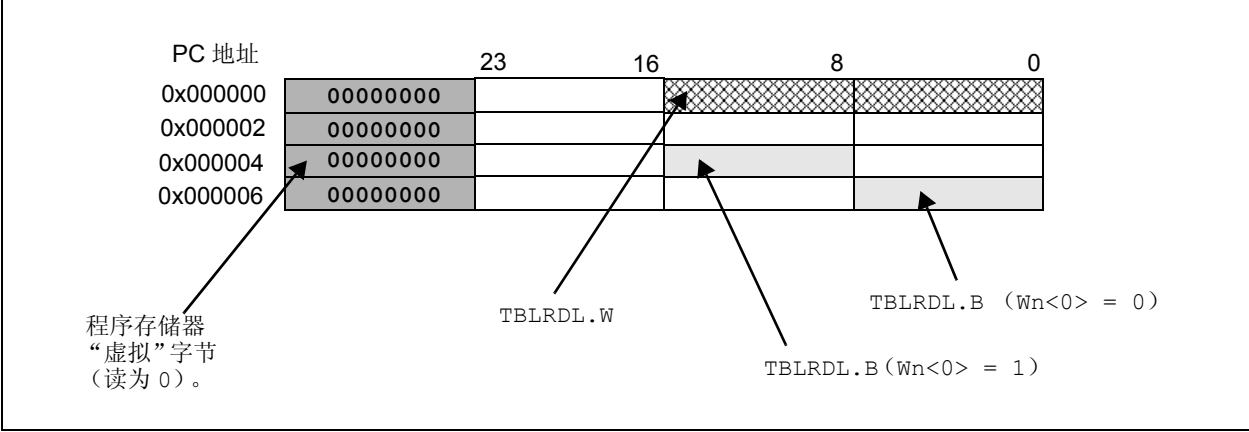
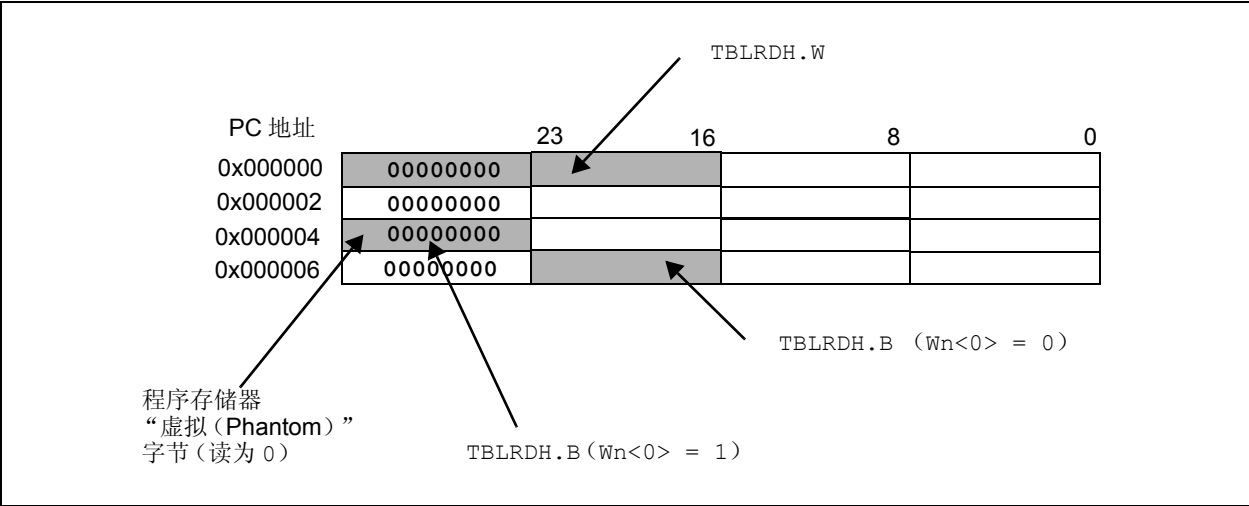


图 3-4: 表访问程序数据 (MSB)



3.1.2 使用程序空间可视性访问程序存储器中的数据

可选择将数据空间的高 32 KB 映射到任何 16K 字程序空间页。这提供了从 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 和 TBLWTL 指令）。

如果数据空间 EA 的 MSb 置 1，并且程序空间可视性使能（方法是在内核控制寄存器 CORCON 中把 PSV 位置 1）时，就能通过数据空间访问程序空间。CORCON 功能的讨论请参见第 2.4 节“DSP 引擎”。

正在执行的指令，如果要对这个区域进行数据访问，就需要增加一个额外的指令周期，因为需要进行两次程序存储空间读取操作。

注意，可寻址数据空间的上半部分始终是 X 数据空间的一部分。因此，当 DSP 操作使用程序空间映射来访问此存储区域时，Y 数据空间通常应该存放 DSP 操作的状态（可变）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（0x8000 及更高）直接映射到对应的程序存储器地址（见图 3-5），但只使用 24 位程序字的低 16 位来存放数据。应该进行适当的设置，使得对高 8 位进行访问的指令都成为非法指令，以维持器件的可靠性。关于指令编码的详细信息，请参见《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

注意，对于每个程序存储字，PC 都将递增 2，数据空间地址的最低有效 15 位将直接映射到相应程序空间地址的最低有效 15 位。剩下的位由程序空间可视性页寄存器（PSVPAG<7:0>）提供，如图 3-5 所示。

注： 在表读 / 写期间，暂时禁止 PSV 访问。

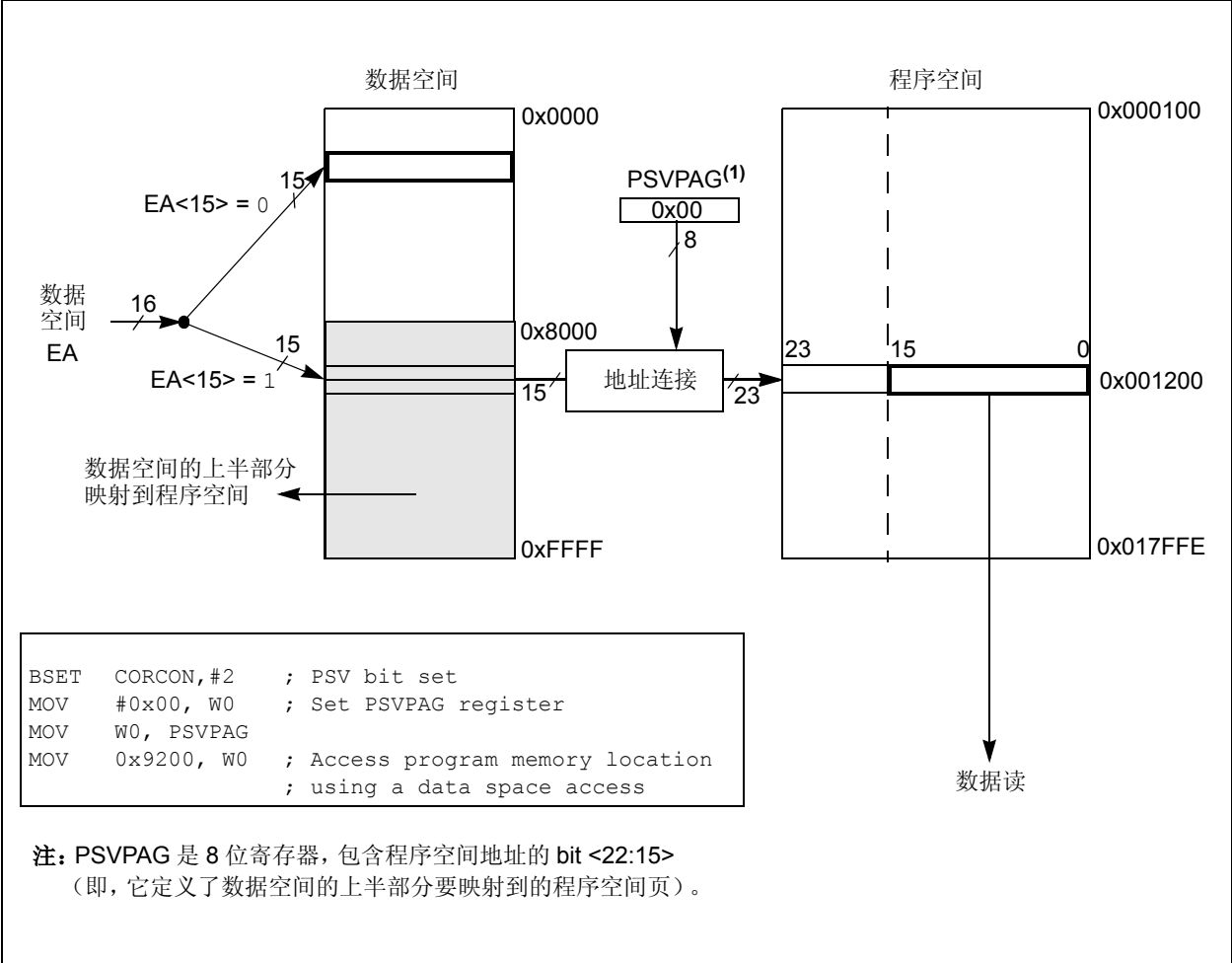
对于使用 PSV 且在 REPEAT 循环之外执行的指令：

- 以下指令都需要在规定的执行时间之外额外增加一个指令周期：
 - 带数据操作数预取的 MAC 类指令
 - MOV 指令
 - MOV.D 指令
- 其他所有指令都需要在规定的指令执行时间之外额外增加两个指令周期。

对于使用 PSV 且在 REPEAT 循环之内执行的指令：

- 以下情况都需要在规定的指令执行时间之外额外增加两个指令周期：
 - 在第一次迭代中执行的指令
 - 在最后一次迭代中执行的指令
 - 由于中断而退出循环之前执行的指令
 - 中断得到处理后再次进入循环时执行的指令
- REPEAT 循环的所有其他各次迭代，都允许使用 PSV 访问数据的指令在一个周期内执行。

图 3-5: 数据空间通过程序空间可视性页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。数据空间可以看作是独立的 (对于某些 DSP 指令而言), 或者看作是统一的线性地址范围 (对于某些 MCU 指令而言)。使用两个地址发生单元 (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

数据存储空间分为两块, X 和 Y 数据空间。此架构的关键之处在于, Y 空间是 X 空间的子集, Y 空间完全包含在 X 空间中。为了提供外在的线性寻址空间, X 和 Y 空间要有连续的地址。

当执行除 MAC 类指令之外的任何指令时, X 块由 64 KB 数据地址空间 (包括所有 Y 地址) 组成。在执行一条 MAC 类指令时, X 块由 64 KB 数据地址空间组成, 但不包括 Y 地址块 (只用来读数据)。换句话说, 所有其他指令把整个数据存储空间当作一个复合的地址空间。MAC 类指令把 Y 地址空间从数据空间中抽取出来, 使用源自 W10 和 W11 的 EA 对其进行寻址。剩下的 X 数据空间则使用 W8 和 W9 来寻址。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 给出了对 MCU 和 DSP 指令如何访问 X 和 Y 数据空间的图形汇总。

dsPIC30F5015/5016

图 3-6: dsPIC30F5015/5016 的数据存储空间映射

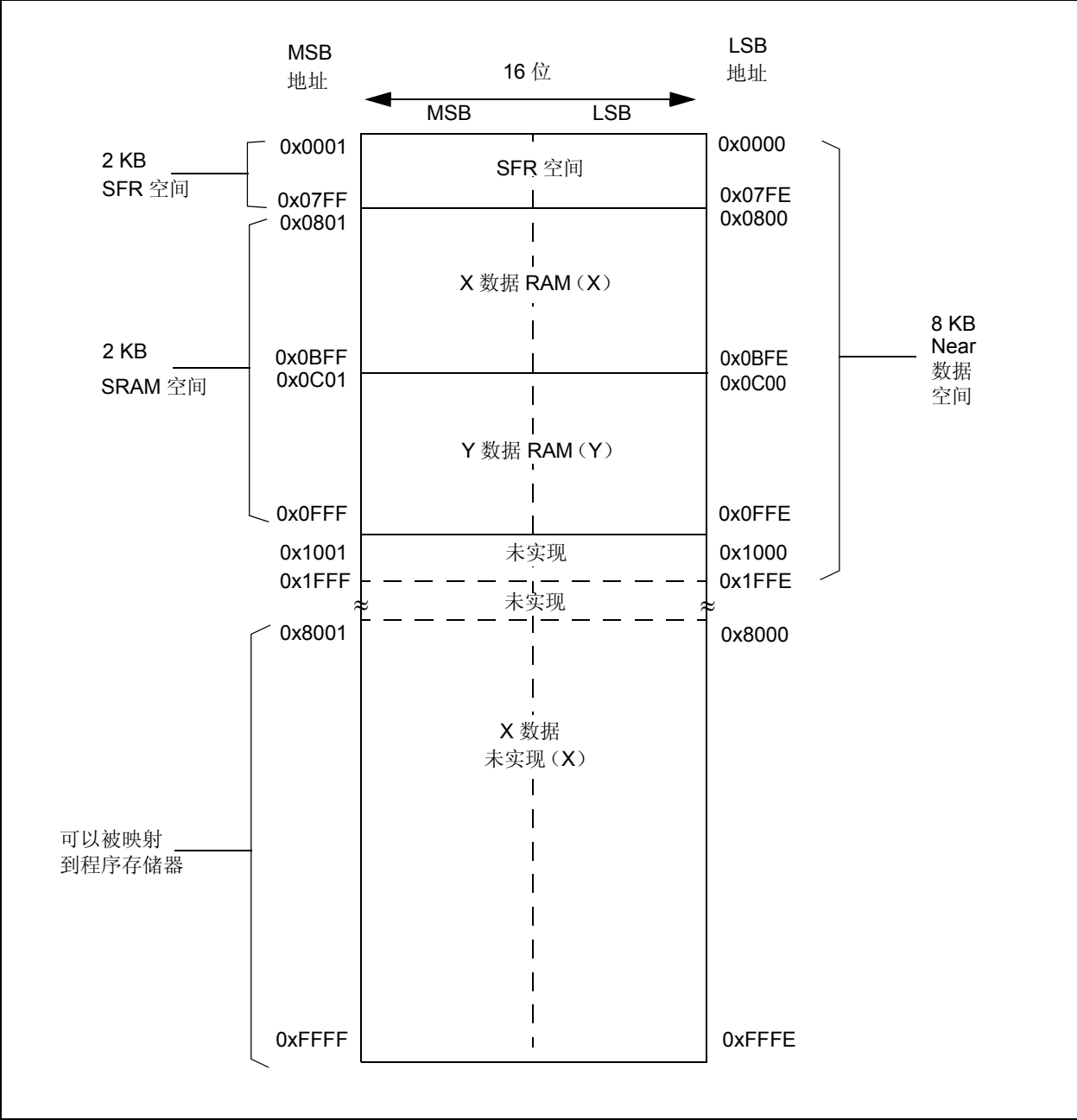
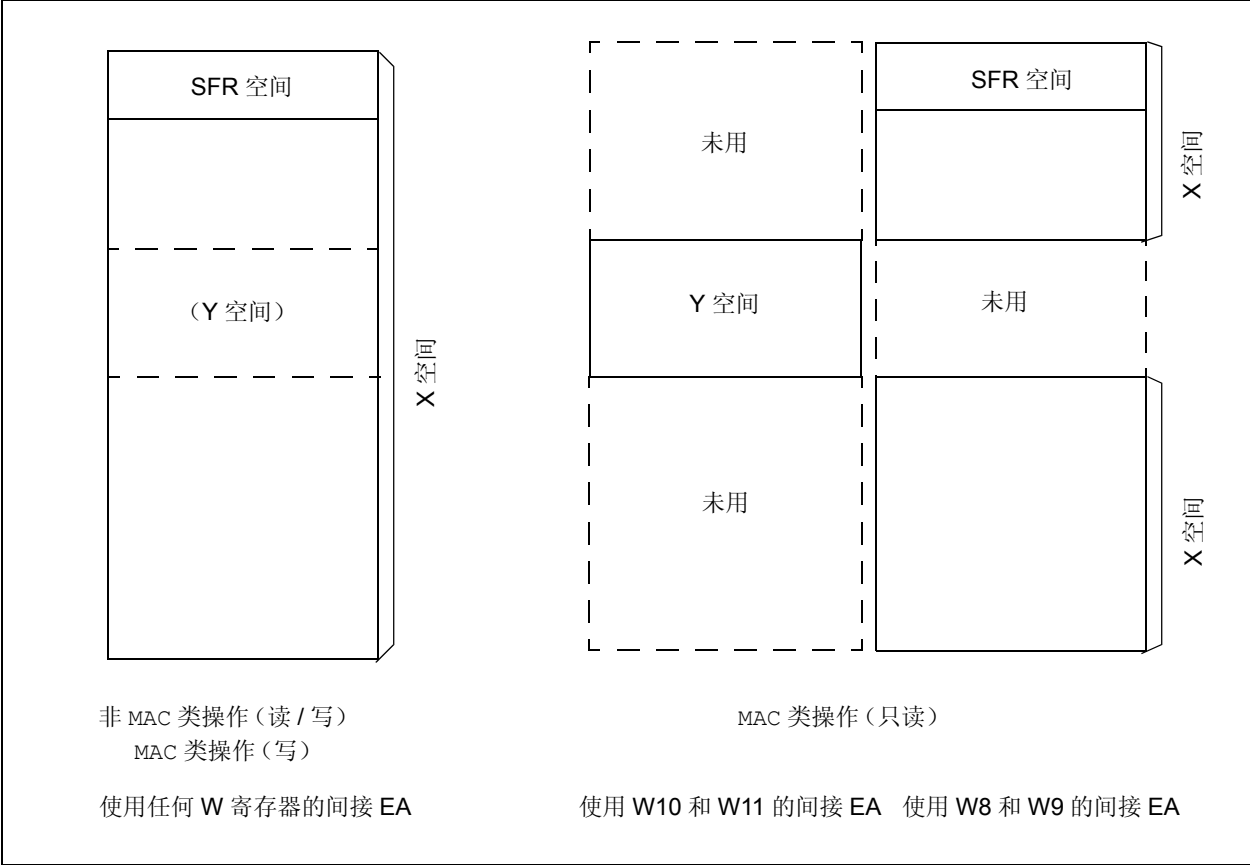


图 3-7: 用于 MCU 和 DSP (MAC 类) 指令的数据空间示例



3.2.2 数据空间

X 数据空间可用于所有指令，并且支持所有寻址模式。读数据总线和写数据总线是独立的。X 读数据总线是所有指令的返回数据路径，它把数据空间视为组合的 X 和 Y 地址空间。对于双操作数读指令（MAC 类），它还是 X 地址空间的数据路径。对于所有指令而言，X 写数据总线是到数据空间的只写路径。

X 数据空间还支持所有指令的模寻址，不过要受到寻址模式的限制。位反转寻址只是在写 X 数据空间时才支持。

MAC 类指令（CLR、ED、EDAC、MAC、MOVSAC、MPY、MPY.N 和 MSC）将 Y 数据空间与 X 数据空间配合使用，提供了两条并行的数据读取路径。Y 总线上不会出现写操作。此类指令使用两个专用的 W 寄存器指针 W10 和 W11，它们始终寻址 Y 数据空间，与 X 数据空间独立；而 W8 和 W9 则始终寻址 X 数据空间。注意，在累加器回写期间，数据地址空间被视为 X 和 Y 数据空间的组合，从而写操作通过 X 总线进行。因此，可以写整个数据空间中的任何地址。

Y 数据空间仅用于与 MAC 类指令相关的数据预取操作。它 also 支持自动循环缓冲区的模寻址。当然，所有其他指令可以通过 X 数据路径把 Y 数据地址空间作为复合线性空间的一部分来进行访问。

图 3-6 给出了 X 和 Y 数据空间之间的边界定义，用户不能设定改变边界。如果 EA 指向其所在地址空间之外的数据，或者指向物理存储器之外的存储单元，将返回全零的字 / 字节。例如，尽管 Y 地址空间对于使用任何寻址模式的所有非 MAC 指令都是可见的，但如果一条 MAC 指令试图使用 W8 或 W9（X 空间指针）从该空间取数据，则将返回 0x0000。

表 3-2: 无效存储器访问的结果

试图进行的操作	返回数据
EA = 未实现的地址	0x0000
在 MAC 指令中，使用 W8 或 W9 访问 Y 数据空间	0x0000
在 MAC 指令中，使用 W10 或 W11 访问 X 数据空间	0x0000

所有有效地址均为 16 位宽，并且指向数据空间内的字节。因此，数据空间地址范围为 64 KB 或 32K 字。

3.2.3 数据空间宽度

内核数据宽度为 16 位。所有内部寄存器均按 16 位宽的字来组织。数据存储空间以字节可寻址的、16 位宽的块来组织。

3.2.4 数据对齐

为了维持与 PIC[®] 器件的后向兼容性，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。数据在数据存储器和寄存器中按字节对齐，而所有数据空间 EA 都将解析为字节。数据字节读取将读取包含字节的整个字，使用任何 EA 的 LSB 来确定要选取的字节。选取的字节将存放到 X 数据路径的 LSB 中（不可能出现来自 Y 数据路径的字节访问，因为 MAC 类指令只能取整字）。也就是说，数据存储器和寄存器被组织为两个并行的字节宽的实体，它们共享（字）地址译码，但写入线独立。数据字节写操作只写阵列或寄存器中与字节地址匹配的那一侧。

这种字节可访问性的结果是，所有的有效地址计算（包括那些由 DSP 操作产生的有效地址，它们只能是字长度的数据）均将在内部进行转换，以适应字对齐的存储空间。例如，内核将识别执行后修改寄存器间接寻址模式 [Ws++] 的结果，对于字节操作产生的值是 Ws + 1，而对于字操作产生的值是 Ws + 2。

所有字访问必须按偶数地址对齐。不支持没有对齐的字数据取，因此在混合字节和字操作时，或者从 8 位 MCU 代码进行转换时，必须要小心。如果试图进行未对齐的读或写，将产生地址错误陷阱。如果在读操作时产生错误，正在执行的指令将完成；而如果在写操作时产生错误，指令仍将执行，但不会进行写入。无论是哪种情况，都会产生陷阱，从而系统和 / 或用户能够检查地址错误发生之前的机器状态。

图 3-8: 数据对齐

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

所有装入 W 寄存器的字节都将被装入相应的 LSB 中。相应的 MSB 不变。

提供了一条符号扩展 (SE) 指令，允许用户把 8 位有符号数据转换为 16 位有符号值。或者，对于 16 位无符号数据，用户可以通过在适当地址处执行一条零扩展 (ZE) 指令清零任何 W 寄存器的 MSB。

尽管大多数指令能够对字或字节大小的数据进行操作，但要注意的是，部分指令（包括 DSP 指令）只对字大小的数据进行操作。

3.2.5 NEAR 数据空间

X 地址存储空间中，在 0x0000 和 0x1FFF 之间保留了一个 8 KB 的“near”数据空间，在所有存储器直接寻址指令中可以通过一个 13 位绝对地址字段来直接访问这个数据空间。其余的 X 地址空间和所有 Y 地址空间都是间接寻址的。此外，使用 MOV 指令可以寻址整个 X 数据空间，这支持通过 16 位地址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 用作堆栈指针。

堆栈指针总是指向第一个可用的空字，并且从低地址向高地址方向增长。它在弹出堆栈之前预递减，而在压入堆栈之后后递增，如图 3-9 所示。注意，对于任何 CALL 指令时的 PC 压栈，在压入堆栈之前，PC 的 MSB 要进行零扩展，从而确保了 MSB 始终是清零的。

注： 在异常处理期间，在 PC 压入堆栈之前，要
先将PC的MSB与SRL寄存器组合在一起。

堆栈指针限制寄存器 (SPLIM) 与堆栈指针相关。复位时 SPLIM 未被初始化。与堆栈指针的情况一样，SPLIM<0> 被强制为 0，这是因为所有堆栈操作必须是字对齐的。每当使用 W15 作为源指针或目标指针产生有效地址 (EA) 时，都会将它与 SPLIM 中的值进行比较。如果堆栈指针 (W15) 的内容与 SPLIM 寄存器的内容相等，则执行压栈操作时不产生堆栈错误陷阱。但在随后的压栈操作时将会产生错误陷阱。因此，当堆栈增长超过 RAM 中地址 0x2000 时，如果要想产生堆栈错误陷阱，用值 0x1FFE 来初始化 SPLIM 即可。

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，从而避免了堆栈干扰特殊功能寄存器 (Special Function Register, SFR) 空间。

在对 SPLIM 寄存器进行写操作之后，不应紧跟着使用 W15 进行间接读操作的指令。

图 3-9: CALL 堆栈帧

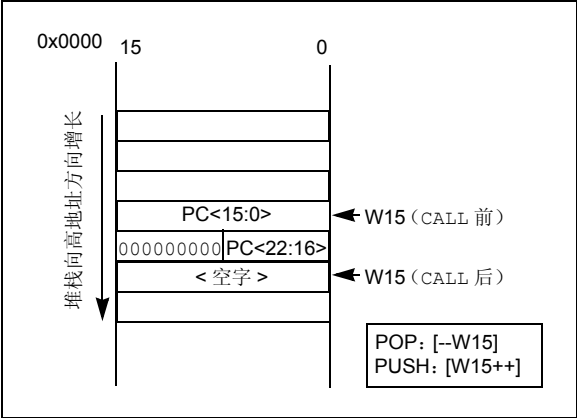


表 3-3: 内核寄存器映射

SFR 名称	地址 (首地址)	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
ACCBH	0028								ACCBH									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									
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																uuuu uuuu uuuu uuu0
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	DOSTARTH							0000 0000 0uuu uuuu
DOENDL	003E	DOENDL																uuuu uuuu uuuu uuu0
DOENDH	0040	—	—	—	—	—	—	—	—	—	DOENDH							0000 0000 0uuu uuuu
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000 0000 0000 0000
CORCON	0044	—	—	—	—	US	EDT	DL2	DL1	DL0	SATA	SATB	SATDW	ACCSAT	IPL3	RND	IF	0000 0000 0010 0000

图注: u = 未初始化位
注: 寄存器中各位的说明, 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 (首地址)	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			
MODCON	0046	XMODEN	YMODEN	—	—	BWM<3:0>					YWM<3:0>					XWM<3:0>				0000 0000 0000 0000	
XMODSRT	0048	XS<15:1>																	0	uuuu uuuu uuuu uuu0	
XMODEND	004A	XE<15:1>																	1	uuuu uuuu uuuu uuu1	
YMODSRT	004C	YS<15:1>																	0	uuuu uuuu uuuu uuu0	
YMODEND	004E	YE<15:1>																	1	uuuu uuuu uuuu uuu1	
XBREV	0050	BREN	XB<14:0>															uuuu uuuu uuuu uuuu			
DISICNT	0052	—	—	DISICNT<13:0>														0000 0000 0000 0000			

图注: u = 未初始化位

注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F5015/5016

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。关于 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)。关于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)：X AGU 和 Y AGU。Y AGU 仅支持 DSP MAC 类指令的字大小的数据读取。DSPIC DSC AGU 支持三种类型的数据寻址：

- 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。位反转寻址只能用于数据空间地址。

4.1 指令寻址模式

表 4-1 给出了基本的寻址模式，这些寻址模式经过优化可支持各指令的具体功能。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 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，它在这些指令中表示为 WREG。目标寄存器通常是同一个文件寄存器或 WREG (MUL 指令除外)，它把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，可以在文件寄存器操作中访问整个数据空间。

4.1.2 MCU 指令

三操作数 MCU 指令的形式如下：

操作数 3 = 操作数 1 < 功能 > 操作数 2

其中，操作数 1 始终是工作寄存器（即，寻址模式只能是寄存器直接寻址），称为 WB。操作数 2 可以是一个 W 寄存器，取自数据存储器或一个 5 位立即数。结果位置可以是 W 寄存器或地址存储单元。MCU 指令支持以下寻址模式：

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

注：并非所有指令都支持上述所有的寻址模式。各条指令可能支持这些寻址模式中的某些模式。

表 4-1： 支持的基本寻址模式

寻址模式	说明
文件寄存器直接寻址	显式地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），WN 的内容形成 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。

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带寄存器偏移量的寄存器间接寻址模式，这也称为寄存器变址寻址模式。

注： 对于 MOV 指令，指令中指定的寻址模式对于源寄存器和目标寄存器 EA，可以是不同的。然而，4 位 WB（寄存器偏移量）字段为源寄存器和目标寄存器所共用（但通常只由其中之一使用）。

概括地说，传送和累加器指令支持以下寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 带寄存器偏移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： 并非所有指令都支持上述所有的寻址模式。各条指令可能支持这些寻址模式中的某些模式。

4.1.4 MAC 指令

双源操作数 DSP 指令（CLR、ED、EDAC、MAC、MPY、MPY.N、MOVSA 和 MSC），也称为 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高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。因此，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 类指令支持以下寻址模式：

- 寄存器间接寻址
- 执行后修改（修改量为 2）的寄存器间接寻址
- 执行后修改（修改量为 4）的寄存器间接寻址
- 执行后修改（修改量为 6）的寄存器间接寻址
- 带寄存器偏移量的寄存器间接寻址（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 指令则使用一个 14 位无符号立即数字段。在一些指令中，例如 ADD ACC，操作数的来源和运算结果已经暗含在操作码中。某些操作，如 NOP，不含任何操作数。

4.2 模寻址

模寻址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。但是，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和结束地址条件，它们可以双向工作（即，在低地址边界和高地址边界上都将进行地址边界检查）。

4.2.1 起始地址和结束地址

模寻址机制要求指定起始和结束地址，并把它们装入 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 计算使用字长度的数据（每个 EA 的 LSB 始终清零）。

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

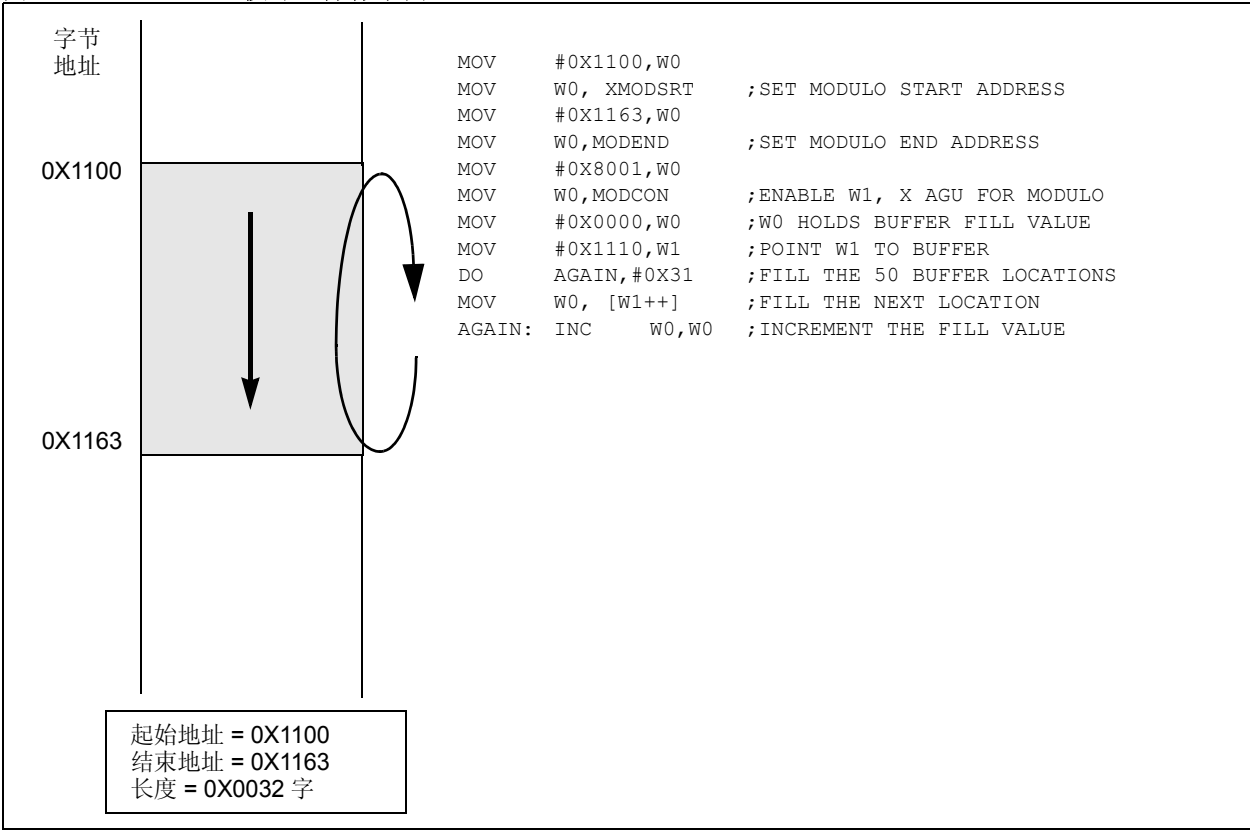
4.2.2 W 地址寄存器选择

模寻址和位反转寻址控制寄存器 MODCON<15:0> 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）存储在 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）存储在 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



DSPIC30F5015/5016

4.2.3 模寻址的应用

模寻址可以应用于任何与 **W** 寄存器相关的有效地址 (**EA**) 计算。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处于边界上, 而且会检查地址是否小于或大于上限 (对于递增缓冲区) 地址、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]**), 会进行模地址修正, 但寄存器的内容保持不变。

4.3 位反转寻址

位反转寻址用于简化基 2 FFT 算法的数据重新排序。它为 **X AGU** 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。地址源和目标仍然是正常的顺序。因此, 惟一需要反转的操作数就是地址修改量。

4.3.1 位反转寻址的实现

位反转寻址的使能方式如下:

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为 $M = 2^N$ 字节, 则数据缓冲区起始地址的最后 “**N**” 位必须为零。

XB<14:0> 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 计算中, 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的数据 (每个 **EA** 的 **LSB** 始终清零)。为了产生兼容地址, 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、且仅对字大小的数据写入, 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的数据, 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, 由于要求的是字大小的数据, **EA** 的 **LSB** 被忽略 (且始终清零)。

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

如果通过将 **BREN** 位 (**XBREV<15>**) 置 1 使能了位反转寻址, 那么, 在写 **XBREV** 寄存器之后, 不应立即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

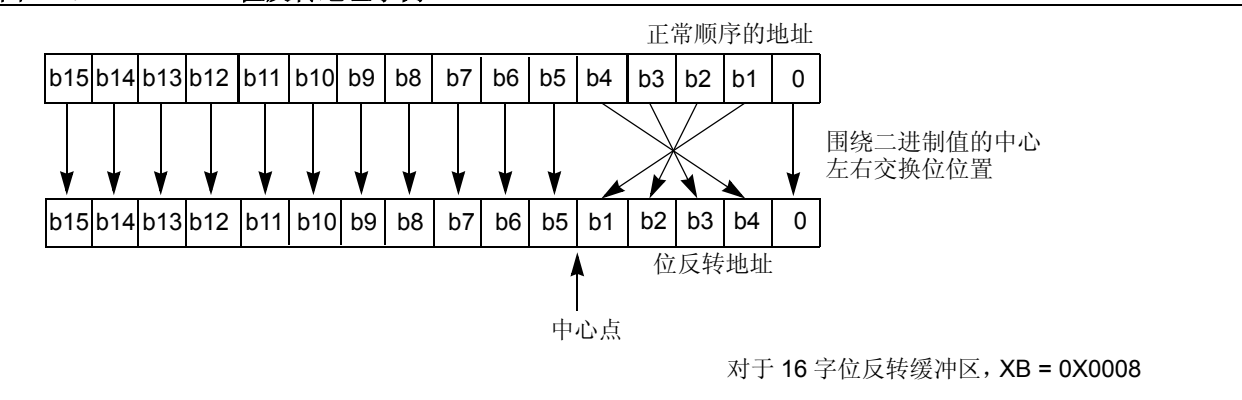


表 4-2: 位反转地址序列 (16 项)

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

DSPIC30F5015/5016

注:

5.0 中断

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。关于 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)。关于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)。

dsPIC30F5015/5016 有 36 个中断源和 4 个处理器异常（陷阱），它们必须按照优先级机制来进行仲裁。

CPU 负责读取中断向量表（Interrupt Vector Table, IVT），并把中断向量中的地址传送给程序计数器。中断向量从程序数据总线向程序计数器的传送，是由程序计数器输入端的 24 位宽多路开关来进行的。

中断向量表（IVT）和备用中断向量表（AIVT）的位置靠近程序存储器的开始（0x000004）。IVT 和 AIVT 如图 5-1 所示。

在中断和处理器异常提交给处理器内核之前，中断控制器负责它们的预处理。使用集中式的特殊功能寄存器对外设中断和陷阱进行允许、优先级分配和控制：

- IFS0<15:0>、IFS1<15:0> 和 IFS2<15:0>
这三个寄存器中保存所有的中断请求标志。这些标志由其对应的外设或外部信号置 1，标志的清除则通过软件完成。
- IEC0<15:0>、IEC1<15:0> 和 IEC2<15:0>
这三个寄存器中保存所有的中断允许控制位。这些控制位用来单独允许来自外设或外部信号的中断。
- IPC0<15:0>...IPC11<7:0>
与 44 个中断中的每一个中断有关的用户可分配优先级，集中存放在这 12 个寄存器中。
- IPL<3:0>
当前 CPU 优先级显式地存放在 IPL 位中。IPL<3> 在 CORCON 寄存器中，而 IPL<2:0> 则存放在处理器内核的状态寄存器（SR）中。
- INTCON1<15:0> 和 INTCON2<15:0>
全局中断控制功能由这两个寄存器提供。
INTCON1 包含处理器异常的控制和状态标志。
INTCON2 寄存器控制外部中断请求信号的操作以及备用中断向量表的使用。

注：当中断条件产生时，不管相应的中断允许位状态如何，中断标志位都将置 1。用户程序应确保在重新允许中断之前，相应的中断标志位已被清零。

用户可以通过 IPCx 寄存器，为所有中断源分配 7 个优先级中的一个，优先级从 1 到 7。每个中断源都与一个中断向量相关，如表 5-1 所示。优先级 7 和 1 分别代表最高和最低的可屏蔽优先级。

注：中断源优先级指定为 0，等同于禁止该中断。

如果 NSTDIS 位（INTCON1<15>）置 1，则禁止中断嵌套。因此，如果正在处理一个中断，将禁止处理新的中断，即使新中断的优先级比当前正在处理的中断优先级高。

注：一旦 NSTDIS 位置为 1，IPL 位就变为只读。

某些中断具有专门的控制位，用于控制诸如边沿或电平触发中断、电平变化中断这样的功能。这些功能的控制仍然由产生中断的外设模块负责。

在执行某些指令期间，如果 DISI 位（INTCON2<14>）保持置 1 的话，那么可以用 DISI 指令来禁止优先级为 6 或更低的中断的处理。

处理中断时，PC 中装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）。这些向量存放在程序存储器从 0x000004 到 0x0000FE 的存储单元中（见图 5-2）。这些存储单元中存有 24 位地址，为了保持鲁棒性（robustness），在正常执行期间当 PC 试图取任何这些字的话，就会产生地址错误陷阱。这避免了执行随机数据；PC 递减时意外进入向量空间、意外地把数据空间地址映射到了向量空间，或者 PC 到达可用程序存储空间的末尾继而返回到 0x000000，都将执行随机数据。执行 GOTO 指令跳转到向量空间，也将产生地址错误陷阱。

dsPIC30F5015/5016

5.1 中断优先级

每个中断源的用户可分配中断优先级位（IP<2:0>）均位于 IPCx 寄存器中每个半字节的最低 3 位中。每个半字节的 bit 3 未使用，读为 0。这些位定义了用户分配给特定中断的优先级。

注： 用户可选择的优先级为从 0 到 7，0 是最低优先级，7 是最高优先级。

由于可能会有多个中断请求源被分配了同一个特定的用户指定优先级，所以提供了一种方法在给定优先级内指定优先级。这种方法称为“自然顺序优先级”。

自然顺序优先级由中断在向量表中的位置决定；只有在具有相同用户分配优先级的多个中断同时等待处理时，才会影响中断操作。

表 5-1 列出了 dsPIC DSC 器件的中断编号和中断源，以及相关的向量编号。

注 1： 自然顺序优先级机制以 0 为最高优先级，以 53 为最低优先级。

2： 自然顺序优先级编号与 INT 编号相同。

用户可以为每个中断分配 7 个优先级之一，这意味着用户可以为具有较低自然顺序优先级的中断分配一个极高的总优先级。

表 5-1: 中断向量表

INT 编号	向量编号	中断源
最高自然顺序优先级		
0	8	INT0——外部中断 0
1	9	IC1——输入捕捉 1
2	10	OC1——输出比较 1
3	11	T1——Timer1
4	12	IC2——输入捕捉 2
5	13	OC2——输出比较 2
6	14	T2——Timer2
7	15	T3——Timer3
8	16	SPI1
9	17	U1RX——UART1 接收器
10	18	U1TX——UART1 发送器
11	19	ADC——ADC 转换完成
12	20	NVM——NVM 写入完成
13	21	SI2C——I ² C™ 从中断
14	22	MI2C——I ² C 主中断
15	23	输入电平变化中断
16	24	INT1——外部中断 1
17	25	保留
18	26	保留
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	保留
25	33	保留
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	保留
31	39	保留
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	保留
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
最低自然顺序优先级		

5.2 复位过程

复位不是真正的异常，因为复位过程中并不涉及中断控制器。在响应强制清零 PC 的复位时，处理器初始化其寄存器。然后，处理器从单元 0x000000 开始执行程序。一条 GOTO 指令存放在第一个程序存储单元中，紧跟着的是 GOTO 指令的地址目标。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 复位的来源

有 6 种错误来源会导致器件复位。

- 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：
试图把未初始化的 W 寄存器用作地址指针将导致复位。
- 非法指令陷阱：
试图执行任何未使用的操作码，将产生非法指令陷阱。注意，如果由于指令流改变而在非法指令执行之前就被从指令流中舍弃的话，取非法指令并不产生非法指令陷阱。
- 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：
多个并发的陷阱条件将导致复位。

5.3 陷阱

可以将陷阱看作不可屏蔽中断，表明出现软件或硬件错误，陷阱遵循预定义的优先级，如图 5-1 所示。陷阱旨在为用户提供一种方法，修正在调试和在应用中工作时的错误操作。

注： 如果用户不想在出现陷阱错误条件时采取纠正措施，则必须在这些向量中装入只包含 RESET 指令的默认陷阱处理程序的地址。否则，如果调用了包含非法地址的向量，将产生地址错误陷阱。

注意，很多陷阱条件只有在发生时才能检测到。因此，在陷阱异常处理之前允许有问题指令完成执行。如果用户选择从错误中恢复，可能不得不修正导致陷阱的错误操作的结果。

陷阱有 8 个固定的优先级：优先级 8 到优先级 15，这意味着在陷阱处理期间，IPL3 始终置 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），这样就禁止了所有中断，但仍然能够处理陷阱。

5.3.1 陷阱的来源

以下陷阱的优先级依次递增。但是，由于所有陷阱都可以嵌套，优先级的作用很小。

数学错误陷阱：

在以下四种情况下，产生数学错误陷阱：

1. 如果试图进行以零作除数的除法操作，它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 进行的算术操作导致 bit 31 溢出且没有使用累加器警戒位时，将产生数学错误陷阱。
3. 如果使能的话，当对累加器 A 或 B 进行的算术操作导致 bit 39 灾难性溢出且所有饱和被禁止时，将产生数学错误陷阱。
4. 如果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。

dsPIC30F5015/5016

地址错误陷阱：

当发生任何下列情形时，将产生陷阱：

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

注： 在 MAC 类指令中，数据空间被分为 X 数据空间和 Y 数据空间，未实现的 X 空间包括所有 Y 空间，而未实现的 Y 空间包括所有 X 空间。

5. 执行“BRA #literal”指令或“GOTO #literal”指令，其中 literal 是未实现的程序存储器地址。
6. 在将 PC 修改为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：

该陷阱在以下条件下产生：

1. 在堆栈指针中装入了一个大于堆栈限制的值（堆栈溢出）：堆栈限制值用户可编程，存放在 SPLIM 寄存器中。
2. 堆栈指针装入了一个小于 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）。这时，图 5-2 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。

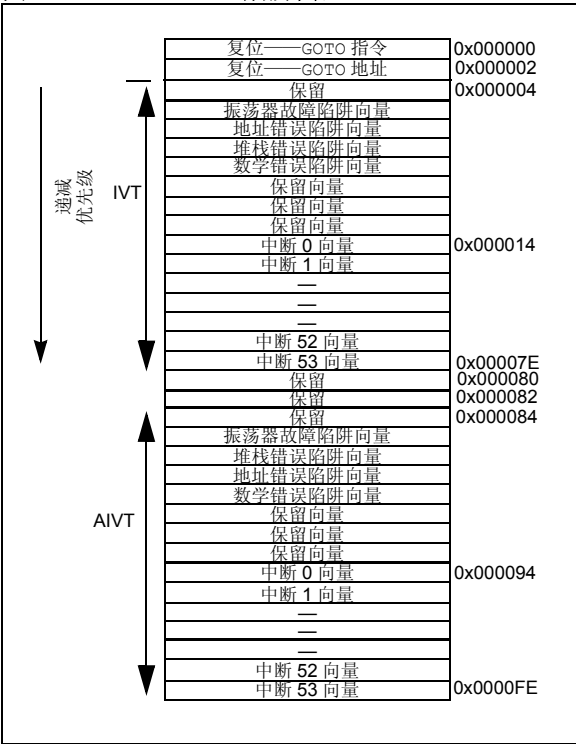
“软”陷阱包括优先级 8 到 11 的异常（含）。算术错误陷阱（优先级 11）就属于这一类陷阱。

“硬”陷阱包括优先级 12 到 15 的异常（含）。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，如果产生了优先级较低的硬陷阱，就会产生硬陷阱冲突。

器件在硬陷阱冲突时自动复位。复位发生时，把 TRAPR 状态位（RCON<15>）置 1，从而在软件中可以检测到该条件。

图 5-1： 陷阱向量



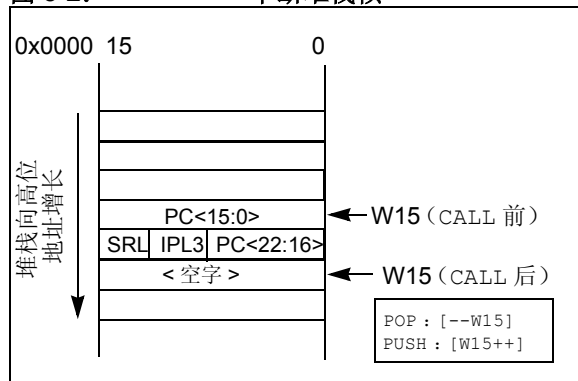
5.4 中断过程

在每个指令周期的开始，IFSx 寄存器都会采样所有的中断事件标志。在 IFSx 寄存器中标志位等于 1，表明有等待处理的中断请求（IRQ）。如果中断允许（IECx）寄存器中相应的位置 1，则 IRQ 会导致中断产生。在余下的指令周期中，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 位中当前处理器优先级高，则处理器将被中断。

随即，处理器将把当前程序计数器和处理器状态寄存器（SRL）的低字节压入堆栈，如图 5-2 所示。状态寄存器低字节包含中断周期开始之前的处理器优先级。然后，处理器把中断的优先级装入状态寄存器。这将禁止所有低优先级的中断，直到中断服务程序完成为止。

图 5-2: 中断堆栈帧



- 注 1:** 通过把一个新值写入 SRL，用户始终能降低优先级。在降低处理器中断优先级之前，为了避免递归中断，中断服务程序必须把 IFSx 寄存器中的中断标志位清零。
- 2:** 在处理中断时，IPL3 位（CORCON<3>）始终被清零。它仅在陷阱执行期间置 1。

RETFIE（从中断返回）指令将把程序计数器和状态寄存器弹出堆栈，使处理器返回到中断处理之前的状态。

5.5 备用向量表

在程序存储器中，中断向量表（IVT）之后是备用中断向量表（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，则所有的中断和异常处理将使用备用向量，而不是默认向量。备用向量与默认向量的组织方式相同。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，而无需重新设定中断向量。该功能还使得可以在运行时在用于评估不同软件算法的应用之间切换。

如果不需要 AIVT，分配给 AIVT 的程序存储空间可以作其他用途。AIVT 不是受保护的区域，用户可以对它自由编程。

5.6 快速现场保护

可以使用影子寄存器来保存现场。为 SR 中的 DC、N、OV、Z 和 C 位，以及寄存器 W0 到 W3 提供了影子寄存器。影子寄存器深度仅为一级。只能使用 PUSH.S 和 POP.S 指令来访问影子寄存器。

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。

如果某个优先级的 ISR 使用 PUSH.S 和 POP.S 指令来快速保存现场，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 5 个外部中断请求信号，INT0-INT4。这些输入是边沿敏感的；它们需要从低至高或从高至低的跳变来产生中断请求。INTCON2 寄存器的 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

如果产生中断时器件处于休眠或空闲模式，中断控制器可以用来把处理器从休眠或空闲模式中唤醒。

如果中断控制器收到已允许的中断请求，且中断请求优先级足够高的话，则标准中断请求将提交给处理器。同时，处理器将从休眠或空闲中唤醒，开始执行处理中断请求所需的中断服务程序（ISR）。

表 5-2: dsPIC30F5015 的中断控制器寄存器映射

SFR 名称	ADR	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	—	—	—	—	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	M2CIF	S2CIF	NVMIF	ADIF	U1TXIF	U1RXIF	SP11IF	T3IF	T2IF	OC2IF	IC2IF	T1IF	OC1IF	IC1IF	INT0IF	0000 0000 0000 0000
IFS1	0086	—	—	IC4IF	IC3IF	C1IF	SP12IF	—	—	INT2IF	T5IF	T4IF	OC4IF	OC3IF	—	—	INT1IF	0000 0000 0000 0000
IFS2	0088	—	—	—	FLTBIF	FLTAIF	—	—	QE1IF	PWMIF	—	INT4IF	INT3IF	—	—	—	—	0000 0000 0000 0000
IEC0	008C	CNIE	M2CIE	S2CIE	NVMIE	ADIE	U1TXIE	U1RXIE	SP11IE	T3IE	T2IE	OC2IE	IC2IE	T1IE	OC1IE	IC1IE	INT0IE	0000 0000 0000 0000
IEC1	008E	—	—	IC4IE	IC3IE	C1IE	SP12IE	—	—	INT2IE	T5IE	T4IE	OC4IE	OC3IE	—	—	INT1IE	0000 0000 0000 0000
IEC2	0090	—	—	—	FLTBIE	FLTAIE	—	—	QE1IE	PWMIE	—	INT4IE	INT3IE	—	—	—	—	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>	—	—	M2CIP<2:0>	—	—	—	—	S2CIP<2:0>	—	—	—	NVMIP<2:0>	—	0100 0100 0100 0100
IPC4	009C	—	—	OC3IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	INT1IP<2:0>	—	0100 0000 0000 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>	—	—	—	—	—	—	—	—	—	—	0100 0100 0000 0000
IPC7	00A2	—	—	—	—	—	—	—	—	—	—	IC4IP<2:0>	—	—	—	IC3IP<2:0>	—	0000 0000 0100 0100
IPC8	00A4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
IPC9	00A6	—	—	PWMIP<2:0>	—	—	—	—	—	—	—	INT41IP<2:0>	—	—	—	INT3IP<2:0>	—	0100 0000 0100 0100
IPC10	00A8	—	—	FLTAIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	QE1IP<2:0>	—	0100 0000 0000 0100
IPC11	00AA	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FLTBIP<2:0>	—	0000 0000 0000 0100

图注: u = 未初始化位
寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 5-3: dsPIC30F5016 的中断控制器寄存器映射

SFR 名称	ADR	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	—	—	—	—	OVBTE	COVTE	—	—	—	—	MATHERR	ADDRERR	STKERR	OSCFail	—	0000 0000 0000 0000
INTCON2	0082	ALTIPT	—	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP	0000 0000 0000 0000
IFS0	0084	CNIF	M2CIF	S2CIF	NVMIF	ADIF	U1TXIF	U1RXIF	SP11IF	T3IF	T2IF	OC2IF	IC2IF	T1IF	OC1IF	IC1IF	INT0IF	0000 0000 0000 0000
IFS1	0086	—	—	—	—	C1IF	SP12IF	—	—	INT2IF	T5IF	T4IF	OC4IF	OC3IF	—	—	INT1IF	0000 0000 0000 0000
IFS2	0088	—	—	—	—	FLTBIF	FLTAIF	—	QE1IF	PWMIF	—	INT4IF	INT3IF	—	—	—	—	0000 0000 0000 0000
IEC0	008C	CNIE	M2CIE	S2CIE	NVMIE	ADIE	U1TXIE	U1RXIE	SP11IE	T3IE	T2IE	OC2IE	IC2IE	T1IE	OC1IE	IC1IE	INT0IE	0000 0000 0000 0000
IEC1	008E	—	—	—	—	C1IE	SP12IE	—	—	INT2IE	T5IE	T4IE	OC4IE	OC3IE	—	—	INT1IE	0000 0000 0000 0000
IEC2	0090	—	—	—	—	FLTBIE	FLTAIE	—	QE1IE	PWMIE	—	INT4IE	INT3IE	—	—	—	—	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>	—	—	—	SP11IP<2:0>	—	0100 0100 0100 0100
IPC3	009A	—	—	CNIP<2:0>	—	—	M2CIP<2:0>	—	—	—	—	S2CIP<2:0>	—	—	—	NVMIP<2:0>	—	0100 0100 0100 0100
IPC4	009C	—	—	OC3IP<2:0>	—	—	—	—	—	—	—	—	—	—	—	INT1IP<2:0>	—	0100 0000 0000 0100
IPC5	009E	—	—	INT2IP<2:0>	—	—	T5IP<2:0>	—	—	—	—	T4IP<2:0>	—	—	—	OC4IP<2:0>	—	0100 0100 0100 0100
IPC6	00A0	—	—	C1IP<2:0>	—	—	SP12IP<2:0>	—	—	—	—	—	—	—	—	—	—	0100 0100 0000 0000
IPC7	00A2	—	—	—	—	—	—	—	—	—	—	IC4IP<2:0>	—	—	—	IC3IP<2:0>	—	0000 0000 0100 0100
IPC8	00A4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0100
IPC9	00A6	—	—	PWMIP<2:0>	—	—	—	—	—	—	—	INT41IP<2:0>	—	—	—	INT3IP<2:0>	—	0100 0000 0100 0100
IPC10	00A8	—	—	FLTAIP<2:0>	—	—	—	—	—	—	—	—	—	—	—	QE1IP<2:0>	—	0100 0000 0000 0100
IPC11	00AA	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FLTBIP<2:0>	—	0000 0000 0000 0100

u = 未初始化位

图注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F5015/5016

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。关于 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)。关于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 系列器件包含用于执行用户代码的内部闪存程序存储器。用户可以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)

dsPIC30F 器件可以在最终的应用电路中进行串行编程。实现很简单，仅需一根编程时钟线（称为 PGC）、一根编程数据线（称为 PGD），以及电源线（VDD）、接地线（VSS）和主复位线（MCLR）。这允许用户在生产电路板时使用未编程器件，仅在产品交付之前才对单片机进行编程，从而可以使用最新版本的固件或者定制固件进行编程。

6.2 运行时自编程 (RTSP)

使用 TBLRD（表读）和 TBLWT（表写）指令来实现 RTSP。

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。

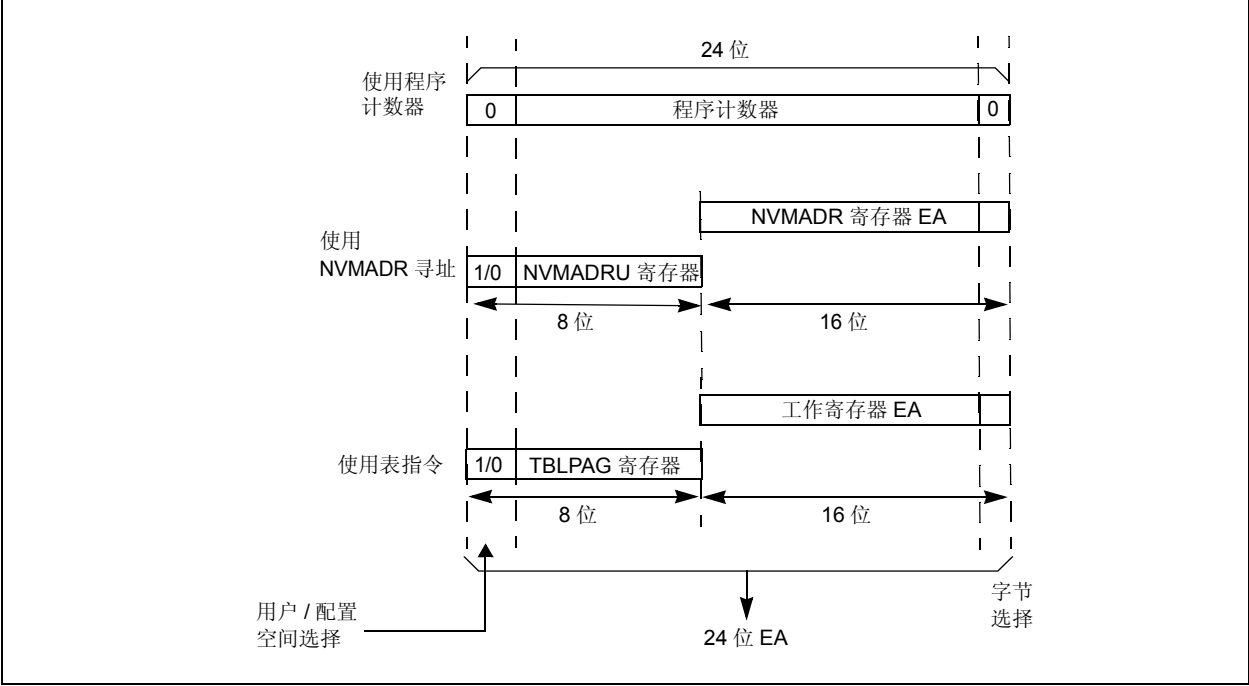
6.3 表指令操作汇总

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

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

图 6-1： 表和 NVM 寄存器的寻址



6.4 RTSP 工作原理

dsPIC30F 闪存程序存储器组织成行和板 (panel)。每行由 32 个指令字 (96 字节) 组成。每个板由 128 行组成 (4K x 24 个指令字)。RTSP 允许用户每次擦除一行 (32 个指令字)、每次编程 32 条指令。

程序存储器的每个板都包含写锁存器, 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入写锁存器中: 指令 0, 指令 1, 依此类推。装入的地址必须始终来自一组偶校验 32 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, 装入写锁存器。通过把 NVMCON 寄存器中的特殊位置 1, 进行编程。32 装入 32 条指令需要 32 条 TBLWTL 和 32 条 TBLWTH 指令。

全部表写操作都是单字写入 (2 个指令周期), 因为只需要写表锁存器。

写锁存器后, 需要启动编程操作对数据编程。

正常工作状态下, 闪存程序存储器在整个 VDD 范围内都是可读写可擦除的。

6.5 RTSP 控制寄存器

有 4 个特殊功能寄存器用来读、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存储块、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 寄存器用来存放有效地址的两个低字节。它捕捉已执行的上一条表指令的 EA<15:0>, 选择要写的行。

6.5.3 NVMADRU 寄存器

NVMADRU 寄存器用来存放有效地址的高字节。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 是只写寄存器, 用于写保护。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, 指定擦除或编程的程序存储器地址。
-----------	--

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，处理器将停顿（等待）直到编程操作完成。把 WR 位（NVMCON<15>）置 1 将开始操作，当操作完成时 WR 位会自动清零。

6.6.1 闪存程序存储器的编程算法

用户一次可以擦除或编程闪存程序存储器的一行。一般过程如下：

1. 读一行闪存程序存储器（32 个指令字），把读出的数据作为数据“镜像”保存到数据 RAM。
2. 用期望的新数据更新数据镜像。
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) 将 WR 位置 1。这将开始擦除周期。
 - f) 在擦除周期中 CPU 将停顿。
 - g) 当擦除周期结束时 WR 位会被清零。

4. 从数据 RAM “镜像”中，把 32 个指令字的数据写入闪存程序存储器写锁寄存器。
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 将停顿。
 - 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空间进行编程。

6.6.2 擦除程序存储器的一行

例 6-1 所示的代码序列，可以用来擦除一行程序存储器（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
```

dsPIC30F5015/5016

6.6.3 装入写锁存器

例 6-2 所示的指令序列，可以用来装入 96 字节的写锁存器。装入由表指针选择的写锁存器，需要 32 条 TBLWTL 指令和 32 条 TBLWTH 指令。

例 6-2: 装入写锁存器

```
; 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 的高字节的内容没有影响。

6.6.4 启动编程序列

出于保护的目的，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。在编程命令执行后，用户必须等待一段编程时间，直至编程完成。编程过程开始之后，紧接着的应该是两条 NOP 指令。

例 6-3: 启动编程过程

```
DISI    #5                          ; Block all interrupts with priority <7
                                              ; for next 5 instructions

MOV      #0x55,W0                    ; Write the 0x55 key
MOV      W0,NVMKEY
MOV      #0xAA,W1                    ;
MOV      W1,NVMKEY                   ; Write the 0xAA key
BSET     NVMCON,#WR                  ; Start 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	全部复位
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
注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F5015/5016

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。关于 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)。关于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 范围内的正常运行期间，数据 EEPROM 存储器是可读写的。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 4 个 SFR，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，用来寻址要访问的 EEPROM 地址单元。TBLRDL 和 TBLWTL 指令用来读写数据 EEPROM。dsPIC30F6010 器件有最多 8 KB（4K 字）的数据 EEPROM，地址范围从 0x7FF000 到 0x7FFFFE。

字写操作之前，应该先擦除相应的存储单元。写操作完成通常需要 2 ms，不过它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，尝试读数据 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写类似。用软件只能将该位置 1 而无法清零。在写操作完成后，由硬件将该位清零。由于无法用软件将 WR 位清零，可避免写操作意外或过早终止。

当 WREN 位置 1 时，允许进行写操作。上电时，WREN 位被清零。当正常工作期间，如果写操作被 MCLR 复位或 WDT 超时复位中断，则 WRERR 位将被置 1。在这些情况下，复位后用户可以检查 WRERR 位并重写存储单元。地址寄存器 NVMADR 保持不变。

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

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
TBLRDL [ W0 ], W4            ; read data EEPROM
```

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 块，并将 NVMCON 寄存器中的 ERASE 位和 WREN 位置 1。将 WR 位置 1，启动擦除操作，如例 7-2 所示。

例 7-2: 数据 EEPROM 块擦除

```
; Select data EEPROM block, ERASE, 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

TBLPAG 和 NVMADR 寄存器必须指向存储块。在 NVMCON 寄存器中，擦除闪存数据块，将 ERASE 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

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

; Start erase cycle by setting WR after writing key sequence
DISI     #5                          ; Block all interrupts with priority <7
                                           ; for next 5 instructions

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

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

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 字。
 - a) 在 NVMCON 寄存器中，选择字擦除数据 EEPROM，并将 WREN 位置 1。
 - b) 把要擦除的字的地址写入 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) 或者查询 NVMIF 位，或者等待 NVMIF 中断。
 - h) 当擦除周期结束时 WR 位会被清零。
2. 把数据字写入数据 EEPROM 的写锁存器中。
3. 把 1 个数据字编程到数据 EEPROM 中。
 - a) 在 NVMCON 寄存器中，选择字编程数据 EEPROM，并将 WREN 位置 1。
 - 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) 或者查询 NVMIF 位，或者等待 NVM 中断。
 - g) 当写周期结束时 WR 位会被清零。

对于每个字，如果上述序列（把 0x55 写入 NVMKEY，把 0xAA 写入 NVMCON，然后将 WR 位置 1）没有严格遵守的话，写操作不会启动。强烈建议在这个代码段执行期间禁止中断。

此外，必须将 NVMCON 中的 WREN 位置 1 来使能写操作。这种机制可防止由于意外执行代码导致误写数据 EEPROM。除了更新 EEPROM 时以外，WREN 位应始终保持清零。WREN 位不会被硬件清零。

一个写过程启动后，将 WREN 位清零不会影响当前的写周期。除非将 WREN 位置 1，否则将禁止 WR 位被置 1。WREN 位必须在前一条指令时被置 1。WR 和 WREN 不能由同一指令置 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。用户可以允许此中断或查询此位。NVMIF 必须用软件清零。

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 字写入

```

; 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

```

7.3.2 写数据 EEPROM 的一个块

要写数据 EEPROM 的一个块，首先要写入全部十六个锁存器，然后设置 NVMCON 寄存器并进行块编程。

例 7-5: 数据 EEPROM 块写入

```
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 NVMDR 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

根据具体应用，将写入存储器的值与原始值相校验是一个很好的编程习惯。在应用中，如果某些位的写次数接近规定极限值，就应该采用写校验。

7.5 防止误写操作的保护措施

有些情况下，用户并不想向器件的数据 EEPROM 存储器写入数据。器件内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。

在欠压、电源毛刺或软件故障期间，写操作的启动序列以及 WREN 位可共同防止意外写操作的发生。

8.0 I/O 端口

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。关于 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)。

所有的器件引脚（除 VDD、VSS、MCLR 和 OSC1/CLKI 之外），均为外设与并行 I/O 端口所共用。

所有 I/O 输入端口都是施密特触发器输入，以增强其抗干扰性。

8.1 并行 I/O (PIO) 端口

如果外设使能，并且外设正在使用其关联引脚时，引脚将不再作为通用 I/O 引脚使用。这时，可以读 I/O 引脚，但是对应并行端口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。

所有的端口引脚，都有三个与端口引脚的操作直接相关的寄存器。数据方向寄存器 (TRISx) 决定引脚是输入还是输出。如果数据方向位是 1，则引脚是输入。复位后，所有端口引脚都被定义为输入。读锁存器 (LATx)，读的是锁存值；写锁存值，写的是锁存器 (LATx)。读端口 (PORTx)，读的是端口引脚值；写端口引脚，写的是锁存器 (LATx)。

对于特定器件无效的任何位及其相关的数据和控制寄存器都将被禁止。这意味着对应的 LATx 和 TRISx 寄存器以及端口引脚都将读为零。

当端口引脚与另一个外设共用或与只定义为输入的功能共用时，它将被视为专用端口，因为没有任何其他竞争的输出资源。INT4 引脚就是例子。

PORTA 的寄存器的格式，如表 8-2 所示。

TRISA（数据方向控制）寄存器控制 RA<7:0> 引脚的方向，以及 INTx 引脚和 VREF 引脚的方向。LATA 寄存器提供数据到输出，它是可读 / 可写的。读 PORTA 寄存器得到输入引脚的状态，而写 PORTA 寄存器修改 LATA 寄存器的内容。

与外设共用引脚的并行 I/O (PIO) 端口一般屈从于外设。外设的输出缓冲数据和控制信号提供给一对多路开关。多路开关选择外设还是相关的端口将拥有 I/O 引脚输出数据和控制信号的所有权。图 8-2 显示了端口是如何与其他外设共用，以及与外设连接的相关 I/O 引脚单元。表 8-1 至表 8-2 给出了从 PORTB 到 PORTG 的共用端口的寄存器的格式。

图 8-1: 专用端口结构框图

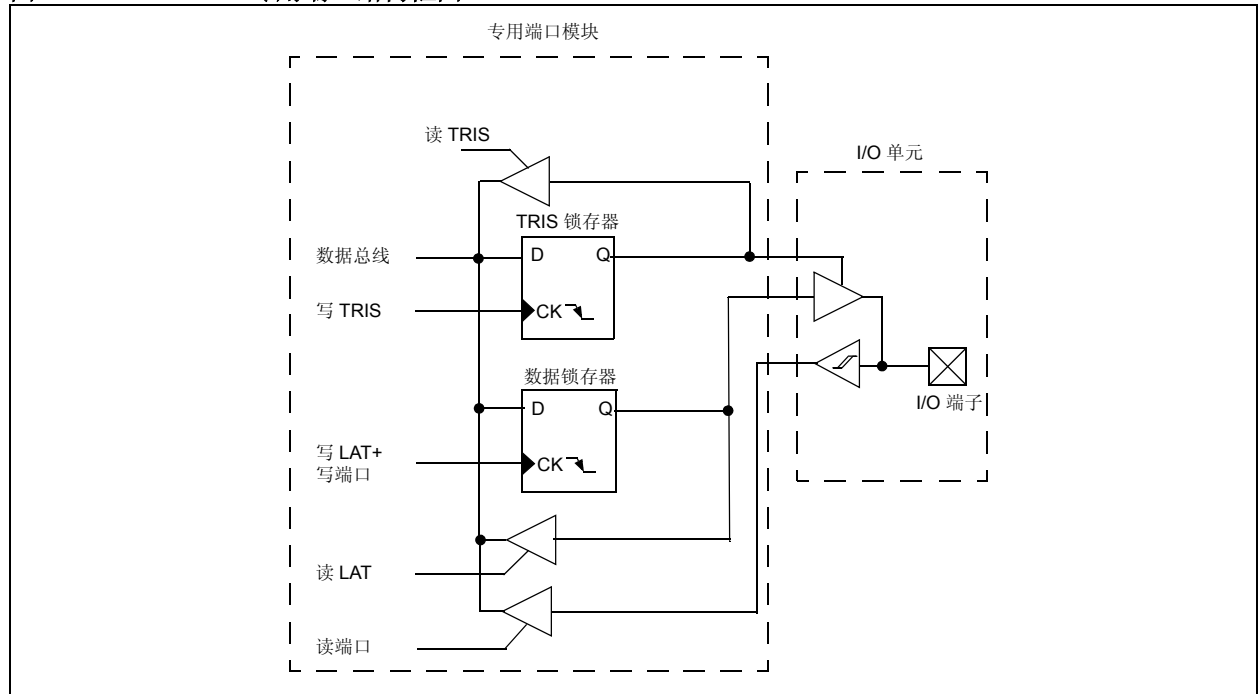
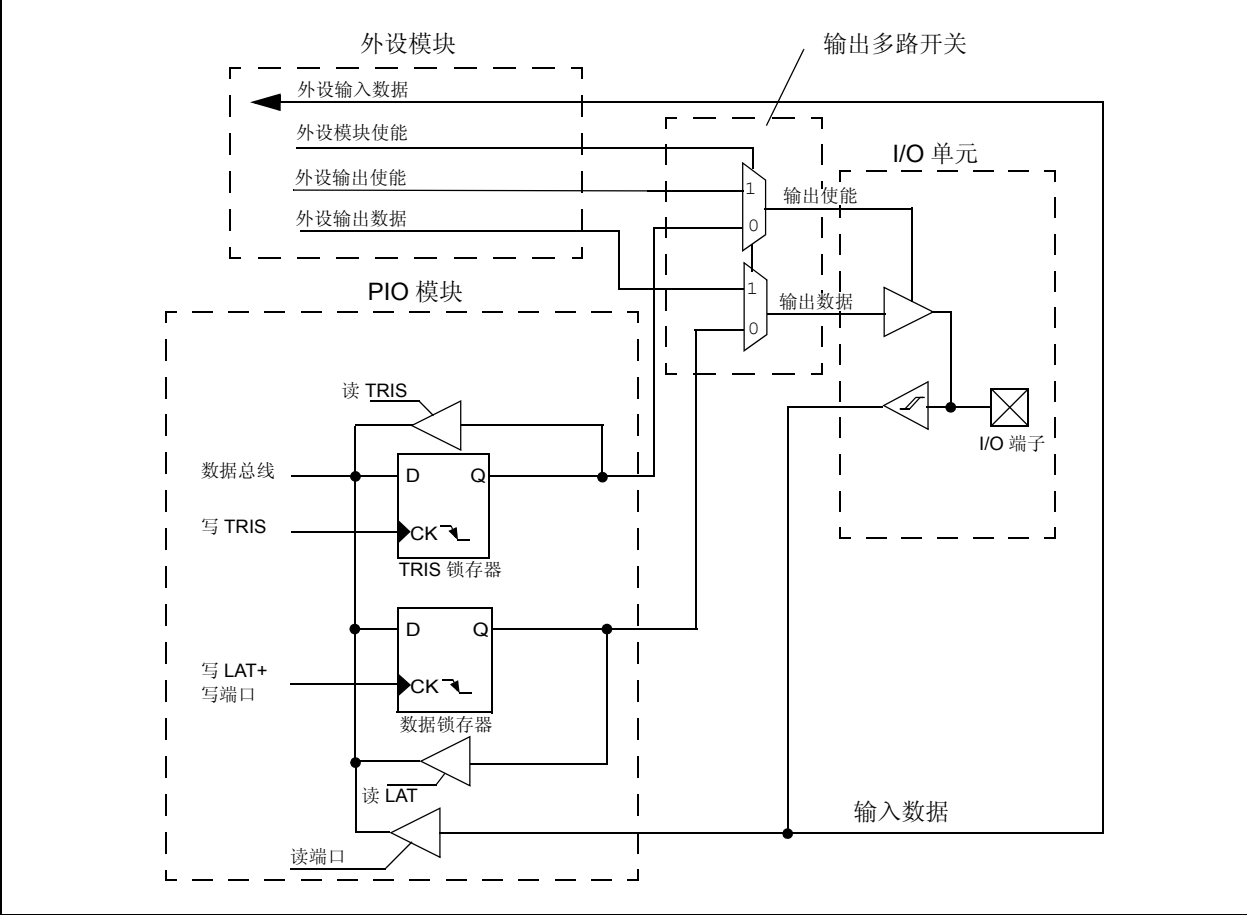


图 8-2: 共用端口结构框图



8.2 配置模拟端口引脚

使用 ADPCFG 和 TRIS 寄存器控制 A/D 端口引脚的操作。如果要将端口引脚用作模拟输入，则对应的 TRIS 位必须置 1（输入）。如果将 TRIS 位清零（输出），则数字输出电平（VOH 或 VOL）将被转换。

当读取端口寄存器时，所有配置为模拟输入通道的引脚都将读为零（低电平）。

配置为数字输入的引脚，不会被当作模拟输入进行转换。对于任何定义为数字输入的引脚（包括 ANx 引脚），加在引脚上的模拟电压可能导致输入缓冲器消耗的电流超出器件规范中规定的值。

8.2.1 I/O 端口写 / 读时序

端口方向改变或端口写操作与同一端口的读操作之间需要一个指令周期。通常该指令是 NOP。

例 8-1: 端口写 / 读示例

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

表 8-1: dsPIC30F5015 的端口寄存器映射

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

u = 未初始化位

注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 8-2: dsPIC30F5016 的端口寄存器映射

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
PORTF	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 = 未初始化位

注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

8.3 输入变化通知模块

输入变化通知模块使 dsPIC30F 器件能够向处理器发出中断请求，以响应所选输入引脚上的状态变化。即使是在休眠模式下、时钟被禁止时，此模块也能检测输入的

状态变化。可选择（使能）22 个外部信号（CN0 到 CN21）产生状态变化时的中断请求。

CN 引脚的位置，请参见引脚图。

表 8-3: dsPIC30F5015 的输入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 = 未初始化位

表 8-4: dsPIC30F5015 的输入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 = 未初始化位

表 8-5: dsPIC30F5016 的输入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 = 未初始化位

表 8-6: dsPIC30F5016 的输入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 = 未初始化位

dsPIC30F5015/5016

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。关于 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 16 位通用 (General Purpose, GP) Timer1 模块及其相关的工作模式。

注： Timer1 是一个 A 类定时器。关于 A 类定时器的规范，请参见此文档中的第 24.0 节“电气特性”。

对于定时器的工作模式，以下各小节将详细说明设置、控制寄存器以及相关的框图。

Timer1 模块是一个 16 位定时器，可以用作实时时钟的时间计数器，或者用作自由运行的时段定时器/计数器。

16 位定时器具有以下模式：

- 16 位定时器
- 16 位同步计数器
- 16 位异步计数器

而且，支持以下操作特性：

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

通过设定 16 位 SFR T1CON 中的相应位来决定这些工作模式。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，定时器在每个指令周期递增，直到等于周期寄存器 PR1 中预先装入的值，然后复位至 0，继续计数。

当 CPU 进入空闲模式时，定时器将停止递增，除非 TSIDL (T1CON<13>) 位 = 0。如果 TSIDL = 1，定时器模块逻辑将在 CPU 空闲模式结束后继续递增。

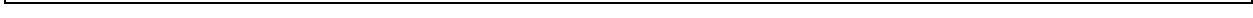
16 位同步计数器模式：在 16 位同步计数器模式下，定时器在外部时钟信号（与内部相位时钟同步）的上升沿递增。定时器计数，直到等于 PR1 中预先装入的值，然后复位至 0，继续计数。

当 CPU 进入空闲模式时，定时器将停止递增，除非相应的 TSIDL 位 = 0。如果 TSIDL = 1，定时器模块逻辑将在 CPU 空闲模式结束后继续递增。

16 位异步计数器模式：在 16 位异步计数器模式下，定时器在外部时钟信号的上升沿递增。定时器计数，直到等于 PR1 中预先装入的值，然后复位至 0，继续计数。

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

--



如果满足所有上述三个条件，它将继续计数，直到

中断台位置 4，则将产生中断

9.4 定时器中断

16 位定时器能够在周期匹配时产生中断。当定时器计数与周期寄存器中匹配时，**T1IF** 位将置 1，将产生中断（如果允许中断的话）。**TxIF** 位必须用软件清零。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，还将在门控信号的下降沿（累加周期的末尾）产生中断。

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** 工作在实时时钟（Real-Time Clock，**RTC**）模式下时，提供了当天时间和事件时间标记的能力。**RTC** 的关键工作特性如下：

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

通过 **T1CON** 控制寄存器中的相应位来决定这些工作模式。

9.5.1 RTC 振荡器操作

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，直至递增到周期寄存器中指定的值，然后复位至 0。

为确保正确工作，必须将 **TSYNC** 位置为逻辑 0（异步模式）。

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

当 **CPU** 进入休眠模式后，如果 32 kHz 外部晶体振荡器处于工作状态并且控制位没有改变的话，**RTC** 将继续工作。为使 **RTC** 在空闲模式下继续工作，应将 **TSIDL** 位置为 0。

9.5.2 RTC 中断

当发生中断事件时，相应的中断标志 **T1IF** 发生将置 1，并将产生中断（如果允许中断的话）。**TxIF** 位必须用软件清零。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。

图 9-2: **TIMER1 LP 振荡器 RTC 的建议元件**

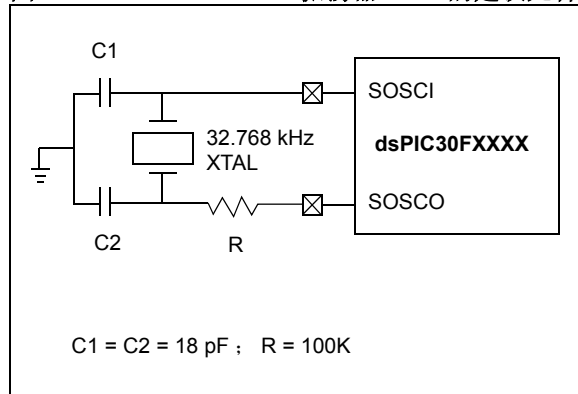


表 9-1: TIMER1 寄存器映射

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 = 未初始化位
寄存器中各位的说明, 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。关于 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 32 位通用 (GP) 定时器模块 (Timer2/3) 及其相关的工作模式。图 10-1 所示为 32 位 Timer2/3 模块的简化框图。图 10-2 和图 10-3 给出了分别将 Timer2/3 配置成的两个独立 16 位定时器 Timer2 和 Timer3 的框图。

注： Timer2 为 B 类定时器，Timer3 为 C 类定时器。关于相应的定时器类型，请参见本文档中的第 24.0 节“电气特性”。

Timer2/3 模块是一个 32 位定时器，可配置为两个 16 位定时器，并且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，以下各小节将详细说明设置、控制寄存器以及相关的框图。

32 位定时器具有以下模式：

- 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 位定时器操作
- 一个 32 位同步计数器

而且，支持以下操作特性：

- ADC 事件触发信号
- 定时器门控操作
- 可选择的预分频比设置
- 空闲和休眠模式下的定时器操作
- 在 32 位周期寄存器匹配时产生中断

通过设置 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 位定时器的最低有效字，而 Timer3 是最高有效字。

注： 对于 32 位定时器操作，T3CON 控制位被忽略。只使用 T2CON 控制位进行设置和控制。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 (T3IF) 反映，而且中断通过 Timer3 中断允许位 (T3IE) 来允许。

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

Timer2 和 Timer3 之间惟一的功能差异是，Timer2 提供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。

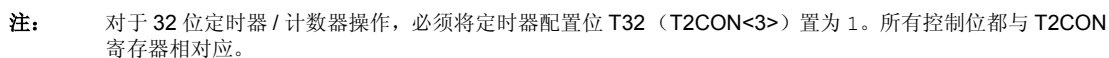
对于一对 Timer2/Timer3 的同步 32 位读操作，读最低有效字 (TMR2 寄存器) 将导致读入最高有效字并锁存到一个 16 位保持寄存器 (称为 TMR3HLD)。

对于同步 32 位写操作，必须首先写入保持寄存器 (TMR3HLD)。如果接着对 TMR2 寄存器执行写操作，则 TMR3HLD 的内容将被传送并锁存到 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，定时器在外部时钟信号 (与内部相位时钟同步) 的上升沿递增。定时器计数，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。

当将定时器配置为同步计数器工作模式时，且 CPU 进入空闲模式，定时器将停止递增，除非 TSIDL (T2CON<13>) 位 = 0。如果 TSIDL = 1，定时器模块逻辑将在 CPU 空闲模式结束后继续递增。

--



10.1 定时器门控操作

可以将 32 位定时器配置为门控时间累加模式。此模式使得当门控输入信号（T2CK 引脚）被置为高电平时，内部 Tcy 能够递增相应的定时器。要使能此模式，必须将控制位 TGATE（T2CON<6>）置 1。在此模式下，Timer2 是发起时钟源。对于 Timer3，TGATE 设置被忽略。必须使能定时器（TON = 1），并且要将定时器设置为使用内部时钟源（TCS = 0）。

外部信号的下降沿会终止计数操作，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）之间发生匹配时，Timer3 将产生特殊的 ADC 触发事件信号。

10.3 定时器预分频器

定时器的输入时钟（Fosc/4 或外部时钟）具有以下预分频比选项：1:1、1:8、1:64 和 1:256，这些预分频比可通过控制位 TCKPS<1:0>（T2CON<5:4>和 T3CON<5:4>）进行选择。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。当出现以下事件之一时，预分频计数器将清零：

- 写 TMR2/TMR3 寄存器
- 将 TON（T2CON<15> 或 T3CON<15>）位设为 0
- 器件复位，如 POR 和 BOR

但是，如果定时器被禁止（TON = 0），由于预分频器时钟已停止，便无法复位 Timer2 预分频器了。

写 T2CON/T3CON 时不会清零 TMR2/TMR3。

10.4 休眠模式下的定时器操作

在 CPU 休眠模式期间，由于内部时钟被禁止，定时器不工作。

10.5 定时器中断

32 位定时器模块可以在发生周期匹配或外部门控信号出现下降沿时产生中断。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（FS0<7>）将置 1，并将产生中断（如果允许中断的话）。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（仅限 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 ⁽¹⁾	—	0000 0000 0000 0000	
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	—	—	—	0000 0000 0000 0000	

图注: u = 未初始化位。

注 1: dsPIC30F5015 的 T2CON 寄存器中没有 TCS。

注 2: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F5015/5016

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。关于 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)。

本节叙述第二个 32 位通用定时器模块 (Timer4/5) 及其相关的工作模式。图 11-1 所示为 32 位 Timer4/5 模块的简化框图。图 11-2 和图 11-3 给出了分别将 Timer4/5 配置成的两个独立 16 位定时器 Timer4 和 Timer5 的框图。

注: Timer4 为 B 类定时器, Timer5 为 C 类定时器。关于相应的定时器类型,请参见本文档的**第 24.0 节“电气特性”**。

Timer4/5 模块的工作与 Timer2/3 模块相似。但是，也存在如下区别：

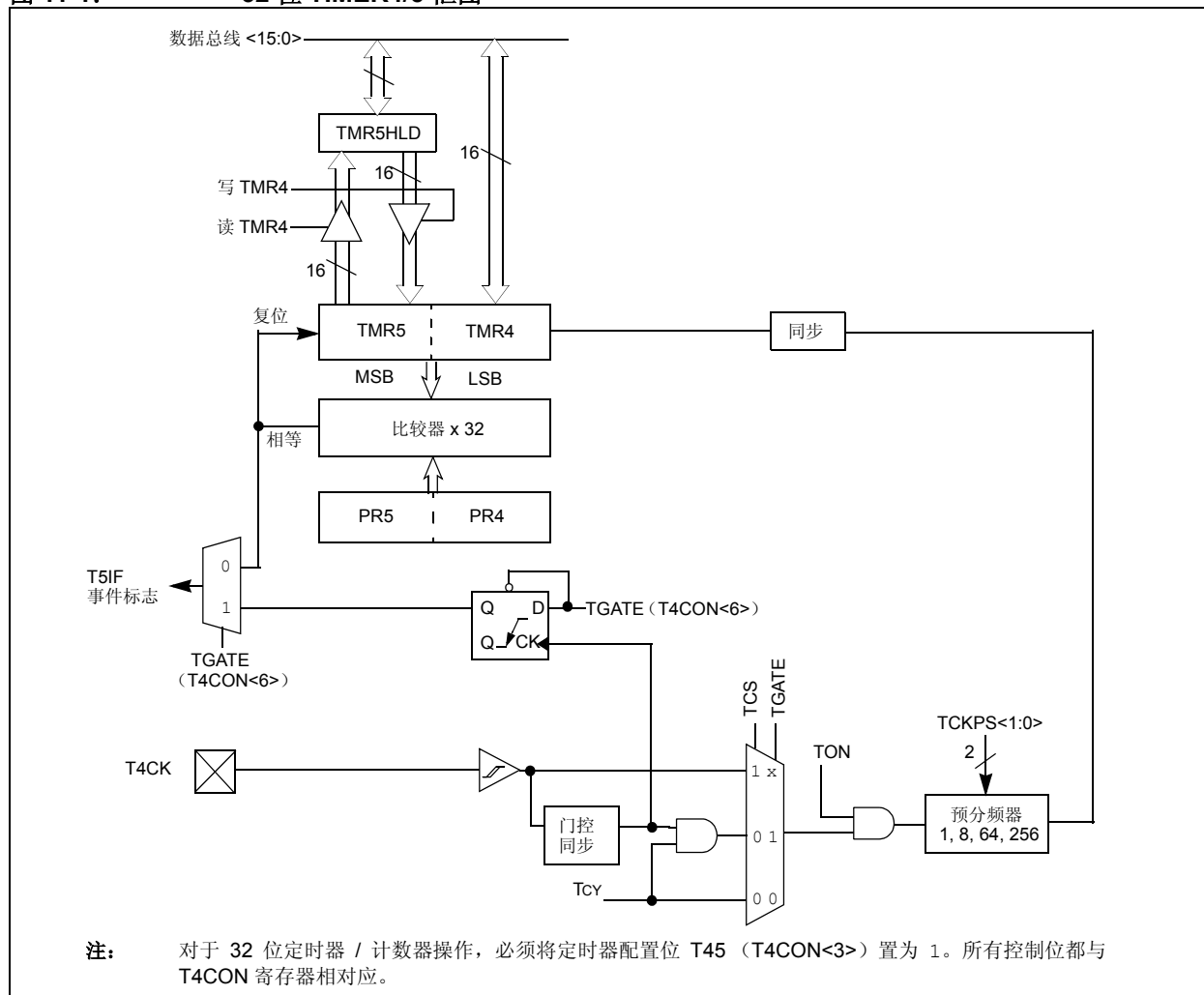
- Timer4/5 模块不支持 ADC 事件触发信号功能
- Timer4/5 不能为例如输入捕捉和输出比较的其他外设模块所使用

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

对于 32 位定时器 / 计数器操作, Timer4 是 32 位定时器的最低有效字, 而 Timer5 是最高有效字。

注: 对于 32 位定时器操作, T5CON 控制位被忽略。只使用 T4CON 控制位进行设置和控制。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 (T5IF) 反映, 而且中断通过 Timer5 中断允许位 (T5IE) 来允许。

图 11-1: 32 位 TIMER4/5 框图



dsPIC30F5015/5016

图 11-2: 16 位 TIMER4 框图 (B 类定时器)

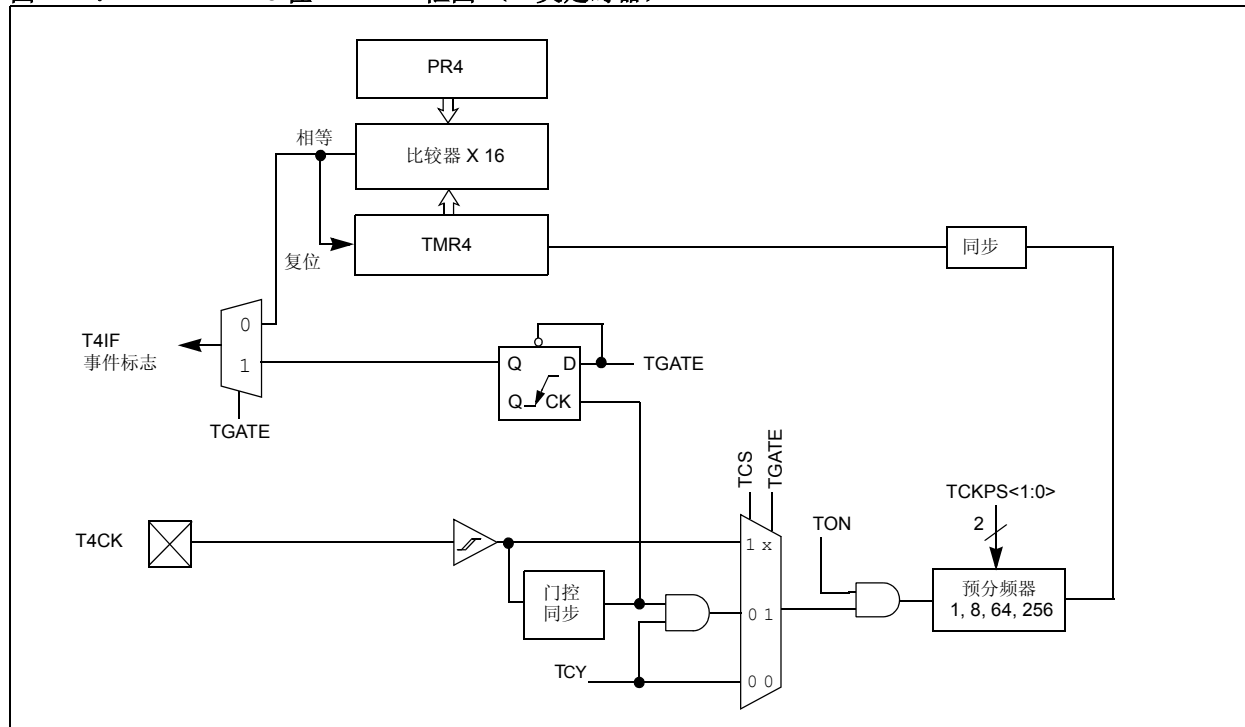


图 11-3: 16 位 TIMER5 框图 (C 类定时器)

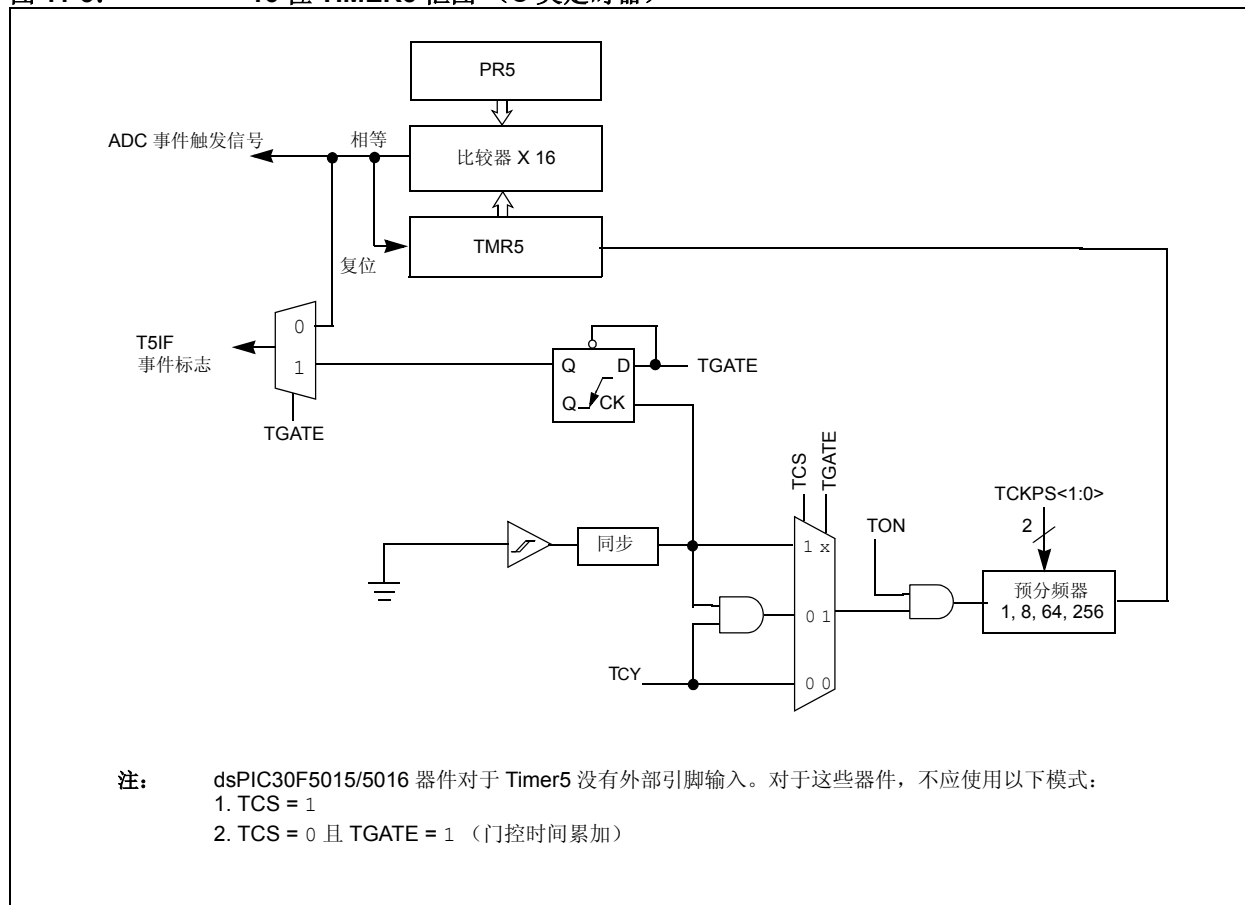


表 11-1: TIMER4/5 寄存器映射

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 (仅限 32 位操作)																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	—	—	—	—	0000 0000 0000 0000

u = 未初始化位

图注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F5015/5016

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。关于 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)。

本节叙述输入捕捉模块及其相关的工作模式。此模块所提供的功能对于需要测量频率（周期）和脉冲的应用是很有用的。图 12-1 所示为输入捕捉模块的框图。输入捕捉可用于以下模式：

- 频率 / 周期 / 脉冲测量
- 附加外部中断源

输入捕捉模块的主要工作特性如下：

- 简单捕捉事件模式
- TIMER2 和 TIMER3 模式选择
- 发生输入捕捉事件时产生中断

通过设置 ICXCON 寄存器（其中 X = 1,2,...,N）中的相应位来决定这些工作模式。DSPIC30F5015/5016 器件具有 8 路捕捉通道。

12.1 简单捕捉事件模式

DSPIC30F 产品系列中的简单捕捉事件有：

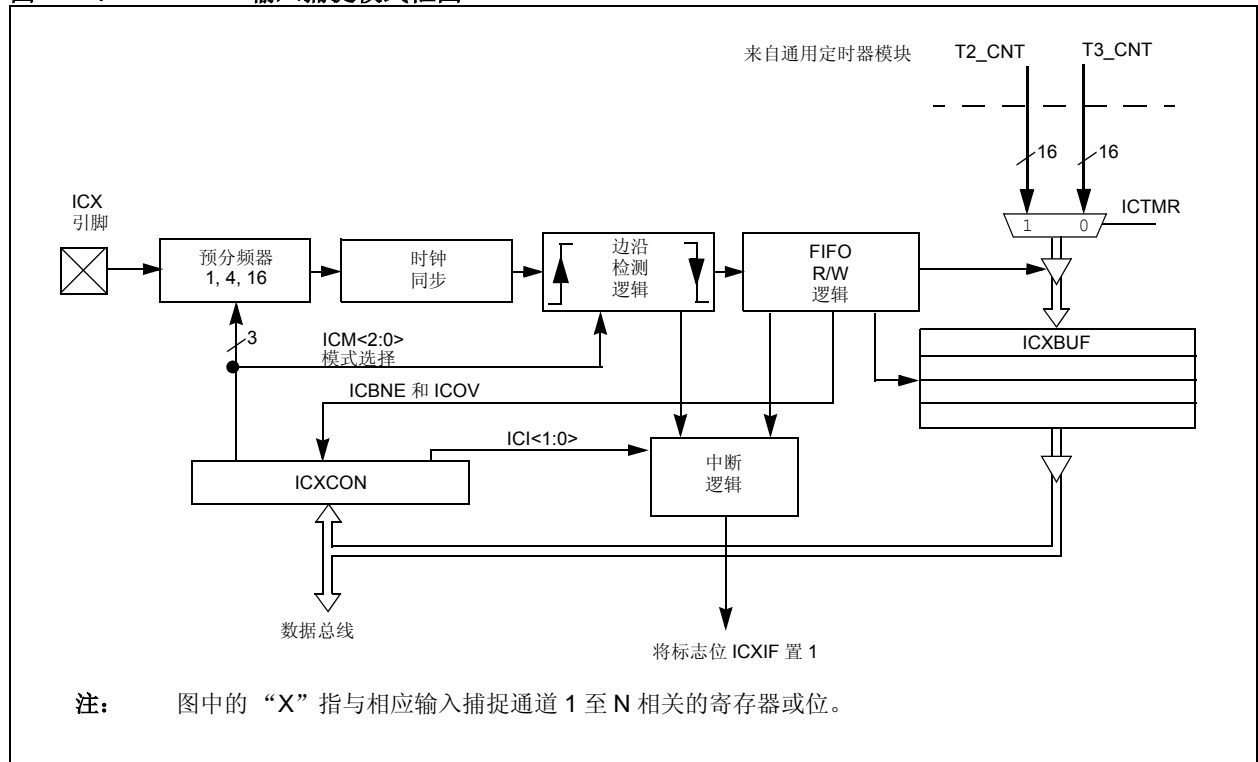
- 在每个下降沿进行捕捉
- 在每个上升沿进行捕捉
- 在每第 4 个上升沿进行捕捉
- 在每第 16 个上升沿进行捕捉
- 在每个上升沿和下降沿进行捕捉

通过设置适当的位 ICM<2:0>（ICXCON<2:0>），选择这些简单输入捕捉模式。

12.1.1 捕捉预分频器

有 4 种输入捕捉预分频器设置，它们由 ICM<2:0>（ICXCON<2:0>）位指定。每当捕捉通道关闭时，预分频计数器将清零。此外，任何复位都将清零预分频计数器。

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关联的 FIFO 缓冲器，该缓冲器深度为 4 级 16 位字。有两个状态标志，提供 FIFO 缓冲器的状态：

- ICBFNE——输入捕捉缓冲器非空
- ICOV——输入捕捉溢出

当发生第一个输入捕捉事件时，ICBFNE 位被置 1，该状态将保持到所有捕捉事件都已从 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步位置。

如果 FIFO 已被 4 个捕捉事件填满，在读 FIFO 之前又发生第 5 个捕捉事件，则将产生溢出，ICOV 位被设置为逻辑 1。第 5 个捕捉事件将丢失，不会存入 FIFO。在将 4 个事件全部从缓冲器中读出之前，不会再捕捉任何事件。

如果 FIFO 已全部读完，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

每个捕捉通道都可以选择两个定时器（TIMER2 或 TIMER3）之一作为时基。

定时器资源的选择通过 SFR 位 ICTMR (ICXCON<7>) 来实现。TIMER3 是输入捕捉模块默认的定时器资源。

12.1.4 霍尔传感器模式

当将输入捕捉模块设置为在每个边沿（上升沿和下降沿，CM<2:0> = 001）进行捕捉时，输入捕捉逻辑将执行以下操作：

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

12.2 休眠和空闲模式期间的输入捕捉操作

当器件处于 CPU 空闲或休眠模式下时，输入捕捉事件将唤醒器件或产生中断（如果允许中断的话）。

如果 ICM<2:0> = 111 且中断允许位置 1，那么当发生捕捉事件时，输入捕捉模块将使 CPU 从休眠或空闲模式唤醒，这与使能的定时器无关。如果已满足处理中断的条件，前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 休眠模式允许输入捕捉模块以简化功能工作。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，必须将模块配置为仅在上升沿产生中断（ICM<2:0> = 111）。在此模式下，4:1 或 16:1 的预分频比设置不适用。

12.2.2 CPU 空闲模式下的输入捕捉

CPU 空闲模式允许输入捕捉模块以完整功能工作。在 CPU 空闲模式下，可以通过 ICI<1:0> 位选择中断模式，还可以通过控制位 ICM<2:0> 选择 4:1 和 16:1 捕捉预分频比设置。此模式要求使能所选的定时器。此外，还必须将 ICSIDL 位设置为逻辑 0。

在 CPU 空闲模式下，如果将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

输入捕捉通道可以根据所选的捕捉事件数来产生中断。所选的捕捉事件数可以通过控制位 ICI<1:0> (ICXCON<6:5>) 进行设置。

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

通过各捕捉通道中断允许（TCXIE）位来允许中断。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 1 捕捉寄存器																UUUU UUUU UUUU UUUU
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC2BUF	0144	输入 2 捕捉寄存器																UUUU UUUU UUUU UUUU
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC3BUF	0148	输入 3 捕捉寄存器																UUUU UUUU UUUU UUUU
IC3CON	014A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC4BUF	014C	输入 4 捕捉寄存器																UUUU UUUU UUUU UUUU
IC4CON	014E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000

U = 未初始化位

图注: 寄存器中各位的说明, 请参见《DSPIC30F 系列参考手册》(DS70046E_CN)。

注:

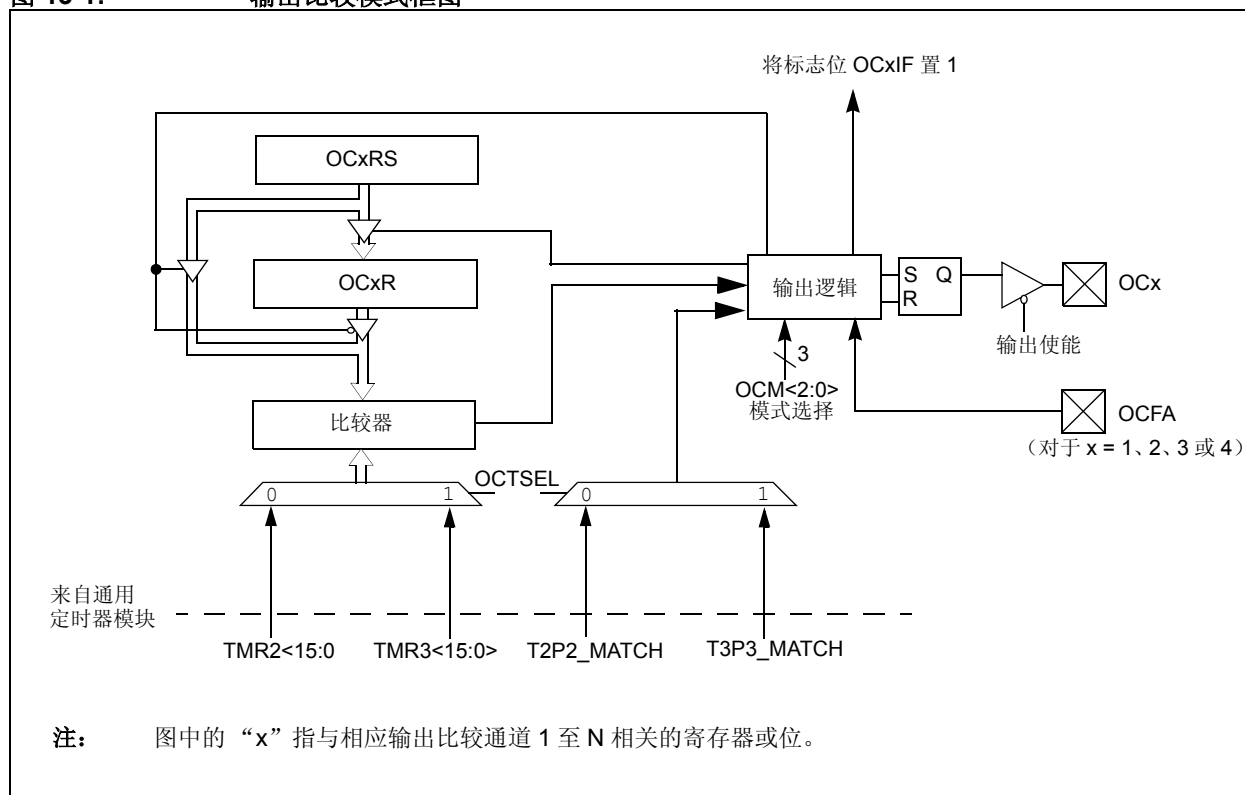
DSPIC30F5015/5016

注:

图 13-1 所示为输出比较模块的框图。

图中的 OCxRS 和 OCxR 表示双比较寄存器。在双比较模式下，OCxR 寄存器用于第一个比较，OCxRS 用于第二个比较。

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。

定时器的选择由 OCTSEL 位（OCxCON<3>）控制。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以下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

这些模式中使用 OCxR 寄存器。OCxR 寄存器装入一个值，然后与所选的递增定时器的计数值作比较。进行比较时，将出现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以下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

为将模块配置为生成单输出脉冲，用户需要执行以下步骤（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，只要再次将写操作设置为 OCM<2:0> = 100 即可。

13.3.2 连续脉冲模式

为将模块配置为生成连续输出脉冲，用户需要执行以下步骤：

- 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。配置为 PWM 工作模式时，OcxR 是主锁存器（只读），OcxRS 是辅助锁存器。这可以使 PWM 跳变时不产生毛刺。

为将输出比较模块配置为 PWM 工作模式，用户需要执行以下步骤：

1. 通过写适当的周期寄存器设置 PWM 周期。
2. 通过写 OCxRS 寄存器设置 PWM 占空比。
3. 将输出比较模块配置为 PWM 工作模式。
4. 设置 TMRx 预分频比并使能定时器，TON（TxCON<15>）= 1。

13.4.1 PWM 的输入引脚故障保护

当控制位 OCM<2:0>（OCxCON<2:0>）= 111 时，所选输出比较通道仍然配置为 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 引脚上检测到逻辑 0，则相应的 PWM 输出引脚被置为高阻输入状态。OCFLT 位（OCxCON<4>）表明是否产生故障条件。此状态将保持到以下事件都已发生：

- 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

PWM 周期可通过写 PRx 寄存器来设定。PWM 周期可由公式 13-1 计算。

公式 13-1: PWM 周期

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

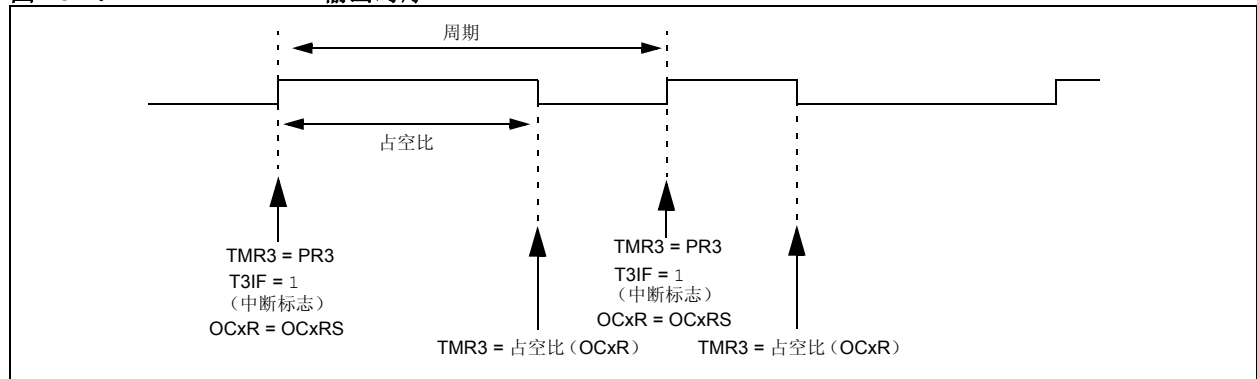
PWM 频率定义为 $1/[\text{PWM 周期}]$ 。

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以下 4 个事件：

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

关键 PWM 周期的比较，请参见图 13-1。为清晰起见，图中使用的是 Timer3。

图 13-1: PWM 输出时序



13.5 CPU 休眠模式期间的输出比较操作

当 CPU 进入休眠模式时，所有内部时钟都停止。因此，当 CPU 进入休眠状态时，输出比较通道将把引脚驱动为 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，则引脚将继续保持高电平。类似地，当 CPU 进入休眠模式时，如果引脚为低电平，则引脚将继续保持低电平。在这两种情况下，当器件唤醒时，输出比较模块都将恢复工作。

13.6 CPU 空闲模式期间的输出比较操作

当 CPU 进入空闲模式时，输出比较模块仍然能够以全功能工作。

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，则输出比较通道在 CPU 空闲模式期间工作。

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，这与所选的匹配模式无关。

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

对于 PWM 模式，当发生事件时，对应的定时器中断标志 (T2IF 或 T3IF) 置 1，并且还将产生中断 (如果允许中断的话)。IF 位位于 IFS0 状态寄存器中，并且必须用软件清零。通过相应的定时器中断允许位 (T2IE 或 T3IE) 来允许中断；该位位于 IEC0 控制寄存器中。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

图注: u = 未初始化位
寄存器中各位的说明, 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。关于 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)。

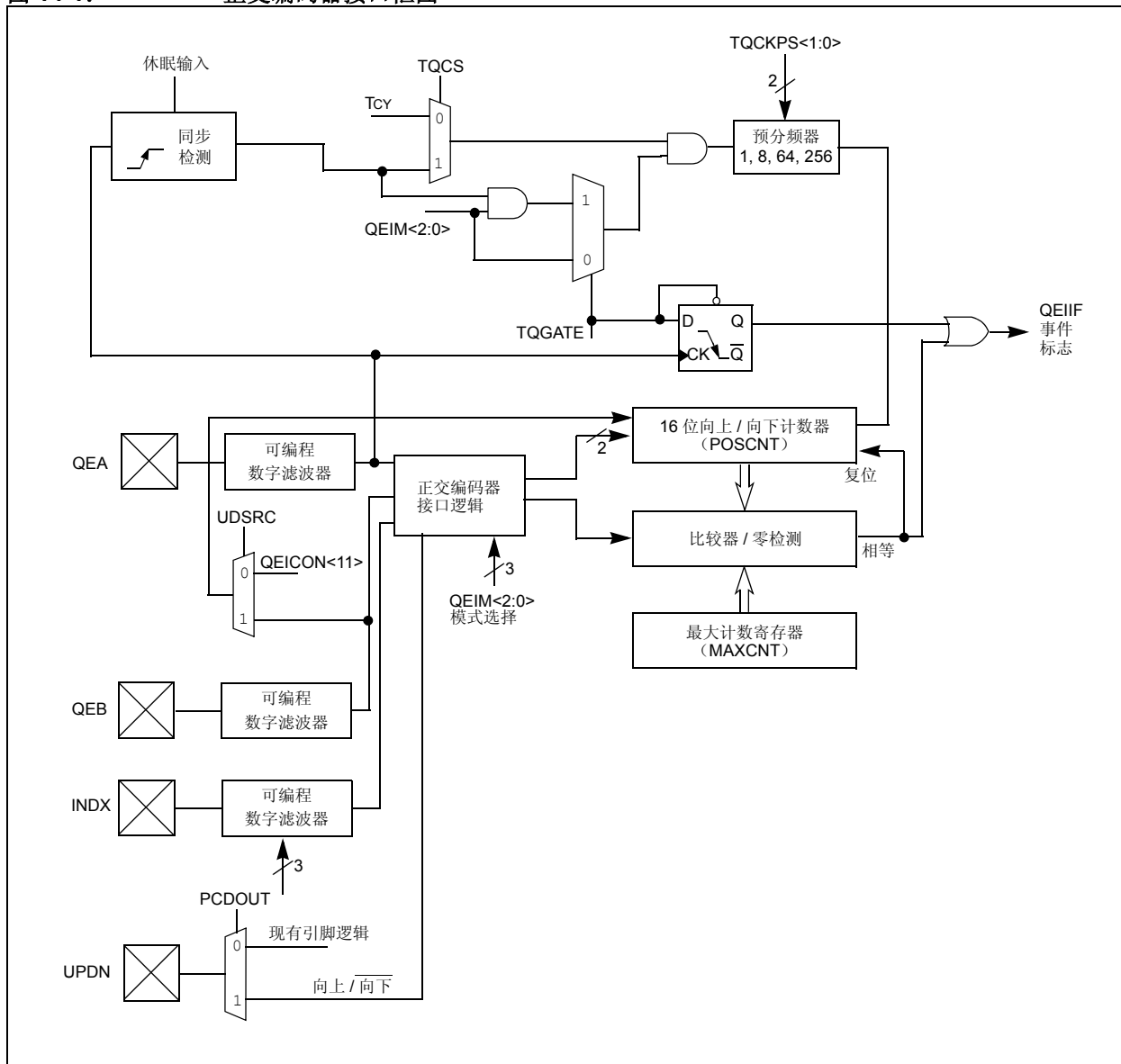
本节介绍正交编码器接口 (Quadrature Encoder Interface, QEI) 模块及其相关的工作模式。QEI 模块提供了与增量式编码器的接口，以获取机械位置数据。

QEI 模块的工作特性如下：

- 3 个输入通道，用于两个相位信号和索引脉冲
- 16 位向上 / 向下位置计数器
- 计数方向状态
- 位置测量 (x2 和 x4) 模式
- 输入端具有可编程数字噪声滤波器
- 备用 16 位定时器 / 计数器模式
- 正交编码器接口中断

通过设置适当的位 QEIM<2:0> (QEICON<10:8>) 来决定这些工作模式。图 14-1 所示为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

典型的增量式编码器（也即光电式编码器）有 3 个输出，分别是：A 相、B 相和索引脉冲。这些信号很有用，在 ACIM 和 SR 电机的位置和速度控制中经常用到。

A 相（QEA）和 B 相（QEB）这两个通道间的关系是惟一的。如果 A 相超前 B 相，则电机的旋转方向被认为是正向的。如果 A 相落后 B 相，则电机的旋转方向被认为是反向的。

第三个通道称为索引脉冲，每转一周产生一个脉冲，作为基准用来确定绝对位置。索引脉冲和 A 相、B 相信号相对应，如下所示。

14.2 16 位向上 / 向下位置计数器模式

16 位向上 / 向下计数器在每个计数脉冲向上或向下计数，该计数脉冲由 A 相和 B 相输入信号的差相产生。计数器作为积分器，它的计数值与位置成正比。计数方向由 UPDN 信号决定，此信号由正交编码器接口逻辑产生。

14.2.1 位置计数器错误检查

在 QEI 中提供了位置计数错误检查功能，由 CNTERR 位（QEICON<15>）表示。仅当位置计数器被配置为在产生索引脉冲时复位的模式（QEIM<2:0> = 110 或 100）时，错误检查才适用。在这些模式下，POSCNT 寄存器的内容与相应值（根据方向不同为 0xFFFF 或 MAXCNT + 1）进行比较。如果检测到这些值，通过将 CNTERR 位置 1 产生错误条件，并产生 QEI 计数错误中断。通过将 CEID 位（DFLTCON<8>）置 1 可以禁止 QEI 计数错误中断。检测到错误后，位置计数器将继续对编码器边沿进行计数。POSCNT 寄存器继续向上 / 向下计数，直到发生自然计满返回 / 下溢。对于自然计满返回 / 下溢事件，不会产生中断。CNTERR 位是读 / 写位，由用户用软件复位。

14.2.2 位置计数器复位

位置计数器复位使能位 POSRES（QEICON<2>）控制检测到索引脉冲时是否复位位置计数器。仅当 QEIM<2:0> = 100 或 110 时，此位才适用。

如果 POSRES 位置为 1，则位置计数器在检测到索引脉冲时复位。如果 POSRES 位置为 0，则位置计数器在检测到索引脉冲时不复位。位置计数器将继续向上或向下计数，并在计满返回或下溢情况发生时复位。

仍然在检测到索引脉冲时产生中断，而不是在位置计数器上溢 / 下溢时产生中断。

14.2.3 计数方向状态

如前面章节所述，QEI 逻辑根据 A 相与 B 相的时间关系产生 UPDN 信号。除输出引脚外，内部 UPDN 信号的状态被提供给 SFR 位 UPDN（QEICON<11>），该位作为只读位。要将信号状态送至 I/O 引脚，则必须将 SFR 位 PCDOUT（QEICON<6>）置为 1。

14.3 位置测量模式

器件支持两种测量模式，称为 x2 和 x4。这些模式通过位于 SFR QEICON<10:8> 中的 QEIM<2:0> 模式选择位进行选择。

当控制位 QEIM<2:0> = 100 或 101 时，选择 x2 测量模式，且 QEI 逻辑仅根据 A 相输入来确定位置计数器递增计数的速率。A 相信号的每个上升沿和下降沿都会使位置计数器进行递增或递减计数。B 相信号仍用于确定计数器的方向，与在 x4 模式下一样。

在 x2 测量模式下，有两种复位位置计数器的方法：

1. 检测到索引脉冲时复位位置计数器，
QEIM<2:0> = 100。
2. 与 MAXCNT 匹配时复位位置计数器，
QEIM<2:0> = 101。

当控制位 QEIM<2:0> = 110 或 111 时，选择 x4 测量模式，且 QEI 逻辑同时检测 A 相和 B 相输入信号的上升沿和下降沿。两种信号的每个边沿都会使位置计数器进行递增或递减计数。

在 x4 测量模式下，有两种复位位置计数器的方法：

1. 检测到索引脉冲时复位位置计数器，
QEIM<2:0> = 110。
2. 与 MAXCNT 匹配时复位位置计数器，
QEIM<2:0> = 111。

x4 测量模式可以为确定电机位置提供更高精度的数据（更多位置计数）。

14.4 可编程数字噪声滤波器

数字噪声滤波器部件用于滤除输入捕捉或正交信号中的噪声。施密特触发器输入和有 3 个时钟周期延时的滤波器组合使用，用以滤除低电平噪声和通常在易产生噪声的应用（如电机系统）中出现的幅度大而持续时间短的尖脉冲噪声。

该滤波器可以确保在 3 个连续的时钟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，数字滤波器的时钟分频由 QECK<2:0>（DFLTCON<6:4>）位设定，并且来自基本指令周期 Tcy。

要能使通道 QEA、QEB 和 INDX 的滤波器输出，必须将 QEOUT 位置 1。在 POR 和 BOR 时，所有通道的滤波器网络被禁止。

14.5 备用 16 位定时器 / 计数器

当 QEI 模块不是配置为 QEI 模式，即 QEIM<2:0> = 001 时，可以将模块配置为简单的 16 位定时器 / 计数器。辅助定时器的设置和控制通过 QEICON SFR 寄存器实现。此定时器的功能与 Timer1 的功能一样。QEA 引脚被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，MAXCNT 寄存器作为周期寄存器。当发生定时器 / 周期寄存器匹配时，QEI 中断标志位置 1。

此定时器和通用定时器的惟一区别在于，此定时器增加了外部向上 / 向下输入选择功能。当将 UPDN 引脚置为高电平时，定时器将递增计数；当将 UPDN 引脚置为低电平时，定时器将递减计数。

注： 改变工作模式（即从 QEI 到定时器，或从定时器到 QEI）时，不会影响定时器 / 位置计数寄存器的内容。

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

此外，控制位 UDSRC（QEICON<0>）可确定定时器计数方向状态是基于写入 UPDN 控制 / 状态位（QEICON<11>）的逻辑状态，还是基于 QEB 引脚状态。当 UDSRC = 1 时，定时器计数方向由 QEB 引脚控制；类似地，当 UDSRC = 0 时，定时器计数方向由 UPDN 位控制。

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

14.6 CPU 休眠模式期间的 QEI 模块操作

14.6.1 CPU 休眠模式期间的 QEI 操作

在 CPU 休眠模式期间，QEI 模块将暂停工作。

14.6.2 CPU 休眠模式期间的定时器操作

在 CPU 休眠模式期间，由于内部时钟被禁止，定时器不工作。

14.7 CPU 空闲模式期间的 QEI 模块操作

由于 QEI 模块可以用作正交编码器接口或 16 位定时器，以下章节将说明这两种模式下的模块操作。

14.7.1 CPU 空闲模式期间的 QEI 操作

当将 CPU 配置为空闲模式时，如果 QEISIDL 位（QEICON<13>）= 0，QEI 模块将继续工作。在执行 POR 和 BOR 时，此位默认为逻辑 0。要在 CPU 空闲模式期间暂停 QEI 模块，应将 QEISIDL 置 1。

14.7.2 CPU 空闲模式期间的定时器操作

当将 CPU 配置为空闲模式，并且将 QEI 模块配置为 16 位定时器模式时，如果 QEISIDL 位 (QEICON<13>) = 0，则 16 位定时器将继续工作。在执行 POR 和 BOR 时，此位默认为逻辑 0。要在 CPU 空闲模式期间暂停定时器模块，应将 QEISIDL 置 1。

如果 QEISIDL 位清零，定时器将像 CPU 没有进入空闲模式一样正常工作。

14.8 正交编码器接口中断

正交编码器接口可以在发生以下事件时产生中断：

- 在 16 位向上/向下位置计数器发生计满返回/下溢时产生中断
- 在检测到合格索引脉冲，或者当 CNTERR 位置 1 时
- 定时器周期匹配事件（上溢 / 下溢）
- 门控累加事件

在发生以上任何事件时，QEI 中断标志位 QEIIF 被置 1。QEIIF 位必须用软件清零。QEIIF 位于 IFS2 状态寄存器中。

通过对应的中断允许位 QEIIE 可以允许中断。QEIIE 位位于 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
DELTCON	0124	—	—	—	—	—	—	IMV<1:0>	CEID	QEOUT	QECK2	QECK1	QECK0	—	—	—	—	0000 0000 0000 0000
POSCNT	0126	位置计数器 <15:0>																0000 0000 0000 0000
MAXCNT	0128	最大计数 <15:0>																1111 1111 1111 1111

图注: u = 未初始化位
寄存器中各位的说明, 请参见 《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F5015/5016

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。关于 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 (Pulse-Width Modulated, PWM) 输出的任务。特别是 PWM 模块还能支持以下电源和电机控制应用：

- 三相交流感应电机 (AC Induction Motor)
- 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)

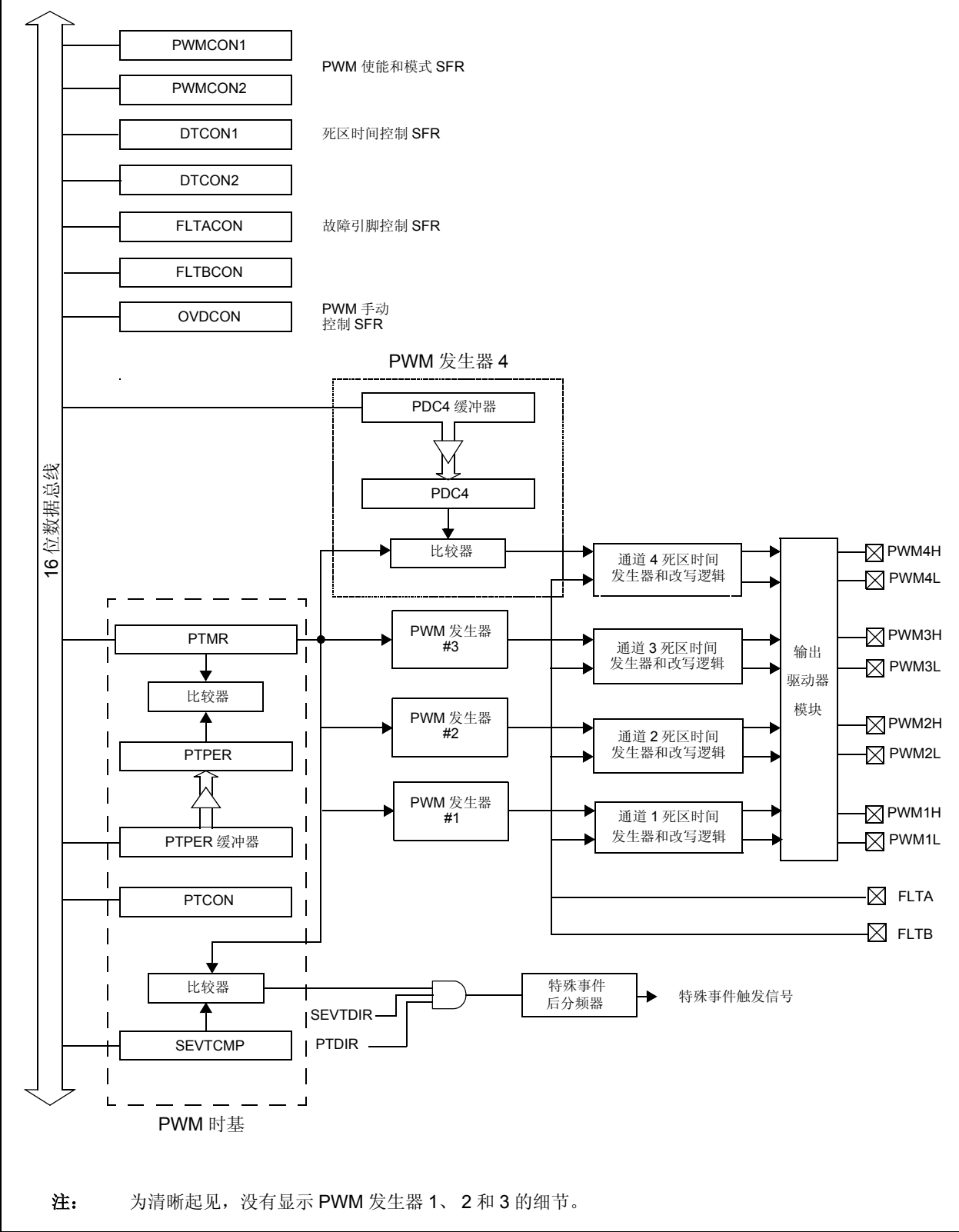
PWM 模块具有以下特性：

- 8 个 PWM I/O 引脚，具有 4 个占空比发生器
- 最高 16 位分辨率
- “运行时” PWM 频率更改
- 边沿和中心对齐输出模式
- 单脉冲生成模式
- 在中心对齐模式下，为非对称更新提供了中断支持
- 对于电子换向电机 (Electrically Commutative Motor, ECM) 操作提供了输出改写控制
- 用于安排其他外设事件的“特殊事件”比较器
- 故障引脚可选择将各 PWM 输出引脚驱动为定义的状态
- 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，编号从 1 到 4。模块具有 8 个 PWM 输出引脚，编号从 PWM1H/PWM1L 到 PWM4H/PWM4L。这 8 个 I/O 引脚分组为高 / 低编号对，分别由后缀 H 或 L 表示。对于互补性负载，低 PWM 引脚总与相应高 I/O 引脚互补。

PWM 模块允许多种工作模式，这些模式对于具体的电源控制应用很有益。

图 15-1: PWM 模块框图



15.1 PWM 时基

PWM 时基由一个带有预分频器和后分频器的 15 位定时器提供。可通过 PTMR SFR 访问该时基。PTMR<15> 为只读状态位 PTDIR，指示 PWM 时基当前的计数方向。如果 PTDIR 清零，则表示 PTMR 正在向上计数。如果 PTDIR 置 1，则表示 PTMR 正在向下计数。通过 PTCON SFR 对 PWM 时基进行配置。通过置 1/ 清零 PTCON SFR 中的 PTEN 位来使能/禁止时基。当 PTEN 位用软件清零时，PTMR 不会清零。

PTPER SFR 用于设置 PTMR 的计数周期。用户必须向 PTPER<14:0> 中写入一个 15 位的值。当 PTMR<14:0> 中的值与 PTPER<14:0> 中的值匹配时，时基将复位为 0，或者在出现下一个时钟周期时反转计数方向。所采取的操作取决于时基的工作模式。

注： 如果将周期寄存器设置为 0x0000，则定时器将停止计数，同时不会产生中断和特殊事件触发信号，即使当特殊事件值也是 0x0000 时亦是如此。如果周期寄存器已为 0x0000，则模块将不能更新周期寄存器；因此，用户必须禁止此模块以便更新周期寄存器。

可以将 PWM 时基配置为 4 种不同的工作模式：

- 自由运行模式
- 单事件模式
- 连续向上 / 向下计数模式
- 带双更新中断的连续向上 / 向下计数模式

这 4 种模式通过 PTCON SFR 中的 PTMOD<1:0> 位进行选择。向上 / 向下计数模式支持产生中心对齐的 PWM。单事件模式使 PWM 模块可以对某些电子换向电机（ECM）进行脉冲控制。

PWM 时基根据 PTCON SFR 中的模式选择位（PTMOD<1:0>）和后分频器位（PTOPS<3:0>）来产生中断。

15.1.1 自由运行模式

在自由运行模式下，PWM 时基将向上计数直到与时基周期寄存器（PTPER）中的值发生匹配。PTMR 寄存器在接下来的输入时钟边沿复位，只要 PTEN 位保持置 1，时基仍将继续向上计数。

当 PWM 时基处于自由运行模式（PTMOD<1:0> = 00）时，在 PTMR 寄存器与 PTPER 寄存器匹配时，产生中断事件，PTMR 寄存器被复位为零。在此定时器模式下，可以使用后分频器选择位以减小中断事件发生的频率。

15.1.2 单事件模式

在单事件计数模式下，PWM 时基在 PTEN 位置 1 时将开始向上计数。当 PTMR 寄存器中的值与 PTPER 寄存器中的值匹配时，PTMR 寄存器在接下来的输入时钟边沿复位，并且 PTEN 位将由硬件清零以停止时基。

当 PWM 时基处于单事件模式（PTMOD<1:0> = 01）时，在 PTMR 寄存器与 PTPER 寄存器匹配时，产生中断事件，PTMR 寄存器在接下来的输入时钟边沿被复位为零，并且 PTEN 位被清零。后分频器选择位对此定时器模式没有影响。

15.1.3 连续向上 / 向下计数模式

在连续向上 / 向下计数模式下，PWM 时基将向上计数直到与 PTPER 寄存器中的值发生匹配。定时器将在接下来的输入时钟边沿开始向下计数。PTCON SFR 中的 PTDIR 位是只读位，指示计数方向。当定时器向下计数时，PTDIR 位置 1。

在向上 / 向下计数模式（PTMOD<1:0> = 10）下，每当 PTMR 寄存器的值变为零时都会产生中断事件，这时 PWM 时基开始向上计数。在此定时器模式下，可以使用后分频器选择位以减小中断事件发生的频率。

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，以及每当发生周期匹配时都产生中断。后分频器选择位对此定时器模式没有影响。

双更新模式为用户提供了两种额外的功能。首先，由于 PWM 占空比在每个周期可更新两次，因此控制循环带宽加倍。其次，可以生成非对称中心对齐的 PWM 波形，这对于在某些电机控制应用中最小化输出波形的失真很有用。

注： 对周期寄存器编程为值 0x0001 会生成连续的中断脉冲，因此应避免这样操作。

15.1.5 PWM 时基预分频器

PTMR（Fosc/4）的输入时钟具有以下预分频选项：1:1、1:4、1:16 或 1:64，通过 PTCON SFR 中的控制位 PTCKPS<1:0> 选择。当出现以下事件之一时，预分频计数器将清零：

- 对 PTMR 寄存器进行写操作
- 对 PTCON 寄存器进行写操作
- 任何器件复位

写 PTCON 时 PTMR 不会清零。

15.1.6 PWM 时基后分频器

用户可以选择通过一个 4 位后分频器（分频比从 1:1 到 1:16）对 PTMR 的匹配输出进行后分频。

当出现以下事件之一时，后分频计数器将清零：

- 对 PTMR 寄存器进行写操作
- 对 PTCON 寄存器进行写操作
- 任何器件复位

写 PTCON 时 PTMR 寄存器不会清零。

15.2 PWM 周期

PTPER 是一个 15 位寄存器，用于设置 PWM 时基的计数周期。PTPER 是双缓冲寄存器。PTPER 缓冲器的内容在发生以下情形时被装入 PTMR 寄存器：

- 自由运行和单事件模式：当 PTMR 寄存器在与 PTPER 寄存器匹配后复位为零时。
- 向上 / 向下计数模式：当 PTMR 寄存器为零时。

当 PWM 时基被禁止（TEN = 0）时，PTPER 缓冲器中保存的值被自动装入 PTMR 寄存器。

PWM 周期可通过公式 15-1 确定：

公式 15-1: PWM 周期

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

如果 PWM 时基被配置为向上 / 向下计数模式之一，PWM 周期可通过公式 15-2 得出。

公式 15-2: PWM 周期 (向上 / 向下模式)

$$T_{PWM} = \frac{2 \cdot T_{CY} \cdot (PTPER + 1)}{(\text{PTMR 预分频值})}$$

给定器件振荡器的最大分辨率（以位为单位）和 PWM 频率可通过公式 15-3 确定：

公式 15-3: PWM 分辨率

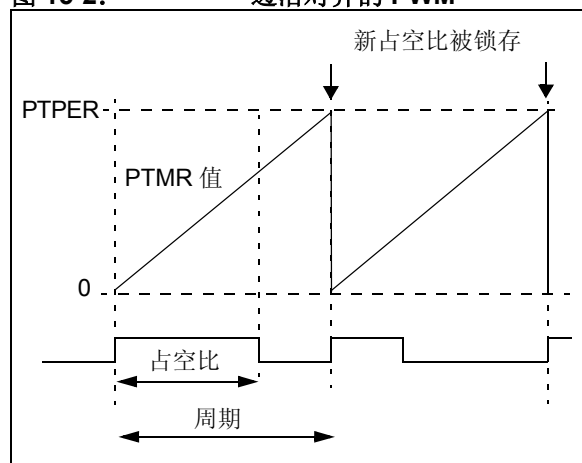
$$\text{分辨率} = \frac{\log(2 \cdot T_{PWM} / T_{CY})}{\log(2)}$$

15.3 边沿对齐的 PWM

当 PWM 时基处于自由运行或单事件模式时，模块产生边沿对齐的 PWM 信号。对于边沿对齐的 PWM 输出，输出的周期由 PTPER 中的值指定，占空比由相应的占空比寄存器指定（见图 15-2）。PWM 输出在周期开始（PTMR = 0）时被驱动为有效；当占空比寄存器中的值与 PTMR 发生匹配时，PWM 输出被驱动为无效。

如果某个占空比寄存器中的值为零，则相应 PWM 引脚的输出在整个 PWM 周期内都将为无效。此外，如果占空比寄存器中的值大于 PRPER 寄存器中保存的值，则 PWM 引脚的输出在整个 PWM 周期内都将有效。

图 15-2: 边沿对齐的 PWM



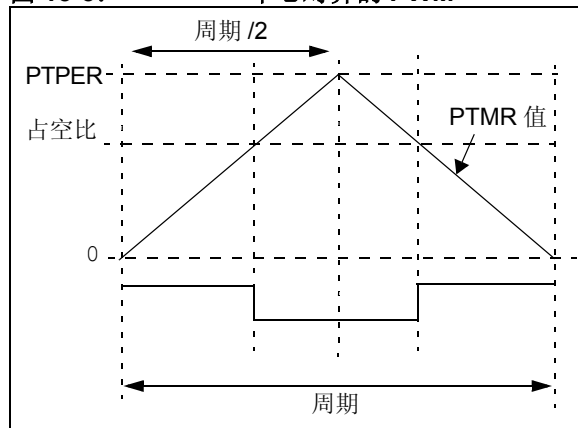
15.4 中心对齐的 PWM

当 PWM 时基配置为向上 / 向下计数模式时，模块将产生中心对齐的 PWM 信号（见图 15-3）。

当占空比寄存器的值与 PTMR 的值匹配，并且 PWM 时基正在向下计数（PTDIR = 1）时，PWM 比较输出被驱动为有效状态。当 PWM 时基正在向上计数（PTDIR = 0），并且 PTMR 寄存器中的值与占空比值匹配时，PWM 比较输出被驱动为无效状态。

如果某个占空比寄存器中的值为零，则相应 PWM 引脚的输出在整个 PWM 周期内都将为无效。此外，如果占空比寄存器中的值等于 PRPER 寄存器中保存的值，则 PWM 引脚的输出在整个 PWM 周期内都将有效。

图 15-3: 中心对齐的 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）用于指定 PWM 模块的占空比值。

每个占空比寄存器中的值确定 PWM 输出处于有效状态的时间量。占空比寄存器为 16 位宽。占空比寄存器的 LSB 确定是否在开始出现 PWM 边沿。因此，PWM 实际上具有双倍分辨率。

15.5.1 占空比寄存器缓冲

这 4 个 PWM 占空比寄存器都是双倍缓冲的，以使 PWM 输出更新时无毛刺。对于每个占空比，都有可由用户访问的占空比寄存器和保存当前 PWM 周期中所使用的实际比较值的占空比寄存器。

对于边沿对齐的 PWM 输出，只要 PTMR 与 PTPER 寄存器发生了匹配，新的占空比值就会更新，同时 PTMR 复位。当 PWM 时基被禁止（PTEN = 0）且 PWMCON2 中的 UDIS 位被清零时，占空比缓冲器的内容会被自动装入占空比寄存器。

当 PWM 时基处于向上 / 向下计数模式时，当 PTMR 寄存器的值为零且 PWM 时基开始向上计数时，新的占空比值被更新。当 PWM 时基被禁止（TEN = 0）时，占空比缓冲器的内容会被自动装入占空比寄存器。

当 PWM 时基处于带双更新的向上 / 向下计数模式时，当 PTMR 寄存器的值为零且 PTMR 寄存器的值与 PTPER 寄存器中的值匹配时，新的占空比值被更新。当 PWM 时基被禁止（TEN = 0）时，占空比缓冲器的内容会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，任何向占空比寄存器的写操作均将立即更新为新的占空比值。此功能为用户提供了一种选择，使其可立即更新当前有效的 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，系统稳定性将由于系统观察到发出系统校正命令之间时间的缩短而提高。

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值，则 PWM 输出将立即变为有效，并对新写入的占空比值保持有效。

15.6 互补 PWM 操作

在互补工作模式下，每对 PWM 输出都构成一对互补 PWM 信号。在器件切换过程中，有一段两个输出均无效的短暂时期，此时可以选择插入一个死区时间（见第 15.7 节“死区时间发生器”）。

在互补模式下，占空比较单元被分配给 PWM 输出使用，如下所示：

- 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

通过将 PWMCON1 SFR 中相应的 PMODx 位清零，可以将相应的一对 PWM I/O 引脚选择为互补模式。在器件复位时，PWM I/O 引脚默认设置为互补模式。

15.7 死区时间发生器

当任何一对 PWM I/O 引脚工作在互补输出模式时，均可使用死区时间生成功能。PWM 输出采用推挽器件电路。由于电源输出设备不可能瞬时完成切换，所以必须在关闭一对互补的 PWM 输出中的一个和开启另一个晶体管之间提供一定的时间。

PWM 模块允许编程两个不同的死区时间。这两个死区时间可以用以下两种方法之一来提高用户灵活性：

- 可以对 PWM 输出信号进行优化使一对互补晶体管中的高边和低边晶体管的关断时间不同。在互补对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。
- 两个死区时间可以单独分配给一对 PWM I/O 引脚。此工作模式使 PWM 模块可以对每一对互补的 PWM I/O 引脚驱动不同的晶体管 / 负载组合。

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 位的向下计数器，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 包含几个控制位，用于将死区时间分配给各个互补输出。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 PWM1L/PWM1H 有效边沿死区时间。
DTS1I	选择 PWM1L/PWM1H 无效边沿死区时间。
DTS2A	选择 PWM2L/PWM2H 有效边沿死区时间。
DTS2I	选择 PWM2L/PWM2H 无效边沿死区时间。
DTS3A	选择 PWM3L/PWM3H 有效边沿死区时间。
DTS3I	选择 PWM3L/PWM3H 无效边沿死区时间。
DTS4A	选择 PWM4L/PWM4H 有效边沿死区时间。
DTS4I	选择 PWM4L/PWM4H 无效边沿死区时间。

15.7.3 死区时间范围

死区时间是由通过指定输入时钟预分频值和 6 位无符号值选择的每个死区时间单元提供的。可以单独设置每个死区时间单元所提供的死区时间量。

通过所提供的 4 个输入时钟预分频器选项，用户可以根据器件的工作频率选择适当的死区时间范围。可以单独为两个死区时间值中分别选择时钟预分频器选项。死区时间的时钟预分频值是使用 DTCON1 SFR 中的 DTAPS<1:0> 和 DTBPS<1:0> 控制位选择的。每个死区时间值均可选择 4 个时钟预分频器选项之一（Tcy、2 Tcy、4 Tcy 或 8 Tcy）。

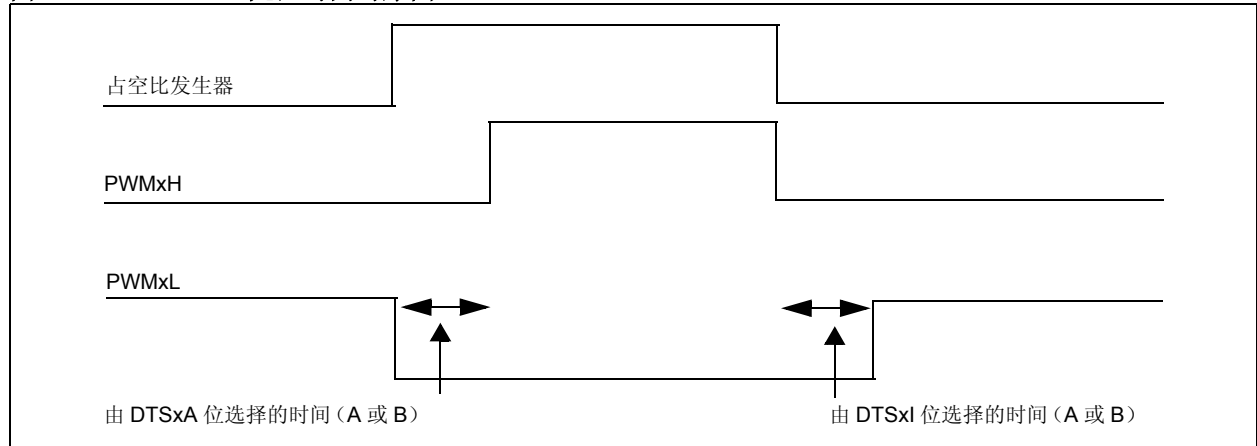
选择预分频值后，通过向 DTCON1 SFR 中装入两个 6 位无符号值，可以调整每个单元的死区时间。

死区时间单元预分频器在发生以下事件时清零：

- 由于发生占空比比较边沿事件而装入向下定时器时。
- 写 DTCON1 或 DTCON2 寄存器时。
- 任何器件复位时。

注： 当 PWM 模块正在工作（PTEN = 1）时，用户不应修改 DTCON1 或 DTCON2 的值。因为这样做可能造成意外结果。

图 15-4: 死区时间时序图



15.8 独立 PWM 输出

在驱动某些类型的负载时需要使用独立 PWM 输出模式。当 PWMCON1 寄存器中的相应 PMOD 位置 1 时，某一对 PWM 输出就处于独立输出模式。当模块工作于独立模式时，相邻 PWM I/O 引脚之间不存在死区控制，允许两个 I/O 引脚同时有效。

在独立模式下，每个占空比发生器同时连接到输出对的两个 PWM I/O 引脚。通过使用相关的占空比寄存器和 OVDCON 寄存器中的相应位，用户可以为每个工作于独立模式的 PWM I/O 引脚选择以下信号输出选项：

- I/O 引脚输出 PWM 信号
- I/O 引脚无效
- I/O 引脚有效

15.9 单脉冲 PWM 操作

当 PTCN 控制位 PTMOD<1:0> = 10 时，PWM 模块产生单脉冲输出。在单脉冲模式下，只能产生边沿对齐的输出。在单脉冲模式下，当 PTEN 位置 1 时，PWM I/O 引脚被驱动为有效状态。当与占空比寄存器发生匹配时，PWM I/O 引脚被驱动为无效状态。当与 PTPER 寄存器发生匹配时，PTMR 寄存器清零，所有的有效 PWM I/O 引脚均被驱动为无效状态，PTEN 位清零，并产生中断。

15.10 PWM 输出改写

PWM 输出改写位可以让用户手动将 PWM I/O 引脚驱动为指定逻辑状态，而不受占空比比较单元的影响。

所有与 PWM 输出改写功能相关的控制位都包含在 OVDCON 寄存器中。OVDCON 寄存器的高半部分包含 8 个位（POVDxH<4:1> 和 POVDxL<4:1>）这些位确定改写哪些 PWM I/O 引脚。OVDCON 寄存器的低半部分也包含 8 个位（POUTxH<4:1> 和 POUTxL<4:1>），当通过 POVD 位改写某个特定输出时，可以使用这些位确定 PWM I/O 引脚的状态。

15.10.1 互补输出模式

当通过 OVDCON 寄存器将 PWMxL 引脚驱动为有效时，输出信号被强制为与输出对中 PWMxH 引脚互补的信号。当手动改写 PWM 通道时，仍然执行死区时间插入。

15.10.2 改写同步

如果 PWMCON2 寄存器中的 OSYNC 位置 1，则所有通过 OVDCON 寄存器执行的输出改写将与 PWM 时基同步。同步输出改写发生在以下时间：

- 边沿对齐模式，当 PTMR 为零时。
- 中心对齐模式，当 PTMR 为零时，以及当 PTMR 与 PTPER 的值匹配时。

15.11 PWM 输出和极性控制

有 3 个与 PWM 模块相关的器件配置位提供 PWM 输出引脚控制，如下：

- HPOL 配置位
- LPOL 配置位
- PWMPIN 配置位

FPORBOR 配置寄存器（见第 20.0 节“系统集成”）中的这 3 个位与位于 PWMCON1 SFR 中的 4 个 PWM 使能位（PWMEN<4:1>）配合工作。配置位和 PWM 使能位确保在发生器件复位后，PWM 引脚处于正确的状态。通过 PWMPIN 配置熔丝，可以选择在器件复位时使能 PWM 模块输出。如果 PWMPIN = 0，PWM 输出在复位时被驱动为无效状态。如果 PWMPIN = 1（默认），PWM 输出将为三态。HPOL 位指定 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 控制位分别用于使能每个高边 PWM 输出引脚和每个低边 PWM 输出引脚。如果没有使能某个 PWM 输出引脚，则将其视为通用 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）。当使能时，可选择通过这些引脚将各 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 SFR 各有 4 个控制位，用来决定某对 PWM I/O 引脚是否要由故障输入引脚控制。要将某一对 PWM I/O 引脚使能为故障改写，需要将 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对 PWM 模块没有影响，可以将引脚用作通用中断或 I/O 引脚。

注：故障引脚逻辑可以独立于 PWM 逻辑工作。如果将 FLTACON/FLTBCON 寄存器中所有的使能位都清零，则可以将故障引脚用作通用中断引脚。各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 特殊功能寄存器各有 8 个位，用来决定各 PWM I/O 引脚在被故障输入改写时的状态。当这些位清零时，PWM I/O 引脚被驱动为无效状态。当相应位置 1 时，PWM I/O 引脚被驱动为有效状态。有效和无效状态与各 PWM I/O 引脚被定义的极性（通过 HPOL 和 LPOL 极性控制位设置）相对应。

当 PWM 模块的一对 I/O 处于互补模式，并且两个引脚都编程为在产生故障条件时驱动为有效时，存在一种特殊情况。在互补模式下，PWMxH 引脚始终优先，因此两个 I/O 引脚不能同时被驱动为有效。

15.12.3 故障引脚优先级

如果两个故障输入引脚都被分配为控制某一对 PWM I/O 引脚，则为故障 A 输入引脚编程的故障状态将优先于故障 B 输入引脚。

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用软件清零。当这两项操作都发生后，PWM 输出将在下一个 PWM 周期开始时或半周期边界返回到正常工作状态。如果中断标志在故障条件结束前清零，则 PWM 模块将等到故障引脚不再有效时才恢复输出。
- **逐个周期模式：**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 输出就会一直保持定义的故障状态。在故障引脚驱动为高电平后，PWM 输出将在下一个 PWM 周期开始时或半周期边界返回到正常工作状态。

各故障输入引脚的工作模式通过 FLTACON 和 FLTBCON 特殊功能寄存器中的 FLTAM 和 FLTBM 控制位选择。

可以用软件对各故障引脚进行手动控制。

15.13 PWM 更新锁定

对于复杂的 PWM 应用，用户可能需要在给定时间对最多 4 个占空比寄存器及时基周期寄存器 PTPER 进行写操作。在某些应用中，必须在新占空比和周期值被装入以供模块使用前写入所有的缓冲寄存器，这是很重要的。

通过在 PWMCON2 SFR 中将 UDIS 控制位置 1 并将 IUE 控制位清零，可以使能 PWM 更新锁定功能。UDIS 位会影响所有的占空比缓冲寄存器和 PWM 时基周期缓冲器 TPER。当 UDIS = 1 时，占空比更改或周期值更改都不起作用。

如果将 IUE 位置 1，则无论 UDI 位的状态如何，对占空比寄存器作出的任何更改均将立即得到更新。PWM 周期寄存器更新（PTPER）不受 IUE 控制位的影响。

15.14 PWM 特殊事件触发器

PWM 模块有一个特殊事件触发器，可以使 A/D 转换与 PWM 时基同步。可以将 A/D 采样和转换时间编程为在 PWM 周期内的任何时间发生。特殊事件触发器可以使用户将采集 A/D 转换结果的时间与占空比值更新的时间之间的延迟减到最小。

PWM 特殊事件触发器有一个 SFR（SEVTVMP）和 5 个控制位控制相应的操作。用于产生特殊事件触发信号的 PTMR 值装入 SEVTCMP 寄存器中。当 PWM 时基处于向上 / 向下计数模式时，还需要一个控制位指定特殊事件触发信号的计数方向。此计数方向通过 SEVTCMP SFR 中的 SEVTDIR 控制位进行选择。如果 SEVTDIR 位清零，则特殊事件触发信号将在 PWM 时基的向上计数周期产生。如果 SEVTDIR 位置 1，则特殊事件触发信号将在 PWM 时基的向下计数周期产生。如果 PWM 时基不配置为向上 / 向下计数模式，则 SEVTDIR 控制位不起作用。

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后分频器，后分频比从 1:1 到 1:16。通过写 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以下事件时清零：

- 对 SEVTCMP 寄存器的任何写操作
- 任何器件复位

15.15 CPU 休眠模式期间的 PWM 操作

故障 A 和故障 B 输入引脚能够将 CPU 从休眠模式唤醒。在休眠模式下，如果任一故障引脚被驱动为低电平，则 PWM 模块将产生中断。

15.16 CPU 空闲模式期间的 PWM 操作

PTCON SFR 包含 PTSIDL 控制位。此位用于确定当器件进入空闲模式时 PWM 模块是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 保持处于空闲模式，模块就停止工作。

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>				PTMOD<1:0>			0000 0000 0000 0000
PTMR	01C2	PTDIR	—	—	—	—	—	—	—	—	PWM 定时计数器值							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	—	—	—	—	—	—	—	—	—	PEN3H	PEN2H	PEN1H	PEN4L	PEN3L	PEN2L	PEN1L	0000 0000 1111 1111
PWMCON2	01CA	—	—	—	—	—	—	—	—	—	—	—	—	—	IUE	OSYNC	UDIS	0000 0000 0000 0000
DTCON1	01CC	DTBPS<1:0>	死区延时 B 值				SEVOPS<3:0>				DTAPS<1:0>				死区延时 A 值			0000 0000 0000 0000
DTCON2	01CE	—	—	—	—	—	—	—	—	—	DTS4A	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 = 未初始化位。

寄存器中各位的说明, 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。关于 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)。

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

16.1 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, 其中移位寄存器用来移入和移出数据。一个控制寄存器 SPIxCON 用来配置模块。此外, 一个状态寄存器 SPIxSTAT 用来指示各种状态条件。

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

在主模式下工作时, SCK 是时钟输出, 但在从模式下它是时钟输入。

一组 8 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, 模块将置 1 SPIROV 位, 表明产生了溢出条件。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 SCL 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：如果模块配置为仅执行发送, 则用户必须读 SPIxBUF, 以避免接收到溢出条件 (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。如果已经向缓冲器寄存器写了任何发送数据, 发送缓冲器的内容将送至 SPIxSR。因此, 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。只要将值写入 SPIxBUF, 便立即发送数据。在最后一个位传送的过程中产生中断。

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 输出被禁止。

提供给模块的时钟为 $F_{osc}/4$ 。然后, 将通过主预分频因子 (PPRE<1:0>) 和辅助预分频因子 (SPRE<2:0>) 对此时钟进行预分频。CKE 位决定发送的时钟边沿, 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 位操作除了发送的位数是 16 而不是 8 以外, 与 8 位操作功能相同。

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 位操作之间的基本区别是: 对于 8 位操作, 数据从 SPIxSR 的 bit 7 发送; 对于 16 位操作, 数据从 SPIxSR 的 bit 15 发送。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。这样可以将 SPI 模块配置为仅输入模式。SDO 也可用于通用 I/O。

图 16-1: SPI 框图

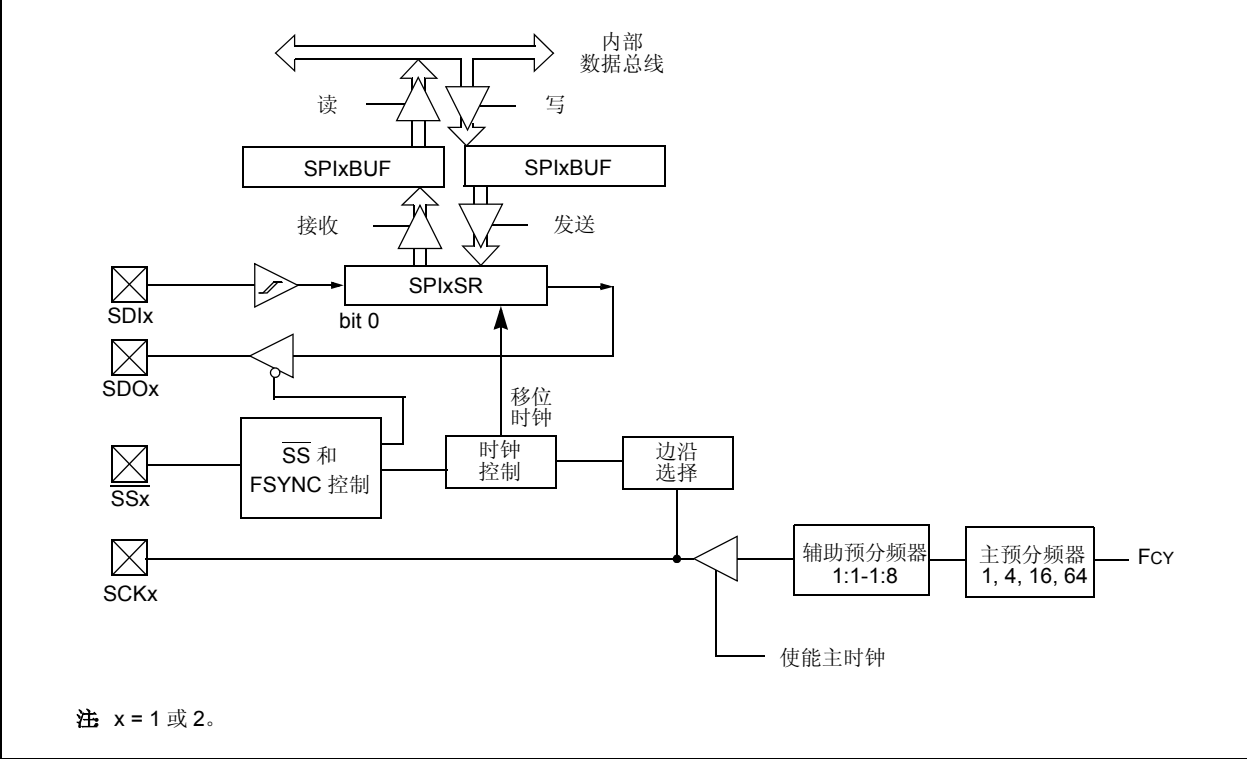
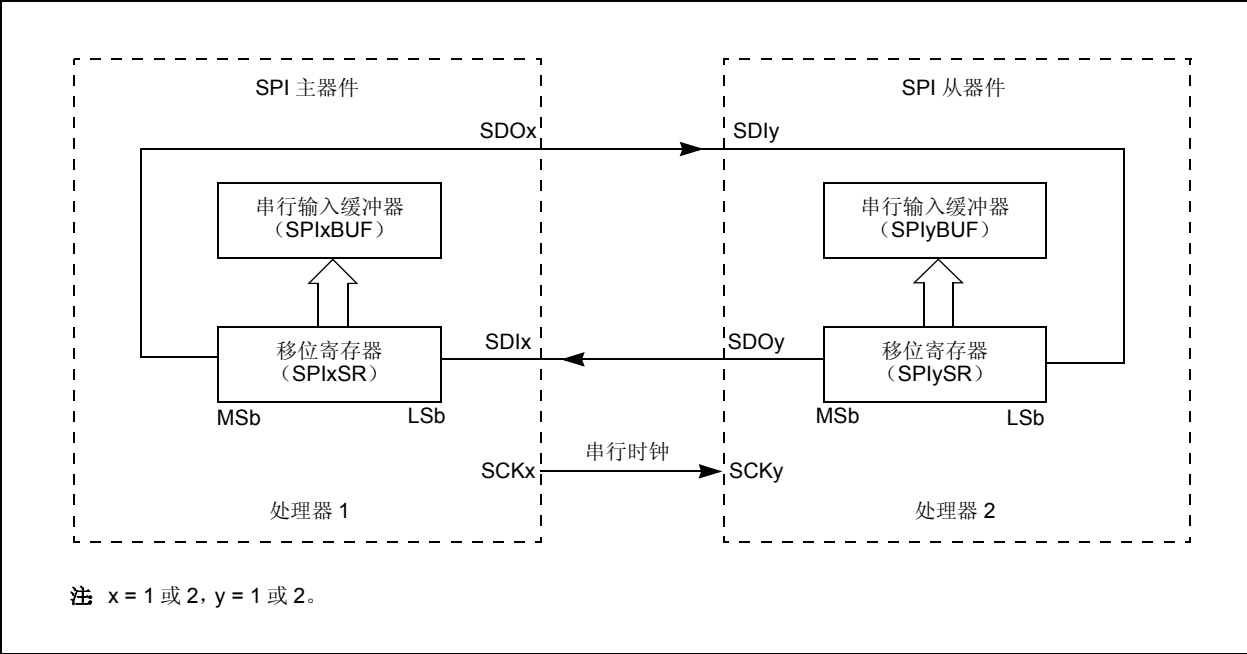


图 16-2: SPI 主 / 从器件连接



16.2 帧 SPI 支持

在主模式或从模式下，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** 引脚是输入还是输出（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 时钟的下一个后续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。必须使能 **SSx** 引脚控制（**SSEN** = 1）将 SPI 配置为 SPI 从模式。当 **SSx** 引脚为低电平时，使能发送和接收且驱动 **SDOx** 引脚。当 **SSx** 引脚变为高电平时，将不再驱动 **SDOx** 引脚。此外，可以重新同步 SPI 模块，所有计数器 / 控制电路均复位。因此，当再次将 **SSx** 置为低电平时，发送 / 接收将从 **MSb** 开始，即使 **SSx** 是在发送 / 接收过程中变为高电平。

16.4 CPU 休眠模式期间的 SPI 操作

在休眠模式期间，SPI 模块关闭。SPI 正在进行处理时，如果 CPU 进入休眠模式，则发送和接收将中止。

在休眠模式下，发送器和接收器将停止。但是，进入或退出休眠模式不会影响寄存器内容。

16.5 CPU 空闲模式期间的 SPI 操作

当器件进入空闲模式时，所有时钟源继续工作。**SPISIDL** 位（**SPIxSTAT<13>**）选择 SPI 模块在空闲模式下是停止工作还是继续工作。如果 **SPISIDL** = 0，则在 CPU 进入空闲模式时，模块将继续工作。如果 **SPISIDL** = 1，则在 CPU 进入空闲模式时，模块将停止工作。

表 16-1: SPI 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 = 未初始化位
注: 寄存器中各位的说明, 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	—	—	—	—	SPI2BF	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 = 未初始化位
注: 寄存器中各位的说明, 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。关于 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（16 位接口）为 I²C 串行通信标准的从模式和多主模式提供完全的硬件支持。

此模块具有以下主要特性：

- I²C 接口支持主 / 从模式操作。
- 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 支持多主器件操作；检测总线冲突并相应地进行仲裁。

17.1 功能说明

硬件完全实现了 I²C 标准和快速模式规范的所有主从功能，以及 7 位和 10 位寻址。

从而，在 I²C 总线上，I²C 模块既可作为主器件工作，也可作为从器件工作。

17.1.1 各种 I²C 模式

支持 I²C 模块工作在以下模式：

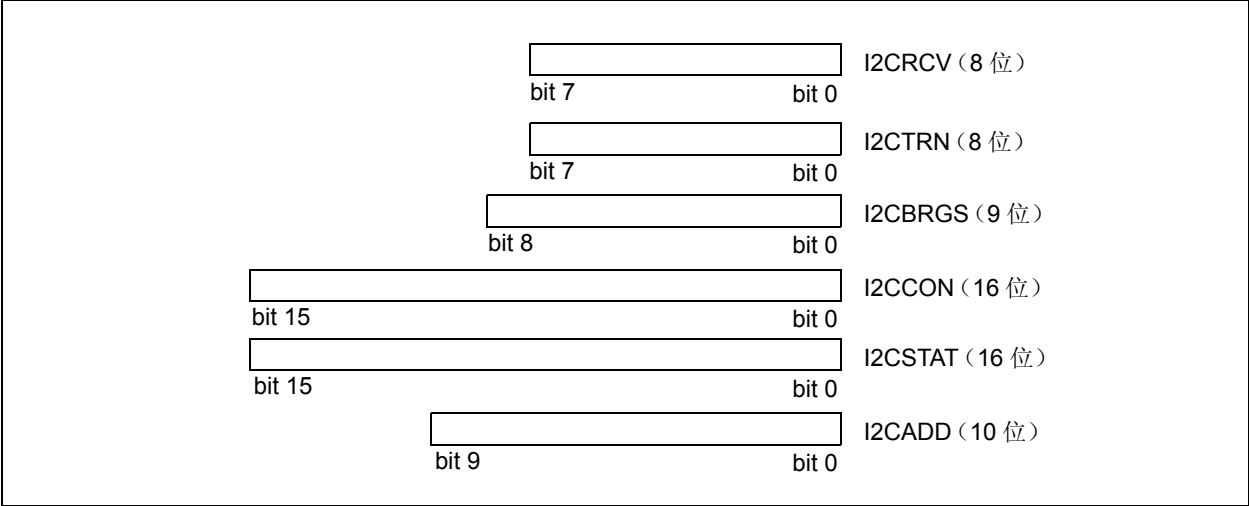
- 带 7 位地址的 I²C 从模式
- 带 10 位地址的 I²C 从模式
- 带 7 位或 10 位地址的 I²C 主模式

I²C 编程模型，请参见图 17-1。

17.1.2 I²C 模式的引脚配置

I²C 有一个双引脚接口；SCL 引脚是时钟线，SDA 引脚是数据线。

图 17-1: 编程模型



17.1.3 I²C 寄存器

I2CCON 和 I2CSTAT 分别为控制和状态寄存器。I2CCON 寄存器是可读写的。I2CSTAT 的低 6 位是只读的，它的其余位是可读写的。

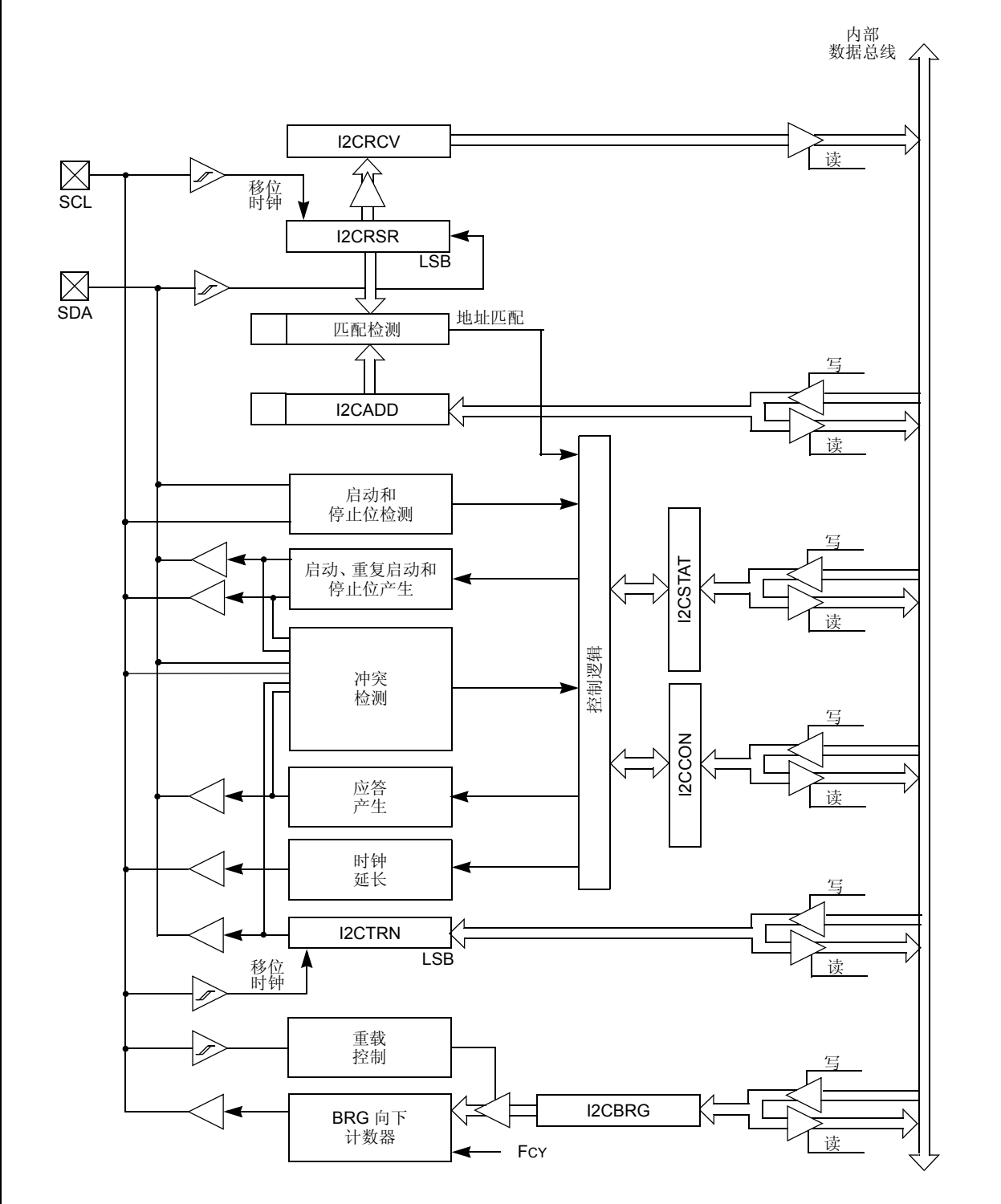
I2CRSR 是用来移动数据的移位寄存器，而 I2CRCV 是读写数据字节的缓冲寄存器。I2CRCV 是接收缓冲器，如图 17-2 所示。I2CTRN 是发送寄存器，发送操作期间向此寄存器写入字节，如图 17-2 所示。

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（Baud Rate Generator, BRG）的重载值。

在进行接收时，I2CRSR 和 I2CRCV 一起形成双缓冲接收器。当 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，在重复启动条件之后，用户只需匹配前 7 位地址。

图 17-2: I²C™ 框图



17.2 I²C 模块地址

I2CADD 寄存器包含从模式地址。这是一个 10 位寄存器。

如果 A10M 位 (I2CCON<10>) 为 0，则模块将地址解释为 7 位地址。当接收到地址时，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，则将地址视为 10 位地址。当接收到地址时，将收到的地址与二进制值“11110A9A8”（其中 A9 和 A8 是 I2CADD 的最高两位）进行比较。如果值匹配，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 模式主代码
0x08-0x77	有效 7 位地址
0x78-0x7b	有效 10 位地址（低 7 位）
0x7c-0x7f	保留

17.3 I²C 7 位从模式操作

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，8 个位将移入 I2CRSR，然后地址将与 I2CADD 进行比较。在 7 位模式 (A10M = 0) 下，位 I2CADD<6:0> 与 I2CRSR<7:1> 进行比较，而 I2CRSR<0> 是 R_W 位。所有进入的位都是在 SCL 的上升沿采样的。

如果地址匹配，将发送一个应答，并且在第 9 个位 (ACK) 的下降沿将从事件中中断标志 (SI2CIF) 置 1。地址匹配不会影响 I2CRCV 缓冲器的内容或 RBF 位。

17.3.1 从模式发送

如果接收到的 R_W 位为 1，串行端口将进入发送模式。模块将在第 9 个位发送 ACK，并使 SCL 保持为 0，直到 CPU 写 I2CTRN 进行响应为止。通过将 SCLREL 位置 1 释放 SCL，而后数据的 8 个位被移出。数据位在 SCL 的下降沿移出，因此 SCL 为高电平时 SDA 有效（见时序图）。将在第 9 个时钟脉冲的下降沿发出中断脉冲，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 位为 0，就将启动接收模式。在 SCL 上升沿采样进入的位。接收到 8 个位之后，如果 I2CRCV 未滿或 I2COV 没有置 1，I2CRSR 中的数据将被送至 I2CRCV。在第 9 个时钟发送 ACK。

如果 RBF 标志置 1，表明 I2CRCV 仍然存有前一次接收到的数据 (RBF = 1)，则不发送 ACK；但是，仍将产生中断脉冲。如果发生溢出，I2CRSR 中的内容不会被装入 I2CRCV。

注： 如果 I2COV 位为 1 且 RBF 标志为 0，将装入 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，用户不会清零 I2COV 位。不发送应答 (ACK = 1)，但 I2CRCV 会被更新。

17.4 I²C 10 位从模式操作

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

I²C 规范要求，对于写操作，必须在一个启动位后跟两个地址字节来寻址从器件。

A10M 控制位用来表明 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 位报文而言，是完全相同的，但比较的位不同。

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（默认 10 位地址）进行比较，而 I2CRSR <2:1> 与 I2CADD <9:8> 进行比较。如果发生匹配且 R_W = 0，则发出中断脉冲。ADD10 位被清零，表明地址部分匹配。如果匹配失败或 R_W = 1，则 ADD10 位被清零，模块返回到空闲状态。

然后，接收地址的低字节，并与 I2CADD <7:0> 进行比较。如果发生地址匹配，则产生中断脉冲并将 ADD10 位置 1，表明 10 位地址完全匹配。如果地址不匹配，ADD10 位清零，模块返回到空闲状态。

17.4.1 10 位模式从发送

一旦以上述方式使用完整的 10 位地址寻址从器件（此状态称为“PRIOR_ADDR_MATCH”），主器件就可以开始发送数据字节，从器件接收。

17.4.2 10 位模式从接收

一旦寻址到从器件，主器件就可以产生重启启动位，复位地址的高字节，并将 R_W 位置 1，但不产生停止位，从而开始从器件发送操作。

17.5 自动时钟延长

在从模式下，模块可以通过时钟延长同步读缓冲器和写主器件。

17.5.1 发送时钟延长

在 10 位和 7 位发送模式下，如果 TBF 位清零（表明缓冲器为空），则在第 9 个时钟的下降沿后将 SCLREL 位清零，就实现了时钟延长。

在从发送模式下，始终进行时钟延长，与 STREN 位无关。

在发送序列的第 9 个时钟之后，发生时钟同步。如果器件在第 9 个时钟的下降沿采样到 ACK，并且 TBF 位仍然清零的话，则 SCLREL 位自动清零。SCLREL 的清零将使 SCL 线置为低电平。在允许继续发送之前，用户中断服务程序必须将 SCLREL 位置 1。通过使 SCL 线保持为低电平，在主器件启动另一个发送序列之前，用户就有时间执行处理中断服务程序并装入 I2CTRN 的内容。

- 注 1:** 如果用户装入 I2CTRN 的内容，在第 9 个时钟下降沿之前将 TBF 位置 1，则 SCLREL 位将不会被清零，也不会发生时钟延长。
- 2:** 不管 TBF 位的状态如何，SCLREL 位都可以用软件置 1。

17.5.2 接收时钟延长

在从接收模式下，I2CCON 寄存器中的 STREN 位可用来使能时钟延长。当 STREN 位置 1 时，SCL 引脚将在每个数据接收序列结束保持为低电平。

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，SCL 线在缓冲寄存器已满时保持低电平。对于 7 位和 10 位寻址模式，延长 SCL 输出的方法相同。

时钟延长将在接收序列的第 9 个时钟后进行。在 ACK 序列末尾的第 9 个时钟的下降沿，如果 RBF 位置 1，SCLREL 位将自动清零，这将强制 SCL 输出保持为低电平。在允许继续接收之前，用户中断服务程序必须将 SCLREL 位置 1。保持 SCL 线为低电平，用户可以在主器件启动另一个接收序列之前，有时间处理中断服务程序并读取 I2CRCV 的内容。这可以防止缓冲器溢出。

- 注 1:** 如果用户读 I2CRCV 的内容，在第 9 个时钟下降沿之前将 RBF 位清零，则 SCLREL 位将不会被清零，也不会发生时钟延长。
- 2:** 不管 RBF 位的状态如何，SCLREL 位都可以用软件置 1。为避免溢出，在下一个接收序列开始之前，用户要注意在中断服务程序中清零 RBF 位。

17.5.4 10 位寻址时的时钟延长（STREN = 1）

在寻址过程中会自动进行时钟延长。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，可以用软件清零 SCLREL 位，从而实现时钟延长的软件控制。逻辑会将对 SCLREL 位的写操作与 SCL 时钟进行同步。清零 SCLREL 位不会使 SCL 输出保持为低电平，直到模块检测到 SCL 输出的下降沿且采样到 SCL 为低电平。在 SCL 线采样为低电平时，如果用户清零 SCLREL 位，则 SCL 输出将保持为低电平。SCL 输出将保持低电平，直到 SCLREL 位置 1，并且 I²C 总线上所有其他器件都已释放了 SCL。这确保了对 SCLREL 位的写操作不会违反 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)。在主器件报文事件完成时，MI2CIF 中断标志被激活。在检测到发往从器件的报文时，SI2CIF 中断标志被激活。

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。根据需要，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)。当此位置 1 时，模块将接受并对所有地址执行操作。

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该以应答信号作出响应。

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

当广播呼叫使能位 (GCEN) 置 1 (I2CCON<7> = 1) 时，将识别广播呼叫地址。在检测到启动位后，将 8 个位移入 I2CRSR，并将地址与 I2CADD 进行比较，同时也与固化在硬件中的广播呼叫地址进行比较。

如果广播呼叫地址匹配，I2CRSR 在第 8 个时钟之后被送至 I2CRCV，RBF 标志在第 9 位 (ACK 位) 的下降沿置 1，主事件中断标志 (MI2CIF) 置 1。

在响应中断时，通过读 I2CRCV 的内容可以检测到中断的来源，从而确定地址是特定于器件的还是广播呼叫地址。

17.11 I²C 主模式支持

用作主器件时，支持以下 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，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 ACK 条件。

17.12 I²C 主模式操作

主器件产生所有的串行时钟脉冲、启动和停止条件。出现停止条件或重复启动条件时，传输结束。由于重复启动条件也是下一次串行传输的开始，此时不会释放 I²C 总线。

在主发送模式下，串行数据通过 SDA 输出，而 SCL 输出串行时钟。发送的第一个字节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。一次发送 8 位串行数据。在每个字节发送完之后，接收到一个 ACK 位。输出启动和停止条件，以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，发送的第一个字节是一个 7 位从地址，后跟一个表明接收的 1。通过 SDA 接收串行数据，而 SCL 输出串行时钟。一次接收 8 位串行数据。在每个字节接收之后，发送一个 ACK 位。启动和停止条件表明发送的开始和结束。

17.12.1 I²C 主发送

只需将值写入 I2CTRN 寄存器，便可以进行数据字节、7 位地址或 10 位地址后半部分的发送。当模块处于等待状态时，用户应该只写 I2CTRN。此操作将使缓冲器满标志 (TBF) 置 1，并允许波特率发生器开始计数，开始下一次发送。在 SCL 的下降沿后，地址 / 数据的每一个位都将被移出至 SDA 引脚上。发送状态标志 TRSTAT (I2CSTAT<14>) 用来表明主器件是否正在进行发送。

17.12.2 I²C 主接收

通过设置接收使能 (RCEN) 位 (I2CCON<3>)，可以使能主模式接收。在 RCEN 位置 1 之前，I²C 模块必须处于空闲状态，否则将忽略 RCEN 位。波特率发生器开始计数，每次计满返回时，在每个时钟的上升沿 SCL 引脚的状态和数据被移入 I2CRSR 中。

17.12.3 波特率发生器

在 I²C 主模式下，BRG 的重载值位于 I2CBRG 寄存器中。BRG 装入该值后，BRG 将递减计数到 0 并停止，直到发生再次装入。如果发生时钟仲裁，例如 SCL 引脚采样为高电平时，将重载 BRG。

dsPIC30F5015/5016

根据 I²C 标准，FSCK 可以为 100 kHz 或 400 kHz。但是，用户可以指定任何波特率，最高到 1 MHz。I2CBRG 的值不能是 0 或 1。

公式 17-1: 串行时钟速率

$$I2CBRG = \left(\frac{FCY}{FSCL} - \frac{FCY}{1,111,111} \right) - 1$$

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）时，就会发生时钟仲裁。当允许 SCL 引脚悬空为高电平时，波特率发生器暂停计数，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入 I2CBRG 的内容并开始计数。在外部器件使时钟保持为低电平时，这可以始终确保 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、总线冲突和总线仲裁

通过总线仲裁实现了对多主器件操作的支持。当主器件在 SDA 引脚上输出地址 / 数据位时，第一个主器件在 SDA 上输出 1（使 SDA 引脚悬空为高电平），而另一个主器件要输出 0，就会发生仲裁。当 SCL 引脚悬空为高电平时，数据应该是稳定的。如果 SDA 上预期数据是 1，而从 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 MI2CIF 脉冲置 1，并且将 I²C 端口的主器件部分复位到空闲状态。

如果正在进行发送时产生总线冲突，将停止发送，清零 TBF 标志，释放 SDA 和 SCL 线，而后可将值写入 I2CTRN。当用户执行 I²C 主事件中断服务程序时，如果 I²C 总线空闲（即 P 位置 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 I2CCON 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，如果停止条件出现，将 MI2CIF 位置 1。

写 I2CTRN 将从第一个数据位开始数据的发送，无论发生总线冲突时发送器停在何处。

在多主器件环境中，在检测到启动和停止条件时产生中断，可以确定总线何时空闲。当 I2CSTAT 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的 I²C 模块操作

17.13.1 CPU 休眠模式期间的 I²C 操作

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。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，则发送将中止。类似地，如果休眠出现在接收过程中，则接收也将中止。

17.13.2 CPU 空闲模式期间的 I²C 操作

对于 I²C，I2CSIDL 位用来选择空闲时模块是停止还是继续工作。如果 I2CSIDL = 0，则在空闲模式下模块将继续工作。如果 I2CSIDL = 1，则在空闲模式下模块将停止工作。

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 = 未初始化位
寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

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。关于 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）。

本节叙述通用异步收发器通信模块。

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）
- 一个或两个停止位
- 完全集成的波特率发生器，具有 16 位预分频器
- 在 30 MHz 指令速率下，波特率范围从 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、帧和缓冲器溢出错误检测
- 支持仅在地址检测（第 9 位 = 1）时产生中断
- 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

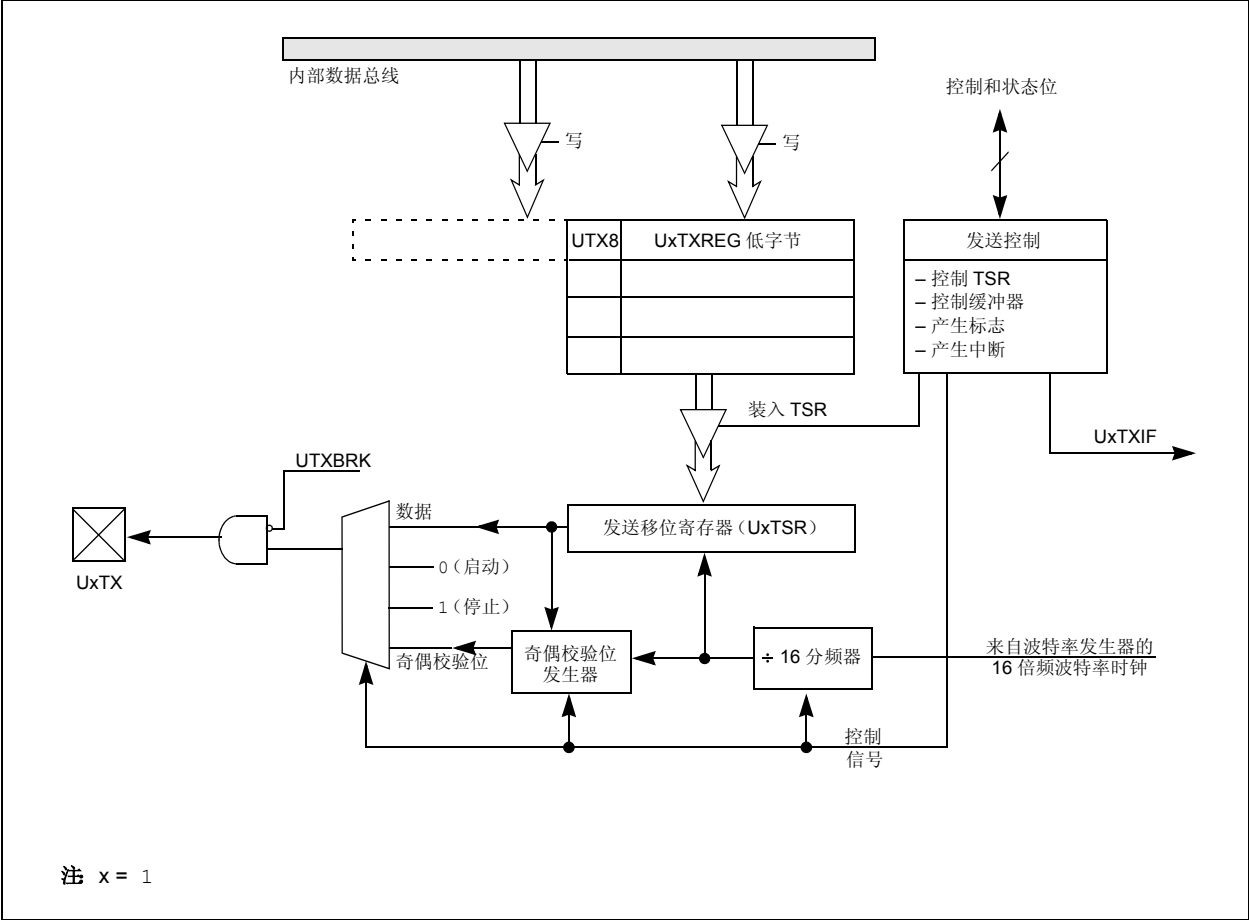
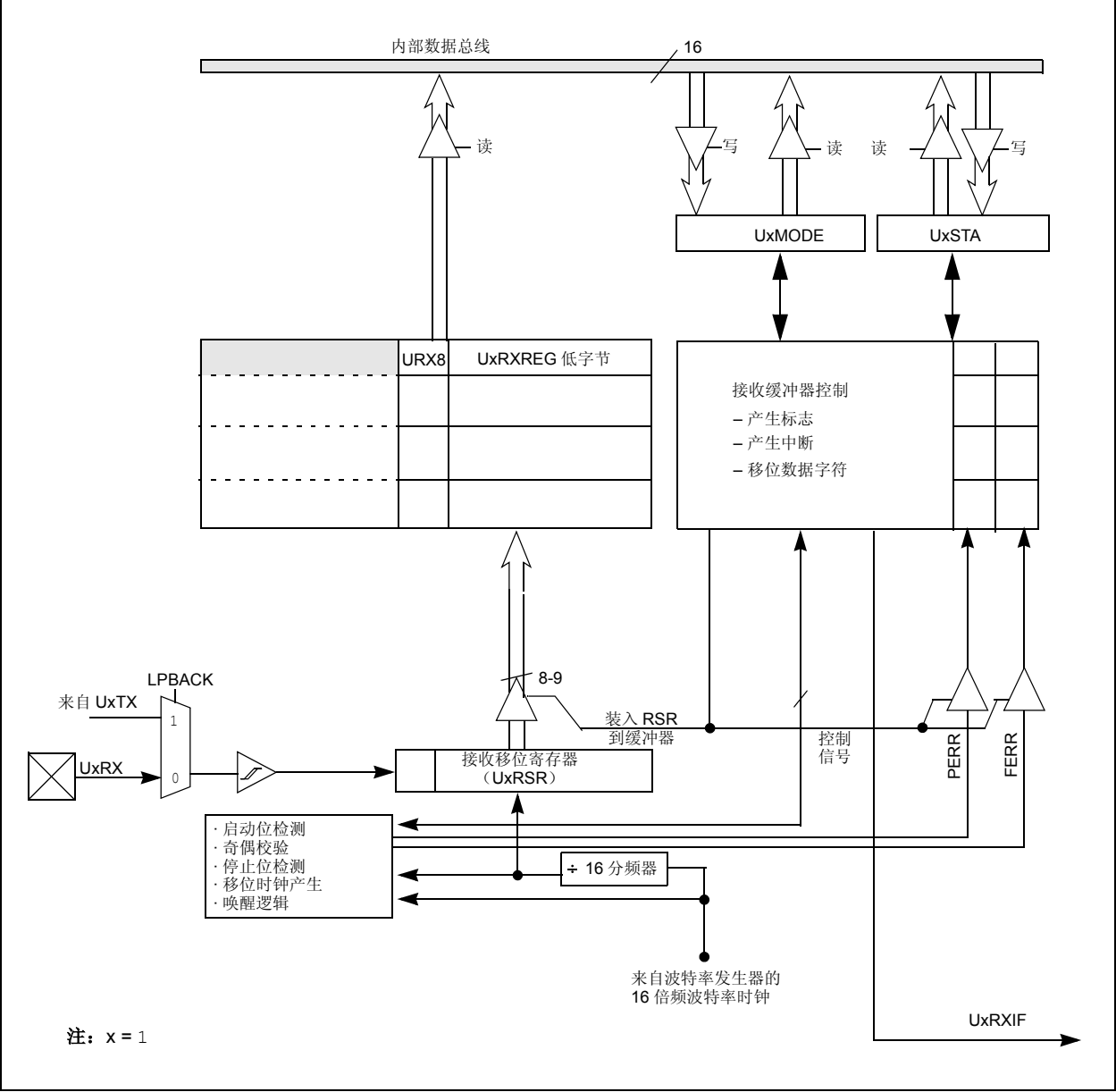


图 18-2: UART 接收器框图



18.2 使能和设置 UART

18.2.1 使能 UART

通过将 **UxMODE** 寄存器（其中 $x = 1$ ）中的 **UARTEN** 位置 1，可以使能 **UART** 模块。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 **LATCH** 寄存器位的设置。在没有发送数据时，**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，可以禁止 **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，受相应端口引脚的锁存器和 **TRIS** 位的控制。

禁止 **UART** 模块会将缓冲器复位为空状态。缓冲器中的所有数据字符都将丢失，同时波特率计数器也将复位。

当 **UART** 模块被禁止时，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 **RIDL** 和 **TRMT** 被置 1。其他控制位，包括 **ADDEN**、**URXISEL<1:0>** 和 **UTXISEL**，以及 **UxMODE** 和 **UxBRG** 寄存器则不受影响。

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复位模块。再次使能 **UART** 将使用同样的配置重新启动 **UART**。

18.2.3 设置数据、奇偶校验和停止位选择

UxMODE 寄存器中的控制位 **PDSEL<1:0>** 用来选择发送时使用的数据长度和奇偶性。数据长度可以是 8 位，具有偶校验位、奇校验位或没有校验位，数据长度也可以是没有校验位的 9 位。

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，必须执行以下步骤：

1. 设置 **UART**：
首先，必须选择数据长度、奇偶校验和停止位的位数。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。此外，必须向 **UxBRG** 寄存器中写入适当的波特率值。
2. 通过将 **UARTEN** (**UxMODE<15>**) 位置 1，使能 **UART**。
3. 将 **UTXEN** 位 (**UxSTA<10>**) 置 1，从而使能发送。

注： 在将 **UARTEN** 位置 1 以使能 **UART** 发送后，必须将 **UTXEN** 位置 1。

4. 将要发送的字节写入 **UxTXREG** 的低字节。此值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。或者，可以在 **UTXEN = 0** 时写入数据字节，而后用户可以将 **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 位数据发送涉及的步骤与 8 位数据发送类似，但是必须将一个 16 位数据字（其中高 7 位始终为零）写入 **UxTXREG** 寄存器。

18.3.3 发送缓冲器 (**UxTXB**)

发送缓冲器为 9 位宽，4 字符深度。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，则 FIFO 不会接受新数据，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，但当器件进入省电模式或从省电模式唤醒时，FIFO 不受影响。

18.3.4 发送中断

发送中断标志（U1TXIF）位于相应的中断标志寄存器中。

发送器通过产生一个边沿将 UxTXIF 位置 1。产生中断的条件取决于 UTXISEL 控制位：

- 如果 UTXISEL = 0，当从发送缓冲器向发送移位寄存器（UxTSR）传送一个字时，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，当从发送缓冲器向发送移位寄存器（UxTSR）传送一个字并且发送缓冲器为空时，将产生中断。

工作期间可以在这两种中断模式之间切换，有时这提供了更大的灵活性。

18.3.5 发送间隔字符

将 UTXBRK 位（UxSTA<11>）置 1 将使 UxTX 线被驱动为逻辑 0。UTXBRK 位改写了所有的发送活动。因此，通常用户应该在将 UTXBRK 置 1 之前等待发送器空闲。

要发送间隔字符，必须用软件将 UTXBRK 位置 1，并保持该位置 1 至少 13 个波特率时钟周期。然后用软件将 UTXBRK 位清零，产生停止位。在重新装入 UxTXB 或开始其他发送活动之前，用户必须等待至少一或两个波特率时钟周期，以确保产生有效的停止位。发送间隔字符不会产生发送中断。

18.4 接收数据

18.4.1 8 位或 9 位数据模式下的接收

在接收 8 位或 9 位数据时，必须执行以下步骤：

- 设置 UART（见第 18.3.1 节“8 位数据模式下的发送”）。
- 使能 UART（见第 18.3.1 节“8 位数据模式下的发送”）。
- 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，确定是否发生了溢出错误。OERR 位必须用软件复位。
- 从 UxRXREG 读取接收到的数据。读 UxRXREG 会将下一个字移动到接收 FIFO 的顶端，并且将更新 PERR 和 FERR 值。

18.4.2 接收缓冲器（UxRXB）

接收缓冲器为 4 字深度。加上接收移位寄存器（UxRSR），用户实际上有一个 5 字深度的 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，读取的值是缓冲器中的原有值，在 FIFO 中不会发生数据移位。

任何器件复位时，FIFO 都将复位。当器件进入省电模式或从省电模式唤醒时，FIFO 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置 1。将接收中断标志置 1 的条件，取决于由 URXISEL<1:0>（UxSTA<7:6>）控制位指定的设置。

- 如果 URXISEL<1:0> = 00 或 01，每当从接收移位寄存器（UxRSR）向接收缓冲器传送一个字时，就会产生中断。接收缓冲器可以有一个或多个字符。
- 如果 URXISEL<1:0> = 10，当从接收移位寄存器（UxRSR）向接收缓冲器传送一个字，使得接收缓冲器中有 3 个字符时，就会产生中断。
- 如果 URXISEL<1:0> = 11，当从接收移位寄存器（UxRSR）向接收缓冲器传送一个字，使得接收缓冲器中有 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模式之间切换，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）

如果满足以下所有条件，OERR 位（UxSTA<1>）将置 1：

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

一旦 OERR 置 1，就不会有数据移入 UxRSR（除非用软件清零 OERR 位或发生复位）。存放在 UxRSR 和 UxRXREG 中的数据继续有效。

18.5.2 帧错误（FERR 位）

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为 1，否则 FERR 也将置 1。FERR 位是只读的，它将和接收到的数据一起被送入缓冲器。此位在发生任何复位时被清零。

18.5.3 奇偶错误（PERR 位）

如果接收到的字奇偶错误的话，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，它将和接收到的数据一起被送入缓冲器。此位在发生任何复位时被清零。

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，RIDL 位（UxSTA<4>）为 0。在停止位完成与检测到下一个启动位之间时，RIDL 位为 1，表明 UART 空闲。

18.5.5 接收间隔

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

如果中止超过 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 中装入零，同时产生中断（如果允许中断），并且 RIDL 位置 1。

当模块接收到长间隔信号，同时接收器检测到启动位、数据位和无效停止位（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。

间隔字符被视为一个全 0 字符，且 FERR 位置 1。间隔字符被装入缓冲器中。只有在接收到停止位之后，才会继续进行接收。注意，当还没有接收到停止位时，RIDL 将变为高电平。

18.6 地址检测模式

通过将 ADDEN 位（UxSTA<5>）置 1 可以使能此特殊模式，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 位置 1 时产生。

18.7 环回模式

通过将 LPBACK 位置 1 可以使能此特殊模式，在该模式下，UxTX 引脚在内部连接到 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。
- 使能发送，如同第 18.3 节“发送数据”中定义的那样。

18.8 波特率发生器

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)

Fcy = 指令时钟速率 (1/Tcy)

波特率由公式 18-1 给出。

公式 18-1: 波特率

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

因此，可能的最大波特率为

Fcy/16（如果 BRG = 0），

可能的最小波特率为

Fcy/(16 * 65536)。

对于在 30 MIPS 工作的 16 位波特率发生器，能够达到的最小波特率为 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选定的捕捉输入。要启用此模式，用户必须通过编程输入捕捉模块，使其检测启动位的上升沿和下降沿。

18.10 CPU 休眠和空闲模式期间的 UART 操作

18.10.1 CPU 休眠模式期间的 UART 操作

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。如果在发送进行期间进入休眠模式，则发送将中止。UxTX 引脚被驱动为逻辑 1。类似地，如果在接收进行期间进入休眠模式，则接收将中止。UxSTA、UxMODE、发送和接收寄存器和缓冲器以及 UxBRG 寄存器不受休眠模式的影响。

在器件进入休眠模式前，如果唤醒位（UxMODE<7>）置 1，则 UxRX 引脚上的下降沿将产生接收中断。接收中断选择模式位（RXISEL）对此功能没有影响。如果接收中断允许，那么这会将器件从休眠中唤醒。为了产生唤醒中断，必须将 UARTEN 位置 1。

18.10.2 CPU 空闲模式期间的 UART 操作

对于 UART，USIDL 位选择当器件进入空闲模式时，模块是停止工作还是继续工作。如果 USIDL = 0，则模块将在空闲模式下继续工作。如果 USIDL = 1，则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 0000 0000 0000			
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
寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

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。关于 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，用来与其他 CAN 模块或单片机之间进行通信。此接口 / 协议设计为允许在噪声环境下进行通信。只有一个 CAN 模块可用。

CAN 模块是一个通信控制器，实现了 CAN 2.0 A/B 协议，如 BOSCH 规范中所定义。模块支持协议的 CAN 1.2、CAN 2.0A、CAN 2.0B Passive 以及 CAN 2.0B Active 版本。模块实现的是完整的 CAN 系统。本数据手册不讨论 CAN 规范。更多详细信息，读者可参见 BOSCH CAN 规范。

模块具有以下特性：

- 实现了 CAN 协议 CAN 1.2、CAN 2.0A 和 CAN 2.0B
- 标准和扩展数据帧
- 0-8 字节数据长度
- 可编程比特率，最高为 1 Mbit/s
- 支持远程帧
- 双缓冲接收器，带两个区分优先级的接收报文存储缓冲器（每个缓冲器可存放最多 8 字节的数据）
- 有 6 个完全（标准 / 扩展标识符）接收过滤器，其中 2 个与高优先级的接收缓冲器相关联，4 个与低优先级接收缓冲器相关联
- 有 2 个完全接收过滤屏蔽寄存器，分别与高优先级接收缓冲器和低优先级接收缓冲器相关联
- 有 3 个发送缓冲器，能够由应用指定优先级，并具有中止能力（每个缓冲器可存放最多 8 字节的数据）
- 可编程唤醒功能，集成了低通滤波器
- 可编程环回模式，支持自检操作
- 通过中断功能，为所有 CAN 接收器和发送器的错误状态发送信号
- 可编程时钟源
- 到定时器模块的可编程链接，可进行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与报文缓冲 / 控制组成。CAN 协议引擎处理 CAN 总线上报文接收和发送的所有功能。发送报文时，首先将报文装入恰当的数据寄存器。通过读取恰当的寄存器，可以检查状态和错误。在 CAN 总线检测到任何报文时，首先检查是否有错，然后与过滤器进行匹配，决定是否接收报文并把它存放到两个接收寄存器之一。

19.2 帧类型

CAN 模块发送各种类型的帧，包括用户发起的数据报文或远程传送请求，以及用于控制目标而自动生成的帧。支持以下帧类型：

• 标准数据帧

当节点希望传送数据时，就将生成标准数据帧。它包括一个 11 位标准标识符 (SID)，但不包括 18 位扩展标识符 (EID)。

• 扩展数据帧

扩展数据帧与标准数据帧类似，但多了扩展标识符。

• 远程帧

目标节点可以向源节点请求发送数据。要做到这一点，目标节点需要发送一个远程帧，其标识符与所需数据帧的标识符相匹配。随后，相应的数据源节点会发送一个数据帧，作为对远程请求的响应。

• 错误帧

节点检测到总线错误，就会生成错误帧。一个错误帧由 2 个字段构成：错误标志字段和错误界定符字段。

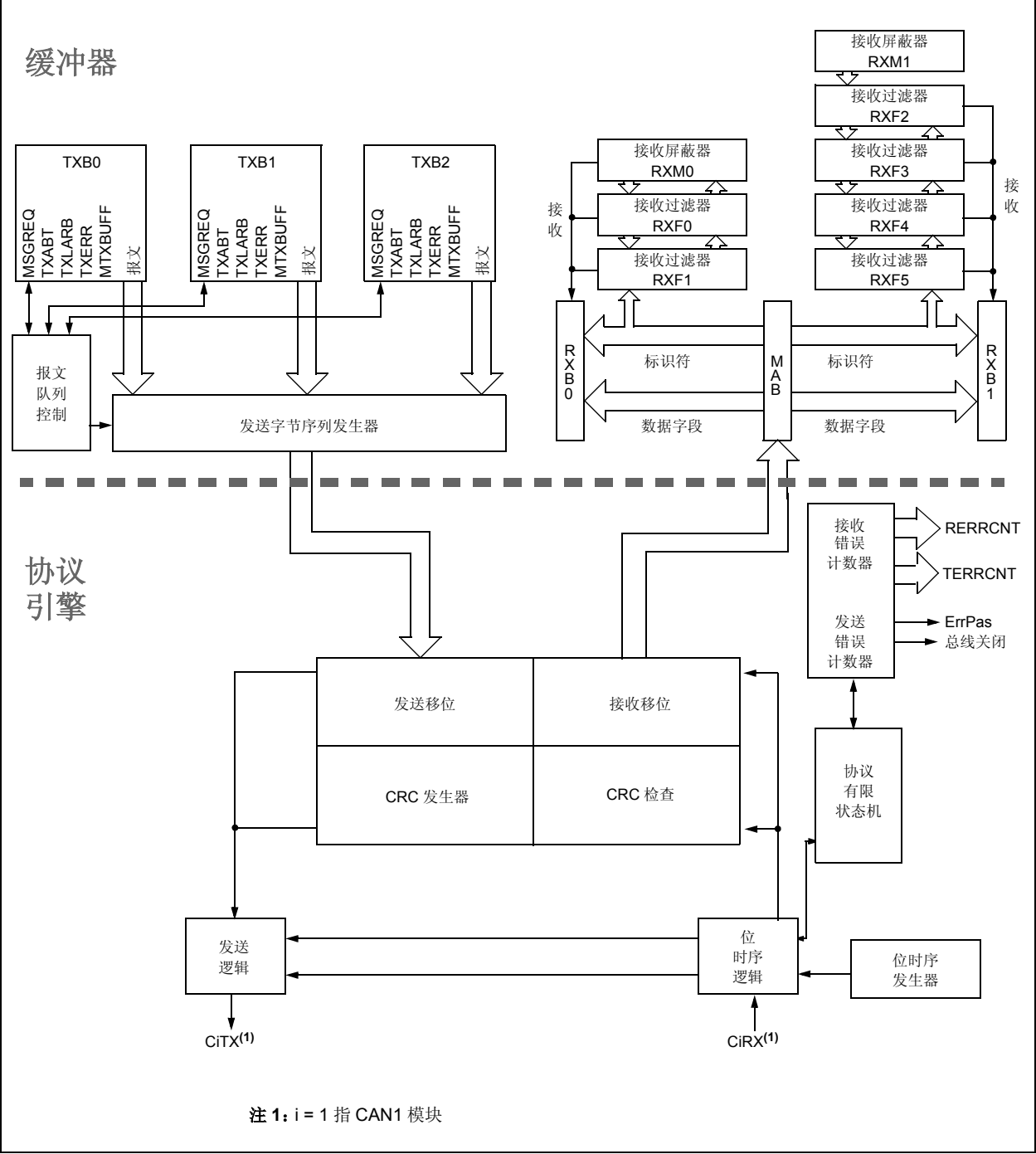
• 过载帧

如果出现以下两种情况之一，节点将生成过载帧。第一种情形，节点在帧间间隔期间检测到显性位，这是非法状态。第二种情形，由于内部状态，节点还不能开始接收下一条报文。节点可生成最多 2 个连续的重载帧，以便延迟接收下一条报文。

• 帧间间隔

帧间间隔将当前帧（无论何种类型）与其后的数据帧或远程帧分隔开来。

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 模块工作在以下几种工作模式之一。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>) 可以确认进入的模式。除非模式改变是可接受的，通常是在由至少 11 个连续隐性位确定的总线空闲时间内，否则模块不会改变模式和 OPMODE 位。

19.3.1 初始化模式

在初始化模式下，模块不进行发送或接收。错误计数器清零，中断标志保持不变。编程器能够访问配置寄存器，而这在其他模式下是受到限制的。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，不能修改所有控制模块配置的寄存器。如果正在发送，不允许 CAN 模块进入配置模式。配置模式就像是一把锁，保护着下列寄存器：

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

19.3.2 禁止模式

在禁止模式下，模块不会进行发送或接收。由于总线活动，模块能够置 1 WAKIF 位，但是，等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，模块将进入模块禁止模式。如果该模块处于活动状态，它将等候 CAN 总线上的 11 位隐性位，检测到总线空闲状态，然后接受模块禁止命令。当 OPMODE<2:0> 位 (CiCTRL<7:5>) = 001 时，这表明模块成功进入了禁止模式。当模块在模块禁止模式下时，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，可以设置模块，在 CiRX 输入线上应用低通滤波器功能。由 WAKFIL 位 (CiCFG2<14>) 使能或禁止滤波器。

注： 通常，如果允许 CAN 模块在某种工作模式下发送，并且在 CAN 模块进入该模式后立即被要求发送，则模块将在开始发送前等待总线上出现 11 个连续隐性位。如果用户在此 11 位期间切换到禁止模式，那么发送会被中止，同时相应的 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，模块被激活，I/O 引脚用作 CAN 总线功能。模块将通过 CxTX 和 CxRX 引脚，发送和接收 CAN 总线报文。

19.3.4 监听模式

如果激活监听模式，则 CAN 总线上的模块是被动的。发送器缓冲器恢复为端口 I/O 功能。接收引脚仍然为输入。对于接收器，不发出错误标志或应答信号。该状态下，错误计数器失效。监听模式可用于检测 CAN 总线上的波特率。要使用监听模式，必须有至少 2 个可以互相通信的节点。

19.3.5 错误识别模式

可以设置模块，使它忽略所有的错误并接收任何报文。通过设置 RXM<1:0> 位 (CiRXnCON<6:5>) 寄存器为 11，激活错误识别模式。在此模式下，报文合成缓冲器中一直到错误发生前的数据将被复制到接收缓冲器中，可通过 CPU 接口读取。

19.3.6 环回模式

如果环回模式被激活，模块将在模块边界把内部发送信号连接到内部接收信号。发送和接收引脚将转换到其 I/O 端口功能。

19.4 报文接收

19.4.1 接收缓冲器

CAN 总线模块有 3 个接收缓冲器。但是，其中一个缓冲器总是用于监视总线是否有进入的报文。这个缓冲器称为报文合成缓冲器（Message Assembly Buffer, MAB）。因此只有 2 个可见的接收缓冲器（RXB0 和 RXB1），基本上可以即时接收来自协议引擎的完整报文。

MAB 将组合所有接收到的报文，这些报文只有在满足接收过滤器判据时才被传送到 RXBn 缓冲器。当接收到报文时，RXnIF 标志（CiINTF<0> 或 CiINRF<1>）将置 1。此位仅在接收到报文时才能被模块置 1。该位在 CPU 处理完缓冲器中的报文后由 CPU 清零。如果 RXnIE 位（CiINTE<0> 或 CiINTE<1>）置 1，当接收到报文时将会产生一个中断。

RXF0 和 RXF1 过滤器，以及 RXM0 屏蔽器与 RXB0 相关联。过滤器 RXF2、RXF3、RXF4 和 RXF5，以及屏蔽器 RXM1 与 RXB1 关联。

19.4.2 报文接收过滤器

报文接收过滤器和屏蔽寄存器用于决定报文合成缓冲器中的一条报文是否应该被装入接收缓冲器中的一个。一旦一条有效报文被接收进入报文合成缓冲器，就会将该报文的标识符字段与过滤值进行比较。如果匹配的话，该报文就会被装入相应的接收缓冲器。

接收过滤器检查进入的报文，根据 RXIDE 位（CiRXnSID<0>）决定如何比较标识符。如果 RXIDE 位清零，报文是标准帧，只与 EXIDE 位（CiRXFnSID<0>）清零的过滤器比较。如果 RXIDE 位置 1，报文是扩展帧，只与 EXIDE 位置 1 的过滤器比较。将 RXM<1:0> 位配置为 01 或 10，可以改写 EXIDE 位。

19.4.3 报文接收过滤屏蔽

屏蔽位基本上决定了哪些位应进行过滤。如果某屏蔽位设置为零，则该位将被自动接收，而与过滤位无关。有 2 个可编程接收过滤屏蔽器与接收缓冲器关联，每个缓冲器有一个过滤屏蔽器与其关联。

19.4.4 接收溢出

当报文合成缓冲器组合了一个有效的接收报文后，该报文通过接收过滤器接收，而此时与该过滤器关联的接收缓冲器还未指定将前一次接收的报文清除时，就会发生溢出。

溢出错误标志 RXnOVR（CiINTF<15> 或 CiINTF<14>）和 ERRIF 位（CiINTF<5>）将置 1，MAB 中的报文会被丢弃。

如果 DBEN 位清零，则 RXB1 和 RXB0 独立工作。在这种情况下，如果 RXB0 中有未读的报文，则要存入 RXB0 的报文将不会被转存到 RXB1，并且 RX0OVR 位将置 1。

如果 DBEN 位置 1，处理 RXB0 溢出的方法就不同了。如果接收到一条将要存入 RXB0 的有效报文，且 RXFUL = 1（RXB0 满）、RXFUL = 0（RXB1 空），RXB0 的报文将装入 RXB1。此时不会为 RXB0 产生溢出错误。如果接收到一条将要存入 RXB0 的有效报文且 RXFUL = 1（表明 RXB0 和 RXB1 都为满），这样报文将被丢弃并产生 RXB1 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（Cyclic Redundancy Check，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。然而，当发生上述错误之一时，接收错误计数器会递增 1。置 1 RXWAR 位（CiINTF<9>），表明接收错误计数器已经达到 CPU 警告上限值 96，接着产生中断。

19.4.6 接收中断

接收中断可划分为三大类，每类包括各种产生中断的条件：

- 接收中断

报文已被成功接收并被装入一个接收缓冲器中。接收到帧结束（End-of-Frame, EOF）字段后，中断立即被激活。读 RXnIF 标志位可知哪个接收缓冲器引起了中断。

- 唤醒中断

CAN 模块已经从禁止模式中唤醒，或者器件已经从休眠模式中唤醒。

- 接收错误中断

接收错误中断由 **ERRIF** 位表明。该位表明产生了错误条件。通过检查 **CAN** 中断状态寄存器中的位 **CiINTF**，可以确定错误的来源。

- 接收到无效的报文

如果在最后一条报文的接收过程中，产生了任何类型的错误，**IVRIF** 位将表明出现错误。

- 接收器溢出

RXnOVR 位表明是否出现溢出。

- 接收器警告

RXWAR 位表明，是否接收错误计数器 (**RERRCNT<7:0>**) 已经到达 **96** 的警告上限。

- 接收器错误被动状态

RXEP 位表明，是否接收错误计数器已经超过了 **127** 的错误被动上限，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 模块有 3 个发送缓冲器。每个缓冲器可容纳 14 字节的数据。其中 8 个字节用于存放发送的报文（最多 8 个字节）。另外 5 个字节用来存放标准或扩展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。如果某报文缓冲器的 **TXPRI<1:0>** (**CiTnCON<1:0>**，其中 $n = 0, 1$ 或 2 ，代表具体的发送缓冲器) 置为 11，则该缓冲器具有最高优先级。如果某个报文缓冲器的 **TXPRI<1:0>** 被置为 10 或 01，则该缓冲器具有中等优先级。如果某个报文缓冲器的 **TXPRI<1:0>** 是 00，则该缓冲器优先级最低。

19.5.3 发送过程

必须把 **TXREQ** 位 (**CiTnCON<3>**) 置 1 来开始发送报文。**CAN** 总线模块解决了由 **TXREQ** 位设定与帧起始 (**Start-of-Frame, SOF**) 之间的任何时序冲突，确保在优先级改变时，能在发送 **SOF** 之前正确解决时序冲突。当 **TXREQ** 置 1 时，**TXABT** (**CiTnCON<6>**)、**TXLARB** (**CiTnCON<5>**) 和 **TXERR** (**CiTnCON<4>**) 标志位将自动清零。

将 **TXREQ** 位置 1 即表示将报文缓冲器标志为排队以等待发送。当模块检测到总线可用时，就开始发送具有最高优先级的报文。

如果首次尝试发送就成功完成，**TXREQ** 位将自动清零，如果 **TXnIE** 位置 1 的话还将产生中断。

如果报文发送失败，错误状态标志位中的一个将置 1，**TXREQ** 位将保持置 1，表示该报文仍然等待发送。如果在报文尝试发送期间遇到出错情况，**TXERR** 位将置 1，出错情况可能会引起中断。如果报文尝试发送期间仲裁失败，**TXLARB** 位将置 1。没有中断产生表明仲裁失败。

19.5.4 中止报文发送

通过清零与各个报文缓冲器关联的 **TXREQ** 位，系统能中止报文发送。置 1 **ABAT** 位 (**CiCTRL<12>**) 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，那么将被中止。当模块置 1 **TXABT** 位时，表明发生了中止，**TXnIF** 标志不会被自动置 1。

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有表示。每个错误将引起错误计数器的值递增 1。一旦错误计数器的值超过 96，**ERRIF** (**CiINTF<5>**) 和 **TXWAR** 位 (**CiINTF<10>**) 将置 1。一旦错误计数器的值超过 96，将产生中断，错误标志寄存器中的 **TXWAR** 位将置 1。

19.5.6 发送中断

发送中断可划分为两大类，每类包括各种产生中断的条件：

- 发送中断

三个发送缓冲器中至少有一个为空（未预定），并且可以装入预定发送的报文。读 **TXnIF** 标志位可知哪一个发送缓冲器可用并引起了中断。

- 发送错误中断

发送错误中断由 **ERRIF** 标志位表明。该标志位表明产生了错误条件。通过检查 **CAN** 中断状态寄存器 **CiINTF** 中的错误标志，可以确定错误的起源。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** 位表示发送错误计数器的值已经达到 CPU 警告的上限值 96。
- 发送器错误被动
- **TXEP** 位（**CiINTF<12>**）用来表示发送错误计数器已经超过了错误被动的上限值 127，模块已经进入了错误被动状态。
- 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，模块已经进入总线关断状态。

19.6 波特率设置

任意特定 **CAN** 总线上的所有节点必须有相同的标称比特率。为了设置波特率，必须初始化下列参数：

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

19.6.1 位时序

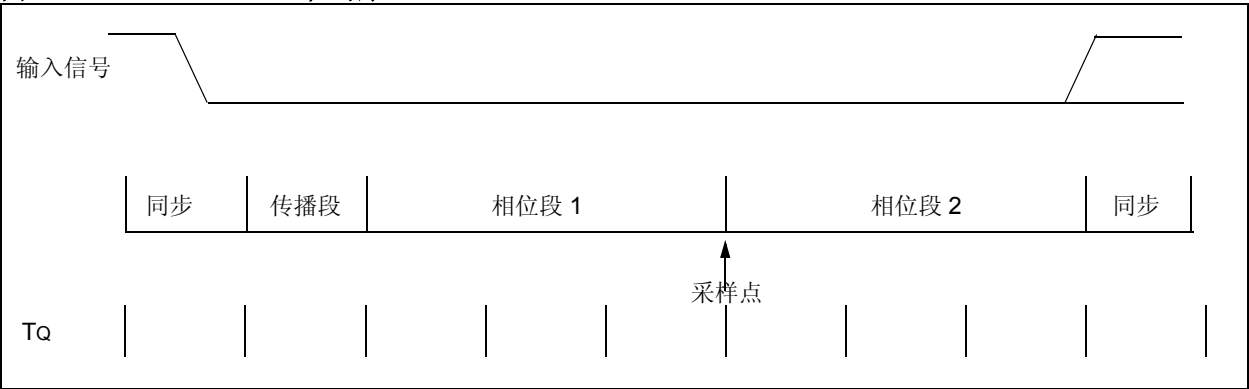
CAN 总线上的所有控制器都必须有相同的波特率和位长度。然而，不同的控制器并不要求使用相同的主振荡器时钟。如果各控制器的时钟频率不同，就不得不通过调节每个段的时间份额数来调节波特率。

可以认为，标称位时间划分成几个互不重叠的时间段。这些段如图 19-2 所示。

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

时间段以及标称位时间由整数个时间单元组成，这些单元称为时间份额或 **Tq**。根据定义，标称位时间最小由 8 个 **Tq** 组成，最大由 25 个 **Tq** 组成。同样根据定义，最小标称位时间是 1 μ s，对应最大 1 MHz 的比特率。

图 19-2: CAN 位时序



19.6.2 预分频器设置

有一个可编程预分频器，它除了固定的二分频用于时钟发生以外，还带有整数预分频比，范围为 1 至 64。其时间份额（TQ）是从振荡器周期得出的固定时间单元，由公式 19-1 给出，其中 FCAN 是 Fcy（如果 CANCKS 位置 1；如该位清零则为 4 Fcy）。

注： FCAN 不能超出 30 MHz。如果 CANCKS = 0，那么 Fcy 不能超出 7.5 MHz。

公式 19-1: 时钟发生的时间份额

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

19.6.3 传播段

这部分的位时间用来补偿网络内的物理延时。这些延时包括总线线路上的信号传播时间以及节点的内部延时。通过设置 PRSEG<2:0> 位（CiCFG2<2:0>），可以设置传播段为 1 Tq 到 8 Tq。

19.6.4 相位段

相位段用于在发送位时间内定位最佳的接收位采样。采样点在相位缓冲段 1 与相位缓冲段 2 之间。这两个段可以通过重新同步加长或缩短。相位缓冲段 1 的末尾决定在一个位周期内的采样点。该段可以编程为 1 Tq 到 8 Tq。相位缓冲段 2 为下一个发送数据跳变提供一个延时。该段可以编程为 1 Tq 到 8 Tq，也可将其定义为相位缓冲段 1 的时间份额与信息处理时间（2 Tq）两者中的较大者。通过设置 SEG1PH<2:0>（CiCFG2<5:3>）位初始化相位缓冲段 1，通过设置 SEG2PH<2:0>（CiCFG2<10:8>）位初始化相位缓冲段 2。

在设置相位缓冲段的长度时，必须满足下面的要求：

- 传播段 + 相位缓冲段 1 >= 相位缓冲段 2

19.6.5 采样点

采样点是个时间点，在此时间点将读取总线电平并将其解释为对应位的值。它的位置在相位缓冲段 1 的末尾。若位时序较慢而且包含很多 Tq，可以在同一个采样点指定总线线路的多个采样。由 CAN 总线决定的电平将对应于三值择多决策的结果。择多采样在采样点进行，且前两次采样相隔 Tq/2。CAN 模块允许用户选择在同一点采样 3 次或 1 次。这可以通过置 1 或清零 SAM 位（CiCFG2<6>）实现。

通常，位采样应该发生在位时间的 60-70% 处，取决于系统参数。

19.6.6 同步

为了补偿不同总线站振荡器频率的相移，每个 CAN 控制器必须要能够与进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，逻辑会将该沿的位置与预期时间（同步段）比较。然后电路将调节相位缓冲段 1 和相位缓冲段 2 的值。有两种机制用来同步。

19.6.6.1 硬同步

只有在总线空闲期间产生一个从“隐性”转变到“显性”的边沿时（表明报文开始），才会进行硬同步。硬同步后，位时间计数器从同步段重新开始计数。硬同步把引起硬同步的沿强制地置于重新开始的位时间同步段之内。如果完成一个硬同步，在该位时间内将不会再有重新同步。

19.6.6.2 重新同步

重新同步可能使相位缓冲段 1 加长或相位缓冲段 2 缩短。相位缓冲段加长或缩短的量有一个上限值，称为同步跳转宽度，由 SJW<1:0> 位（CiCFG1<7:6>）指定。同步跳转宽度的值将被加到相位缓冲段 1 或从相位缓冲段 2 减去。重新同步跳转宽度可以设置为 1 Tq 到 4 Tq。

在设置 SJW<1:0> 位时，必须满足以下要求：

- 相位缓冲段 2 > 同步跳转宽度

表 19-1: 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	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EXIDE	000u uuuu uuuu uuuu
C1RXF0EIDH	0302	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXF0EIDL	0304	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF1SID	0308	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EXIDE	000u uuuu uuuu uuuu
C1RXF1EIDH	030A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXF1EIDL	030C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF2SID	0310	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EXIDE	000u uuuu uuuu uuuu
C1RXF2EIDH	0312	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXF2EIDL	0314	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF3SID	0318	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EXIDE	000u uuuu uuuu uuuu
C1RXF3EIDH	031A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXF3EIDL	031C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF4SID	0320	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EXIDE	000u uuuu uuuu uuuu
C1RXF4EIDH	0322	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXF4EIDL	0324	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF5SID	0328	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EXIDE	000u uuuu uuuu uuuu
C1RXF5EIDH	032A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXF5EIDL	032C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM0SID	0330	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MIDE	000u uuuu uuuu uuuu
C1RXM0EIDH	0332	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXM0EIDL	0334	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM1SID	0338	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MIDE	000u uuuu uuuu uuuu
C1RXM1EIDH	033A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 uuuu uuuu uuuu
C1RXM1EIDL	033C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1TX2SID	0340	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX2EID	0342	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu 0000 uuuu uuuu
C1TX2DLC	0344	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uuuu uuuu u000
C1TX2B1	0346	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uuuu uuuu uuuu
C1TX2B2	0348	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uuuu uuuu uuuu
C1TX2B3	034A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uuuu uuuu uuuu
C1TX2B4	034C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uuuu uuuu uuuu
C1TX2CON	034E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TXPRI<1:0>	0000 0000 0000 0000
C1TX1SID	0350	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX1EID	0352	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu 0000 uuuu uuuu
C1TX1DLC	0354	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uuuu uuuu u000
C1TX1B1	0356	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	uuuu uuuu uuuu uuuu

u = 未初始化

寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

图注:
注:

表 19-1: 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																uuuu uuuu uuuu uuuu
C1TX1B3	035A	发送缓冲器 1 字节 5																uuuu uuuu uuuu uuuu
C1TX1B4	035C	发送缓冲器 1 字节 7																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>																uuuu u000 uuuu uuuu
C1TX0EID	0362	发送缓冲器 0 扩展标识符 <17:14>																uuuu 0000 uuuu uuuu
C1TX0DLC	0364	发送缓冲器 0 扩展标识符 <5:0>																uuuu uuuu uuuu u000
C1TX0B1	0366	发送缓冲器 0 字节 1																uuuu uuuu uuuu uuuu
C1TX0B2	0368	发送缓冲器 0 字节 3																uuuu uuuu uuuu uuuu
C1TX0B3	036A	发送缓冲器 0 字节 5																uuuu uuuu uuuu uuuu
C1TX0B4	036C	发送缓冲器 0 字节 7																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>									000u uuuu uuuu uuuu
C1RX1EID	0372	—	—	—	—	—	—	—	接收缓冲器 1 扩展标识符 <17:6>									0000 uuuu uuuu uuuu
C1RX1DLC	0374	接收缓冲器 1 扩展标识符 <5:0>																uuuu uuuu 000u uuuu
C1RX1B1	0376	接收缓冲器 1 字节 1																uuuu uuuu uuuu uuuu
C1RX1B2	0378	接收缓冲器 1 字节 3																uuuu uuuu uuuu uuuu
C1RX1B3	037A	接收缓冲器 1 字节 5																uuuu uuuu uuuu uuuu
C1RX1B4	037C	接收缓冲器 1 字节 7																uuuu uuuu uuuu uuuu
C1RX1CON	037E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTTRO	FILHIT<2:0>		0000 0000 0000 0000	
C1RX0SID	0380	—	—	—	—	—	—	接收缓冲器 0 标准标识符 <10:0>									000u uuuu uuuu uuuu	
C1RX0EID	0382	—	—	—	—	—	—	接收缓冲器 0 扩展标识符 <17:6>									0000 uuuu uuuu uuuu	
C1RX0DLC	0384	接收缓冲器 0 扩展标识符 <5:0>																uuuu uuuu 000u uuuu
C1RX0B1	0386	接收缓冲器 0 字节 1																uuuu uuuu uuuu uuuu
C1RX0B2	0388	接收缓冲器 0 字节 3																uuuu uuuu uuuu uuuu
C1RX0B3	038A	接收缓冲器 0 字节 5																uuuu uuuu uuuu uuuu
C1RX0B4	038C	接收缓冲器 0 字节 7																uuuu uuuu uuuu uuuu
C1RX0CON	038E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTTRO	DBEN	JTOFF	FILHITO	
C1CTRL	0390	CANCAP	CSIDLE	ABAT	CANCKS	REQOP<2:0>	—	—	—	OPMODE<2:0>		—		ICODE<2:0>		—		
C1CFG1	0392	—	—	—	—	—	—	—	—	SJM<1:0>		BRP<5:0>						
C1CFG2	0394	—	WAKFIL	—	—	—	—	—	—	SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>			
C1INTF	0396	RX0OVR	RX1OVR	TXBO	TXEP	RXEP	TXWAR	RXWAR	EWARN	IVRIE	WAKIF	ERRIF	TX2IF	TX1IF	TX0IF	RX1IF	RX0IF	
C1INTE	0398	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	TX2IE	TX1IE	TX0IE	RX1IE	RX0IE	
C1EC	039A	发送错误计数器寄存器																0000 0000 0000 0000

u = 未初始化

寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

图注:

注:

dsPIC30F5015/5016

注:

20.0 系统集成

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。关于 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)。关于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，提供省电工作模式以及提供代码保护：

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

dsPIC30F 器件具备一个看门狗定时器，它可以通过配置位永久使能，或用软件进行控制。它依靠自身的 RC 振荡器运行，以便提升可靠性。两个定时器提供了上电时必需的延时。一个是振荡器起振定时器 (OST)，旨在确保芯片在晶体振荡器达到稳定之前始终处于复位状态。另一个是上电延时定时器 (PWRT)，仅在上电时提供延时，用来在电源稳定过程中使器件保持在复位状态。有了这两个片上定时器，大多数应用不再需要外部复位电路。

休眠模式旨在提供极低电流的掉电模式。通过外部复位、看门狗定时器唤醒或中断，用户可将器件从休眠中唤醒。一些振荡器功能仍然可用，从而器件能够适用于范围宽广的各种应用。在空闲模式下，时钟源仍然处于工作状态，但 CPU 停止。RC 振荡器可降低系统成本，而 LP 晶振功能可降低功耗。

20.1 振荡器系统概述

dsPIC30F 振荡器系统包含以下模块和功能：

- 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
- 在各种时钟源之间进行切换的时钟切换机制
- 可编程时钟后分频器，可节省系统功耗
- 故障保护时钟监视器 (FSCM)，可检测时钟故障并采取故障保护措施
- 时钟控制寄存器 OSCCON
- 用于主振荡器选择的配置位

配置位决定了上电复位 (POR) 和欠压复位 (BOR) 时的时钟源。于是，时钟源可以在允许的时钟源之间切换。OSCCON 寄存器控制时钟切换并反映系统时钟相关状态位。

表 20-1 给出了 dsPIC30F 振荡器工作模式的汇总。图 20-1 给出了振荡器系统的简化图。

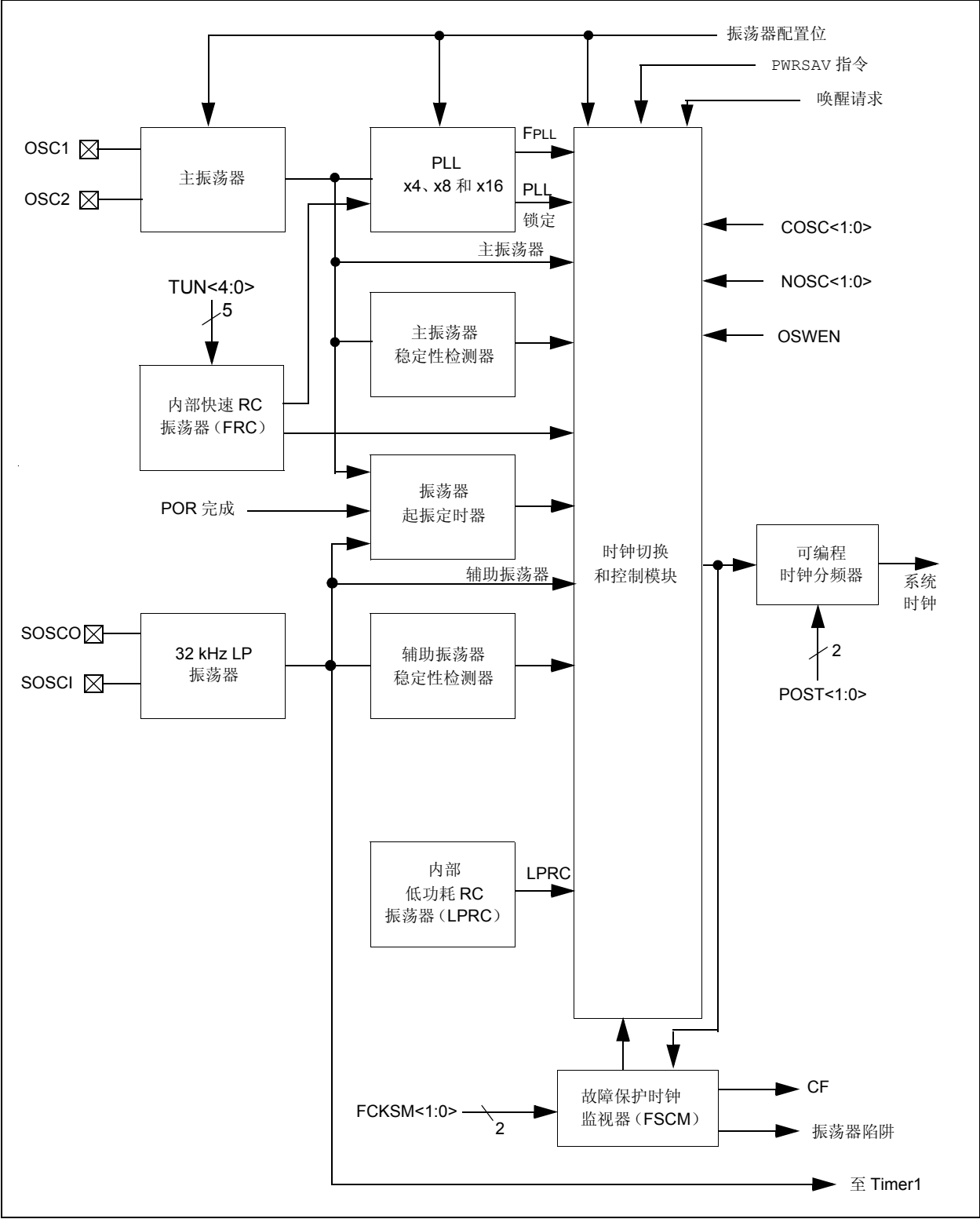
dsPIC30F5015/5016

表 20-1: 振荡器工作模式

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

- 注
- 1: 再高的话, 就将超出器件工作频率范围。
 - 2: LP 振荡器可以很方便地共用为系统时钟, 以及共用为 Timer1 的实时时钟。
 - 3: 高于此值将超出 PLL 输入范围。
 - 4: 低于此值将超出 PLL 输入范围。
 - 5: 需要外部 R 和 C。最大工作频率 4 MHz。

图 20-1: 振荡器系统框图



dsPIC30F5015/5016

20.2 振荡器配置

20.2.1 初始时钟源选择

当器件从上电复位或欠压复位中退出时，器件将选择其时钟源，选择是基于：

- a) FOS<2:0> 配置位，选择四个振荡器组中的一个。
- b) 以及 FPR<4:0> 配置位，在主组内选择 16 个振荡器之一。

选择如表 20-2 所示。

表 20-2: 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
ECIO, PLL 4x	PLL	1	1	1	0	1	1	0	1	I/O
ECIO, PLL 8x	PLL	1	1	1	0	1	1	1	0	I/O
ECIO, PLL 16x	PLL	1	1	1	0	1	1	1	1	I/O
FRC, PLL 4x	PLL	1	1	1	0	0	0	0	1	I/O
FRC, PLL 8x	PLL	1	1	1	0	1	0	1	0	I/O
FRC, PLL 16x	PLL	1	1	1	0	0	0	1	1	I/O
XT, PLL 4x	PLL	1	1	1	0	0	1	0	1	OSC2
XT, PLL 8x	PLL	1	1	1	0	0	1	1	0	OSC2
XT, PLL 16x	PLL	1	1	1	0	0	1	1	1	OSC2
HS/2, PLL 4x	PLL	1	1	1	1	0	0	0	1	OSC2
HS/2, PLL 8x	PLL	1	1	1	1	0	0	1	0	OSC2
HS/2, PLL 16x	PLL	1	1	1	1	0	0	1	1	OSC2
HS/3, PLL 4x	PLL	1	1	1	1	0	1	0	1	OSC2
HS/3, PLL 8x	PLL	1	1	1	1	0	1	1	0	OSC2
HS/3, PLL 16x	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 和 2)
FRC	内部 FRC	0	0	1	x	x	x	x	x	(注 1 和 2)
LPRC	内部 LPRC	0	1	0	x	x	x	x	x	(注 1 和 2)

注 1: OSC2 引脚功能由 FPR<4:0> 决定。

2: OSC1 引脚不能用作 I/O 引脚，即使是始终选用辅助振荡器或内部时钟源也是如此。

20.2.2 振荡器起振定时器（OST）

为了确保晶体振荡器（或陶瓷谐振器）已经起振并稳定工作，提供了振荡器起振定时器。它是一个简单的 10 位计数器，在计数 1024 个 T_{osc} 周期后，再将振荡器时钟提供给系统其余部分使用。超时周期用 T_{ost} 表示。每次振荡器不得不重启时（即在 POR 和 BOR 时或从休眠中唤醒时）都需要一段 T_{ost} 时间。对于主振荡器，振荡器起振定时器适用于 LP、XT、XTL 和 HS 振荡器模式（从休眠中唤醒、POR 以及 BOR 时）。

20.2.3 LP 振荡器控制

下面的两个因素控制着 LP 振荡器的使能：

- 1. 当前振荡器组选择位 $COSC<2:0>$
- 2. LPOSCEN 位（OSCCON 寄存器）

如果 $LPOSCEN = 1$ ，则 LP 振荡器工作（即便是在休眠模式期间）。LP 振荡器用作器件时钟，如果：

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

如果 LP 振荡器始终保持工作状态，可以快速切换到 32 kHz 系统时钟进行低功耗工作。返回到较快的主振荡器仍然需要一段起振时间。

20.2.4 锁相环（PLL）

PLL 把主振荡器产生的时钟进行倍频处理。PLL 的增益可以选择为 $\times 4$ 、 $\times 8$ 和 $\times 16$ 。表 20-3 汇总了输入和输出频率范围。

表 20-3: PLL 频率范围

F_{in}	PLL 倍频器	F_{out}
4 MHz-10 MHz	4 倍频	16 MHz-40 MHz
4 MHz-10 MHz	8 倍频	32 MHz-80 MHz
4 MHz-7.5 MHz	16 倍频	64 MHz-120 MHz

PLL 具备一个锁定输出，当 PLL 进入锁相状态时它将变为有效。如果锁相环失锁（例如，由于噪声的缘故），锁定信号将失效。该信号的状态由 OSCCON 寄存器中只读的 LOCK 位反映。

20.2.5 快速 RC 振荡器（FRC）

FRC 振荡器是快速的（标称值 7.37 MHz $\pm 2\%$ ）内部 RC 振荡器。在不使用外部晶振、陶瓷谐振器或 RC 网络的情况下，使用该振荡器提供合理的器件运行速度。FRC 振荡器可以配合 PLL 使用，以获得更高的时钟频率。

当 OSCCON 寄存器中的当前振荡器选择控制位（ $OSCCON<14:12>$ ）设置为 001 时，dsPIC30F 器件将依靠 FRC 振荡器工作。

用户使用 $TUN<4:0>$ （ $OSCTUN<4:0>$ ）指定的 6 个位域，来对内部快速 RC 振荡器（标称值 7.37 MHz）进行调整。对于 FRC 振荡器，用户可调整范围为以出厂校准设定为中心的 +12.6%（930 kHz）和 -13%（960 kHz），调整最小单位（步长）是 0.4%（见表 20-4）。

如果 $OSCCON<14:12>$ 设置为 111，且 $FPR<4:0>$ 设置为 00101、00110 或 00111，那么可分别使用 4、8 或 16 倍频的 PLL 倍频器。

注：使用 16 倍频 PLL 时，FRC 振荡器调整后的频率，不得大于 7.5 MHz。

表 20-4: FRC 调整

$TUN<5:0>$ 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（振荡器运行在校准频率）
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%

20.2.6 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, 振荡频率标称值为 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。在为功耗要求严格但对时序精度要求不高的应用提供低频率时钟源时, 也可使用 LPRC 振荡器。

因为 LPRC 振荡器是 PWRT 的时钟源, 所以它在上电复位时总是使能的。在 PWRT 超时后, 如果以下条件之一为真, 则 LPRC 振荡器将保持为开启状态:

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

如果以上条件都不为真, 则 LPRC 将在 PWRT 超时后关闭。

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

2: 注意, OSC1 引脚不能用作 I/O 引脚, 即便是始终选用辅助振荡器或内部时钟源。

20.2.7 故障保护时钟监视器

故障保护时钟监视器 (FSCM) 使得器件即使在出现振荡器故障时仍能继续工作。恰当地设置 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), 可使能 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), 不再受 SWDTEN 位的控制。

在发生振荡器故障时, FSCM 会产生时钟故障陷阱事件, 并将系统时钟切换到 FRC 振荡器。之后, 用户可以选择尝试重新启动振荡器或执行受控关闭。用户可以把陷阱当作热复位来处理, 这只要把复位地址装入振荡器陷阱向量即可。出现故障时, 一旦识别出时钟故障, CF (时钟故障) 状态位 (OSCCON<3>) 也将置 1。

出现时钟故障时, WDT 不受影响, 继续靠 LPRC 时钟运行。

从 POR、BOR 或休眠退出后, 如果振荡器起振非常慢的话, 则可能出现这样的情况, 在振荡器启动之前 PWRT 定时器就已到期。在这样的情形下, FSCM 将被激活, 启动一个时钟故障陷阱, FRC 振荡器选择将装入 COSC<2:0> 位中。这实际上会关闭正在尝试启动的原始振荡器。

在时钟故障陷阱 ISR 中, 用户可以检测到这样的情况并重启振荡器。

检测到时钟故障时, FSCM 模块将发起至 FRC 振荡器的时钟切换, 方法如下:

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

出于时钟切换的目的, 时钟源划分为四类:

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

用户可以在这些功能类型之间进行切换, 但不能在同一个类型内进行切换。如果选择了主时钟源, 则在主时钟源内的选择始终由 FPR<4:0> 配置位决定。

与时钟切换相关的控制和状态位在 OSCCON 寄存器中。

- COSC<2:0>: 只读状态位, 总是反映当前有效的振荡器类型。
- NOSC<2:0>: 写控制位, 表明新选择的振荡器类型。
 - 在 POR 和 BOR 时, 将把配置位 FOS<2:0> 的值装入 COSC<2:0> 和 NOSC<2:0> 中。
- LOCK: LOCK 状态位, 表明 PLL 是否锁定。
- CF: 只读状态位, 表明是否检测到时钟故障。
- OSWEN: 控制位, 当时钟切换过程启动时, 从 0 变为 1。清零 OSWEN 控制位将中止正在进行的时钟切换 (用于挂起情形)。

如果配置位 FCKSM<1:0> = 1x, 那么时钟切换和故障保护时钟监视器功能被禁止。这是默认的配置位设定。

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

注: 当故障保护时钟监视器使能时, 应用程序不应试图切换到低于 100 kHz 频率的时钟。如果进行时钟切换的话, 器件可能产生一个振荡器故障陷阱并切换到快速 RC 振荡器。

20.2.8 防止意外写入 OSCCON

因为 OSCCON 寄存器控制时钟切换和时钟分频，所以有意将它的写入操作设计得很困难。

要写 OSCCON 的低字节，必须执行以下代码序列，不要在其中插入任何其他指令：

```
Byte Write 0x46 to OSCCON low
Byte Write 0x57 to OSCCON low
```

字节写可在一个指令周期内完成。写期望值或使用位操作指令。

要写 OSCCON 的高字节，必须执行以下代码序列，不要在其中插入任何其他指令：

```
Byte Write "0x78" to OSCCON high
Byte Write "0x9A" to OSCCON high
```

字节写可在一个指令周期内完成。写期望值或使用位操作指令。

20.3 复位

dsPIC30F 有以下几种不同类型的复位：

- a) 上电复位 (POR)
- b) 正常工作期间的 MCLR 复位
- c) 休眠期间的 MCLR 复位
- d) 看门狗定时器 (WDT) 复位 (正常工作期间)
- e) 可编程欠压复位 (BOR)
- f) RESET 指令
- g) 由陷阱锁定 (TRAPR) 导致的复位
- h) 由非法操作码导致的复位，或者是由于把未初始化的 W 寄存器用作地址指针 (IOPUWR) 而导致的复位

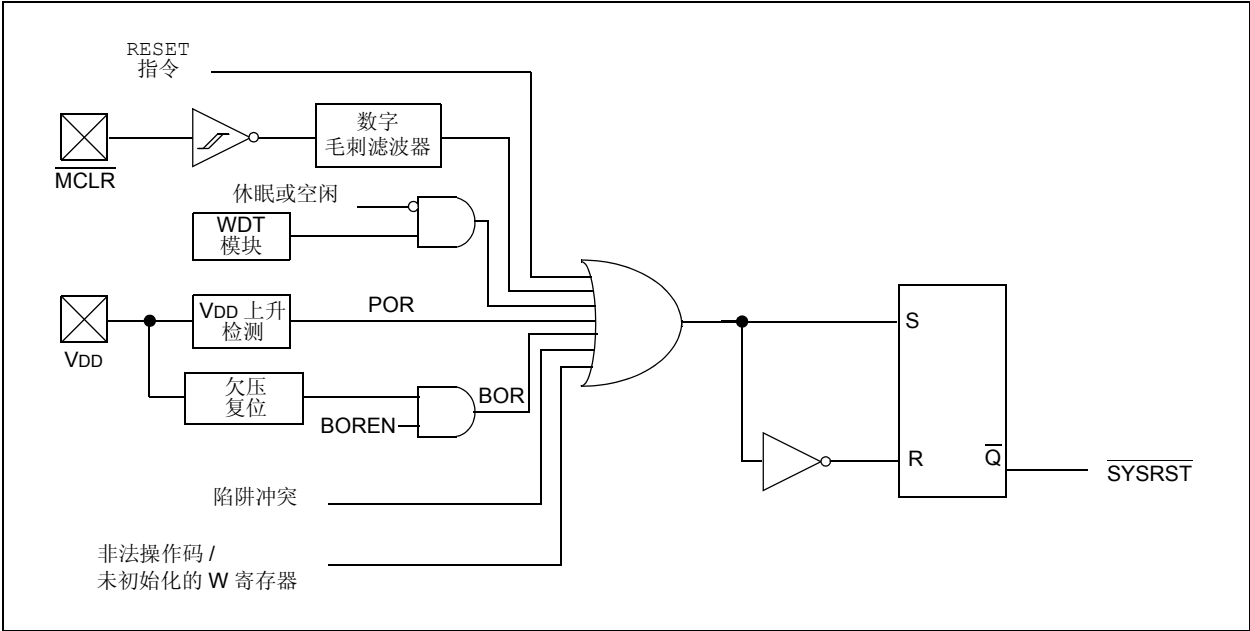
各种复位条件以不同的方式影响不同的寄存器。大多数寄存器不受 WDT 唤醒的影响，这是因为 WDT 唤醒被视为对正常工作的恢复。在不同的复位条件下，将以不同的方式置 1 或清零 RCON 寄存器中的状态位，如表 20-5 所示。可在软件中使用这些状态位判断复位的性质。

片上复位电路的框图，如图 20-2 所示。

在 MCLR 复位路径上，提供了一个 MCLR 噪声滤波器。该滤波器检测并滤除小脉冲。

内部产生的复位不会将 MCLR 引脚驱动为低电平。

图 20-2: 复位系统框图



dsPIC30F5015/5016

20.3.1 POR: 上电复位

当检测到VDD上升时，上电事件将产生内部POR脉冲。复位脉冲将在POR电路门限电压（VPOR）处产生，标称门限值是1.85V。器件电源电压特性曲线必须满足规定的起始电压和上升速度要求。POR脉冲将复位POR定时器，并使器件进入复位状态。POR还将选择器件时钟源，时钟源由振荡器配置熔丝位指定。

POR电路将插入一个标称值为10 μs的短暂延时TPOR，以确保器件偏压电路达到稳定。此外，还要施加用户选择的上电延时（TPWRT）。TPWRT参数由器件配置位设定，可以是0 ms（无延时）、4 ms、16 ms或64 ms。器件上电的总延时为TPOR + TPWRT。在这些延时结束后，SYSRST将在Q1时钟的下一个上升沿反相，PC将跳转到复位向量。

SYSRST信号时序如图20-3至图20-5所示。

图 20-3: 上电时的延时时序（MCLR 连接到 VDD）

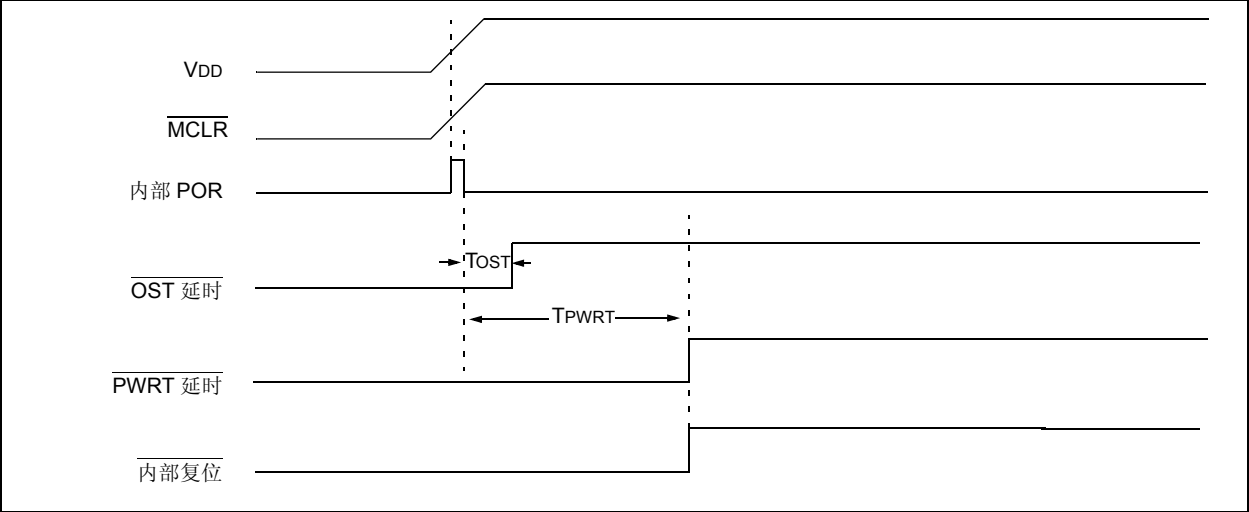


图 20-4: 上电时的延时时序（MCLR 不连接到 VDD）：情形 1

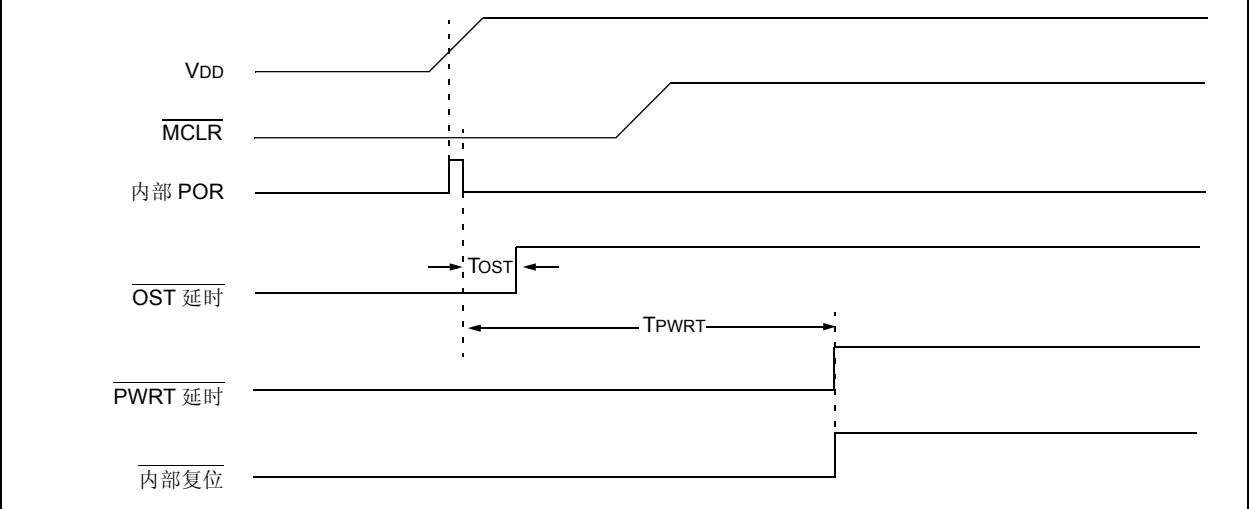
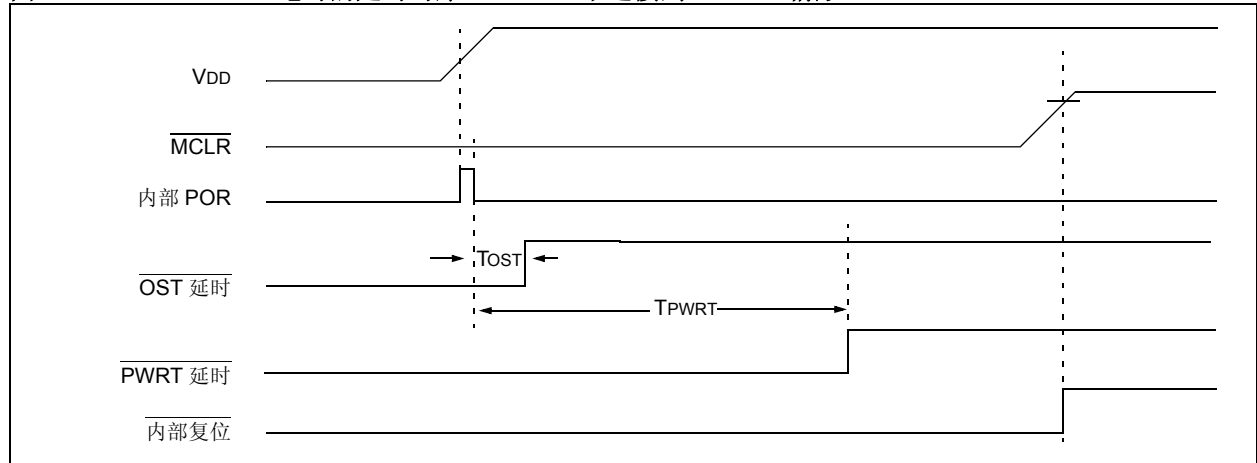


图 20-5: 上电时的延时时序 ($\overline{\text{MCLR}}$ 不连接到 VDD): 情形 2

20.3.1.1 具有长晶振起振时间的 POR (FSCM 使能)

振荡器起振电路没有连接到 POR 电路。某些晶振电路 (尤其是低频晶振) 的起振时间会相对较长。从而, 在 POR 定时器和 PWRT 超时后, 可能会出现下列情况:

- 振荡电路未起振。
- 振荡器起振定时器还没有超时 (如果使用晶振的话)。
- PLL 还未锁定 (如果使用了 PLL 的话)。

如果 FSCM 使能且出现上述条件之一时, 就将产生时钟故障陷阱。器件将自动切换到 FRC 振荡器, 而用户可在陷阱 ISR 中切换到期望的晶体振荡器。

20.3.1.2 FSCM 和 PWRT 禁止时的操作

如果 FSCM 禁止且上电延时定时器 (PWRT) 也禁止的话, 上电时器件将快速退出复位状态。如果时钟源是 FRC、LPRC、EXTRC 或 EC, 它将立即激活。

如果 FSCM 禁止而系统时钟还没有启动, 器件将在复位向量处处于冻结状态, 直到系统时钟启动为止。从用户的角度看, 器件看上去仍然在复位状态, 直到系统时钟可用为止。

20.3.2 BOR: 可编程欠压复位

BOR (欠压复位) 模块是基于内部参考电压电路的。BOR 模块的主要用途是在发生欠压条件时产生器件复位。欠压条件通常由交流电源线上的干扰信号 (即, 由于电源传输线路不良造成的交流周期波形丢失, 或者由于接入大负载时过电流造成电压下降) 产生的。

BOR 模块允许选择以下电压跳变点之一:

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

注: 这里所述的 BOR 电压跳变点都是标称值, 仅供设计参考。

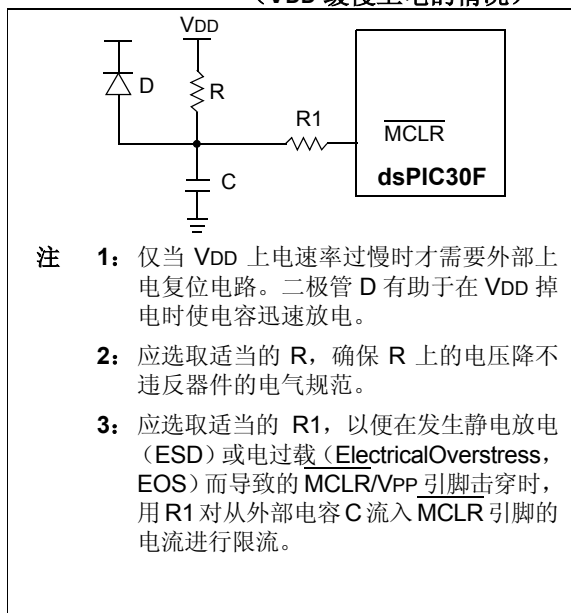
BOR 将产生复位器件的复位脉冲。BOR 会根据器件配置位 ($\text{FOS}<2:0>$ 和 $\text{FPR}<4:0>$) 的值选择时钟源。此外, 如果选择了振荡器模式, BOR 将激活振荡器起振定时器 (OST)。系统时钟将保持到 OST 超时。如果使用了 PLL, 则时钟将被保持到 LOCK 位 ($\text{OSCCON}<5>$) 置 1。

dsPIC30F5015/5016

同时，在内部复位信号释放之前，还要施加 POR 延时 (T_{POR}) 和 PWRT 延时 (T_{pwrt})。如果 $T_{pwrt} = 0$ 且正在使用的是晶体振荡器的话，那么将施加 $T_{fscm} = 100 \mu s$ 的标称延时。这样，总延时为 ($T_{POR} + T_{fscm}$)。

BOR 状态位 ($RCON<1>$) 将置 1，以表明发生了 BOR。使能时，BOR 电路将在休眠或空闲模式下继续工作，当 V_{DD} 下降到 BOR 门限电压以下时将复位器件。

图 20-6: 外部上电复位电路
(V_{DD} 缓慢上电的情况)



注: 专用监控器件，如 MCP1XX 和 MCP8XX 等，也可以用作外部上电复位电路。

表 20-5 所示为 RCON 寄存器的复位状态。由于 RCON 寄存器中的控制位都是可读写的，表中的信息意味着：表中的所有位状态，与在“条件”这一列指定动作之前的位状态刚好相反。

表 20-5: RCON 寄存器的初始化状态，情形 1

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	IDLE	SLEEP	POR	BOR
上电复位	0x000000	0	0	0	0	0	0	0	1	1
欠压复位	0x000000	0	0	0	0	0	0	0	0	1
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	0	0	0	0
正常工作期间的软件复位	0x000000	0	0	0	1	0	0	0	0	0
休眠期间的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	0	1	0	0
空闲期间的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	1	0	0	0
WDT 超时复位	0x000000	0	0	0	0	1	0	0	0	0
WDT 唤醒	PC + 2	0	0	0	0	1	0	1	0	0
通过中断从休眠状态唤醒	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

注 1: 如果是由于中断引发的唤醒，则把相应的中断向量装入 PC。

表 20-6 所示为 RCON 寄存器位状态的另一个例子。在此情形下，没有假设用户在“条件”这一列中指定动作之前已置 1/ 清零相应的位。

表 20-6: RCON 寄存器的初始化状态，情形 2

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	IDLE	SLEEP	POR	BOR
上电复位	0x000000	0	0	0	0	0	0	0	1	1
欠压复位	0x000000	u	u	u	u	u	u	u	0	1
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	0	0	0	0	u	u
正常工作期间的软件复位	0x000000	u	u	0	1	0	0	0	u	u
休眠期间的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	u	0	0	1	u	u
空闲期间的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	u	0	1	0	u	u
WDT 超时复位	0x000000	u	u	0	0	1	0	0	u	u
WDT 唤醒	PC + 2	u	u	u	u	1	u	1	u	u
通过中断从休眠状态唤醒	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 = 不变

注 1: 如果是由于中断引发的唤醒，则把相应的中断向量装入 PC。

20.4 看门狗定时器 (WDT)

20.4.1 看门狗定时器操作

看门狗定时器 (WDT) 的主要功能是在软件故障时使处理器复位。WDT 是独立运行的定时器，它使用片上 RC 振荡器，不需要外部元件。因此，即使主处理器时钟（例如晶振）出现故障，WDT 定时器仍将继续工作。

20.4.2 使能和禁止 WDT

看门狗定时器的使能或禁止，只能通过配置寄存器 FWDTE 中的配置位 (FWDTE) 来进行。

设置 FWDTE = 1，使能看门狗定时器。使能是在对器件进行编程时完成的。默认情况下，在擦除芯片后，FWDTE 位 = 1。任何能够对 dsPIC30F 器件进行编程的编程器，都能够对这个配置位及其他配置位进行编程。

如果使能的话，WDT 将递增，直到它溢出或“超时”。WDT 超时将强制器件复位（除非是在休眠期间）。要防止 WDT 超时，用户必须使用 CLRWDTE 指令清零看门狗定时器。

如果休眠期间 WDT 超时，器件将唤醒。RCON 寄存器中的 WDT0 位将清零，表明 WDT 超时导致唤醒。

设置 FWDTE = 0，允许用户软件通过 SWDTE (RCON<5>) 控制位来使能 / 禁止看门狗定时器。

20.5 省电模式

通过执行一条特殊的指令 PWRSAV，可以进入两个省电状态。

这两个模式是：休眠和空闲。

PWRSAV 指令的格式如下：

PWRSAV <parameter>，其中“parameter”定义是空闲还是休眠模式。

20.5.1 休眠模式

在休眠模式下，CPU 和外设的时钟关闭。如果在使用片上振荡器，它也将关闭。

休眠期间，故障保护时钟监视器不起作用，因为没有时钟需要监视。不过，如果休眠期间 WDT 工作的话，LPRC 时钟将保持为活动状态。

欠压保护电路，如果使能的话，在休眠期间仍然起作用。

至少发生以下事件之一时，处理器将从休眠模式唤醒：

- 任何中断，如果已单独允许并且满足所需的优先级
- 任何复位 (POR、BOR 和 MCLR)
- WDT 超时

从休眠模式唤醒时，处理器将重新启动进入休眠模式之前处于活动状态的时钟。当时钟切换使能时，COSC<2:0> 位将决定唤醒时所使用的振荡器源。如果时钟切换禁止，则只有一个系统时钟。

注： 如果发生 POR 或 BOR，则根据 FOS<2:0> 和 FPR<4:0> 配置位来选择振荡器。

如果时钟源是振荡器，器件的时钟将一直关闭，直到 OST 超时为止（这表明振荡器已经稳定）。如果使用了 PLL，系统时钟关闭，直到 LOCK = 1 为止（这表明 PLL 已经稳定）。无论是哪种情况，都要加上 Tpor、Tlock 和 Tpwrt 延时。

如果使用了 EC、FRC、LPRC 或 EXTRC 振荡器，那么要加上 Tpor 的延时 (~ 10 μs)。这是从休眠中唤醒时的最小延时。

此外，如果在休眠期间 LP 振荡器处于工作状态，并且 LP 是唤醒时使用的振荡器，那么起振延时等于 Tpor。不会有 PWRT 延时和 OST 定时器延时。为了使从休眠中唤醒的起振延时最小，在进入休眠之前，应该选择这些快速唤醒选项之一。

任何中断，如果被单独允许（通过相应的 IE 位）并且优先级占先的话，将唤醒处理器。处理器将处理中断，转移至 ISR。唤醒时 RCON 寄存器中的休眠状态位将置 1。

注： 尽管要加上各种延时 (Tpor、Tlock 和 Tpwrt)，在延时结束后，晶振（以及 PLL）可能还未处于工作状态（例如，低频晶振）。在这种情形下，如果 FSCM 使能，那么器件将检测到这是一个时钟故障，并随即处理时钟故障陷阱，这样 FRC 振荡器将使能，用户就不得不重新使能晶振。如果 FSCM 没有使能，那么器件将直接暂停代码的执行，直到时钟稳定为止，且将继续停留在休眠模式，直到振荡器时钟已经启动。

所有的复位都将把处理器从休眠模式中唤醒。除 POR 之外的任何复位，都将置 1 休眠状态位。POR 时，休眠状态位清零。

如果看门狗定时器使能，则在 WDT 超时时处理器将从休眠模式中唤醒。休眠和 WDT0 状态位都将置 1。

20.5.2 空闲模式

在空闲模式下，CPU 的时钟关闭，而外设继续运行。休眠模式不同的是，时钟源继续保持为活动状态。

对于一些外设而言，每个模块中有一个与外设对应的控制位，控制外设空闲期间是否工作。

如果时钟故障检测使能的话，LPRC 故障保护时钟继续保持为活动状态。

在发生以下事件之一时，处理器将从空闲模式唤醒：

- 任何中断，如果已允许（IE 位为 1）并且满足所需的优先级
- 任何复位（POR、BOR 或 MCLR）
- WDT 超时

在从空闲模式唤醒时，时钟再次供 CPU 使用且指令立即开始执行，从 PWRSAV 指令之后的第一条指令处开始。

任何被单独允许（使用相应的 IE 位）且优先级占先的中断都将唤醒处理器。处理器将处理中断，转移至 ISR。唤醒时 RCON 寄存器中的空闲状态位将置 1。

除 POR 之外的任何复位，都将把空闲状态位置 1。POR 时，空闲状态位清零。

如果看门狗定时器使能，则处理器在 WDT 超时时将从空闲模式中唤醒。空闲和 WDTO 状态位都将置 1。

与从休眠中唤醒不同的是，从空闲中唤醒时不存在延时。

20.6 器件配置寄存器

每个器件配置寄存器中的配置位指定某些器件模式，配置位通过器件编程器或者器件的在线串行编程（ICSP）功能来进行编程。虽然每个器件配置寄存器都是 24 位寄存器，但只有低 16 位可用来保存配置数据。有 4 个器件配置寄存器可供用户使用：

1. FOSC（0xF80000）：振荡器配置寄存器
2. FWDT（0xF80002）：看门狗定时器配置寄存器
3. FBORPOR（0xF80004）：BOR 和 POR 配置寄存器
4. FGS（0xF8000A）：通用代码段配置寄存器

如果在器件编程器中选择了器件，配置位的位置将被自动处理。可以在源代码（这取决于使用的语言工具）中或者通过编程接口来指定配置位的期望状态。在对器件编程之后，应用软件可以通过表读指令读取配置位的值。更多信息，请参见器件的编程规范。

注： 如果设置了代码保护配置熔丝位（FGS<GCP> 和 FGS<GWRP>），那么只有在电压 VDD ≥ 4.5V 时才可以擦除整个代码保护的器件。

20.7 外设模块禁止（PMD）寄存器

外设模块禁止（Peripheral Module Disable, PMD）寄存器提供了一种禁止外设模块的方法，即，停止提供给模块的所有时钟源。通过对应的 PMD 控制位禁止某个外设时，外设将处于最低功耗状态。与外设相关的控制和状态寄存器也会被禁止，所以读写这些寄存器的操作都无效。

外设模块使能的条件是：PMD 寄存器中相关位清零，且特定的 dsPIC DSC 器件支持该外设。如果器件中存在相应的外设模块，则默认情况下在 PMD 寄存器中使能外设。

注： 如果 PMD 位置 1，则对应的模块在一个指令周期的延时后被禁止。类似地，如果 PMD 位清零，对应的模块将在一个指令周期的延时后被使能（假定已经配置模块控制寄存器，使能了模块的操作）。

20.8 在线调试器

如果选择 MPLAB® ICD 2 作为调试器，则使能在线调试功能。该功能与 MPLAB IDE 配合使用时，可进行简单的调试功能。当使能器件的这个功能时，某些资源不能用于通常的用途。这些资源包括数据 RAM 的前 80 字节和两个 I/O 引脚。

用户使用 MPLAB IDE 中的配置选项，可选择 4 对调试 I/O 引脚中的一对。这些引脚为 EMUD/EMUC、EMUD1/EMUC1、EMUD2/EMUC2 和 MUD3/EMUC3。

无论选取哪对调试引脚，选取的 EMUD 引脚都是仿真 / 调试数据线，而 EMUC 引脚则是仿真 / 调试时钟线。这些引脚将与 Microchip 提供的 MPLAB ICD 2 模块接口。MPLAB ICD 2 使用选取的调试 I/O 引脚对，发送命令、接收响应，发送和接收数据。要使用器件的在线调试功能，应用设计必须实现至 MCLR、VDD、VSS、PGC、PGD 以及选取的 EMUDx/EMUCx 引脚对的 ICSP 连接。

这样就产生了两种可能性：

1. 如果选取 EMUD/EMUC 作为调试 I/O 引脚对，那么只需要一个 5 引脚的接口，这是因为，在所有的 dsPIC30F 器件上，EMUD 和 EMUC 引脚功能均可与 PGD 和 PGC 引脚功能复用。
2. 如果选取了 EMUD1/EMUC1、EMUD2/EMUC2 或 EMUD3/EMUC3 作为调试 I/O 引脚对，那么需要 7 引脚的接口，这是因为，EMUDx/EMUCx 引脚功能（x = 1、2 或 3）不能与 PGD 和 PGC 引脚功能复用。

表 20-7: dsPIC30F5015/5016 的系统集成寄存器映射

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
RCON	0740	TRAPR	IOPUWR	BGST	—	—	—	—	—	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	取决于复位类型。
OSCCON	0742	—	COSC<2:0>		—	—	NOSC<2:0>		—	POST<1:0>	LOCK	—	—	CF	—	LPOSCEN	OSWEN	取决于配置位。
OSCTUN	0744	—	—	—	—	—	—	—	—	—	—	—	TUN<4:0>					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: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 20-8: 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		—	—	—	PWMMPIN	HPOL	LPOL	BOREN	—	BORV<1:0>		—	—	FPWRT<1:0>	
FGS	F8000A	—	—		—	—	—	—	—	—	—	—	—	—	—	GCP	GWRP	

图注: u = 未初始化位

注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

注:

21.0 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。关于 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) 能够把模拟输入信号转换为 10 位数字信号。模块是基于逐次逼近寄存器 (Successive Approximation Register, SAR) 结构的，最大采样速率为 1 Msps。ADC 模块有多达 16 路模拟输入，经过多路开关连接到 4 个采样保持放大器。采样保持放大器的输出是转换器的输入，由转换器产生结果。可以用软件选择器件电源电压 (AVDD/AVSS)，或 (VREF+/VREF-) 引脚电压作为模拟参考电压。ADC 有一个独特之处，即它在器件处于休眠模式期间也能工作。

ADC 模块有 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)

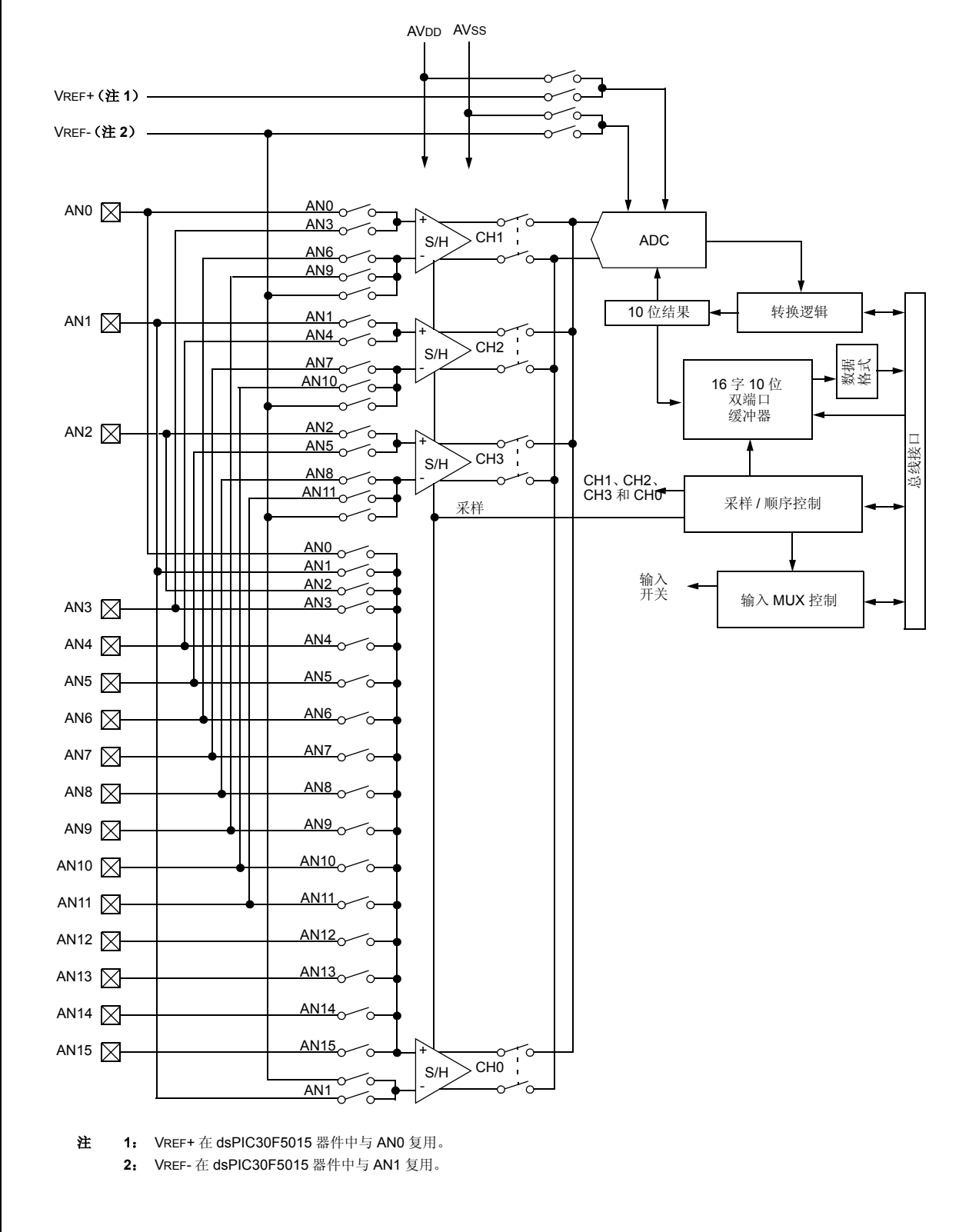
ADCON1、ADCON2 和 ADCON3 寄存器，控制 ADC 模块的工作。ADCHS 寄存器选择转换的输入通道。ADPCFG 寄存器配置端口引脚，使端口引脚用作模拟输入或数字 I/O。ADCSSL 寄存器选择扫描的输入。

注：当 ADON = 1 时，不要写 SSRC<2:0>、ASAM、SMPI<3:0>、BUFM 和 ALTS 位，也不要写 ADCON3 和 ADCSSL 寄存器。因为这将导致不确定的结果。

图 21-1 给出了 A/D 模块的框图。

dsPIC30F5015/5016

图 21-1: 10 位高速 A/D 功能框图



21.1 ADC 结果缓冲器

该模块有一个 16 字双端口只读缓冲器，用于缓存 A/D 转换结果，它们分别称为 ADCBUF0、ADCBUF1、...、ADCBUF6 和 ADCBUFF。RAM 为 10 位宽，但被读取时，其读取值为不同格式的 16 位字。用户软件不能写这 16 个 ADC 结果缓冲寄存器（ADCBUF0 到 ADCBUFF）。

21.2 转换过程

配置好 ADC 模块之后，把 SAMP 位置 1 就将启动采样。各种触发源，例如可编程的位、定时器超时以及外部事件等，均可终止采样并启动转换。当 A/D 转换完成时，转换结果将存放到 ADCBUF0 至 ADCBUFF 中，在进行了一定数量的采样后（采样数量由 SMPI 位指定），A/D 中断标志位 ADIF 和 DONE 位将置 1。

进行 A/D 转换时，应遵循以下步骤：

1. 配置 ADC 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（如需要）：
 - 清零 ADIF 位
 - 选择 A/D 中断优先级
3. 启动采样
4. 等待一段必要的采样时间。
5. 触发采样结束，启动转换
6. 等待 A/D 转换完成，可通过以下两种方法之一来判断转换是否完成：
 - 等待 A/D 中断
 - 等待 DONE 位置 1
7. 读 A/D 结果缓冲器，必要时清零 ADIF。

21.3 选择转换序列

有几组控制位用来选择时序，时序包括 A/D 把输入连接到采样 / 保持通道、对通道进行转换、写缓冲存储器并产生中断。该时序由采样时钟控制。

SIMSAM 位控制多个通道的采样 / 转换时序。如果 SIMSAM 位为 0，将用两个或四个采样时钟对两个或四个选定通道采样并按顺序转换。如果 SIMSAM 位为 1，将用一个采样时钟对两个或四个选定通道同时采样。随后将按顺序转换通道。很明显，如果只选中了 1 个通道，SIMSAM 位不适用。

CHPS 位选择采样的通道数。范围为 1、2 或 4 个通道。如果 CHPS 选择 1 个通道，将在采样时钟周期内采样并转换 CH0 通道。结果存储在缓冲器中。如果 CHPS 选择 2 个通道，将采样并转换 CH0 和 CH1 通道。如果 CHPS 选择 4 个通道，将采样并转换 CH0、CH1、CH2 和 CH3 通道。

SMPI 位选择采集 / 转换几次后产生中断。可以选择每次中断采样 1 次到每次中断采样 16 次。

用户对 CHPS 和 SMPI 位进行组合设定时，无法指定每次中断采样超过 16 次或 8 次（取决于 BUFM 位）。BUFM 位置 1 时，16 字的结果缓冲器（ADCBUF0...ADCBUFF）被拆分成两个 8 字的组。每次中断事件发生时，轮流写 8 字的缓冲器。BUFM 位的使用取决于中断后有多少时间可用于将数据移出缓冲器，这取决于应用程序。

如果处理器能够迅速清空一个满的缓冲器，并且用时少于其进行一个通道的采集和转换的时间，则 BUFM 位可以是 0，且每次中断可以进行最多 16 次转换。处理器将只有一个采样和转换时间，用来传送 16 个转换结果。

如果在采样转换时间之内，处理器不能清空缓冲器，那么 BUFM 应该为 1。例如，如果 $SMPI<3:0> = 0111$ ，则 8 个转换结果将装入到一半缓冲器中，随即产生中断。后续 8 个转换结果将装入到缓冲器的另一半中去。在两次中断之间，处理器有完整的时间来传送 8 个转换结果。

ALTS 位用来在采样序列期间轮换所选择的输入。输入多路开关有两组采样输入：MUX A 和 MUX B。如果 ALTS 位为 0，只选择 MUX A 输入进行采样。如果 ALTS 位为 1 而且在第一个采样/转换序列中 $SMPI<3:0> = 0000$ ，那么先选择 MUX A 输入，在下一个采样 / 转换序列中，选择 MUX B 输入。

对于 MUX A 组中所选数量的模拟输入而言，CSCNA 位（ $ADCON2<10>$ ）将允许 CH0 通道输入被轮流扫描。使用 ADCSSL 寄存器选择输入。如果 ADCSSL 寄存器中某一位是 1，则相应的输入被选取。每次中断之后，总是按照输入编号从低到高的顺序，开始扫描输入。如果所选输入的数量，大于每次中断的采样数，将不使用编号大于每次中断采样数的输入。

21.4 设置转换触发源启动

转换触发源将终止采样，启动请求的转换。

SSRC<2:0> 位选择转换触发源。

SSRC 位提供了最多 5 个备用转换触发源。

当 SSRC<2:0> = 000 时，转换触发源由软件控制。清零 SAMP 位，将引发转换触发事件。

当 SSRC<2:0> = 111（自动开始模式）时，转换触发源由 A/D 时钟控制。SAMC 位选择采样开始到转换开始之间的 A/D 时钟数。对于多通道转换，这提供了最快的转换速率。SAMC 必须至少为 1 个时钟周期。

其他触发源可以来自定时器模块、电机控制 PWM 模块或外部中断。

注： 要让 ADC 在最大指定转换速度下工作，应选择“自动转换触发器”选项（SSRC = 111），自动采样时间位应设置为 1 TAD（SAMC = 00001）。该配置将提供 13 TAD 的总转换周期（采样 + 转换）。

使用任何其他转换触发器都将导致更多的 TAD 周期，以便将外部事件与 ADC 同步。

21.5 中止转换

转换期间清零 ADON 位，将中止当前转换，并停止采样序列。不会使用部分完成的 A/D 转换采样来更新 ADCBUF。此即，ADCBUF 将继续保留上次完成的转换值（或最后一次写入 ADCBUF 寄存器的值）。

如果 ADON 位的清零与自动开始同时发生，清零有更高的优先级。

中止 A/D 转换后需要 2 TAD 的等待时间，才能置 1 SAMP 位开始下次采样。

如果指定顺序采样，A/D 将在收到响应下个被转换通道的采样脉冲时继续。如果指定同步采样，A/D 将以下个多通道组转换序列继续。

21.6 选择 A/D 转换时钟

ADC 转换需要 12 TAD。可通过软件、使用 6 位计数器来选择 A/D 转换时钟源。TAD 有 64 个可能的选择。

公式 21-1: ADC 转换时钟

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

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

将 ADRC 位置 1，选择内部 RC 振荡器。

为使 A/D 转换正确，A/D 转换时钟（TAD）的选择，必须确保最小 TAD 为 83.33 ns（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 21-1 举例说明如何进行 ADCS<5:0> 位的计算，假设器件工作速度是 30 MIPS。

例 21-1: A/D 转换时钟计算

$$TAD = 84 \text{ ns}$$

$$T_{CY} = 33 \text{ ns (30 MIPS)}$$

$$\begin{aligned} ADCS<5:0> &= 2 \frac{TAD}{T_{CY}} - 1 \\ &= 2 \frac{84 \text{ ns}}{33 \text{ ns}} - 1 \\ &= 4.09 \end{aligned}$$

因此，

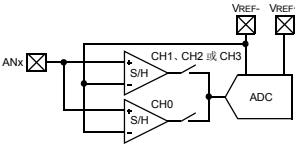
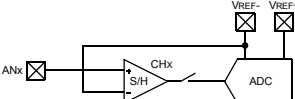
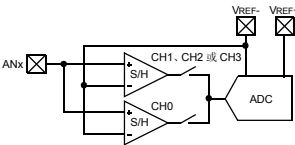
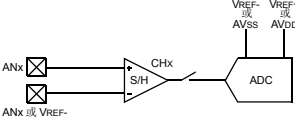
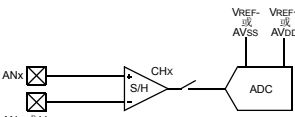
$$\text{取 } ADCS<5:0> = 5$$

$$\begin{aligned} \text{Actual TAD} &= \frac{T_{CY}}{2} (ADCS<5:0> + 1) \\ &= \frac{33 \text{ ns}}{2} (9 + 1) \\ &= 99 \text{ ns} \end{aligned}$$

21.7 ADC 速度

dsPIC30F 10 位 ADC 规范允许最大允许采样速率为 1 Msps。表 21-1 汇总了 dsPIC30F 10 位 ADC 的转换速率及要求的工作条件。

表 21-1: 10 位转换速率参数

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

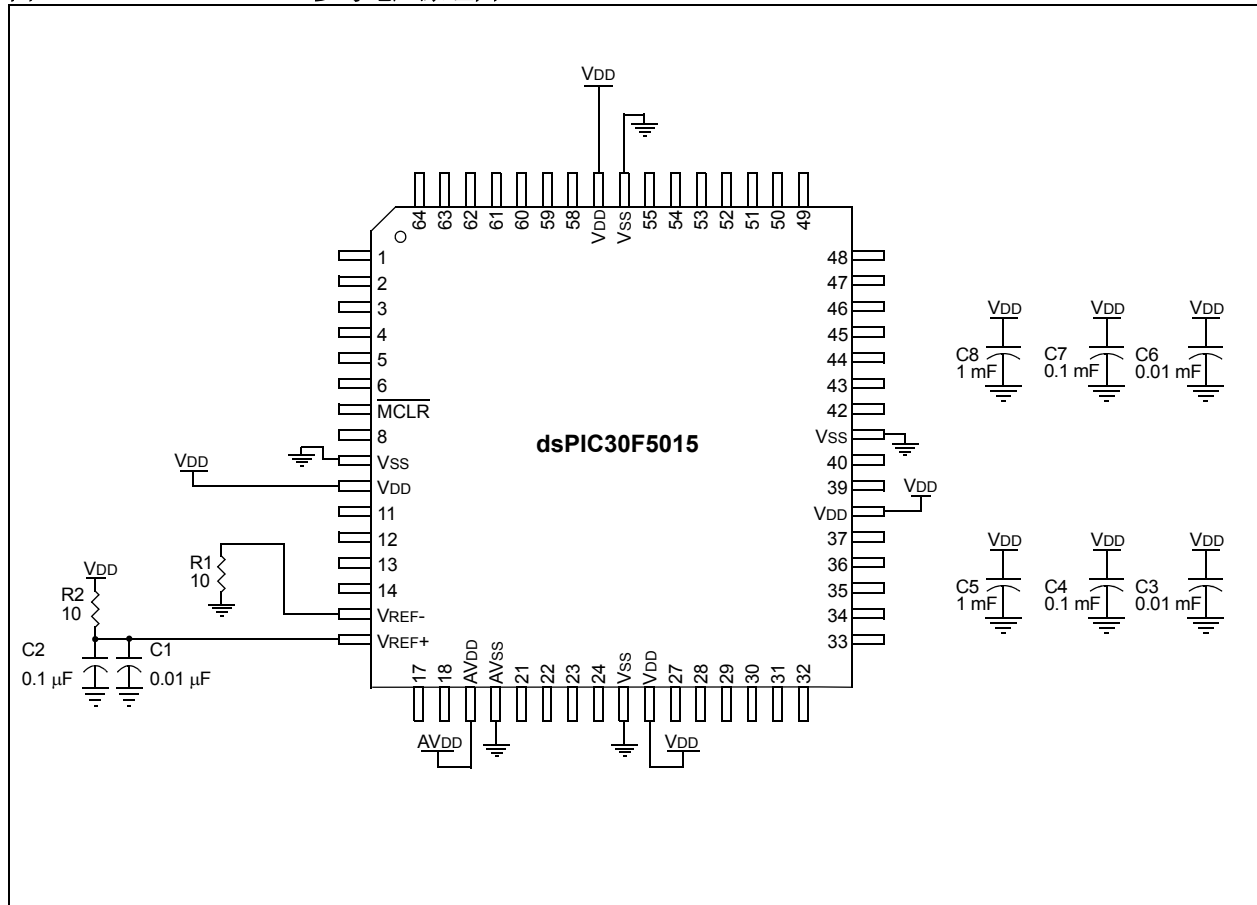
注 1: 为了正确工作，必须使用 VREF- 和 VREF+ 引脚。建议电路见图 21-2。

dsPIC30F5015/5016

配置指导原则给出了超过500 ksps的转换速率所需的设置值，因为它们需要使用外部 VREF 引脚，配置步骤也有所不同。此处省略了对转换速率来说不太重要的配置细节。

转换速度超过 500 ksps 的建议电路如下图所示。

图 21-2: ADC 参考电压原理图



21.7.1 1 Msps 配置指导原则

1 Msps 操作的配置取决于是对单输入引脚进行采样，还是对多引脚进行采样。

21.7.1.1 单模拟输入

对单模拟输入的 1 Msps 转换，必须使能至少两个采样保持（S/H）通道。必须配置模拟输入复用器，使同一输入引脚连接到采样保持通道。ADC 转换 S/H 通道上保持的值，同时第二个 S/H 通道采集新的输入样本。

21.7.1.2 多模拟输入

ADC 也可用于通过多个采样保持通道对多个模拟输入采样。在这种情况下，总共 1 Msps 的转换速率在不同输入信号间分配。例如，4 个输入可按每个信号 250 ksps 的速率采样，或者两个输入按每个信号 500 ksps 的速率采样。此配置中必须使用顺序采样，以保证每个输入都有足够的采样时间。

21.7.1.3 1 Msps 配置项目

为了获得 1 Msps 的转换速率，需要进行如下配置：

- 符合表 21-2 中所述条件
- 根据图 21-2 所示的建议电路，连接外部 VREF+ 和 VREF- 引脚
- 设置 ADCON1 寄存器中的 SSRC<2:0> = 111，使能自动转换选择
- 把 ADCON1 寄存器中的 ASAM 控制位置 1，使能自动采样
- 把 ADCON1 寄存器中的 SIMSAM 位清零，使能顺序采样
- 写 ADCON2 寄存器中的 CHPS<1:0> 控制位，使能至少两个采样保持通道
- 写 ADCON2 寄存器中的 SMPI<3:0> 控制位，设置写入所需的中断之间的转换次数至少设置 SMPI<3:0> = 0001，因为至少应使能两个采样保持通道
- 写 ADCON3 寄存器中的 ADCS<5:0> 控制位，将 A/D 时钟周期配置为：

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

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

21.7.2 750 ksps 配置指南

为了获得 750 ksps 的转换速率，需要进行如下配置：此配置假设对单模拟输入采样。

- 符合表 21-2 中所述条件
- 根据图 21-2 所示的建议电路，连接外部 VREF+ 和 VREF- 引脚
- 设置 ADCON1 寄存器中的 SSRC<2:0> = 111，使能自动转换选择
- 把 ADCON1 寄存器中的 ASAM 控制位置 1，使能自动采样
- 设置 ADCON2 寄存器中的 CHPS<1:0> = 00，使能一个采样保持通道
- 写 ADCON2 寄存器中的 SMPI<3:0> 控制位，设置写入所需的中断之间的转换次数
- 写 ADCON3 寄存器中的 ADCS<5:0> 控制位，将 A/D 时钟周期配置为：

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

- 写 SAMC<4:0> = 00010，将采样时间配置为 2 TAD

21.7.3 600 ksps 配置指南

600 ksps 操作的配置取决于是对单输入引脚进行采样，还是对多引脚进行采样。

21.7.3.1 单模拟输入

对单模拟输入的 600 ksps 转换，必须使能至少两个采样保持通道。必须配置模拟输入复用器，使同一输入引脚连接到采样保持通道。ADC 转换 S/H 通道上保持的值，同时第二个 S/H 通道采集新的输入样本。

21.7.3.2 多模拟输入

ADC 也可用于通过多个采样保持通道对多个模拟输入采样。在这种情况下，总共 600 ksps 的转换速率在不同输入信号间分配。例如，四个输入可按每个信号 150 ksps 的速率采样，或者两个输入按每个信号 300 ksps 的速率采样。此配置中必须使用顺序采样，保证每个输入都有足够的采样时间。

21.7.3.3 600 ksps 配置项目

为了获得 600 ksps 的转换速率，需要进行如下配置：

- 符合表 21-2 中所述条件
- 根据图 21-2 所示的建议电路，连接外部 VREF+ 和 VREF- 引脚
- 设置 ADCON1 寄存器中的 SSRC<2:0> = 111，使能自动转换选择
- 把 ADCON1 寄存器中的 ASAM 控制位置 1，使能自动采样
- 把 ADCON1 寄存器中的 SIMSAM 位清零，使能顺序采样
- 写 ADCON2 寄存器中的 CHPS<1:0> 控制位，使能至少两个采样保持通道
- 写 ADCON2 寄存器中的 SMPI<3:0> 控制位，设置写入所需的中断之间的转换次数至少设置 SMPI<3:0> = 0001，因为至少应使能两个采样保持通道
- 写 ADCON3 寄存器中的 ADCS<5:0> 控制位，将 A/D 时钟周期配置为：

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

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

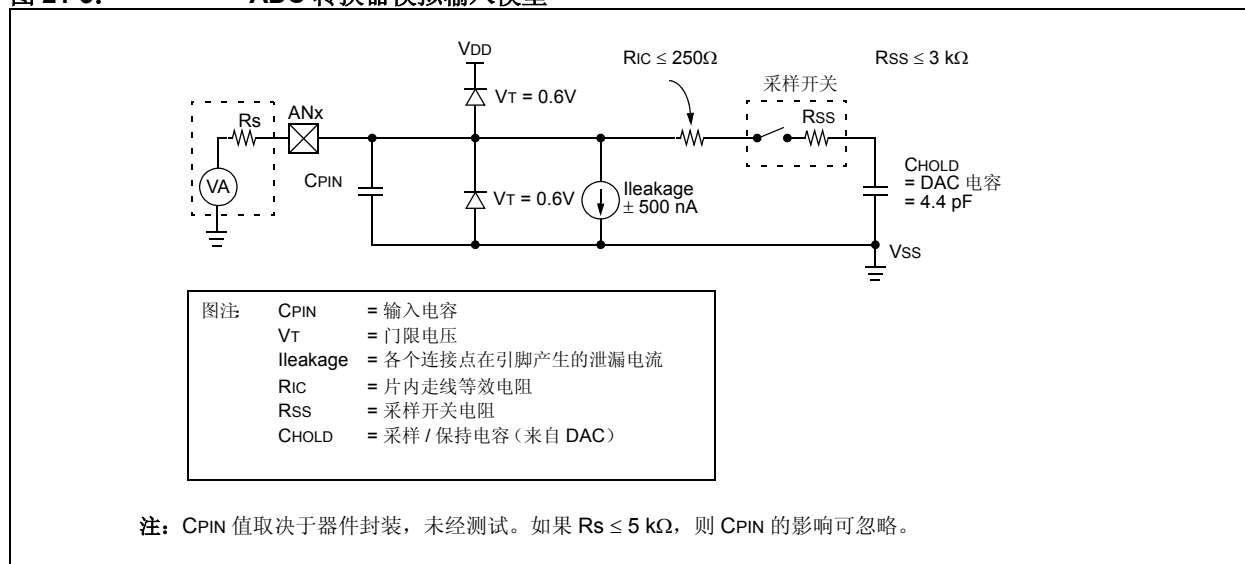
21.8 ADC 采集要求

10 位 ADC 的模拟输入模型如图 21-3 所示。ADC 的总采样时间由内部放大器稳定时间、器件 V_{DD} 和保持电容充电时间决定。

为了使 ADC 达到其规定精度，必须使充电保持电容（CHOLD）充满电至模拟输入引脚上的电平。模拟输出源阻抗（ R_s ）、片内走线等效电阻（ R_{ic} ）和内部采样开关阻抗（ R_{ss} ），共同直接影响着电容 CHOLD 充电所需的时间。所以总阻抗必须足够小，从而在选择的采样时间内能够对保持电容充分充电。为了把引脚泄漏电流对 ADC 转换器精度的影响降到最低，最大建议源阻抗 R_s 为 $5\text{ k}\Omega$ （对最高 500 ksp/s 的转换速率）和 500Ω （对最高 1 Msps 的转换速率）。在选择（更改）模拟输入通道之后，该采样功能必须在启动转换之前完成。在每次采样操作之前，内部保持电容将处于放电状态。

用户必须允许转换之间有至少 1 TAD 的采样时间 T_{SAMP} ，保证每个样本都被采集。该采样时间可通过置 1/ 清零 SAMP 位在软件中手动控制，也可以由 ADC 自动控制。在自动配置中，用户必须允许转换触发之间有足够的的时间，以满足最小采样时间的要求。TAD 和采样时间的要求请参见表 24-40。

图 21-3: ADC 转换器模拟输入模型



21.9 模块的掉电模式

模块有 3 种内部功耗模式。当 ADON 位为 1 时，模块处于工作模式；模块充分供电，正常工作。当 ADON 为 0 时，模块处于关闭模式。电路的数字和模拟部分被禁用，以最大程度地省电。为了从关闭模式返回到工作模式，用户必须等待 ADC 电路稳定下来。

21.10 CPU 休眠和空闲模式期间的 ADC 操作

21.10.1 CPU 休眠模式期间的 ADC 操作

当器件进入休眠模式时，模块的所有时钟源被关闭并保持为逻辑 0。

如果在转换过程中进入休眠状态，转换会中止。从休眠模式退出时，转换器将不会恢复部分完成的转换。

器件进入或退出休眠模式不会影响寄存器的内容。

如果将 ADC 时钟源设置成 RC (ADRC = 1)，ADC 模块可以在休眠模式下工作。在选择了 RC 时钟源时，ADC 模块在启动转换之前，要先等待一个指令周期。这将使 SLEEP 指令得以执行，从而从转换中消除了所有的数字开关噪声。转换完成后，DONE 位将置 1，转换结果送入 ADCBUF 寄存器。

如果允许了 A/D 中断，器件将从休眠中唤醒。如果未允许 A/D 中断，则即使 ADON 位保持置 1，A/D 模块也将被关闭。

21.10.2 CPU 空闲模式期间的 ADC 操作

ADSIDL 位选择 A/D 模块在空闲模式下是停止还是继续工作。如果 ADSIDL = 0，则模块将在空闲模式时继续工作。如果 ADSIDL = 1，则模块将在空闲模式下停止工作。

21.11 复位的影响

器件复位将强制所有寄存器进入复位状态。这将迫使 ADC 模块关闭，中止任何正在进行的转换和采样序列。ADCBUF 寄存器中的值不会改变。上电复位后，ADC 转换结果寄存器中的数据未知。

21.12 输出格式

ADC 转换结果是 10 位宽的，数据缓冲区 RAM 也是 10 位宽的。10 位数据可以用以下四种格式之一读取。FORM<1:0> 位选择格式。每种输出格式都将转化为数据总线上的 16 位结果。

写数据总是右对齐的（整数）格式。

图 21-4: ADC 输出数据格式

RAM 内容:										d09	d08	d07	d06	d05	d04	d03	d02	d01	d00						
读至总线:																									
有符号小数 （1.15）										$\overline{\text{d09}}$	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
小数 （1.15）										d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
有符号整数										$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	$\overline{\text{d09}}$	d08	d07	d06	d05	d04	d03	d02	d01	d00
整数										0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00

21.13 配置模拟端口引脚

使用 ADPCFG 和 TRIS 寄存器控制 ADC 端口引脚的工作。如果要将端口引脚用作模拟输入，则对应的 TRIS 位必须置 1（输入）。如果将 TRIS 位清零（输出），则数字输出电平（V_{OH} 或 V_{OL}）将被转换。

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

在读取端口寄存器时，所有配置为模拟输入通道的引脚均读为 0。

配置为数字输入的引脚，不会被当作模拟输入进行转换。对于任何定义为数字输入的引脚（包括 AN_x 引脚），加在引脚上的模拟电压可能导致输入缓冲器消耗的电流超出器件规范。

21.14 连接注意事项

模拟输入在 V_{DD} 和 V_{SS} 之间连接有二极管进行 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向偏置，而且如果超过输入电流规范可能会损坏器件。

有时可外接 RC 滤波器来进行输入信号的抗混叠滤波。R 元件的选择要确保满足采样时间要求。应保证使任何通过高阻抗连接到模拟输入引脚上的外部元件（如电容和齐纳二极管等）都只有很小的引脚泄漏电流。

表 21-2: 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	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF1	0282	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF2	0284	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF3	0286	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF4	0288	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF5	028A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF6	028C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF7	028E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF8	0290	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUF9	0292	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUFA	0294	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUFB	0296	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUFC	0298	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUFD	029A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUFE	029C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCBUFF	029E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 00uu uuuu uuuu
ADCON1	02A0	ADON	—	ADSIDL	—	—	—	FORM<1:0>	—	SSRC<2:0>	—	—	—	SIMSAM	ASAM	SAMP	DONE	0000 0000 0000 0000
ADCON2	02A2	—	—	—	—	—	—	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	—	—	CH0SB<3:0>	—	CH123NA<1:0>	—	CH123SA	CH0NA	—	—	—	—	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

u = 未初始化位

图注: 寄存器中各位的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F5015/5016

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。关于 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)。关于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 指令集增强、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 8 位的操作码及一个或多个操作数；操作码表明指令类型，操作数进一步指定指令的操作。

指令集是高度正交的，分为 5 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，不带任何地址修改量
- 第二个源操作数通常是寄存器 Ws，带或不带地址修改量
- 运算结果的目标寄存器，通常是寄存器 Wd，带或不带地址修改量

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
- 目标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）均有两个操作数：

- W 寄存器（带或不带地址修改量）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，不带任何地址修改量
- 第二个源操作数是立即数
- 操作结果的目标寄存器（仅在与第一个源操作数不同时）通常是寄存器 Wd（带或不带地址修改量）

MAC 类 DSP 指令可使用下列操作数：

- 要使用的累加器（A 或 B）（必需的操作数）
- 要用作两个操作数的 W 寄存器
- X 和 Y 地址空间预取操作
- X 和 Y 地址空间预取目标
- 累加器回写目标

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（必需）
- 源操作数或目标操作数（分别由 Wso 或 Wdo 指定），带或不带地址修改量
- 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。如果第二个字被其自身作为一条指令执行，则它将作为一条 NOP 指令执行。

dsPIC30F5015/5016

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改变了程序计数器。对于上述两种特殊情况，指令执行需要两个指令周期，在第二个指令周期中执行一条 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： 关于指令集的更多详细信息，请参见《dsPIC30F/33F 程序员参考手册》（DS70157B-CN）。

表 22-1: 操作码说明中使用的符号

字段	说明
#text	由 text 定义的立即数
(text)	text 的内容
[text]	由 text 寻址的存储单元
{ }	可选字段或操作
<n:m>	寄存器位域
.b	字节模式选择
.d	双字模式选择
.s	影子寄存器选择
.w	字模式选择（默认）
Acc	两个累加器 {A, B} 中的一个
AWB	累加器回写目标地址寄存器，目标地址寄存器 $\in \{W13, [W13] + = 2\}$
bit4	4 位宽位选择字段（用在字寻址指令中） $\in \{0...15\}$
C, DC, N, OV, Z	MCU 状态位：进位、半进位、负、溢出和零标志位
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 $\in \{0..0.31\}$
lit8	8 位无符号立即数 $\in \{0..0.255\}$
lit10	10 位无符号立即数，字节模式时 $\in \{0...255\}$ ，字模式时 $\in \{0:1023\}$
lit14	14 位无符号立即数 $\in \{0..0.16384\}$
lit16	16 位无符号立即数 $\in \{0..0.65535\}$
lit23	23 位无符号立即数 $\in \{0...8388608\}$ ；LSB 必须为 0
无	字段不需要项，可以为空白
OA, OB, SA, SB	DSP 状态位：ACCA 溢出、ACCB 溢出、ACCA 饱和和 ACCB 饱和
PC	程序计数器
Slit10	10 位有符号立即数 $\in \{-512...511\}$
Slit16	16 位有符号立即数 $\in \{-32768..0.32767\}$
Slit6	6 位有符号立即数 $\in \{-16..0.16\}$

表 22-1: 操作码说明中使用的符号 (续)

字段	说明
Wb	基本 W 寄存器 $\in \{W0..W15\}$
Wd	目标 W 寄存器 $\in \{Wd, [Wd], [Wd++] , [Wd--], [++Wd], [--Wd]\}$
Wdo	目标 W 寄存器 $\in \{Wnd, [Wnd], [Wnd++] , [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb]\}$
Wm, Wn	被除数、除数工作寄存器对 (直接寻址)
Wm*Wm	平方指令的被乘数和乘数工作寄存器对 $\in \{W4*W4, W5*W5, W6*W6, W7*W7\}$
Wm*Wn	DSP 指令的被乘数和乘数工作寄存器对 $\in \{W4*W5, W4*W6, W4*W7, W5*W6, W5*W7, W6*W7\}$
Wn	16 个工作寄存器中的一个 $\in \{W0..W15\}$
Wnd	16 个目标工作寄存器中的一个 $\in \{W0..W15\}$
Wns	16 个源工作寄存器中的一个 $\in \{W0..W15\}$
WREG	W0 (用于文件寄存器指令中的工作寄存器)
Ws	源 W 寄存器 $\in \{Ws, [Ws], [Ws++] , [Ws--], [++Ws], [--Ws]\}$
Wso	源 W 寄存器 $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb]\}$
Wx	DSP 指令的 X 数据空间预取地址寄存器 $\in \{[W8]+ = 6, [W8]+ = 4, [W8]+ = 2, [W8], [W8]- = 6, [W8]- = 4, [W8]- = 2, [W9]+ = 6, [W9]+ = 4, [W9]+ = 2, [W9], [W9]- = 6, [W9]- = 4, [W9]- = 2, [W9+W12], \text{none}\}$
Wxd	DSP 指令的 X 数据空间预取目标寄存器 $\in \{W4..W7\}$
Wy	DSP 指令的 Y 数据空间预取地址寄存器 $\in \{[W10]+ = 6, [W10]+ = 4, [W10]+ = 2, [W10], [W10]- = 6, [W10]- = 4, [W10]- = 2, [W11]+ = 6, [W11]+ = 4, [W11]+ = 2, [W11], [W11]- = 6, [W11]- = 4, [W11]- = 2, [W11+W12], \text{none}\}$
Wyd	DSP 指令的 Y 数据空间预取目标寄存器 $\in \{W4..W7\}$

dsPIC30F5015/5016

表 22-2: 指令集概述

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
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	$Wd = Wb + Ws$	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	$Wnd =$ 算术右移 Wb, 移位位数为 Wns	1	1	N,Z
		ASR Wb, #lit5, Wnd	$Wnd =$ 算术右移 Wb, 移位位数为 lit5	1	1	N,Z
5	BCLR	BCLR f, #bit4	位清零 f	1	1	无
		BCLR Ws, #bit4	位清零 Ws	1	1	无
6	BRA	BRA C, Expr	如果有进位, 则转移	1	1 (2)	无
		BRA GE, Expr	如果大于或等于, 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	如果没有进位, 则转移	1	1 (2)	无
		BRA NN, Expr	如果不为负, 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	如果不为零, 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 位置 1	1	1	无
		BSET Ws, #bit4	将 Ws 位置 1	1	1	无
8	BSW	BSW.C Ws, Wb	把 C 位写入 Ws<Wb>	1	1	无
		BSW.Z Ws, Wb	把 Z 位写入 Ws<Wb>	1	1	无
9	BTG	BTG f, #bit4	将 f 位翻转	1	1	无
		BTG Ws, #bit4	WS 位翻转	1	1	无
10	BTSC	BTSC f, #bit4	位测试 f, 如果为零则跳过	1	1 (2 或 3)	无
		BTSC Ws, #bit4	位测试 Ws, 如果为零则跳过	1	1 (2 或 3)	无

表 22-2: 指令集概述 (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	位测试 f, 如果为 1 则跳过	1	1 (2 或 3)	无
		BTSS Ws, #bit4	位测试 Ws, 如果为 1 则跳过	1	1 (2 或 3)	无
12	BTST	BTST f, #bit4	位测试 f	1	1	Z
		BTST.C Ws, #bit4	位测试 Ws, 并将被测试位拷贝至 C	1	1	C
		BTST.Z Ws, #bit4	位测试 Ws, 并将被测试位的反码拷贝至 Z	1	1	Z
		BTST.C Ws, Wb	位测试 Ws<Wb>, 并将被测试位拷贝至 C	1	1	C
		BTST.Z Ws, Wb	位测试 Ws<Wb>, 并将被测试位的反码拷贝至 Z	1	1	Z
13	BTSTS	BTSTS f, #bit4	位测试, 然后将 f 置 1	1	1	Z
		BTSTS. Ws, #bit4 C	位测试 Ws, 将被测试位拷贝至 C, 随后 将被测试位置 1	1	1	C
		BTSTS. Ws, #bit4 Z	位测试 Ws, 将被测试位的反码拷贝至 Z, 随后将被测试位置 1	1	1	Z
14	CALL	CALL lit23	调用子程序	2	2	无
		CALL Wn	间接调用子程序	1	2	无
15	CLR	CLR f	f = 0x0000	1	1	无
		CLR WREG	WREG = 0x0000	1	1	无
		CLR Ws	Ws = 0x0000	1	1	无
		CLR Acc, Wx, Wxd, Wy, Wyd, AWB	清零累加器	1	1	OA,OB,SA,SB
16	CLRWDI	CLRWDI	清零看门狗定时器	1	1	WDTO, Sleep
17	COM	COM f	f = f̄	1	1	N, Z
		COM f, WREG	WREG = f̄	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	比较 f 与 WREG, 带借位	1	1	C, DC, N, OV, Z
		CPB Wb, #lit5	比较 Wb 与 lit5, 带借位	1	1	C, DC, N, OV, Z
		CPB Wb, Ws	比较 Wb 与 Ws, 带借位 (Wb - Ws - C̄)	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	禁止中断, 且持续时间为 k 个指令周期	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 循环到 PC + Expr, 执行次数为 lit14 + 1 次	2	2	无
		DO Wn, Expr	执行 DO 循环到 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

dsPIC30F5015/5016

表 22-2: 指令集概述 (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	搜索自左 (MSb) 起第一个位变化	1	1	C
36	FF1L	FF1L Ws, Wnd	搜索自左 (MSb) 起第一个 1	1	1	C
37	FF1R	FF1R Ws, Wnd	搜索自右 (LSb) 起第一个 1	1	1	C
38	GOTO	GOTO Expr	跳转到地址	2	2	无
		GOTO Wn	间接转移	1	2	无
39	INC	INC f	$f = f + 1$	1	1	C, DC, N, OV, Z
		INC f, WREG	$WREG = f + 1$	1	1	C, DC, N, OV, Z
		INC Ws, Wd	$Wd = Ws + 1$	1	1	C, DC, N, OV, Z
40	INC2	INC2 f	$f = f + 2$	1	1	C, DC, N, OV, Z
		INC2 f, WREG	$WREG = f + 2$	1	1	C, DC, N, OV, Z
		INC2 Ws, Wd	$Wd = Ws + 2$	1	1	C, DC, N, OV, Z
41	IOR	IOR f	$f = f .IOR.WREG$	1	1	N, Z
		IOR f, WREG	$WREG = f .IOR.WREG$	1	1	N, Z
		IOR #lit10, Wn	$Wd = lit10 .IOR.Wd$	1	1	N, Z
		IOR Wb, Ws, Wd	$Wd = Wb .IOR.Ws$	1	1	N, Z
		IOR Wb, #lit5, Wd	$Wd = Wb .IOR.lit5$	1	1	N, Z
42	LAC	LAC Wso, #Slit4, Acc	装入累加器	1	1	OA, OB, OAB, SA, SB, SAB
43	LNK	LNK #lit14	链接帧指针	1	1	无
44	LSR	LSR f	$f = \text{逻辑右移 } f$	1	1	C, N, OV, Z
		LSR f, WREG	$WREG = \text{逻辑右移 } f$	1	1	C, N, OV, Z
		LSR Ws, Wd	$Wd = \text{逻辑右移 } Ws$	1	1	C, N, OV, Z
		LSR Wb, Wns, Wnd	$Wnd = \text{逻辑右移 } Wb, \text{ 移位位数为 } Wns$	1	1	N, Z
		LSR Wb, #lit5, Wnd	$Wnd = \text{逻辑右移 } Wb, \text{ 移位位数为 } lit5$	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	传送 f 至 Wn	1	1	无
		MOV f	传送 f 至 f	1	1	N, Z
		MOV f, WREG	传送 f 至 WREG	1	1	N, Z
		MOV #lit16, Wn	传送 16 位立即数至 Wn	1	1	无
		MOV.b #lit8, Wn	传送 8 位立即数至 Wn	1	1	无
		MOV Wn, f	传送 Wn 至 f	1	1	无
		MOV Wso, Wdo	传送 Ws 至 Wd	1	1	无
		MOV WREG, f	传送 WREG 至 f	1	1	N, Z
		MOV.D Wns, Wd	双字传送 W(ns):W(ns+1) 至 Wd	1	2	无
47	MOVSAC	MOVSAC Ws, Wnd	双字传送 Ws 至 W(nd+1):W(nd)	1	2	无
		MOVSAC Acc, Wx, Wxd, Wy, Wyd, AWB	预取并保存累加器	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	$-(Wm \times Wn)$, 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	$\{Wnd+1, Wnd\} = \text{signed}(Wb) * \text{signed}(Ws)$	1	1	无
		MUL.SU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{signed}(Wb) * \text{unsigned}(Ws)$	1	1	无
		MUL.US Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{unsigned}(Wb) * \text{signed}(Ws)$	1	1	无
		MUL.UU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{unsigned}(Wb) * \text{unsigned}(Ws)$	1	1	无
		MUL.SU Wb, #lit5, Wnd	$\{Wnd+1, Wnd\} = \text{signed}(Wb) * \text{unsigned}(lit5)$	1	1	无
		MUL.UU Wb, #lit5, Wnd	$\{Wnd+1, Wnd\} = \text{unsigned}(Wb) * \text{unsigned}(lit5)$	1	1	无
		MUL f	$W3:W2 = f * WREG$	1	1	无

表 22-2: 指令集概述 (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 Acc	累加器求补	1	1	OA,OB,OAB, SA,SB,SAB
		NEG f	$f = \bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG f, WREG	WREG = $\bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG Ws, Wd	Wd = $\bar{Ws} + 1$	1	1	C,DC,N,OV,Z
53	NOP	NOP	空操作	1	1	无
		NOPR	空操作	1	1	无
54	POP	POP f	从栈顶 (TOS) 弹出至 f	1	1	无
		POP Wdo	从栈顶 (TOS) 弹出至 Wdo	1	1	无
		POP.D Wnd	从栈顶 (TOS) 弹出至 W(nd):W(nd+1)	1	2	无
		POP.S	把影子寄存器弹出堆栈	1	1	全部
55	PUSH	PUSH f	把 f 压入栈顶 (TOS)	1	1	无
		PUSH Wso	把 Wso 压入栈顶 (TOS)	1	1	无
		PUSH.D Wns	把 W(ns):W(ns+1) 压入栈顶 (TOS)	1	2	无
		PUSH.S	把影子寄存器压入堆栈	1	1	无
56	PWRSV	PWRSV #lit1	进入休眠或空闲模式	1	1	WDTO,Sleep
57	RCALL	RCALL Expr	相对调用	1	2	无
		RCALL Wn	计算调用	1	2	无
589	REPEAT	REPEAT #lit14	重复执行下一条指令 lit14 + 1 次	1	1	无
		REPEAT Wn	重复执行下一条指令 (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	返回并将立即数送入 Wn	1	3 (2)	无
62	RETURN	RETURN	从子程序返回	1	3 (2)	无
63	RLC	RLC f	f = 带进位位循环左移 f	1	1	C,N,Z
		RLC f, WREG	WREG = 带进位位循环左移 f	1	1	C,N,Z
		RLC Ws, Wd	Wd = 带进位位循环左移 Ws	1	1	C,N,Z
64	RLNC	RLNC f	f = 不带进位位循环左移 f	1	1	N,Z
		RLNC f, WREG	WREG = 不带进位位循环左移 f	1	1	N,Z
		RLNC Ws, Wd	Wd = 不带进位位循环左移 Ws	1	1	N,Z
65	RRC	RRC f	f = 带进位位循环右移 f	1	1	C,N,Z
		RRC f, WREG	WREG = 带进位位循环右移 f	1	1	C,N,Z
		RRC Ws, Wd	Wd = 带进位位循环右移 Ws	1	1	C,N,Z
66	RRNC	RRNC f	f = 不带进位位循环右移 f	1	1	N,Z
		RRNC f, WREG	WREG = 不带进位位循环右移 f	1	1	N,Z
		RRNC Ws, Wd	Wd = 不带进位位循环右移 Ws	1	1	N,Z
67	SAC	SAC Acc, #Slit4, Wdo	保存累加器	1	1	无
		SAC.R Acc, #Slit4, Wdo	保存舍入后的累加器	1	1	无
68	SE	SE Ws, Wnd	Wnd = 符号扩展后的 Ws	1	1	C,N,Z
69	SETM	SETM f	f = 0xFFFF	1	1	无
		SETM WREG	WREG = 0xFFFF	1	1	无
		SETM Ws	Ws = 0xFFFF	1	1	无
70	SFTAC	SFTAC Acc, Wn	算术移位累加器 (Wn) 次	1	1	OA,OB,OAB, SA,SB,SAB
		SFTAC Acc, #Slit6	算术移位累加器 Slit6 次	1	1	OA,OB,OAB, SA,SB,SAB
71	SL	SL f	f = 左移 f	1	1	C,N,OV,Z
		SL f, WREG	WREG = 左移 f	1	1	C,N,OV,Z
		SL Ws, Wd	Wd = 左移 Ws	1	1	C,N,OV,Z
		SL Wb, Wns, Wnd	Wnd = 左移 Wb, 移位位数为 Wns	1	1	N,Z
		SL Wb, #lit5, Wnd	Wnd = 左移 Wb, 移位位数为 lit5	1	1	N,Z

dsPIC30F5015/5016

表 22-2: 指令集概述 (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 Acc	从累加器减去	1	1	OA,OB,OAB, SA,SB,SAB
		SUB f	f = f WREG	1	1	C,DC,N,OV,Z
		SUB f, WREG	WREG = f WREG	1	1	C,DC,N,OV,Z
		SUB #lit10, Wn	Wn = Wn lit10	1	1	C,DC,N,OV,Z
		SUB Wb, Ws, Wd	Wd = Wb Ws	1	1	C,DC,N,OV,Z
		SUB Wb, #lit5, Wd	Wd = Wb lit5	1	1	C,DC,N,OV,Z
73	SUBB	SUBB f	f = f WREG ($\bar{C}$)	1	1	C,DC,N,OV,Z
		SUBB f, WREG	WREG = f WREG ($\bar{C}$)	1	1	C,DC,N,OV,Z
		SUBB #lit10, Wn	Wn = Wn lit10 (C)	1	1	C,DC,N,OV,Z
		SUBB Wb, Ws, Wd	Wd = Wb Ws (C)	1	1	C,DC,N,OV,Z
		SUBB Wb, #lit5, Wd	Wd = Wb lit5 (C)	1	1	C,DC,N,OV,Z
74	SUBR	SUBR f	f = WREG f	1	1	C,DC,N,OV,Z
		SUBR f, WREG	WREG = WREG f	1	1	C,DC,N,OV,Z
		SUBR Wb, Ws, Wd	Wd = Ws Wb	1	1	C,DC,N,OV,Z
		SUBR Wb, #lit5, Wd	Wd = lit5 Wb	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR f	f = WREG f (C)	1	1	C,DC,N,OV,Z
		SUBBR f, WREG	WREG = WREG f (C)	1	1	C,DC,N,OV,Z
		SUBBR Wb, Ws, Wd	Wd = Ws Wb (C)	1	1	C,DC,N,OV,Z
		SUBBR Wb, #lit5, Wd	Wd = lit5 Wb (C)	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b Wn	Wn = 半字节交换 Wn	1	1	无
		SWAP Wn	Wn = 字节交换 Wn	1	1	无
77	TBLRDH	TBLRDH Ws, Wd	读程序存储单元 <23:16> 至 Wd<7:0>	1	2	无
78	TBLRDL	TBLRDL Ws, Wd	读程序存储单元 <15:0> 至 Wd	1	2	无
79	TBLWTH	TBLWTH Ws, Wd	写 Ws<7:0> 至程序存储单元 <23:16>	1	2	无
80	TBLWTL	TBLWTL Ws, Wd	写 Ws 至程序存储单元 <15:0>	1	2	无
81	ULNK	ULNK	取消帧指针链接	1	1	无
82	XOR	XOR f	f = f.XOR.WREG	1	1	N,Z
		XOR f, WREG	WREG = f.XOR.WREG	1	1	N,Z
		XOR #lit10, Wn	Wd = lit10.XOR.Wd	1	1	N,Z
		XOR Wb, Ws, Wd	Wd = Wb.XOR.Ws	1	1	N,Z
		XOR Wb, #lit5, Wd	Wd = Wb.XOR.lit5	1	1	N,Z
83	ZE	ZE Ws, Wnd	Wnd = 零扩展后的 Ws	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，或其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特征：

- 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® 和 dsPIC® 闪存单片机进行调试和编程。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、SEEEVAL® 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) 以及最新的 “*Product Selector Guide (产品选型指南)*” (DS00148)。

24.0 电气特性

本节概述了 dsPIC30F 系列器件的电气特性。将来如果有更多信息，会在本文档的未来版本中提供。

关于 dsPIC30F 架构和内核的详细信息，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。器件长时间工作在这些额定最大值条件下，其可靠性可能受到影响。不建议使器件在或超过本规范指定的工作参数的条件下工作。

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
任一引脚（除 VDD 和 MCLR 外）相对于 Vss 的电压 (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
流出 Vss 引脚的最大电流	300 mA
流入 VDD 引脚的最大电流 (2)	250 mA
输入箝位电流，Iik (Vi < 0 或 Vi > VDD)	±20 mA
输出箝位电流，Iok (Vo < 0 或 Vo > VDD)	±20 mA
任一 I/O 引脚的最大输出灌电流	25 mA
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 (2)	200 mA

- 注 1: MCLR/VPP 引脚上若出现低于 Vss 的尖峰电压，感应电流超过 80 mA，可能导致锁死。因此，在 MCLR/VPP 引脚上施加低电平时，应使用一个 50-100Ω 的串联电阻，而不是将该引脚直接与 Vss 连接。
- 2: 最大允许电流由器件最大功耗的函数决定。请参见表 24-6。

† 注：如果运行条件超过了上述“绝对最大额定值”，即可能对器件造成永久性损坏。上述值仅为运行条件的极大值，我们不建议器件运行在该规范范围以外。器件长时间工作在最大额定值条件下，其可靠性可能受到影响。

dsPIC30F5015/5016

24.1 直流特性

表 24-1: dsPIC30F5015 的工作 MIPS 与电压

VDD 范围 (伏特)	温度范围 (°C)	最大 MIPS	
		dsPIC30F5015-30I	dsPIC30F5015-20E
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: dsPIC30F5016 的工作 MIPS 与电压

VDD 范围 (伏特)	温度范围 (°C)	最大 MIPS	
		dsPIC30F5016-30I	dsPIC30F5016-20E
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: dsPIC30F5015/5016 的热工作条件

参数	符号	最小值	典型值	最大值	单位
dsPIC30F5015-30I/dsPIC30F5016-30I					
工作结点温度范围	TJ	-40	—	+125	°C
工作环境温度范围	TA	-40	—	+85	°C
dsPIC30F5015-20E/dsPIC30F5016-20E					
工作结点温度范围	TJ	-40	—	+150	°C
工作环境温度范围	TA	-40	—	+125	°C
功耗: 内部芯片功耗: $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$ I/O 引脚功耗: $P_{I/O} = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	PD	PINT + PI/O			W
最大允许功耗	PDMAX	(TJ - TA)/θJA			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 (12x12x1mm)	θJA	39	—	°C/W	1
封装热阻, 64 引脚 TQFP (10x10x1mm)	θJA	39	—	°C/W	1

注 1: 通过封装模拟获得结点与环境的热阻值 θJA。

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	供电电压	2.5	—	5.5	V	工业级温度范围
DC11	VDD	供电电压	3.0	—	5.5	V	扩展级温度范围
DC12	VDR	RAM 数据保持电压 (3)	1.5	—	—	V	
DC16	VPOR	确保内部上电复位信号的 VDD 启动电压	—	VSS	—	V	
DC17	SVDD	确保内部上电复位信号的 VDD 上升率	0.05	—	—	V/ms	0.1s 内为 0-5V 60 ms 内为 0-3V

注 1: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。
2: 这些参数为特性值, 但生产时未经测试。
3: 这是在不丢失 RAM 数据的前提下, VDD 的下限值。

dsPIC30F5015/5016

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	5.6	10	mA	25°C	3.3V	0.128 MIPS LPRC (512 kHz)
DC31b	5.7	10	mA	85°C		
DC31c	5.5	10	mA	125°C		
DC31e	14	23	mA	25°C	5V	
DC31f	15	23	mA	85°C		
DC31g	15	23	mA	125°C		
DC30a	10	21	mA	25°C	3.3V	(1.8 MIPS) FRC (7.37 MHz)
DC30b	12	21	mA	85°C		
DC30c	14	21	mA	125°C		
DC30e	23	38	mA	25°C	5V	
DC30f	24	38	mA	85°C		
DC30g	25	38	mA	125°C		
DC23a	30	50	mA	25°C	3.3V	4 MIPS
DC23b	32	50	mA	85°C		
DC23c	35	50	mA	125°C		
DC23e	32	53	mA	25°C	5V	
DC23f	34	53	mA	85°C		
DC23g	35	53	mA	125°C		
DC24a	35	60	mA	25°C	3.3V	10 MIPS
DC24b	37	60	mA	85°C		
DC24c	40	60	mA	125°C		
DC24e	62	95	mA	25°C	5V	
DC24f	63	95	mA	85°C		
DC24g	65	95	mA	125°C		
DC27a	63	95	mA	25°C	3.3V	20 MIPS
DC27b	65	95	mA	85°C		
DC27d	108	145	mA	25°C		
DC27e	127	145	mA	85°C	5V	
DC27f	130	145	mA	125°C		
DC29a	151	200	mA	25°C	5V	
DC29b	170	200	mA	85°C		

注 1: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。
2: 供电电流主要受工作电压和频率的影响。其他因素, 如 I/O 引脚负载和开关速率、振荡器类型、内部代码执行模式和温度等, 也会对电流消耗产生影响。所有 IDD 测量值的测试条件为: OSC1 由外部方波驱动, 轨到轨摆幅。所有 I/O 引脚均配置为输入并上拉至 VDD。MCLR = VDD, WDT、FSCM、LVD 和 BOR 被禁止。CPU、SRAM、程序存储器和数据存储器处于工作状态。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⁾	最大值	单位	条件			
工作电流 (IDD) ⁽²⁾							
DC51a	5.2	9	mA	25°C	3.3V	0.128 MIPS LPRC (512 kHz)	
DC51b	5.3	9	mA	85°C			
DC51c	5.4	9	mA	125°C			
DC51e	13	22	mA	25°C	5V		
DC51f	14	22	mA	85°C			
DC51g	15	22	mA	125°C			
DC50a	8.1	13	mA	25°C	3.3V	(1.8 MIPS) FRC (7.37 MHz)	
DC50b	8.2	13	mA	85°C			
DC50c	8.3	13	mA	125°C			
DC50e	19	32	mA	25°C	5V		
DC50f	20	32	mA	85°C			
DC50g	21	32	mA	125°C			
DC43a	12	26	mA	25°C	3.3V	4 MIPS	
DC43b	14	26	mA	85°C			
DC43c	17	26	mA	125°C			
DC43e	25	42	mA	25°C	5V		
DC43f	26	42	mA	85°C			
DC43g	28	42	mA	125°C			
DC44a	23	40	mA	25°C	3.3V	10 MIPS	
DC44b	25	40	mA	85°C			
DC44c	26	40	mA	125°C			
DC44e	42	68	mA	25°C	5V		
DC44f	43	68	mA	85°C			
DC44g	45	68	mA	125°C			
DC47a	40	55	mA	25°C	3.3V	20 MIPS	
DC47b	40	55	mA	85°C			
DC47d	70	85	mA	25°C	5V		
DC47e	72	85	mA	85°C			
DC47f	74	85	mA	125°C			
DC49a	98	120	mA	25°C	5V		30 MIPS
DC49b	103	120	mA	85°C			

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

2: 基本 IDLE 电流是在内核不工作、时钟工作以及所有模块关闭的条件下测得的。

dsPIC30F5015/5016

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		
DC60b	0.7	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	16	90	μA	125°C				
DC61a	10	30	μA	25°C	3.3V	看门狗定时器电流: ΔI _{WDT} ⁽³⁾		
DC61b	12	30	μA	85°C				
DC61c	12	30	μA	125°C				
DC61e	20	40	μA	25°C	5V			
DC61f	22	40	μA	85°C				
DC61g	23	40	μA	125°C				
DC62a	4	10	μA	25°C	3.3V	Timer 1, 32 kHz 晶振: ΔI _{T132} ⁽³⁾		
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	33	65	μA	25°C	3.3V	BOR 开启: ΔI _{BOR} ⁽³⁾		
DC63b	38	65	μA	85°C				
DC63c	39	65	μA	125°C				
DC63e	38	70	μA	25°C	5V			
DC63f	41	70	μA	85°C				
DC63g	42	70	μA	125°C				

注 1: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

2: 这些参数为特性值, 但生产时未经测试。

3: Δ 电流是在模块使能时额外消耗的电流。应将该电流加到基本 IPD 电流上去。

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 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (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 ⁽¹⁾	最大值	单位	条件
DI10	V _{IL}	输入低电压 ⁽²⁾ I/O 引脚: 带施密特触发缓冲器	V _{SS}	—	0.2 V _{DD}	V	SMbus 禁止 SMbus 使能
DI15		MCLR	V _{SS}	—	0.2 V _{DD}	V	
DI16		OSC1 (在 XT、HS 和 LP 模式下)	V _{SS}	—	0.2 V _{DD}	V	
DI17		OSC1 (在 RC 模式下) ⁽³⁾	V _{SS}	—	0.3 V _{DD}	V	
DI18		SDA 和 SCL	V _{SS}	—	0.3 V _{DD}	V	
DI19		SDA 和 SCL	V _{SS}	—	0.2 V _{DD}	V	
DI20	V _{IH}	输入高电压 ⁽²⁾ I/O 引脚: 带施密特触发缓冲器	0.8 V _{DD}	—	V _{DD}	V	SMbus 禁止 SMbus 使能
DI25		MCLR	0.8 V _{DD}	—	V _{DD}	V	
DI26		OSC1 (在 XT、HS 和 LP 模式下)	0.7 V _{DD}	—	V _{DD}	V	
DI27		OSC1 (在 RC 模式下) ⁽³⁾	0.9 V _{DD}	—	V _{DD}	V	
DI28		SDA 和 SCL	0.7 V _{DD}	—	V _{DD}	V	
DI29		SDA 和 SCL	0.8 V _{DD}	—	V _{DD}	V	
DI30	ICNPU	CNxx 上拉电流 ⁽²⁾	50	250	400	μA	V _{DD} = 5V, V _{PIN} = V _{SS}
DI50	I _{IL}	输入泄漏电流 ⁽²⁾⁽⁴⁾⁽⁵⁾ I/O 端口	—	0.01	±1	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态
DI51		模拟输入引脚	—	0.50	—	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态
DI55		MCLR	—	0.05	±5	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD}
DI56		OSC1	—	0.05	±5	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , XT、 HS 和 LP 振荡器模式

- 注 1: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。
- 2: 这些参数为特性值, 但生产时未经测试。
- 3: 在 RC 振荡器配置中, OSC1/CLKI 引脚是施密特触发器输入。不推荐在 RC 模式下使用外部时钟驱动 dsPIC30F 器件。
- 4: MCLR 引脚上的泄漏电流主要取决于所施加电压。规定电压为正常工作条件下的电压。在不同的输入电压下可能测得更高的泄漏电流。
- 5: 负电流定义为引脚的拉电流。

dsPIC30F5015/5016

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	IOI = 8.5 mA, VDD = 5V
			—	—	TBD	V	IOI = 2.0 mA, VDD = 3V
DO16		OSC2/CLKO (RC 或 EC 振荡器模式)	—	—	0.6	V	IOI = 1.6 mA, VDD = 5V
			—	—	TBD	V	IOI = 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
			TBD	—	—	V	IOH = -2.0 mA, VDD = 3V
DO26		OSC2/CLKO (RC 或 EC 振荡器模式)	VDD - 0.7	—	—	V	IOH = -1.3 mA, VDD = 5V
			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	当外部时钟用于驱动 OSC1 时处于 XTL、XT、HS 和 LP 模式下。
DO56	CIO	所有 I/O 引脚和 OSC2	—	—	50	pF	RC 或 EC 振荡器模式
DO58	CB	SCL 和 SDA	—	—	400	pF	在 I ² C™ 模式

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 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

2: 这些参数为特性值, 但生产时未经测试。

图 24-1: 欠压复位特性

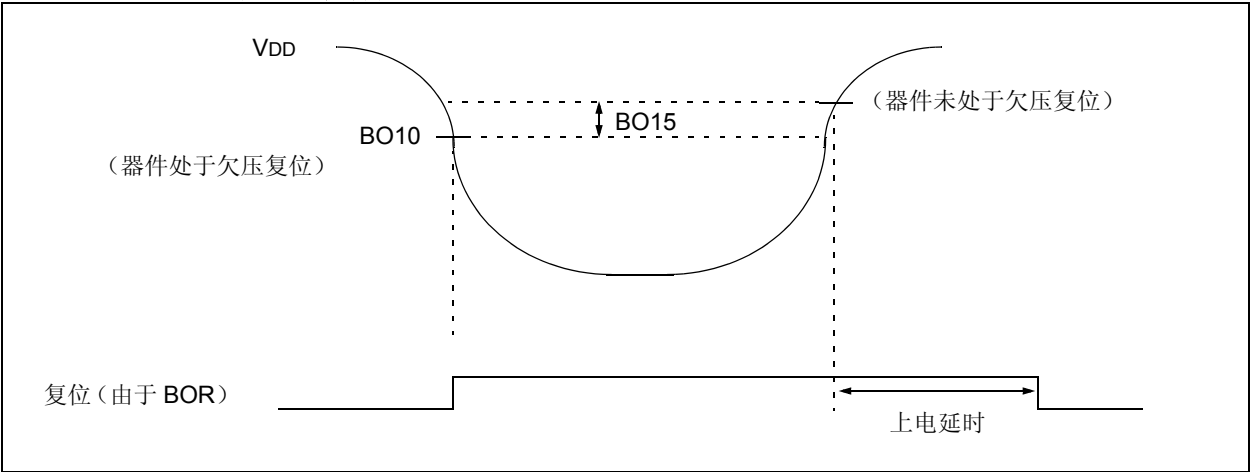


表 24-11: 电气特性: BOR

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 从低至高跳变时的 BOR 电压 ⁽²⁾	BORV = 11 ⁽³⁾	—	—	—	V	不在工作范围中
			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 和 25°C 条件下的值。参数仅供设计参考, 未经测试。
 2: 这些参数为特性值, 但生产时未经测试。
 3: 值 11 不在有效工作范围之内。

表 24-12: 直流特性: 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 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (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 ⁽¹⁾	最大值	单位	条件
D120	ED	数据 EEPROM 存储器 ⁽²⁾ 字节耐擦写次数	100K	1M	—	E/W	$-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ 使用 EECON 来读 / 写 V _{MIN} = 最小工作电压
D121	V _{DRW}		V _{MIN}	—	5.5	V	
D122	T _{DEW}		—	2	—	ms	
D123	T _{RETD}		40	100	—	年	
D124	I _{DEW}		—	10	30	mA	
D130	EP	程序闪存存储器 ⁽²⁾ 单元耐擦写次数	10K	100K	—	E/W	$-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ V _{MIN} = 最小工作电压
D131	V _{PR}		V _{MIN}	—	5.5	V	
D132	V _{EB}		4.5	—	5.5	V	
D133	V _{PEW}		3.0	—	5.5	V	
D134	T _{PEW}		—	2	—	ms	
D135	T _{RETD}		40	100	—	年	
D136	T _{EB}		—	4	—	ms	
D137	I _{PEW}		—	10	30	mA	
D138	I _{EB}		—	10	30	mA	

- 注 1: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。
 2: 这些参数为特性值, 但生产时未经测试。

dsPIC30F5015/5016

24.2 交流特性和时序参数

本小节包含的信息，对 dsPIC30F 器件的交流特性和时序参数进行了定义。

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)
	直流规范第 24.0 节描述了工作电压 V_{DD} 的范围。

图 24-2: 器件时序规范的负载条件

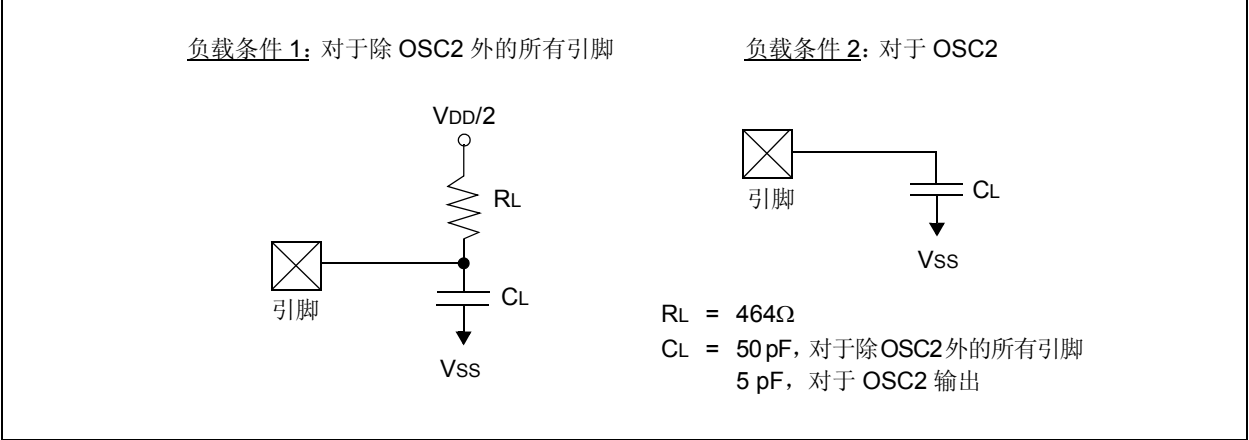


图 24-3: 外部时钟时序

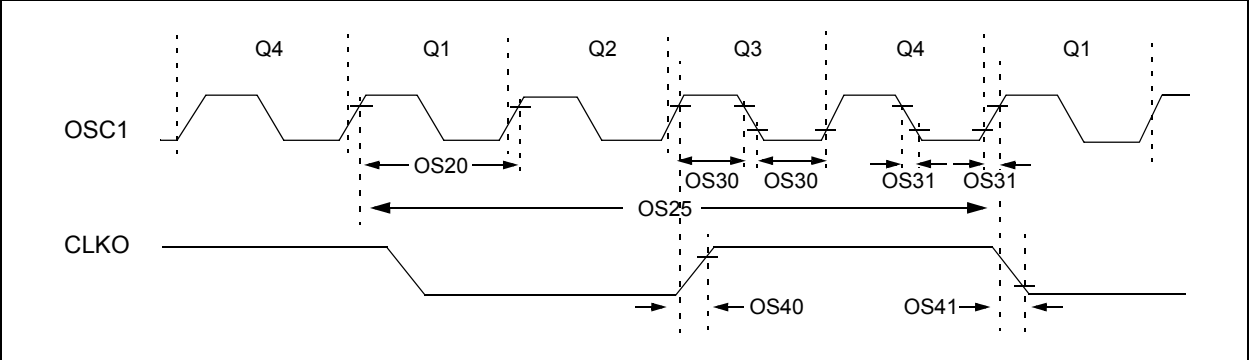


表 24-14: 外部时钟时序要求

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 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (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 ⁽¹⁾	最大值	单位	条件
OS10	Fosc	外部 CLKI 频率 ⁽²⁾ (仅在 EC 模式下允许外部时钟)	DC 4 4 4	— — — —	40 10 10 7.5 ⁽³⁾	MHz MHz MHz MHz	EC EC 带 4 倍频 PLL EC 带 8 倍频 PLL EC 带 16 倍频 PLL
		振荡器频率 ⁽²⁾	DC 0.4 4 4 4 4 10 10 10 12 ⁽⁴⁾ 12 ⁽⁴⁾ 12 ⁽⁴⁾ —	— — — — — — — — — — — — 32.768	4 4 10 10 10 7.5 ⁽³⁾ 25 20 ⁽⁴⁾ 20 ⁽⁴⁾ 15 ⁽³⁾ 25 25 22.5 ⁽³⁾ —	MHz MHz MHz MHz MHz MHz MHz MHz MHz MHz MHz MHz kHz	RC XTL XT XT 带 4 倍频 PLL XT 带 8 倍频 PLL XT 带 16 倍频 PLL HS HS/2 带 4 倍频 PLL HS/2 带 8 倍频 PLL HS/2 带 16 倍频 PLL HS/3 带 4 倍频 PLL HS/3 带 8 倍频 PLL HS/3 带 16 倍频 PLL LP
OS20	Tosc	Tosc = 1/Fosc	—	—	—	—	Fosc 值见参数 OS10
OS25	Tcy	指令周期时间 ⁽²⁾⁽⁵⁾	33	—	DC	ns	见表 24-16
OS30	TosL, TosH	外部时钟 ⁽²⁾ 输入 (OSC1) 的高电平或低电平时间	.45 x Tosc	—	—	ns	EC
OS31	TosR, TosF	外部时钟 ⁽²⁾ 输入 (OSC1) 的上升或下降时间	—	—	20	ns	EC
OS40	TckR	CLKO 上升时间 ⁽²⁾⁽⁶⁾	—	—	—	ns	见参数 D031
OS41	TckF	CLKO 下降时间 ⁽²⁾⁽⁶⁾	—	—	—	ns	见参数 D032

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

2: 这些参数为特性值, 但生产时未经测试。

3: 受到 PLL 输出频率范围的限制。

4: 受到 PLL 输入频率范围的限制。

5: 指令周期时间 (Tcy) 等于输入振荡器时基周期的 4 倍。所有规定值均为基于针对特定振荡器类型, 器件在标准工作条件下执行代码时的特性数据。超出这些规定的限定值, 可能导致振荡器运行不稳定和 / 或导致电流消耗超出预期值。所有器件在测试“最小”值时, 都在 OSC1/CLKI 引脚连接了外部时钟。当使用了外部时钟输入时, 所有器件的“最大”周期时间限制为“DC”(无时钟)。

6: 测量在 RC 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。

dsPIC30F5015/5016

表 24-15: PLL 时钟时序规范 (VDD = 2.5 至 5.5 V)

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	EC 带 4 倍频 PLL
			4	—	10	MHz	EC 带 8 倍频 PLL
			4	—	7.5 ⁽⁴⁾	MHz	EC 带 16 倍频 PLL
			4	—	10	MHz	XT 带 4 倍频 PLL
			4	—	10	MHz	XT 带 8 倍频 PLL
			4	—	7.5 ⁽⁴⁾	MHz	XT 带 16 倍频 PLL
			5 ⁽³⁾	—	10	MHz	HS/2 带 4 倍频 PLL
			5 ⁽³⁾	—	10	MHz	HS/2 带 8 倍频 PLL
			5 ⁽³⁾	—	7.5 ⁽⁴⁾	MHz	HS/2 带 16 倍频 PLL
			4	—	8.33 ⁽³⁾	MHz	HS/3 带 4 倍频 PLL
			4	—	8.33 ⁽³⁾	MHz	HS/3 带 8 倍频 PLL
			4	—	7.5 ⁽⁴⁾	MHz	HS/3 带 16 倍频 PLL
OS51	FSYS	片上 PLL 输出 ⁽²⁾	16	—	120	MHz	带 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: 这些参数为特性值, 但生产时未经测试。

2: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

3: 受到振荡器频率范围的限制。

4: 受到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	4 倍频 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
	8 倍频 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
	16 倍频 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
		—	0.632	0.956	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5 至 5.5V

注 1: 这些参数为特性值, 但生产时未经测试。

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⁾ 带 4 倍频 PLL	MIPS ⁽³⁾ 带 8 倍频 PLL	MIPS ⁽³⁾ 带 16 倍频 PLL
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: 指令执行周期时间：Tcy = 1/MIPS。
 3: 指令执行频率：MIPS = (Fosc * PLLx)/4 （因为每个指令周期有 4 个 Q 时钟）。

dsPIC30F5015/5016

表 24-18: 交流特性: 内部 FRC 抖动、精度和漂移⁽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	
	FRC 频率 = 7.37 MHz ⁽¹⁾ 时的内部 FRC 抖动						
OS62	FRC	—	±0.04	±0.16	%	-40°C ≤ Ta ≤ +85°C	VDD = 3.0-3.6V
		—	±0.07	±0.23	%	-40°C ≤ Ta ≤ +125°C	VDD = 4.5-5.5V
	FRC 带 4 倍频 PLL	—	±0.31	±0.62	%	-40°C ≤ Ta ≤ +85°C	VDD = 3.0-3.6V
		—	±0.34	±0.77	%	-40°C ≤ Ta ≤ +125°C	VDD = 4.5-5.5V
	FRC 带 8 倍频 PLL	—	±0.44	±0.87	%	-40°C ≤ Ta ≤ +85°C	VDD = 3.0-3.6V
		—	±0.48	±1.08	%	-40°C ≤ Ta ≤ +125°C	VDD = 4.5-5.5V
FRC 带 16 倍频 PLL	—	±0.71	±1.23	%	-40°C ≤ Ta ≤ +125°C	VDD = 4.5-5.5V	
	FRC 频率 = 7.37 MHz ⁽¹⁾ 时的内部 FRC 精度						
OS63	FRC	—	—	±0.5	%	-40°C ≤ Ta ≤ +125°C	VDD = 3.0-5.5V
	FRC 频率 = 7.37 MHz ⁽¹⁾ 时的内部 FRC 漂移						
OS64		-0.7	—	0.5	%	-40°C ≤ Ta ≤ +85°C	VDD = 3.0-3.6V
		-0.7	—	0.7	%	-40°C ≤ Ta ≤ +125°C	VDD = 3.0-3.6V
		-0.7	—	0.5	%	-40°C ≤ Ta ≤ +85°C	VDD = 4.5-5.5V
		-0.7	—	0.7	%	-40°C ≤ Ta ≤ +125°C	VDD = 4.5-5.5V

注 1: 频率在 25°C 和 5V 条件下校准。TUN 位可用来补偿温度漂移。

2: 加上抖动、精度和漂移百分比的绝对值, 可以计算出总的 FRC 变化。

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
	频率 = 512 kHz ⁽¹⁾ 时的 LPRC					
OS65		-35	—	+35	%	—

注 1: LPRC 频率随 VDD 的变化而变化。

图 24-4: CLKO 和 I/O 时序特性

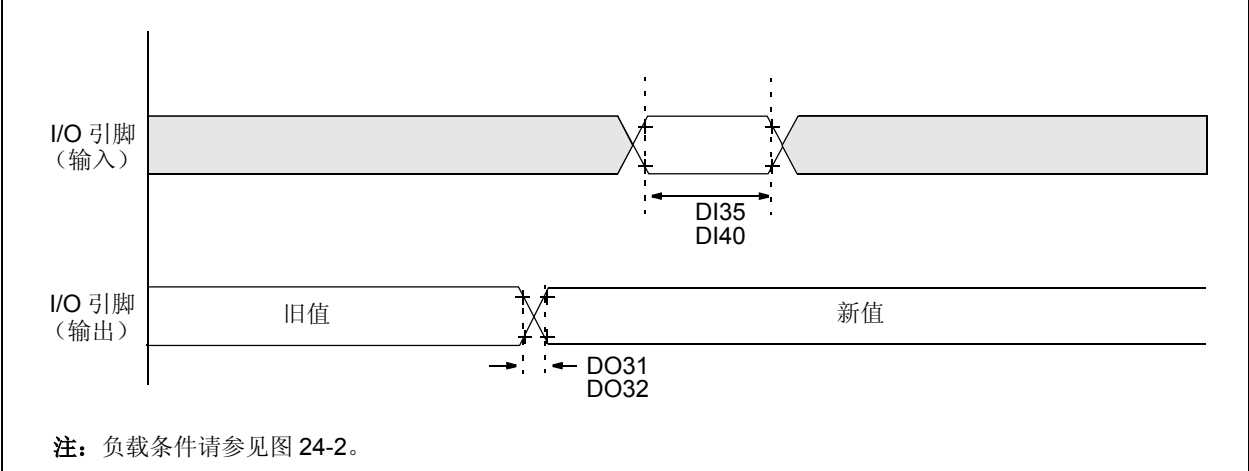


表 24-20: CLKO 和 I/O 时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)(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	—	—	ns	—

- 注 1: 这些参数是与任何内部时钟边沿无关的异步事件。
- 2: 测量是在 RC 模式和 EC 模式下进行的, 其中 CLKO 输出为 4 x TOSC。
- 3: 这些参数为特性值, 但生产时未经测试。
- 4: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。

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

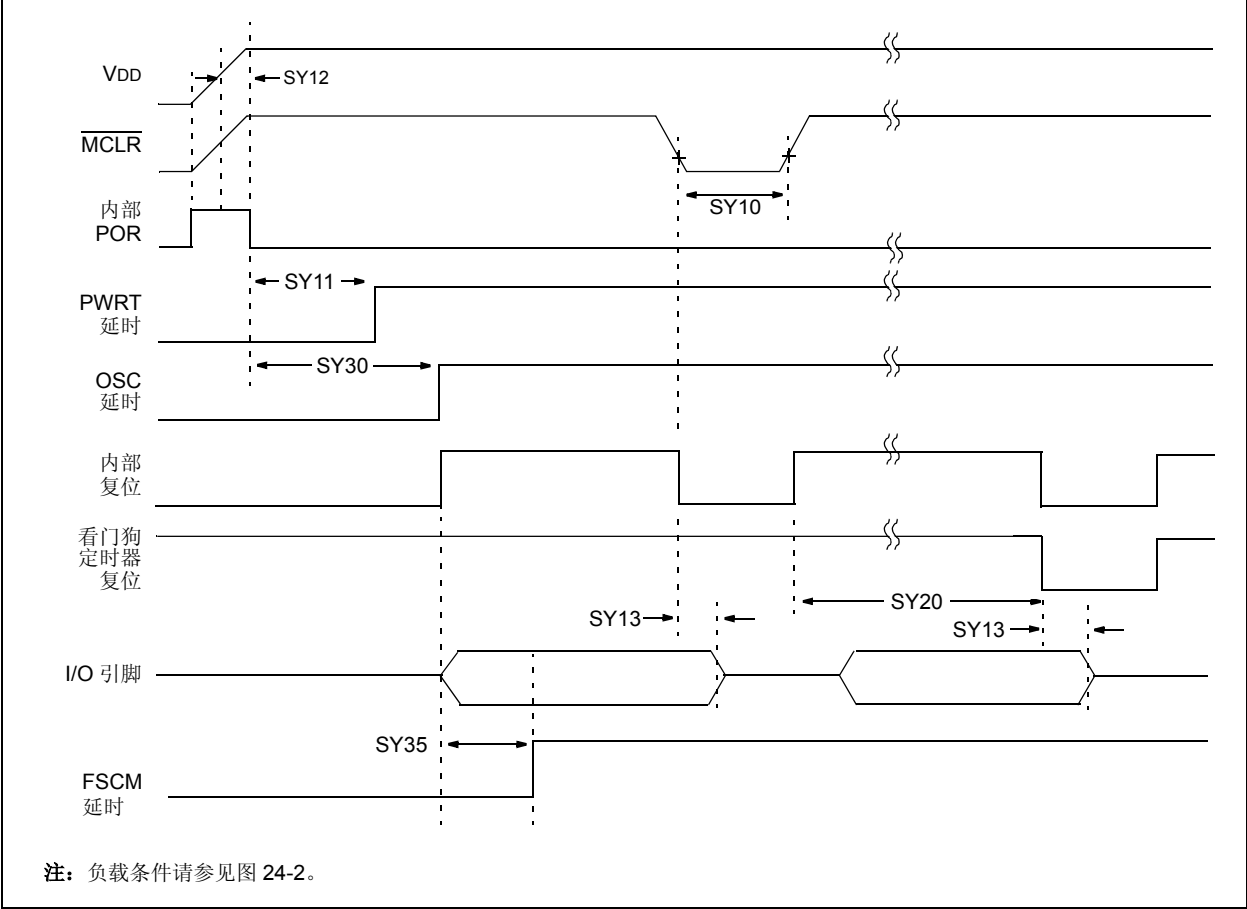


表 24-21: 复位、看门狗定时器、振荡器起振定时器、上电延时定时器和欠压复位时序要求

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 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (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
SY10	TmCL	MCLR 脉冲宽度 (低电平)	2	—	—	μs	-40°C 至 $+85^{\circ}\text{C}$
SY11	TPWRT	上电延时定时器周期	3 12 50	4 16 64	6 22 90	ms	-40°C 至 $+85^{\circ}\text{C}$ 用户可编程
SY12	TPOR	上电复位延时 ⁽⁴⁾	3	10	30	μs	-40°C 至 $+85^{\circ}\text{C}$
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	$V_{DD} = 5\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
	TWDT2		1.4	2.1	2.8	ms	$V_{DD} = 3\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
SY25	TBOR	欠压复位脉冲宽度 ⁽³⁾	100	—	—	μs	$V_{DD} \leq V_{BOR}$ (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 $+85^{\circ}\text{C}$

- 注 1: 这些参数为特性值, 但生产时未经测试。
 2: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。
 3: BOR 请参见图 24-1 和表 24-11。
 4: 设计特性值, 未经测试。

图 24-6: 带隙启动时间特性

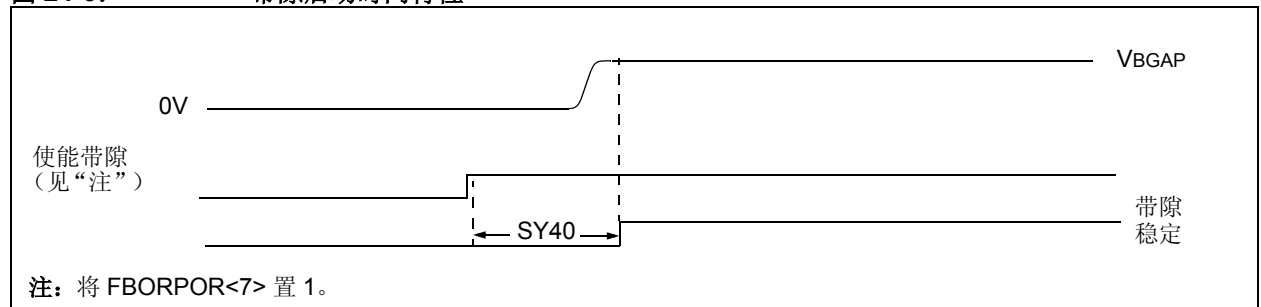


表 24-22: 带隙启动时间要求

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 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (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	最大值	单位	条件
SY40	TBGAP	带隙启动时间	—	40	65	μs	定义为从带隙使能到带隙参考电压稳定之间的时间 (RCON<13> 状态位)。

- 注 1: 这些参数为特性值, 但生产时未经测试。

dsPIC30F5015/5016

图 24-7: TIMER1、2、3、4 和 5 外部时钟时序特性

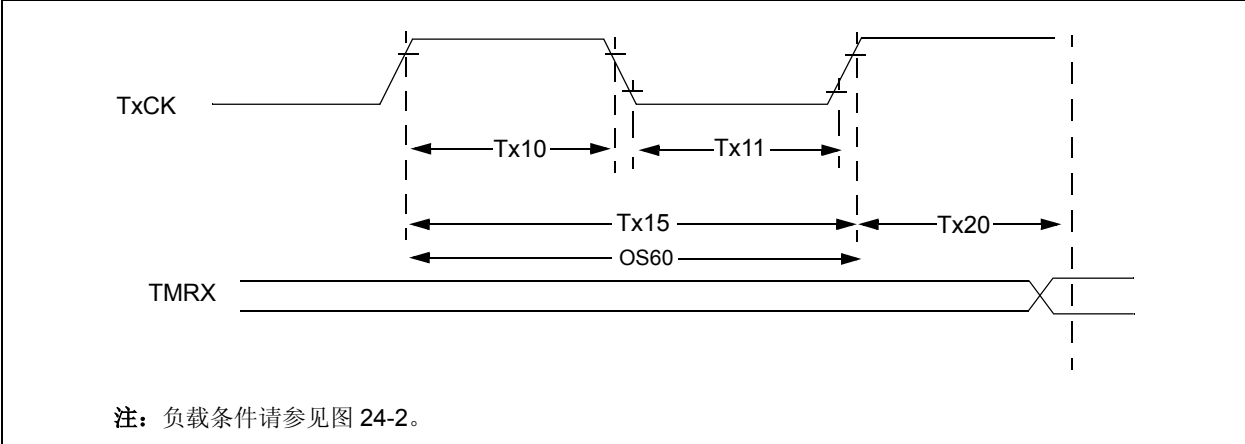


表 24-23: TIMER1 外部时钟时序要求⁽¹⁾

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
TA10	T _{TXH}	TxCK 高电平时间	0.5 T _{CY} + 20	—	—	ns	还必须满足参数 TA15
		同步, 无预分频器	10	—	—	ns	
		同步, 带预分频器	10	—	—	ns	
TA11	T _{TXL}	TxCK 低电平时间	0.5 T _{CY} + 20	—	—	ns	还必须满足参数 TA15
		同步, 无预分频器	10	—	—	ns	
		同步, 带预分频器	10	—	—	ns	
TA15	T _{TXP}	TxCK 输入周期	T _{CY} + 10	—	—	ns	N = 预分频值 (1, 8, 64, 256)
		同步, 无预分频器	取如下二者中较大值: 20 ns 或 (T _{CY} + 40)/N	—	—	—	
		同步, 带预分频器	20	—	—	ns	
OS60	F _{t1}	SOSC1/T1CK 振荡器输入频率范围 (通过置 1 TCS 位 (T1CON, bit 1) 使能振荡器)	DC	—	50	kHz	
TA20	T _{CKEXTMRL}	从外部 TxCK 时钟边沿到定时器递增的延时	0.5 T _{CY}		1.5 T _{CY}	—	

注 1: Timer1 是 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 ≤ 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	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 T _{CY} + 20	—	—	ns	还必须满足参数 TB15
			同步, 带预分频器	10	—	—	ns	
TB11	TtxL	TxCK 低电平时间	同步, 无预分频器	0.5 T _{CY} + 20	—	—	ns	还必须满足参数 TB15
			同步, 带预分频器	10	—	—	ns	
TB15	TtxP	TxCK 输入周期	同步, 无预分频器	T _{CY} + 10	—	—	ns	N = 预分频值 (1, 8, 64, 256)
			同步, 带预分频器	取如下二者 中较大值: 20 ns 或 (T _{CY} + 40)/N				
TB20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增的延时		0.5 T _{CY}	—	1.5 T _{CY}	—	

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 ≤ 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	典型值	最大值	单位	条件
TC10	TtxH	TxCK 高电平时间	同步	0.5 T _{CY} + 20	—	—	ns	还必须满足参数 TC15
TC11	TtxL	TxCK 低电平时间	同步	0.5 T _{CY} + 20	—	—	ns	还必须满足参数 TC15
TC15	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, 带预分频器	取如下二者 中较大值: 20 ns 或 (T _{CY} + 40)/N				
TC20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增的延时		0.5 T _{CY}	—	1.5 T _{CY}	—	

dsPIC30F5015/5016

图 24-8: TIMERQ (QEI 模块) 外部时钟时序特性

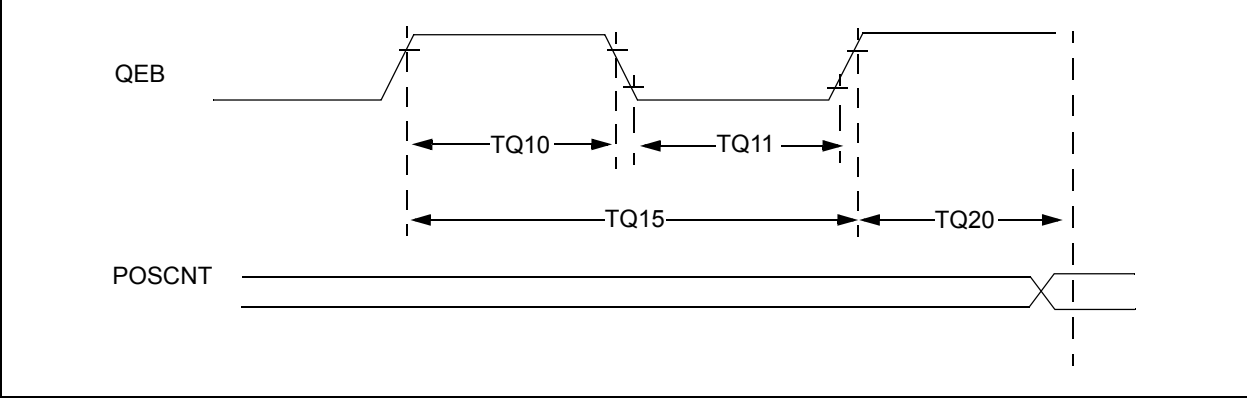


表 24-26: QEI 模块外部时钟时序要求

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, 带预分频器	Tcy + 20		—	ns 还必须满足参数 TQ15
TQ11	TtQL	TQCK 低电平时间	同步, 带预分频器	Tcy + 20		—	ns 还必须满足参数 TQ15
TQ15	TtQP	TQCP 输入周期	同步, 带预分频器	2 * Tcy + 40		—	ns —
TQ20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增的延时		0.5 Tcy		1.5 Tcy	ns —

注 1: 这些参数为特性值, 但生产时未经测试。

图 24-9: 输入捕捉 (CAPx) 时序特性

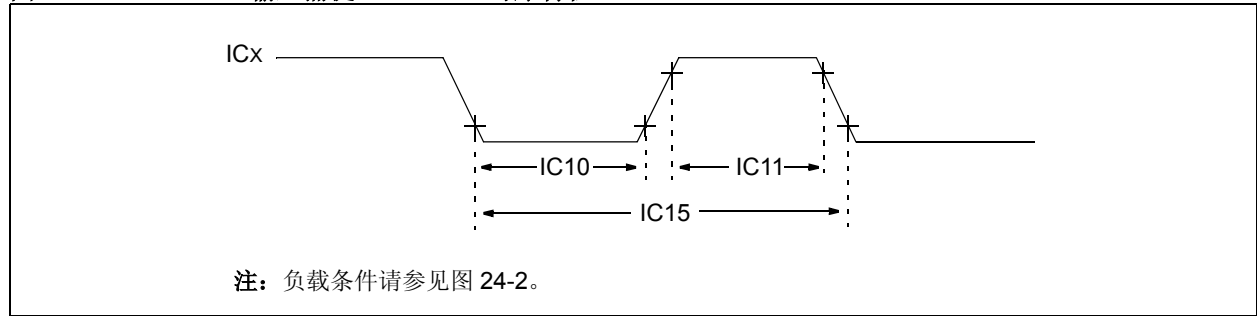


表 24-27: 输入捕捉时序要求

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	
			带预分频器	10	—	ns	
IC11	TccH	ICx 输入高电平时间	无预分频器	$0.5 T_{CY} + 20$	—	ns	
			带预分频器	10	—	ns	
IC15	TccP	ICx 输入周期		$(2 T_{CY} + 40)/N$	—	ns	N = 预分频值 (1、4 或 16)

注 1: 这些参数为特性值, 但生产时未经测试。

图 24-10: 输出捕捉模块 (OCx) 时序特性

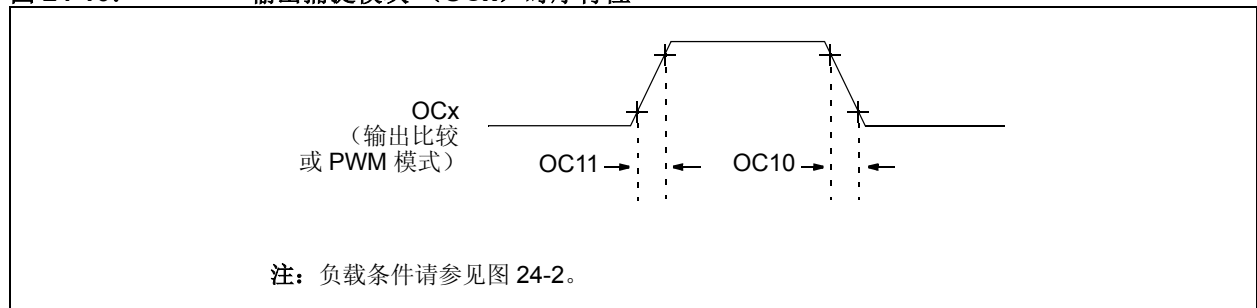


表 24-28: 输出比较模块时序要求

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 D032
OC11	TccR	OCx 输出上升时间	—	—	—	ns	见参数 D031

注 1: 这些参数为特性值, 但生产时未经测试。

2: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

dsPIC30F5015/5016

图 24-11: OC/PWM 模块时序特性

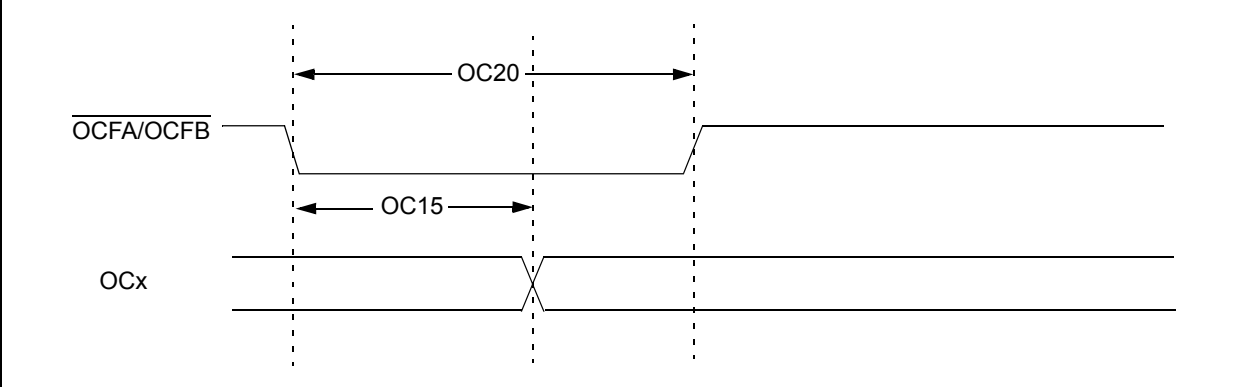


表 24-29: 简单 OC/PWM 模式时序要求

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	T _{FD}	故障输入到 PWM I/O 变化的时间	—	—	50	ns	—
OC20	T _{FLT}	故障输入脉冲宽度	50	—	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。
2: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。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

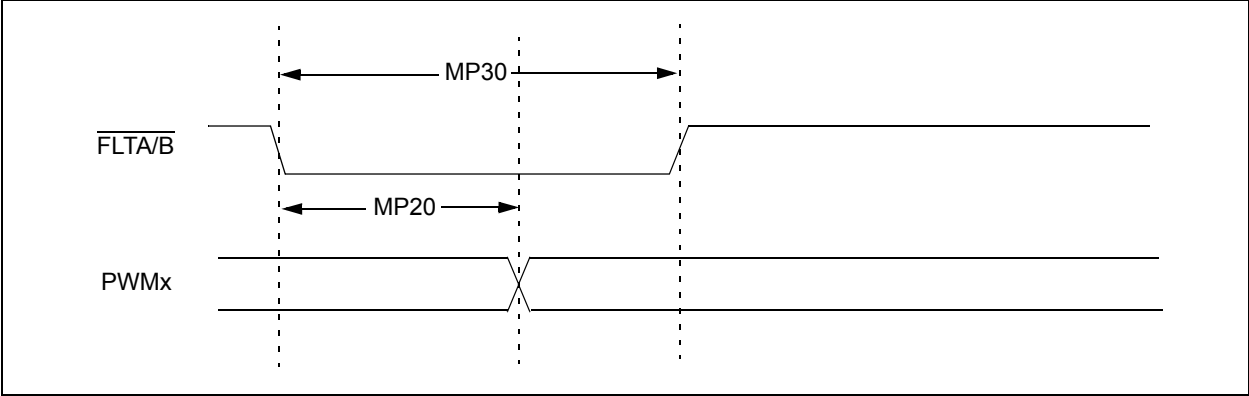


图 24-13: 电机控制 PWM 模块时序特性

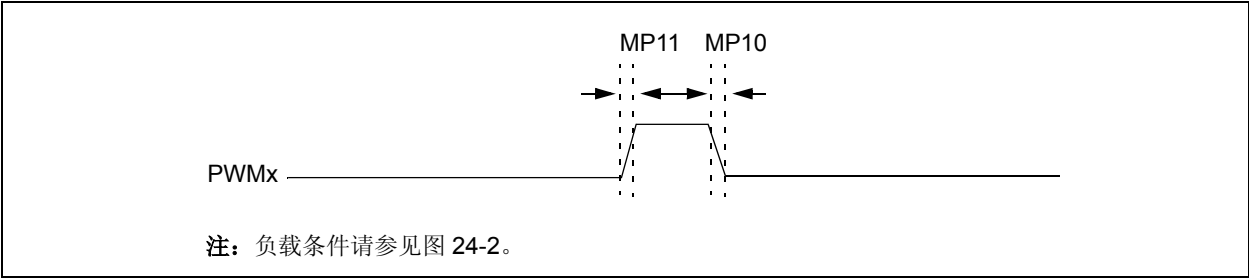


表 24-30: 电机控制 PWM 模块时序要求

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 D032
MP11	TRPWM	PWM 输出上升时间	—	—	—	ns	见参数 D031
MP20	TFD	故障输入 ↓ 到 PWM I/O 变化的时间	—	—	50	ns	—
MP30	TFH	最小脉冲宽度	50	—	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。
2: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

dsPIC30F5015/5016

图 24-14: QEA/QEB 输入特性

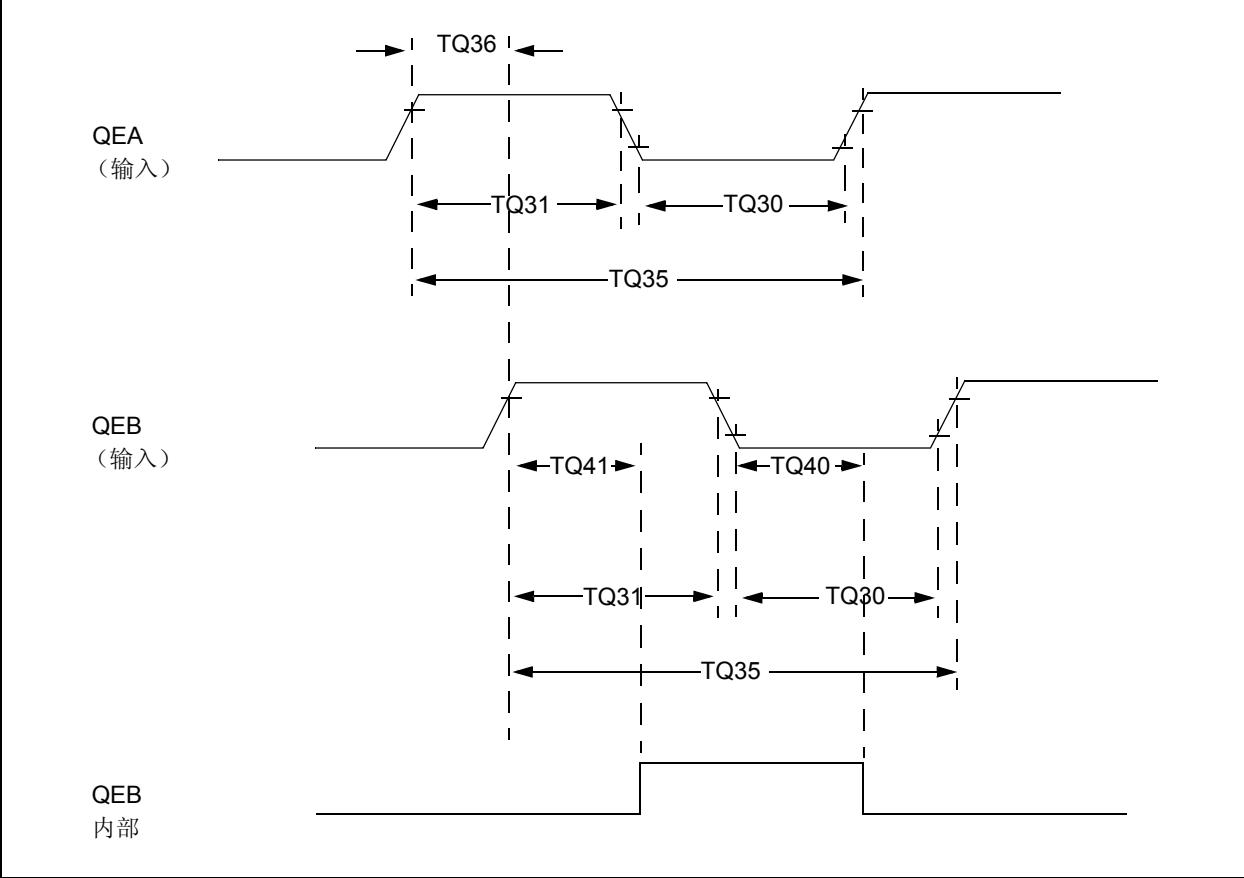


表 24-31: 正交编码器时序要求

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°C ≤ TA ≤ +85°C (工业级)			
			-40°C ≤ TA ≤ +125°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	正交相位周期	3 TCY	—	ns	—
TQ40	TQuFL	滤波器识别低电平的时间, 使用数字滤波器	3 * N * TCY	—	ns	N = 1、2、4、16、32、64、128 和 256 (注 2)
TQ41	TQuFH	滤波器识别高电平的时间, 使用数字滤波器	3 * N * TCY	—	ns	N = 1、2、4、16、32、64、128 和 256 (注 2)

注 1: 这些参数为特性值, 但生产时未经测试。

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 (QE1)”章节。

图 24-15: QEI 模块索引脉冲时序特性

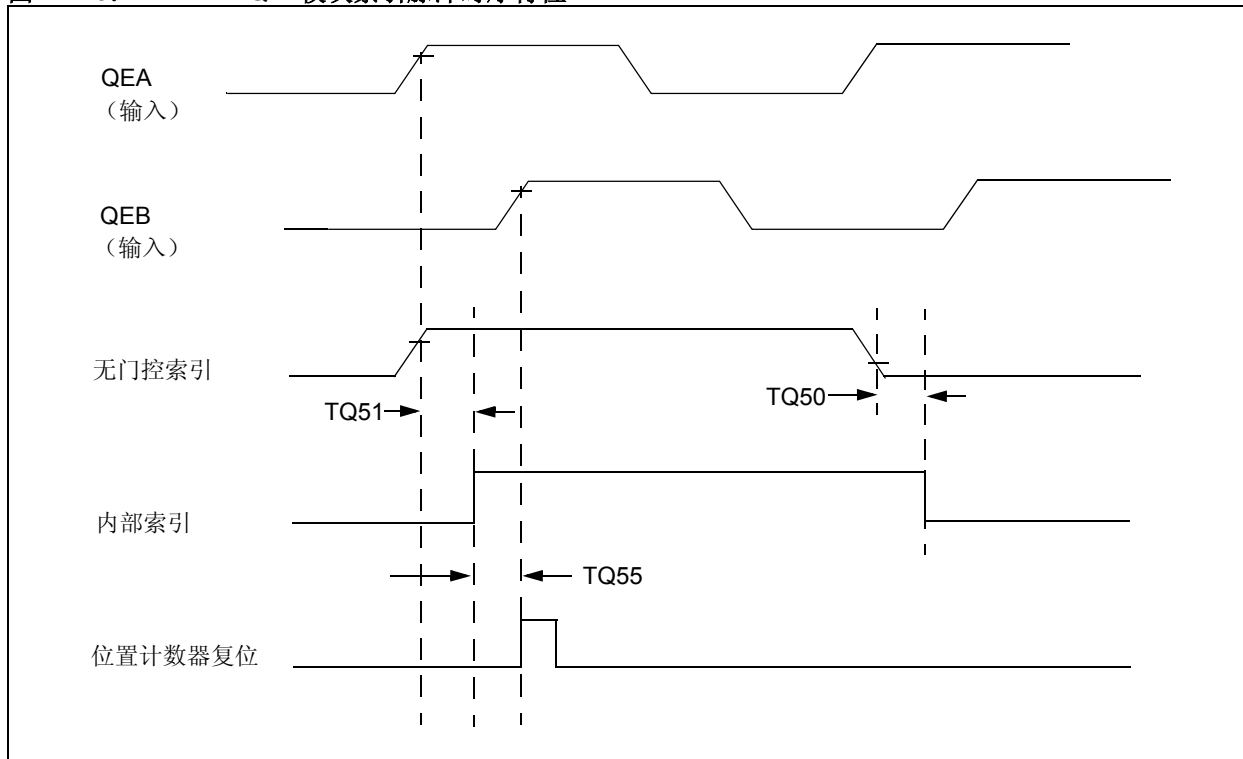


表 24-32: QEI 索引脉冲时序要求

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	滤波器识别低电平的时间, 使用数字滤波器	$3 * N * T_{CY}$	—	ns	N = 1、2、4、16、32、64、128 和 256 (注 2)
TQ51	TqiH	滤波器识别高电平的时间, 使用数字滤波器	$3 * N * T_{CY}$	—	ns	N = 1、2、4、16、32、64、128 和 256 (注 2)
TQ55	Tqidxr	识别索引脉冲到位置计数器复位的时间 (无门控索引)	$3 T_{CY}$	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。

2: 只有位置计数器复位时序中显示索引脉冲与 QEA 和 QEB 对齐。只在正向时序中显示 (QEA 超前 QEB)。反向适用同样的时序 (QEA 滞后 QEB), 但是在下降沿识别索引脉冲。

dsPIC30F5015/5016

图 24-16: SPI 模块主模式 (CKE = 0) 时序特性

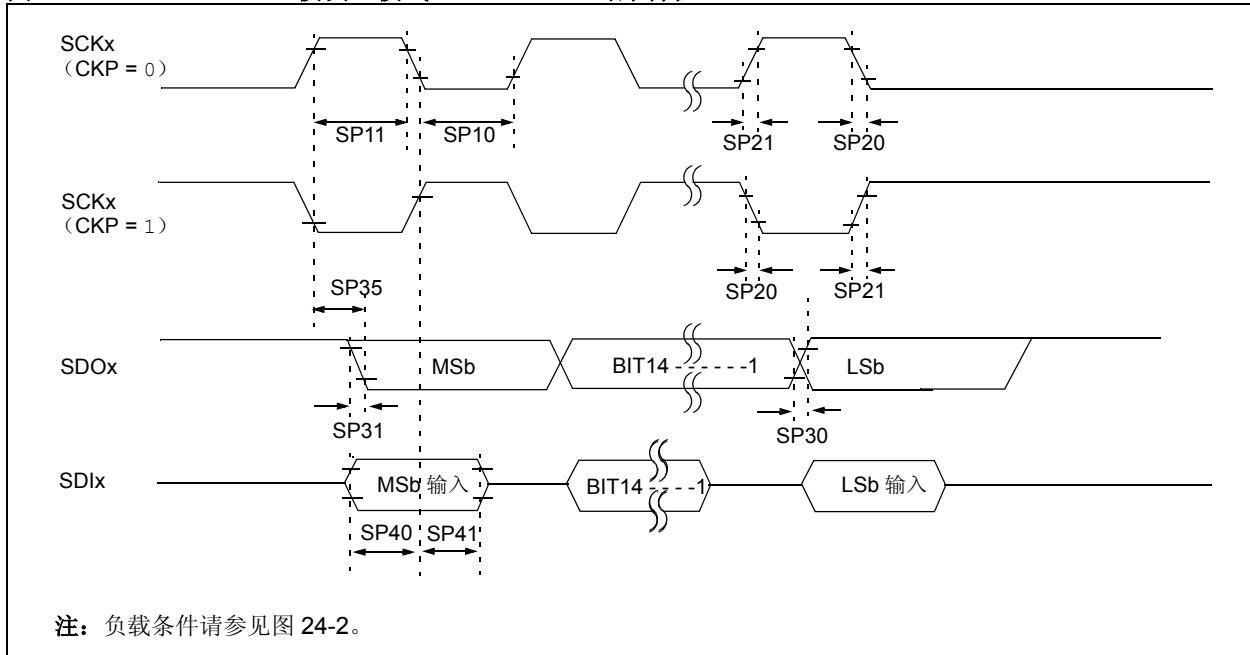


表 24-33: SPI 主模式 (CKE = 0) 时序要求

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
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 D032
SP21	TscR	SCKx 输出上升时间 ⁽⁴⁾	—	—	—	ns	见参数 D031
SP30	TdoF	SDOx 数据输出下降时间 ⁽⁴⁾	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 ⁽⁴⁾	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	在 SCKx 边沿之后 SDOx 数据输出有效的的时间	—	—	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: 这些参数为特性值, 但生产时未经测试。
- 2: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。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该规定。
- 4: 假设所有 SPI 引脚上的负载电容均为 50 pF。

图 24-17: SPI 模块主模式 (CKE = 1) 时序特性

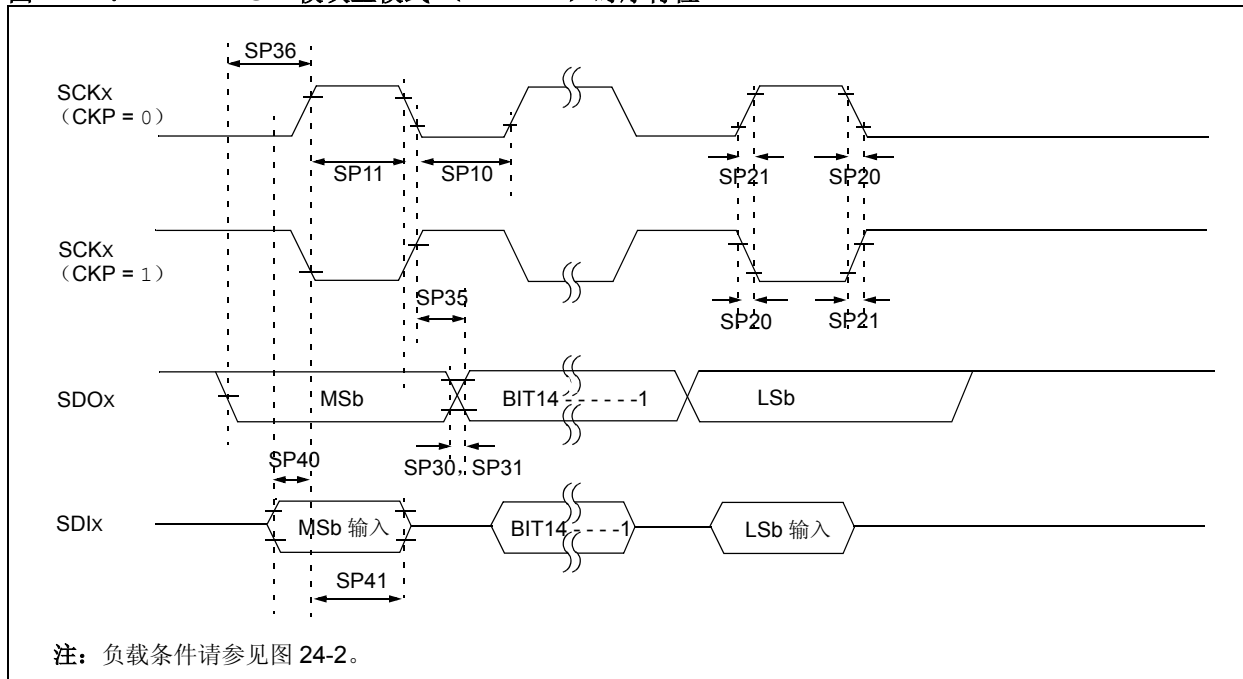


表 24-34: SPI 模块主模式 (CKE = 1) 时序要求

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
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 D032
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 D031
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 D031
SP35	Tsch2doV, TscL2doV	在 SCKx 边沿之后 SDOx 数据输出有效的的时间	—	—	—	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: 这些参数为特性值, 但生产时未经测试。

2: 除非另外声明, 否则“典型值”栏中的数据均为 5V 和 25°C 条件下的值。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该规定。

4: 假设所有 SPI 引脚上的负载电容均为 50 pF。

dsPIC30F5015/5016

图 24-18: SPI 模块从模式 (CKE = 0) 时序特性

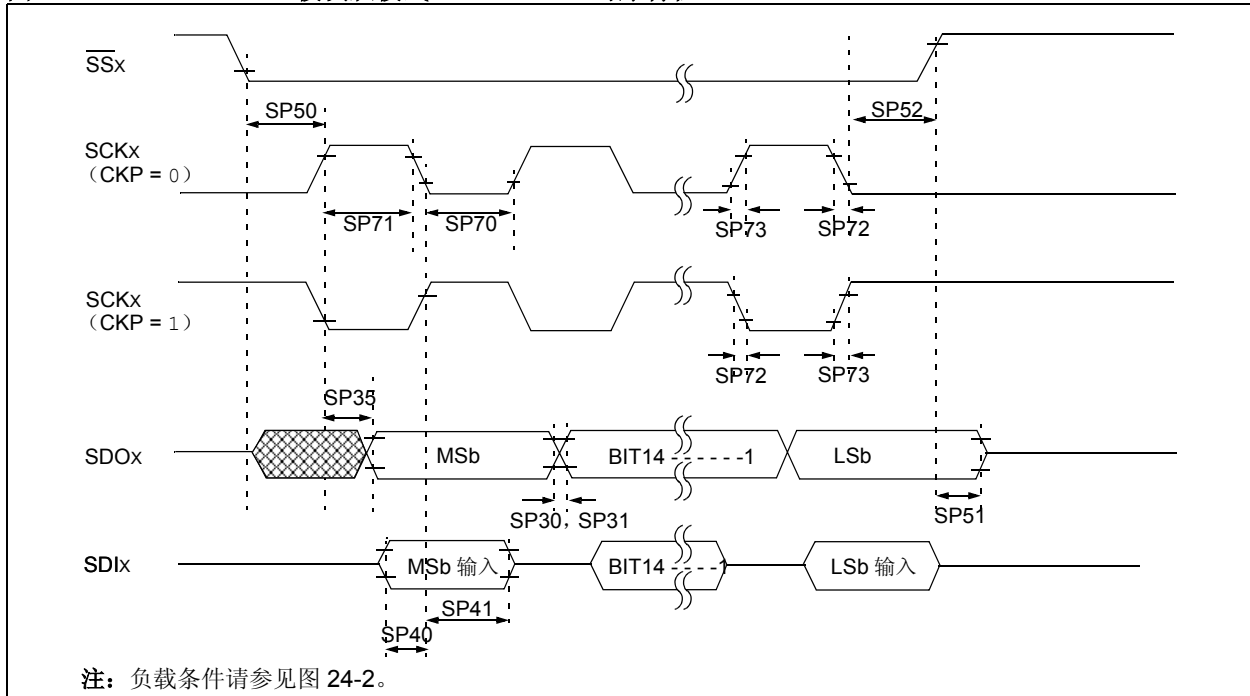


表 24-35: SPI 模块从模式 (CKE = 0) 时序要求

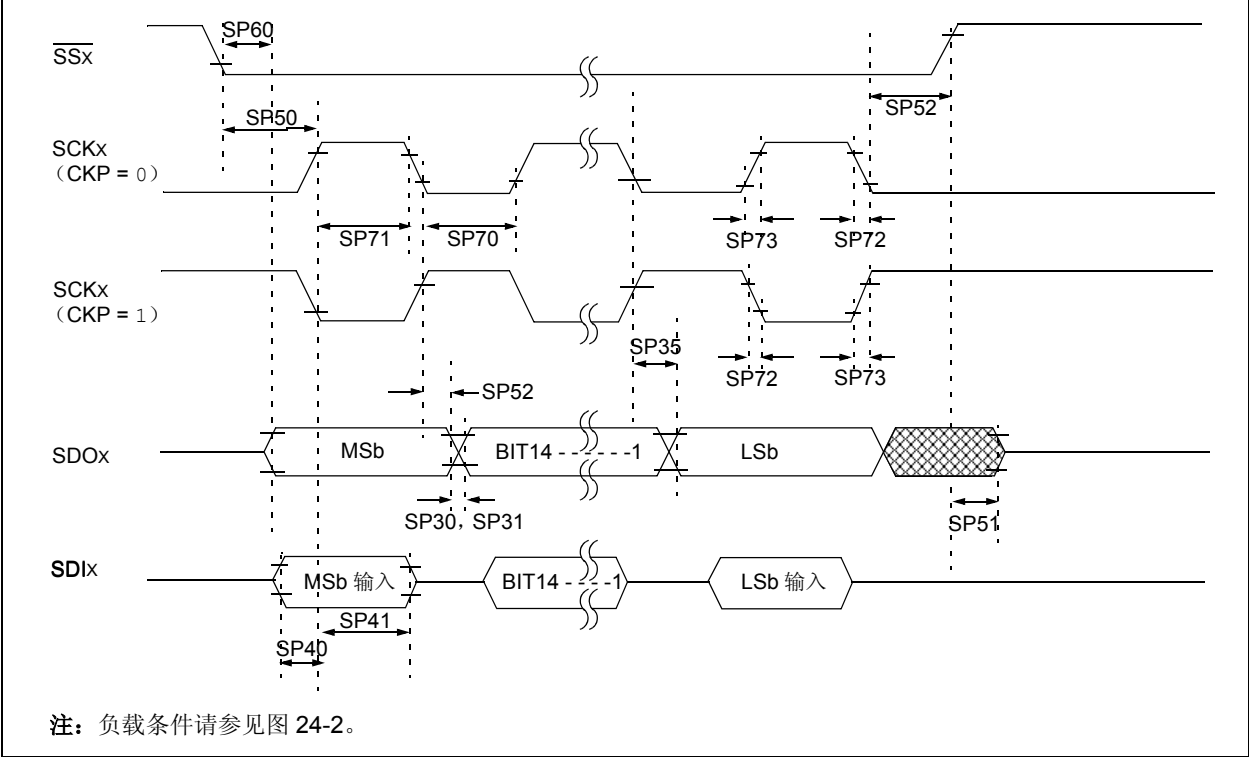
交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 工作温度					
		-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	—
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	—
SP72	TscF	SCKx 输入下降时间 (3)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	25	ns	—
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	在 SCKx 边沿之后 SDOx 数据输出有效的时间	—	—	30	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—
SP50	TssL2scH, TssL2scL	SSx↓ 到 SCKx↑ 或 SCKx↓ 输入的时间	120	—	—	ns	—
SP51	TssH2doZ	SSx↑ 到 SDOx 输出高阻态的时间 (3)	10	—	50	ns	—
SP52	TscH2ssH, TscL2ssH	在 SCK 边沿之后到出现 SSx 的时间	1.5 Tcy + 40	—	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。

2: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25C 条件下的值。参数仅供设计参考, 未经测试。

3: 假设所有 SPI 引脚上的负载电容均为 50 pF。

图 24-19: SPI 模块从模式 (CKE = 1) 时序特性



dsPIC30F5015/5016

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°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 D032
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	在 SCKx 边沿之后 SDOx 数据输出有效的的时间	—	—	30	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—
SP50	TssL2scH, TssL2scL	$\overline{SSx}$ ↓ 到 SCKx↓ 或 SCKx↑ 输入的时间	120	—	—	ns	—
SP51	TssH2doZ	$\overline{SSx}$ ↑ 到 SDOx 输出高阻态的时间 (4)	10	—	50	ns	—
SP52	TscH2ssH, TscL2ssH	在 SCKx 边沿之后到出现 $\overline{SSx}$ ↑ 的时间	1.5 Tcy + 40	—	—	ns	—
SP60	TssL2doV	在 $\overline{SSx}$ 边沿之后 SDOx 数据输出有效的的时间	—	—	50	ns	—

- 注 1: 这些参数为特性值, 但生产时未经测试。
 2: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。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该规定。
 4: 假设所有 SPI 引脚上的负载电容均为 50 pF。

图 24-20: I²C™ 总线启动 / 停止位时序特性（主模式）

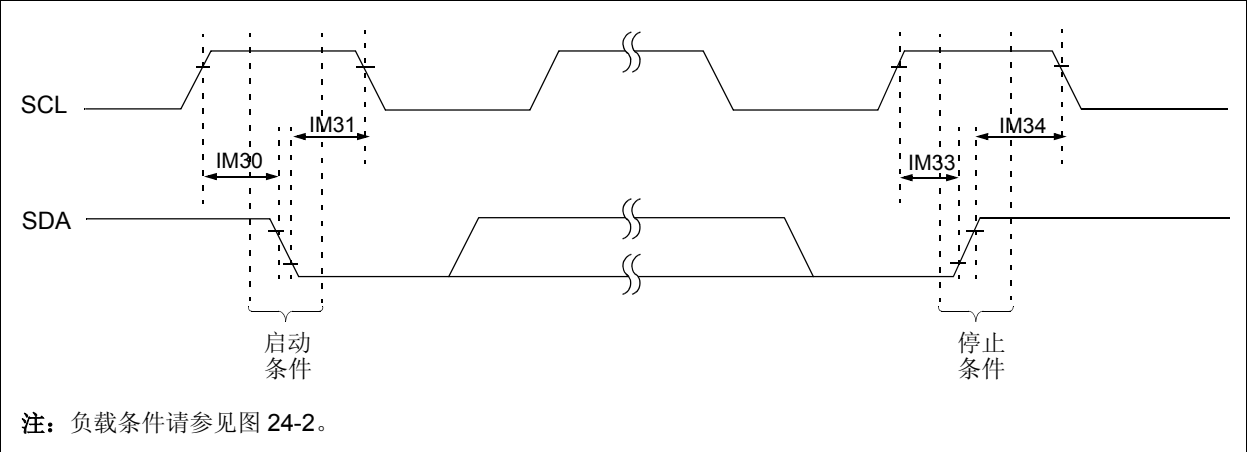
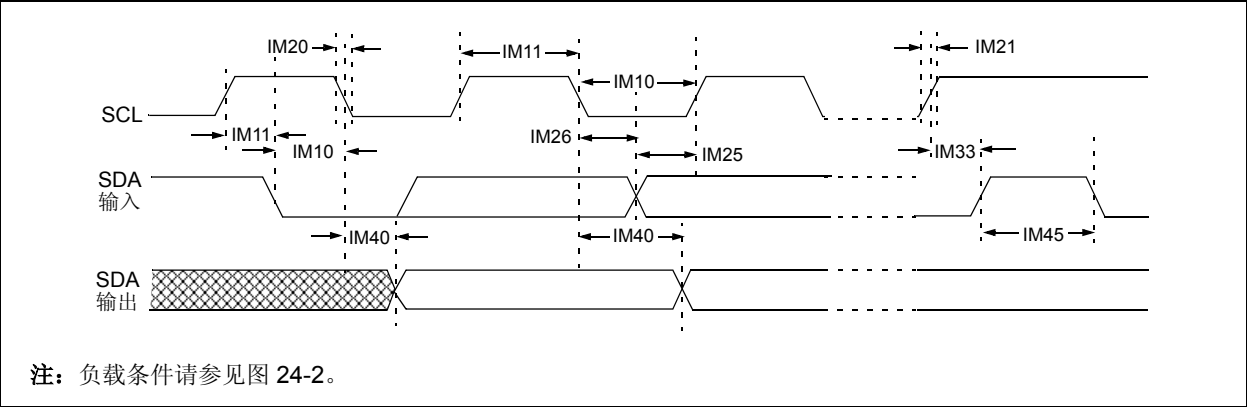


图 24-21: I²C™ 总线数据时序特性（主模式）



dsPIC30F5015/5016

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	C _B 值规定在 10 pF 至 400 pF 之间
			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	C _B 值规定在 10 pF 至 400 pF 之间
			400 kHz 模式	20 + 0.1 C _B	300	ns	
			1 MHz 模式 ⁽²⁾	—	300	ns	
IM25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 ⁽²⁾	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	这个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	时钟输出有效时间	100 kHz 模式	—	3500	ns	—
			400 kHz 模式	—	1000	ns	—
			1 MHz 模式 ⁽²⁾	—	—	ns	—
IM45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	μs	在启动一个新的传输前总线必须保持空闲的时间
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 ⁽²⁾	TBD	—	μs	
IM50	C _B	总线容性负载		—	400	pF	—

图注: TBD = 待定

注 1: BRG 为 I²C 波特率发生器的值。请参见《dsPIC30F 系列参考手册》中的 “I²C™” 章节。

2: 所有 I²C 引脚的最大引脚电容为 10 pF（仅对于 1 MHz 模式）。

图 24-22: I²C™ 总线启动 / 停止位时序特性 (从模式)

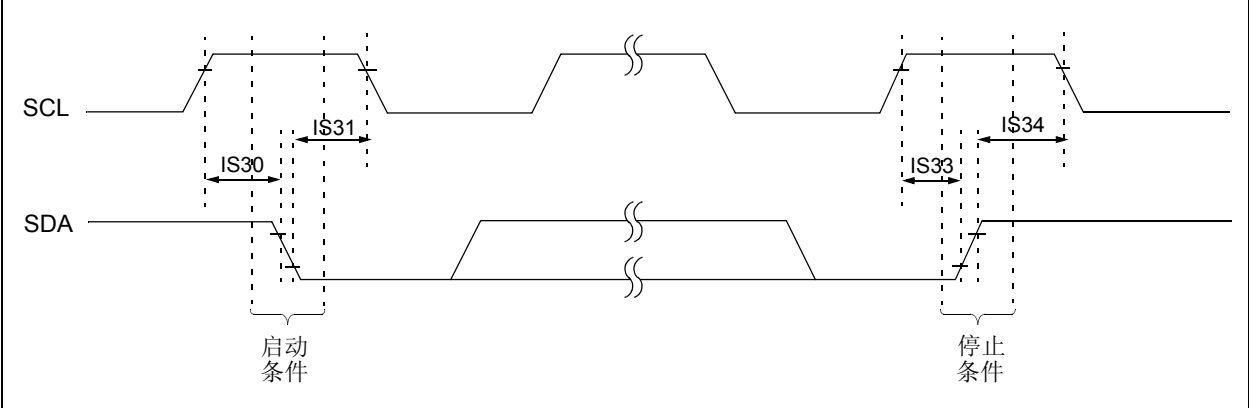
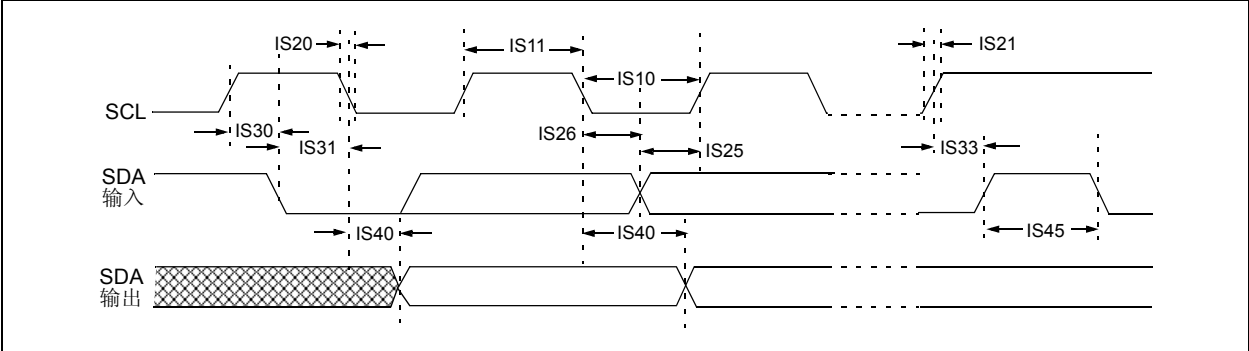


图 24-23: I²C™ 总线数据时序特性 (从模式)



dsPIC30F5015/5016

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	器件工作频率不得低于 1.5 MHz
			400 kHz 模式	1.3	—	μs	器件工作频率不得低于 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	器件工作频率不得低于 1.5 MHz
			400 kHz 模式	0.6	—	μs	器件工作频率不得低于 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	Cb 值规定在 10 pF 至 400 pF 之间
			400 kHz 模式	20 + 0.1 Cb	300	ns	
			1 MHz 模式 ⁽¹⁾	—	100	ns	
IS21	Tr:SCL	SDA 和 SCL 上升时间	100 kHz 模式	—	1000	ns	Cb 值规定在 10 pF 至 400 pF 之间
			400 kHz 模式	20 + 0.1 Cb	300	ns	
			1 MHz 模式 ⁽¹⁾	—	300	ns	
IS25	TSU:DAT	数据输入 建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 ⁽¹⁾	100	—	ns	
IS26	THD:DAT	数据输入 保持时间	100 kHz 模式	0	—	ns	—
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 ⁽¹⁾	0	0.3	μs	
IS30	TSU:STA	启动条件 建立时间	100 kHz 模式	4.7	—	μs	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	这个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	时钟输出 有效时间	100 kHz 模式	0	3500	ns	—
			400 kHz 模式	0	1000	ns	
			1 MHz 模式 ⁽¹⁾	0	350	ns	
IS45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	μs	在启动一个新的传输前总 线必须保持空闲的时间
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 ⁽¹⁾	0.5	—	μs	
IS50	Cb	总线容性负载		—	400	pF	—

注 1: 所有 I²C 引脚的最大引脚电容均为 10 pF（仅对于 1 MHz 模式）。

图 24-24: CAN 模块 I/O 时序特性

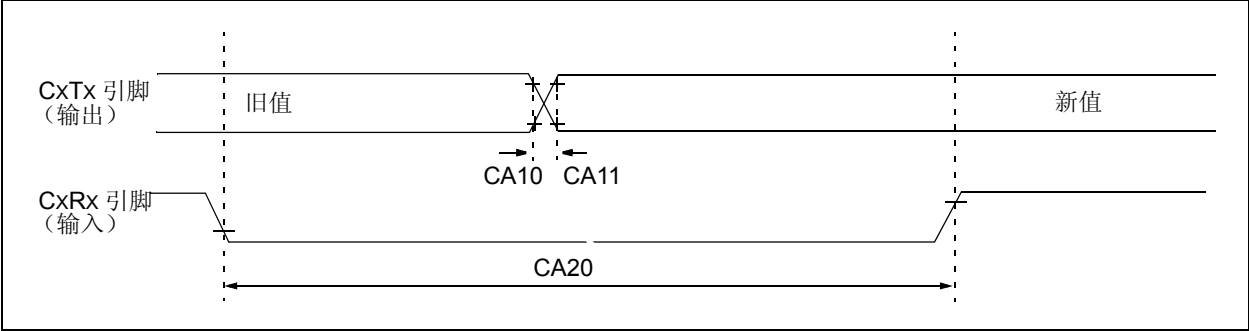


表 24-39: CAN 模块 I/O 时序要求

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 D032
CA11	TioR	端口输出上升时间	—	—	—	ns	见参数 D031
CA20	Tcwf	触发 CAN 唤醒滤波器的脉冲宽度	500			ns	—

注 1: 这些参数为特性值, 但生产时未经测试。
2: 除非另外声明, 否则 “典型值” 栏中的数据均为 5V 和 25°C 条件下的值。参数仅供设计参考, 未经测试。

dsPIC30F5015/5016

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
器件电源							
AD01	AVDD	模块 VDD 电压	VDD - 0.3 或 2.7 的 较大值		VDD + 0.3 或 5.5 的较 小值	V	—
AD02	AVSS	模块 VSS 电压	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 操作 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	模拟电压源阻抗的推荐值	—	—	—	Ω	请参见表 21-1
DC 精度							
AD20	Nr	分辨率	10 数据位			位	—
AD21	INL	积分非线性 ⁽²⁾	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD21A	INL	积分非线性 ⁽²⁾	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD22	DNL	微分非线性 ⁽²⁾	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD22A	DNL	微分非线性 ⁽²⁾	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD23	GERR	增益误差 ⁽²⁾	—	±5	±6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD23A	GERR	增益误差 ⁽²⁾	—	±5	±6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V

注 1: 这些参数为特性值, 但生产时未经测试。

2: 测量采用外部 VREF+, VREF- 用作 ADC 参考电压。

3: A/D 转换结果不会因输入电压的增加而减小, 并且不会丢失编码。

表 24-40: 10 位高速 A/D 模块规范⁽¹⁾ (续)

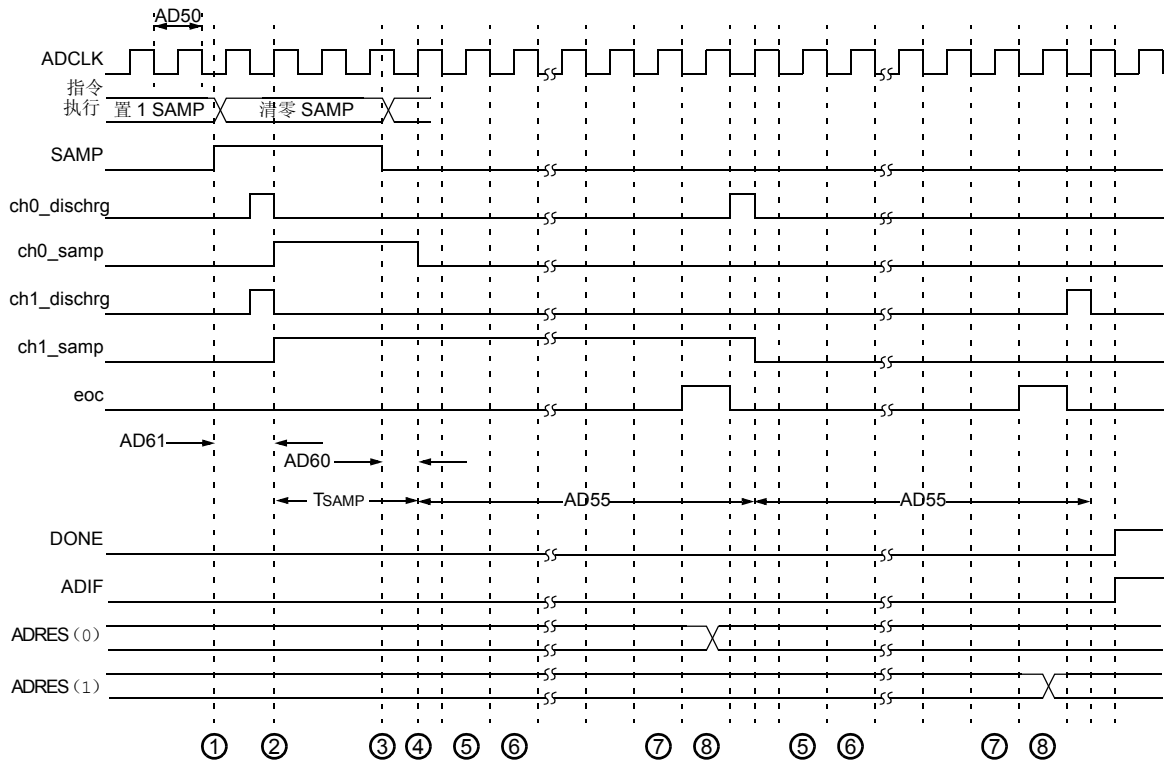
交流特性			标准工作条件: 2.7V 至 5.5V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)				
参数 编号	符号	特性	最小值	典型值	最大值	单位	条件
AD24	E _{OFF}	失调误差 ⁽²⁾	±1	±2	±3	LSb	V _{INL} = AV _{SS} = V _{REFL} = 0V, AV _{DD} = V _{REFH} = 5V
AD24A	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	无杂散信号动态范围	—	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: 这些参数为特性值, 但生产时未经测试。

2: 测量采用外部 V_{REF+}, V_{REF-} 用作 ADC 参考电压。

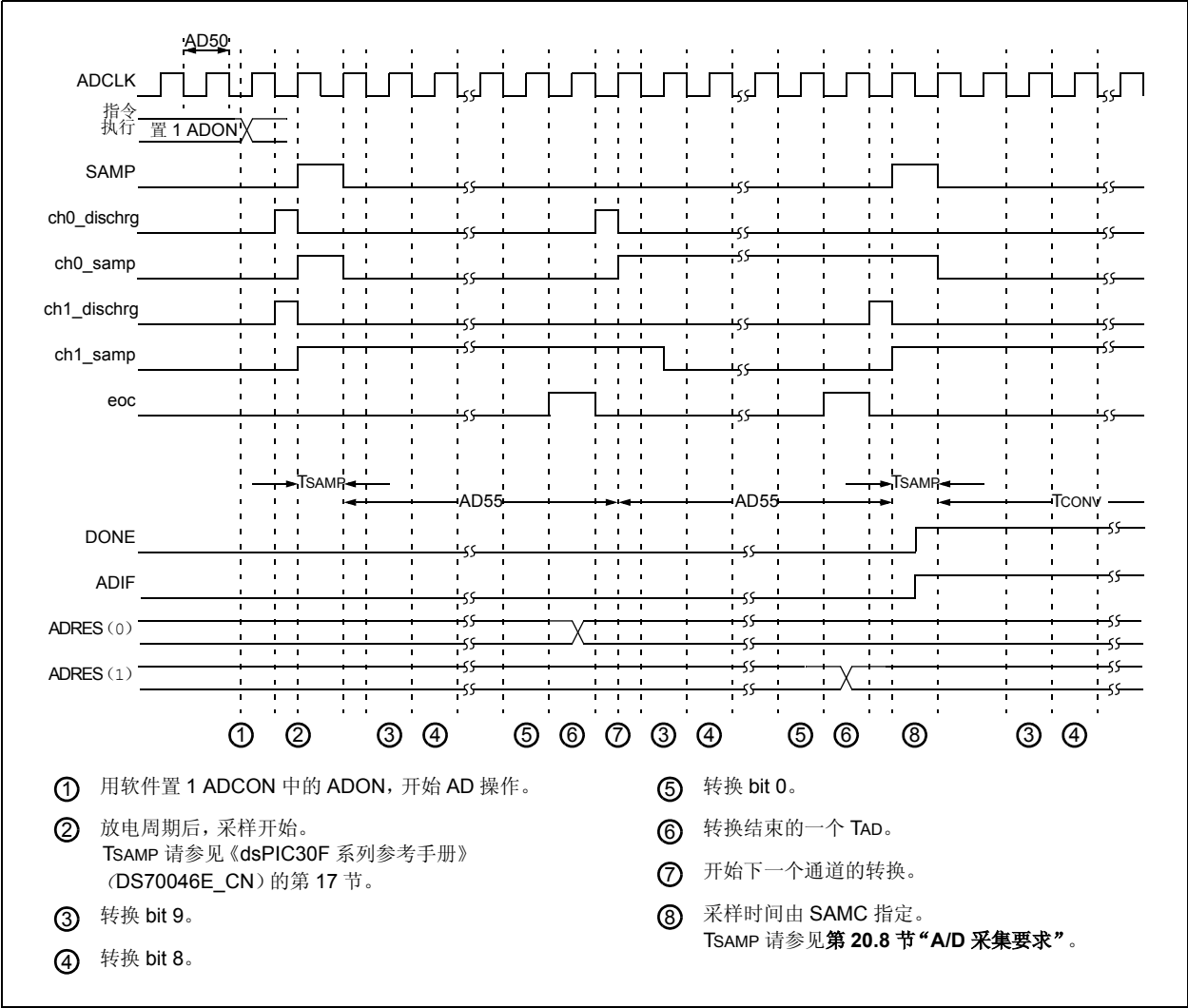
3: A/D 转换结果不会因输入电压的增加而减小, 并且不会丢失编码。

图 24-25: 10 位高速 A/D 转换时序特性
(CHPS = 01, SIMSAM = 0, ASAM = 0, SSRC = 000)



- ① 用软件置 1 ADCON 中的 SAMP，开始采样。
- ② 放电周期后，采样开始。TsAMP 请参见第 20.8 节“A/D 采集要求”。
- ③ 用软件清零 ADCON 中的 SAMP，开始转换。
- ④ 采样结束，转换序列开始。
- ⑤ 转换 bit 9。
- ⑥ 转换 bit 8。
- ⑦ 转换 bit 0。
- ⑧ 转换结束的一个 TAD。

图 24-26: 10 位高速 A/D 转换时序特性
(CHPS = 01, SIMSAM = 0, ASAM = 1, SSRC = 111, SAMC = 00001)



dsPIC30F5015/5016

表 24-41: 10 位高速 A/D 转换时序要求

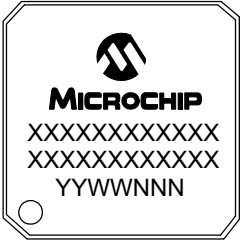
交流特性			标准工作条件: 2.7V 至 5.5V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
时钟参数							
AD50	TAD	A/D 时钟周期	—	83.33 ⁽²⁾	—	ns	见表 21-1 ⁽³⁾
AD51	tRC	A/D 内部 RC 振荡器周期	700	900	1100	ns	—
转换速率							
AD55	tCONV	转换时间	—	12 TAD	—	—	—
AD56	FCNV	吞吐率	—	1.0	—	Msp/s	见表 21-1 ⁽³⁾
AD57	TSAMP	采样时间	—	1 TAD	—	—	见表 21-1 ⁽³⁾
时序参数							
AD60	tPCS	从采样触发到转换开始的时间 ⁽³⁾	—	1.0 TAD	—	—	未选择自动转换触发器 (SSRC = 111)
AD61	tPSS	从采样位 (SAMP) 置 1 到采样开始的时间	0.5 TAD	—	1.5 TAD	—	—
AD62	tCSS	转换完成到采样开始 (ASAM = 1) 的时间 ⁽⁴⁾	—	0.5 TAD	—	ns	—
AD63	tDPU	从 A/D 关闭到 A/D 开启用来稳定模拟级的时间 ⁽⁴⁾	—	20	—	μs	—

- 注 1: 这些参数为特性值, 但生产时未经测试。
2: 工作温度: -40°C 至 +85°C。
3: 因为采样电容最终将放电, 低于 10 kHz 的时钟速率可能影响线性性能, 尤其在温度较高时。
4: 设计特性值, 未经测试。

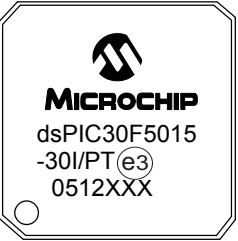
25.0 封装信息

25.1 封装标识信息

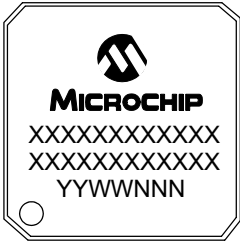
64 引脚 TQFP



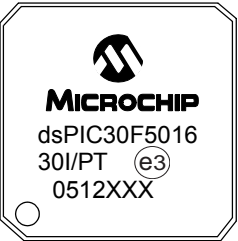
示例



80 引脚 TQFP



示例

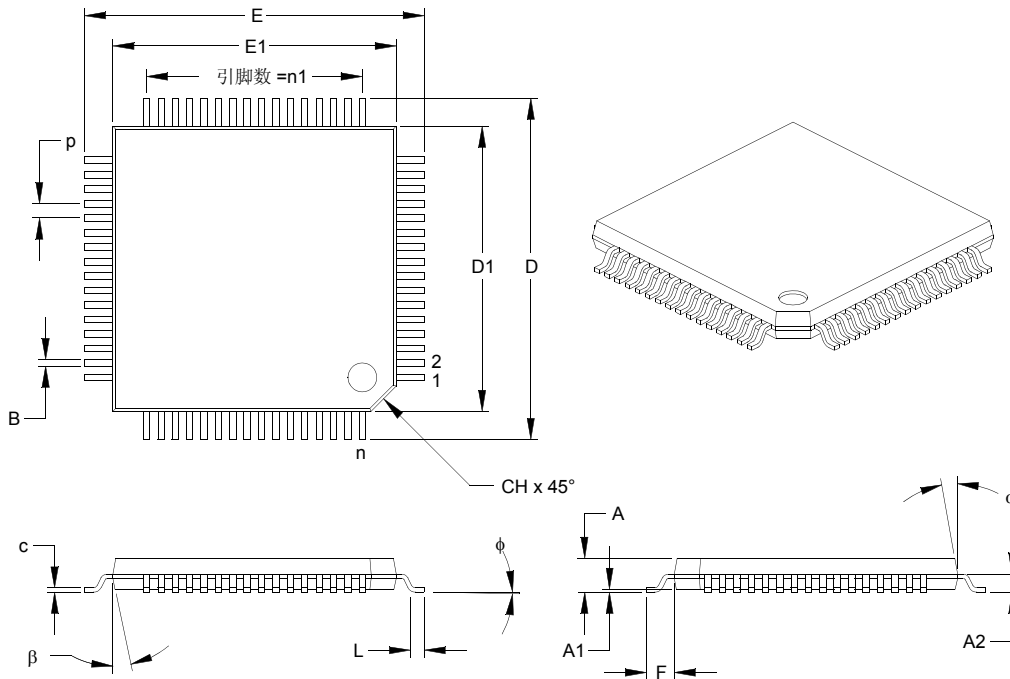


图注：	XX...X	客户指定信息
	Y	年份代码（公历年份的最后一位）
	YY	年份代码（公历年份的最后两位）
	WW	星期代码（1月1日的星期代码为“01”）
	NNN	字母数字的追踪代码
	(e3)	雾锡（Matte Tin, Sn）的无铅 JEDEC 标志
	*	本封装为无铅封装。无铅 JEDEC 标志（(e3)）标示于此种封装的外包装上。
注：如果 Microchip 器件编号不能在一行中完全标出，它将换行继续标出。因此限制了用户指定信息的可用字符数。		

dsPIC30F5015/5016

64 引脚塑封薄型正方扁平封装（PT）10x10x1 mm 主体，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		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
引脚投影长度	F	.039 REF.			1.00 REF.		
底脚倾斜角	phi	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 切角斜面	CH	.025	.035	.045	0.64	0.89	1.14
塑模顶部锥度	alpha	5	10	15	5	10	15
塑模底部锥度	beta	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

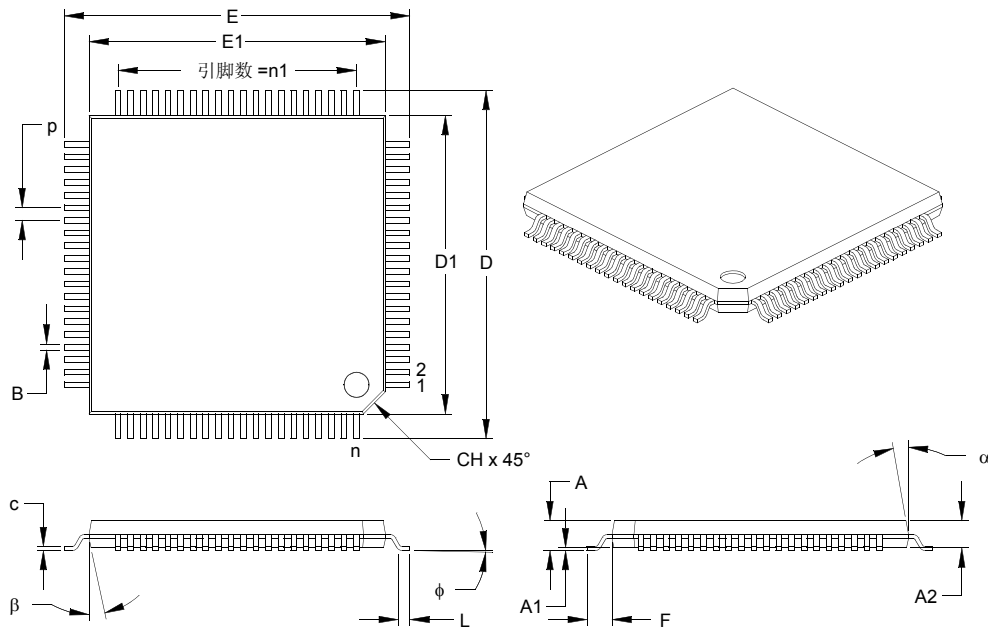
等同于 JEDEC 号：MS-026

图号 C04-085

修订于 07-22-05

80 引脚塑封薄型正方扁平封装（PT） 12x12x1 mm 主体， 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			0.50 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
引脚投影长度	F	.039 REF.			1.00 REF.		
底脚倾斜角	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 切角斜面	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。理论上显示的是没有公差的确切值。

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

dsPIC30F5015/5016

注:

附录 A: 版本历史

版本 A (2005 年 7 月)

dsPIC30F5015 和 dsPIC30F5016 器件数据手册的初始版本。

版本 B (2006 年 9 月)

数据手册的版本 B 反映了如下变化:

- 删除了基本指令 CP1 (见表 21-2)
- 支持的 I²C 从动地址 (见表 17-1)
- ADC 转换时钟选择 (见第 21.0 节 “10 位高速模数转换器 (ADC) 模块”)
- 修改了电器特性部分的:
 - 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)
 - 看门狗定时器限制 (见表 24-21)

dsPIC30F5015/5016

注:

索引

符号

10 位高速模数 (A/D) 转换器模块	149
16 位向上 / 向下位置计数器模式	88
错误检查	88
计数方向状态	88
8 输出 PWM	
寄存器映射	102

A

A/D

ADCHS	149
ADCON1	149
ADCON2	149
ADCON3	149
ADCSSL	149
ADPCFG	149
CPU 空闲模式期间的操作	157
CPU 休眠模式期间的操作	157
采集要求	156
掉电模式	157
复位的影响	157
寄存器映射	159
结果缓冲器	151
连接注意事项	158
配置模拟端口引脚	158
设置转换触发源启动	152
输出格式	157
选择转换时钟	152
选择转换序列	151
中止转换	152
转换过程	151

B

BOR。请参见欠压复位。

版本历史	217
备用 16 位定时器 / 计数器	89
备用向量表 (AIVT)	45
编程模型	16
变更通知客户服务	225
表访问程序数据 (MSB)	26

C

CAN

报文发送	127
错误	127
发送缓冲器	127
发送优先级	127
过程	127
中断	128
中止	127
报文接收	126
接收错误	126
接收过滤屏蔽	126
接收过滤器	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
工作模式	125
初始化	125
错误识别	125
环回	125

监听	125
禁止模式	125
正常	125
同步	129
位时序	128
相位段	129
预分频器设置	129
传播段	129
CAN 模块	123
CAN1 寄存器映射	130
概述	123
帧类型	123
CPU 架构概述	15
C 编译器	
MPLAB C18	170
MPLAB C30	170
操作码说明中使用的符号	162
程序地址空间	23
表指令	
TBLRDH	25
TBLRDL	25
TBLWTH	25
TBLWTL	25
存储器映射	23
构成	24
使用表指令访问程序存储器中的数据	25
使用程序空间可视性访问程序存储器中的数据	26
数据访问自，地址生成	24
程序计数器	16
程序空间可视性	
映射到程序空间的操作	27
除法支持	18
从休眠和空闲模式唤醒	45
从休眠状态唤醒	133
存储器构成	23
D	
dsPIC30F5015 端口	
寄存器映射	61
dsPIC30F5016 端口	
寄存器映射	62
DSP 引擎	18
乘法器	20
数据累加器和加法器 / 减法器	20
累加器回写	21
舍入逻辑	21
数据空间写饱和	22
溢出和饱和	20
代码保护	133
代码示例	
擦除程序存储器的一行	51
读数据 EEPROM	55
启动编程过程	52
数据 EEPROM 块擦除	56
数据 EEPROM 块写入	58
数据 EEPROM 字擦除	56
数据 EEPROM 字写入	57
装入写锁存器	52
地址发生器单元	35
电机控制 PWM 模块	93
电气特性	173
掉电电流 (IPD)	178
读者反馈	226
端口写 / 读示例	60

dsPIC30F5015/5016

F

防止意外写入 OSCCON	139
封装	
标识	213
信息	213
复位	133, 139
BOR, 可编程	141
POR	140
POR, FSCM 和 PWRT 禁止时的操作	141
具有长晶振起振时间的 POR	141
复位过程	43
复位的来源	43

G

公式	
A/D 转换时钟	152
PWM 分辨率	96
PWM 周期	96
PWM 周期 (向上 / 向下模式)	96
波特率	119
串行时钟速率	112
时钟发生的时间份额	129
工作电流 (IDD)	176

H

汇编器	
MPASM 汇编器	170

I

I/O 端口	59
并行 I/O (PIO)	59
I ² C 10 位从模式操作	109
发送	110
接收	110
I ² C 7 位从模式操作	109
发送	109
接收	109
I ² C 模块	107
CPU 休眠和空闲模式期间的操作	112
IPMI 支持	111
编程模型	107
地址	109
各种模式	107
功能说明	107
广播呼叫地址支持	111
寄存器	107
寄存器映射	113
软件控制的时钟延长 (STREN = 1)	110
斜率控制	111
引脚配置	107
中断	111
主模式操作	111
主模式支持	111
I ² C 主模式	
波特率发生器	111
多主器件通信、总线冲突和总线仲裁	112
发送	111
接收	111
时钟仲裁	112

J

简单捕捉事件模式	
Timer2 和 Timer3 选择模式	80
捕捉缓冲器操作	80
捕捉预分频器	79
霍尔传感器模式	80

简单 PWM 模式	84
输入引脚故障保护	84
周期	85
简单输出比较匹配模式	84
交流	
温度和电压规范	182
交流特性	182
PLL 抖动	184
负载条件	182
内部 FRC 抖动、精度和漂移	186
内部 LPRC 精度	186

K

开发支持	169
看门狗定时器 (WDT)	133, 144
操作	144
使能和禁止	144
勘误表	5
可编程	133
可编程数字噪声滤波器	89
客户通知服务	225
客户支持	225
空闲电流 (I _{IDLE})	177
快速现场保护	45
框图	
10 位高速 A/D 功能	150
16 位 Timer2 (B 类定时器)	71
16 位 Timer4 (B 类定时器)	76
16 位 Timer3 (C 类定时器)	71
16 位 Timer5 (C 类定时器)	76
16 位 Timer1 模块 (A 类定时器)	66
32 位 Timer2/3	70
32 位 Timer4/5	75
CAN 缓冲器和协议引擎	124
dsPIC30F5015	8
dsPIC30F5016	11
DSP 引擎	19
I ² C	108
PWM 模块	94
SPI	104
SPI 主 / 从器件连接	104
UART 发送器	115
UART 接收器	116
编程模型	17
复位系统	139
共用端口结构	60
输出比较模式	83
输入捕捉模式	79
外部上电复位电路	142
振荡器系统	135
正交编码器接口	87
专用端口结构	59

M

Microchip 因特网网站	225
MPLAB ASM30 汇编器、链接器和库管理器	170
MPLAB ICD 2 在线调试器	171
MPLAB ICE 2000 高性能通用在线仿真器	171
MPLAB PM3 器件编程器	171
MPLAB REAL ICE 在线仿真器系统	171
MPLAB 集成开发环境软件	169
MPLINK 目标链接器 /MPLIB 目标库管理器	170
寻址	
W 地址寄存器选择	37
操作示例	37
起始地址和结束地址	37

应用	38	CALL 堆栈帧	31
N		软件模拟器 (MPLAB SIM)	170
NVM		S	
寄存器映射	53	SPI 模块	103
内核		CPU 空闲模式期间的操作	105
寄存器映射	32	CPU 休眠模式期间的操作	105
内核概述	15	SDOx 禁止	103
P		SPI1 寄存器映射	106
PICSTART 2 开发编程器	172	SPI2 寄存器映射	106
PICSTART Plus 开发编程器	172	从选择同步	105
POR. 请参见上电复位。		功能说明	103
PWM		帧 SPI 支持	105
CPU 空闲模式期间的操作	101	字和字节通信	103
CPU 休眠模式期间的操作	101	状态寄存器	16
边沿对齐	96	闪存程序存储器	49
单脉冲操作	99	编程算法	51
独立输出	99	表指令操作汇总	49
更新锁定	101	擦除一行	51
故障引脚	100	操作	51
故障状态	100	启动编程序列	52
使能位	100	装入写锁存器	52
输入模式	100	上电复位 (POR)	133
优先级	100	上电延时定时器 (PWRT)	133
互补操作	98	振荡器起振定时器 (OST)	133
时基	95	省电模式	144
单事件模式	95	空闲	145
后分频器	96	休眠	144
连续向上 / 向下计数模式	95	省电模式 (休眠和空闲)	133
双更新模式	96	时序规范	
预分频器	96	PLL 时钟	184
自由运行模式	95	时序图	
输出改写	99	10 位高速 A/D 转换 (CHPS = 01, SIMSAM = 0, ASAM = 0, SSRC = 000)	210
互补输出模式	99	10 位高速 A/D 转换 (CHPS = 01, SIMSAM = 0, ASAM = 1, SSRC = 00001)	211
同步	99	CAN 模块 I/O	207
输出和极性控制	100	CAN 位	128
输出引脚控制	100	CLKOUT 和 I/O	187
死区时间发生器	98	I ² C 总线启动 / 停止位 (从模式)	205
范围	98	I ² C 总线启动 / 停止位 (主模式)	203
分配	98	I ² C 总线数据 (从模式)	205
选择位	98	I ² C 总线数据 (主模式)	203
特殊事件触发器	101	OC/PWM 模块	194
后分频器	101	PWM 输出	85
占空比比较单元	97	QEA/QEB 输入特性	196
寄存器缓冲	97	QEI 模块索引脉冲	197
立即更新	97	SPI 从模式 (CKE = 0)	200
中心对齐	97	SPI 从模式 (CKE = 1)	201
周期	96	SPI 主模式 (CKE = 0)	198
配置模拟端口引脚	60	SPI 主模式 (CKE = 1)	199
Q		TimerQ (QEI 模块) 外部时钟	192
器件概述	7	Timer1、2、3、4 和 5 外部时钟	190
器件 ID 存储单元	133	边沿对齐的 PWM	96
器件配置		带隙启动时间	189
寄存器映射	147	电机控制 PWM 模块	195
器件配置寄存器	145	电机控制 PWM 模块故障	195
FBORPOR	145	复位、看门狗定时器、振荡器起振定时器和上电延时定时器	188
FGS	145	欠压复位	180
FOSC	145	上电时的延时时序 (MCLR 不连接到 VDD), 情形 1	140
FWDTC	145	上电时的延时时序 (MCLR 不连接到 VDD), 情形 2	141
欠压复位 (BOR)	133	上电时的延时时序 (MCLR 连接到 VDD)	140
R		输出比较 (OCx)	193
RCON 寄存器的初始化状态, 情形 1	143		
RCON 寄存器的初始化状态, 情形 2	143		
软件堆栈指针, 帧指针	16		

dsPIC30F5015/5016

输入捕捉 (CAPx)	193	单脉冲模式	84
死区时间	99	连续脉冲模式	84
外部时钟	182		
中心对齐的 PWM	97	T	
时序要求		Timer2/3 模块	69
A/D 转换	212	16 位模式	69
CAN 模块 I/O	207	32 位定时器模式	69
CLKOUT 和 I/O	187	32 位同步计数器模式	69
I ² C 总线数据 (从模式)	206	ADC 事件触发信号	72
I ² C 总线数据 (主模式)	204	定时器预分频器	72
QE1 模块索引脉冲	197	寄存器映射	73
QE1 模块外部时钟	192	门控操作	72
SPI 从模式 (CKE = 0)	200	休眠模式下的操作	72
SPI 从模式 (CKE = 1)	202	中断	72
SPI 主模式 (CKE = 0)	198	Timer4/5 模块	75
SPI 主模式 (CKE = 1)	199	寄存器映射	77
Timer2 和 Timer4 外部时钟	191	Timer1 模块	65
Timer3 和 Timer5 外部时钟	191	16 位定时器模式	65
Timer1 外部时钟	190	16 位同步计数器模式	65
带隙启动时间	189	16 位异步计数器模式	65
电机控制 PWM 模块	195	寄存器映射	68
复位、看门狗定时器、振荡器起振定时器、		门控操作	66
上电延时定时器和欠压复位	189	实时时钟	67
简单 OC/PWM 模式	194	振荡器操作	67
输出比较	193	中断	67
输入捕捉	193	休眠模式下的操作	66
外部时钟	183	预分频器	66
正交编码器	196	中断	67
输出比较模块	83	桶形移位寄存器	22
CPU 空闲模式期间的操作	85	通用异步收发器 (UART) 模块	115
CPU 休眠模式期间的操作	85		
Timer2、Timer3 选择模式	84	U	
寄存器映射	86	UART	
中断	85	CPU 休眠和空闲模式期间的操作	120
数据地址空间	27	UART1 寄存器映射	121
MCU 和 DSP (MAC 类) 指令示例	29	波特率发生器	119
Near 数据空间	31	地址检测模式	119
存储器映射	27, 28	发送数据	117
对齐	30	发送缓冲器 (UxTXB)	117
对齐 (图)	30	在 8 位数据模式下	117
空间	30	在 9 位数据模式下	117
宽度	30	中断	118
软件堆栈	31	环回模式	119
无效存储器访问的结果	30	接收错误处理	118
数据 EEPROM 存储器	55	接收缓冲器溢出错误 (OERR 位)	118
擦除	56	接收间隔	119
擦除, 块	56	空闲状态	119
擦除, 字	56	奇偶性错误 (PERR 位)	119
读	55	帧错误 (FERR 位)	119
防止误写操作的保护措施	58	接收数据	118
写	57	接收缓冲器 (UxRXB)	118
写校验	58	在 8 位或 9 位数据模式下	118
写, 块	58	中断	118
写, 字	57	禁止	117
输入变化通知模块	63	模块概述	115
寄存器映射 (dsPIC30F5015 的 Bit 15-8)	63	设置数据、奇偶性和停止位选择	117
寄存器映射 (dsPIC30F5015 的 Bit 7-0)	63	使能	117
寄存器映射 (dsPIC30F5016 的 Bit 15-8)	63	自动波特率支持	120
寄存器映射 (dsPIC30F5016 的 Bit 7-0)	63		
输入捕捉模块	79	W	
寄存器映射	81	WWW 地址	225
简单捕捉事件模式	79	WWW, 在线支持	5
休眠和空闲模式期间的操作	80	外设模块禁止 (PMD) 寄存器	146
中断	80	位反转寻址	38
双输出比较匹配模式	84	XBREV 寄存器的修改量	39
		示例	38

实现	38	dsPIC30F5016	174
序列表（16 项）	39	空闲电流（IDLE）	177
位置测量模式	88	热工作条件	174
X		温度和电压规范	175
系统集成	133	中断	41
寄存器映射	147	过程	45
陷阱	43	控制器	
来源	43	dsPIC30F5015 的寄存器映射	46
向量	44	dsPIC30F5016 的寄存器映射	47
硬陷阱和软陷阱	44	外部请求	45
Y		优先级	42
引脚说明		中断堆栈帧	45
dsPIC30F5015	9	中断向量表（IVT）	45
dsPIC30F5016	12	自动时钟延长	110
因特网地址	225	10 位寻址模式时（STREN = 1）	110
运行时自编程（RTSP）	49	7 位寻址模式时（STREN = 1）	110
工作原理	50	发送模式	110
控制寄存器	50	接收模式	110
NVMADR	50		
NVMADRU	50		
NVMCON	50		
NVMKEY	50		
Z			
在线串行编程（ICSP）	49, 133		
在线调试器	146		
振荡器			
工作模式（表）	134		
系统概述	133		
振荡器配置	136		
LP 振荡器控制	137		
初始时钟源选择	136		
低功耗 RC（LPRC）	138		
故障保护时钟监视器	138		
快速 RC 振荡器（FRC）	137		
起振定时器（OST）	137		
锁相环（PLL）	137		
振荡器选择	133		
正交编码器接口（QEI）	87		
CPU 空闲模式期间的操作	89		
CPU 空闲模式期间的定时器操作	90		
CPU 休眠模式期间的操作	89		
CPU 休眠模式期间的定时器操作	89		
寄存器映射	91		
逻辑	88		
中断	90		
指令集			
概述	164		
汇总	161		
指令寻址模式	35		
MAC 指令	36		
MCU 指令	35		
其他指令	36		
文件寄存器指令	35		
支持的基本模式	35		
传送指令和累加器指令	36		
直流特性	174		
I/O 引脚输出规范	180		
I/O 引脚输入规范	179		
程序和 EEPROM	181		
掉电电流（IPD）	178		
工作电流（IDD）	176		
工作 MIPS 与电压	174		
dsPIC30F5015	174		

dsPIC30F5015/5016

注:

MICROCHIP 网站

Microchip 网站 (www.microchip.com) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的因特网浏览器即可访问。网站提供以下信息:

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及存档软件
- **一般技术支持**——常见问题 (FAQ)、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排、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 的客户通知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)”服务并按照注册说明完成注册。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助:

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过 <http://support.microchip.com> 获得网上技术支持。

dsPIC30F5015/5016

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： dsPIC30F5015/5016

文献编号：

DS70149B_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？

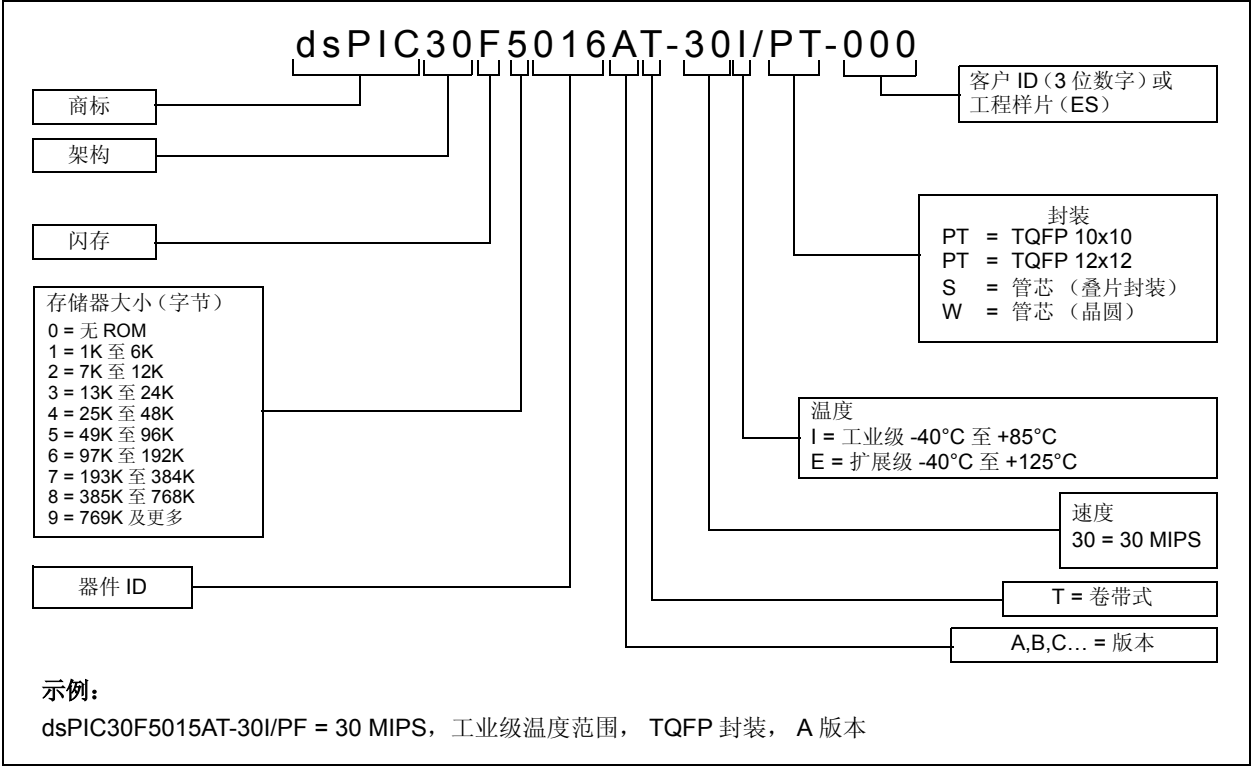
5. 您认为本文档中可以删减哪些内容，而又不会影响整体使用效果？

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

全球销售及服务中心

美洲

公司总部 **Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277

技术支持:
<http://support.microchip.com>
网址: www.microchip.com

亚特兰大 Atlanta
Duluth, GA

Tel: 678-957-9614
Fax: 678-957-1455

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 Kokomo
Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

加拿大多伦多 Toronto
Mississauga, Ontario,
Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 **Asia Pacific Office**
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 北京
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 - 成都
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 福州
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

中国 - 香港特别行政区
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 南京
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 - 顺德
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

中国 - 武汉
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 西安
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

台湾地区 - 高雄
Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾地区 - 台北
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

台湾地区 - 新竹
Tel: 886-3-572-9526
Fax: 886-3-572-6459

亚太地区

澳大利亚 Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

印度 India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama
Tel: 81-45-471-6166
Fax: 81-45-471-6122

韩国 Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark-Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

10/05/07

1.0	器件概述	7
2.0	CPU 架构概述	15
3.0	存储器构成	23
4.0	地址发生器单元	35
5.0	中断	41
6.0	闪存程序存储器	49
7.0	数据 EEPROM 存储器	55
8.0	I/O 端口	59
9.0	Timer1 模块	65
10.0	Timer2/3 模块	69
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 (QEI) 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	系统集成	133
21.0	12 位高速模数转换器 (ADC) 模块	149
22.0	指令集汇总	161
23.0	开发支持	169
24.0	电气特性	173
25.0	封装信息	213
附录 A:		217
索引		219
Microchip 网站		225
变更通知客户服务		225
客户支持		225
读者反馈表		226
产品标识体系		227