

## 带互连的高速 PWM、ADC、PGA 和比较器且 适用于数字电源应用的 16 位数字信号控制器

### 工作条件

- 3.0V 至 3.6V, -40°C 至 +85°C, DC 至 70 MIPS
- 3.0V 至 3.6V, -40°C 至 +125°C, DC 至 60 MIPS

### 闪存架构

- 具有在线更新功能的双分区闪存程序存储器：
  - 支持工作时编程
  - 支持分区软交换

### 内核：16 位 dsPIC33E CPU

- 高效代码型（C 和汇编）架构
- 两个 40 位宽累加器
- 带双数据取操作的单周期（MAC/MPY）
- 单周期混合符号乘法和硬件除法
- 32 位乘法支持
- 4 个附加工作寄存器组（减少现场切换）

### 时钟管理

- $\pm 0.9\%$  精度内部振荡器
- 可编程 PLL 和振荡器时钟源
- 故障保护时钟监视器（Fail-Safe Clock Monitor, FSCM）
- 独立看门狗定时器（Watchdog Timer, WDT）
- 快速唤醒和启动

### 功耗管理

- 低功耗管理模式（休眠、空闲和打盹）
- 集成上电复位和欠压复位
- 0.5 mA/MHz 动态电流（典型值）
- 20  $\mu$ A IPD 电流（典型值）

### 高速 PWM

- 8 个 PWM 发生器（每个发生器两个输出）
- 每个 PWM 具有独立的时基和占空比
- 1.04 ns PWM 分辨率（频率、占空比、死区和相位）
- 支持中心对齐输出模式、冗余输出模式、互补输出模式和真正独立输出模式
- 独立的故障输入和限流输入
- 输出改写控制
- 为交流 / 直流、直流 / 直流、逆变器、PFC 和照明提供 PWM 支持

### 高级模拟特性

- 高速 ADC 模块：
  - 12 位分辨率，带有 4 个专用 SAR ADC 内核和 1 个共用 SAR ADC 内核
  - 可为每个 ADC 内核配置分辨率（最高 12 位）
  - 12 位分辨率时每个通道的转换速率最高为 3.25 Msps
  - 11 至 22 个单端输入
  - 每个模拟通道都具有专用的结果缓冲区
  - 灵活、独立的 ADC 触发源
  - 两个数字比较器
  - 两个用于提高分辨率的过采样滤波器
- 4 个带滞后的轨到轨比较器：
  - 每个模拟比较器均带有专用的 12 位数模转换器（Digital-to-Analog Converter, DAC）
  - 最多两个 DAC 参考输出
  - 最多两个外部参考输入
- 两个可编程增益放大器：
  - 单端或独立地参考
  - 5 个可选增益（4x、8x、16x、32x 和 64x）
  - 40 MHz 增益带宽

### 互连的 SMPS 外设

- 减少与 CPU 的交互以提高性能
- 用于 ADC 转换的灵活 PWM 触发选项
- 高速比较器截断 PWM（典型值为 15 ns）：
  - 支持逐周期电流模式控制
  - 电流复位模式（可变频率）

### 定时器 / 输出比较 / 输入捕捉

- 5 个 16 位定时器 / 计数器，最多两个 32 位定时器 / 计数器
- 4 个输出比较（Output Compare, OC）模块，可配置为定时器 / 计数器
- 4 个输入捕捉（Input Capture, IC）模块

# dsPIC33EPXXGS70X/80X 系列

## 通信接口

- 两个 UART 模块（15 Mbps）：
  - 支持 LIN/J2602 协议和 IrDA®
- 3 个可变宽度 SPI 模块，具有以下工作模式：
  - 3 线 SPI
  - 8x16 或 8x8 FIFO 模式
  - I<sup>2</sup>S 模式
- 两个带有 SMBus 支持的 I<sup>2</sup>C 模块（最高 1 Mbaud）
- 最多两个 CAN 模块
- 四通道 DMA

## 输入 / 输出

- 恒流源（标称值为 10 μA）
- 最大灌电流 / 拉电流分别为 12 mA/15 mA；对于标准 V<sub>OH</sub>/V<sub>OL</sub> 取决于具体引脚
- 5V 耐压引脚
- 可选的漏极开路上拉和下拉
- 所有 I/O 引脚上均具有外部中断
- 外设引脚选择（Peripheral Pin Select, PPS），支持通过 6 个虚拟 I/O 进行功能重映射

## 认证和 B 类支持

- 支持 AEC-Q100 REVG（1 级，-40°C 至 +125°C）
- B 类安全库，IEC 60730
- 6x6x0.55 mm UQFN 封装经过特殊设计和优化，有助于通过 IPC9592B 第 2 级温度循环认证

## 调试器开发支持

- 在线和在应用编程
- 5 个程序断点和 3 个复杂数据断点
- 符合 IEEE 1149.2 的（JTAG）边界扫描
- 跟踪和运行时观察

## 数字外设

- 4 个可配置逻辑单元
- 外设触发信号发生器

器件	引脚	程序存储器 (字节)	RAM (字节)	可重映射的外设										I <sup>2</sup> C	CLC	PTG	12 位 ADC		PGA	DMA	模拟比较器	DAC 输出	恒流源	封装
				通用 I/O (GPIO)	定时器 (1)	输入捕捉	输出比较	UART	SPI	PWM (2)	外部中断 (3)	CAN	参考时钟				模拟输入	S&H 电路						
dsPIC33EP128GS702	28	128K	8K	20	5	4	4	2	3	8x2	4	0	1	2	4	1	11	5	2	0	4	1	1	SOIC, QFN-S, UQFN
dsPIC33EP64GS804	44	64K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	QFN, TQFP
dsPIC33EP128GS704	44	128K	8K	33	5	4	4	2	3	8x2	4	0	1	2	4	1	17	5	2	0	4	1	1	
dsPIC33EP128GS804	44	128K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	TQFP
dsPIC33EP64GS805	48	64K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	
dsPIC33EP128GS705	48	128K	8K	33	5	4	4	2	3	8x2	4	0	1	2	4	1	17	5	2	0	4	1	1	TQFP
dsPIC33EP128GS805	48	128K	8K	33	5	4	4	2	3	8x2	4	2	1	2	4	1	17	5	2	4	4	1	1	
dsPIC33EP64GS806	64	64K	8K	51	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	TQFP
dsPIC33EP128GS706	64	128K	8K	51	5	4	4	2	3	8x2	4	0	1	2	4	1	22	5	2	0	4	2	1	
dsPIC33EP128GS806	64	128K	8K	51	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	TQFP
dsPIC33EP64GS708	80	64K	8K	67	5	4	4	2	3	8x2	4	0	1	2	4	1	22	5	2	0	4	2	1	
dsPIC33EP64GS808	80	64K	8K	67	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	TQFP
dsPIC33EP128GS708	80	128K	8K	67	5	4	4	2	3	8x2	4	0	1	2	4	1	22	5	2	0	4	2	1	
dsPIC33EP128GS808	80	128K	8K	67	5	4	4	2	3	8x2	4	2	1	2	4	1	22	5	2	4	4	2	1	TQFP

注 1: Timer1、Timer2 和 Timer3 的外部时钟是可重映射的。

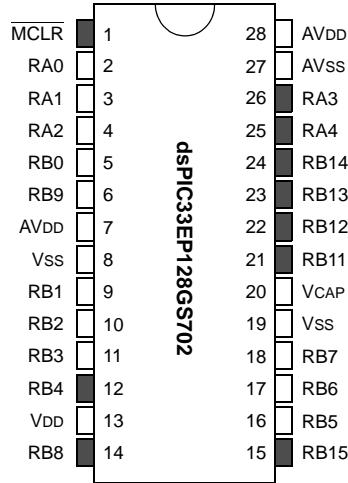
2: 在 28/44/48 引脚器件上，PWM4 至 PWM8 是可重映射的；在 64 引脚器件上，只有 PWM7/PWM8 是可重映射的。

3: 外部中断 INT0 和 INT4 是不可重映射的。

# dsPIC33EPXXGS70X/80X 系列

## 引脚图

### 28 引脚 SOIC



引脚	引脚功能	引脚	引脚功能
1	MCLR	15	PGEC3/SCL2/RP47/RB15
2	AN0/CMP1A/PGA1P1/RP16/RA0	16	TDO/AN19/PGA2N2/RP37/RB5
3	AN1/CMP1B/PGA1P2/PGA2P1/RP17/RA1	17	PGED1/TDI/AN20/SCL1/RP38/RB6
4	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/RP18/RA2	18	PGEC1/AN21/SDA1/RP39/RB7
5	AN3/CMP1D/CMP2B/PGA2P3/RP32/RB0	19	VSS
6	AN4/CMP2C/CMP3A/ISRC4/RP41/RB9	20	VCAP
7	AVDD	21	TMS/PWM3H/RP43/RB11
8	VSS	22	TCK/PWM3L/RP44/RB12
9	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/RP33/RB1	23	PWM2H/RP45/RB13
10	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/RP34/RB2	24	PWM2L/RP46/RB14
11	PGED2/DACOUT1/AN18/INT0/RP35/RB3	25	PWM1H/RP20/RA4
12	PGEC2/ADTRG31/EXTREF1/RP36/RB4	26	PWM1L/RP19/RA3
13	VDD	27	AVSS
14	PGED3/SDA2/FLT31/RP40/RB8	28	AVDD

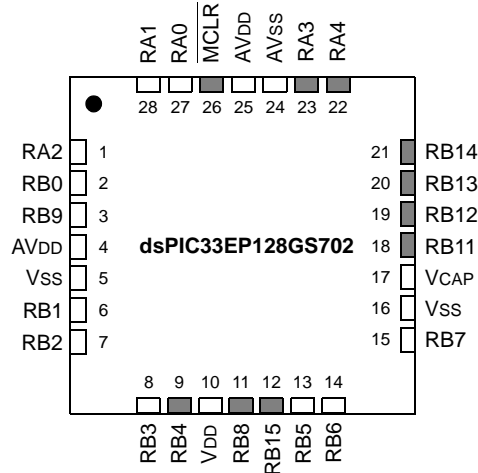
图注：阴影引脚最高可承受 5 VDC。

RPn 表示可重映射外设功能。有关可重映射源的完整列表，请参见表 11-12 和表 11-13。

# dsPIC33EPXXGS70X/80X 系列

## 引脚图 (续)

### 28 引脚 QFN-S 和 UQFN



引脚	引脚功能	引脚	引脚功能
1	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/ <b>RP18</b> /RA2	15	PGEC1/AN21/SDA1/ <b>RP39</b> /RB7
2	AN3/CMP1D/CMP2B/PGA2P3/ <b>RP32</b> /RB0	16	VSS
3	AN4/CMP2C/CMP3A/ISRC4/ <b>RP41</b> /RB9	17	VCAP
4	AVDD	18	TMS/PWM3H/ <b>RP46</b> /RB11
5	VSS	19	TCK/PWM3L/ <b>RP44</b> /RB12
6	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/ <b>RP33</b> /RB1	20	PWM2H/ <b>RP45</b> /RB13
7	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/ <b>RP34</b> /RB2	21	PWM2L/ <b>RP46</b> /RB14
8	PGED2/DACOUT1/AN18/INT0/ <b>RP35</b> /RB3	22	PWM1H/ <b>RP20</b> /RA4
9	PGEC2/ADTRG31/EXTREF1/ <b>RP36</b> /RB4	23	PWM1L/ <b>RP19</b> /RA3
10	VDD	24	AVSS
11	PGED3/SDA2/FLT31/ <b>RP40</b> /RB8	25	AVDD
12	PGEC3/SCL2/ <b>RP47</b> /RB15	26	MCLR
13	TDO/AN19/PGA2N2/ <b>RP37</b> /RB5	27	AN0/CMP1A/PGA1P1/ <b>RP16</b> /RA0
14	PGED1/TDI/AN20/SCL1/ <b>RP38</b> /RB6	28	AN1/CMP1B/PGA1P2/PGA2P1/ <b>RP17</b> /RA1

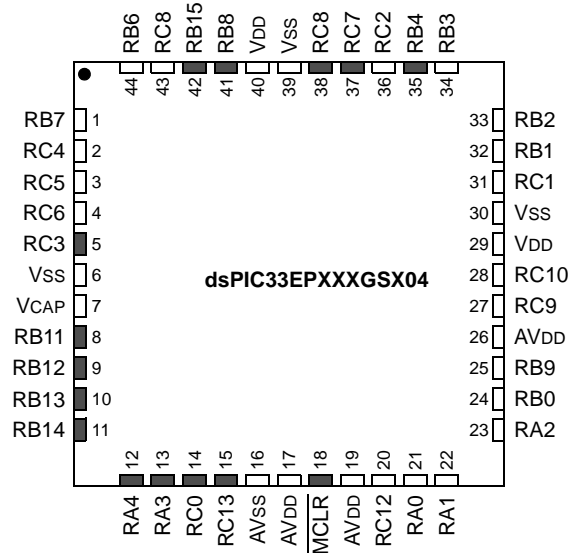
图注: 阴影引脚最高可承受 5 VDC。

RPn 表示可重映射外设功能。有关可重映射源的完整列表, 请参见表 11-12 和表 11-13。

# dsPIC33EPXXGS70X/80X 系列

## 引脚图 (续)

### 44 引脚 QFN 和 TQFP



引脚	引脚功能	引脚	引脚功能
1	PGEC1/AN21/SDA1/RP39/RB7	23	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/RP18/RA2
2	AN1ALT/RP52/RC4	24	AN3/CMP1D/CMP2B/PGA2P3/RP32/RB0
3	AN0ALT/RP53/RC5	25	AN4/CMP2C/CMP3A/ISRC4/RP41/RB9
4	AN17/RP54/RC6	26	AVDD
5	RP51/RC3	27	AN11/PGA1N3/RP57/RC9
6	Vss	28	EXTREF2/AN10/PGA1P4/RP58/RC10
7	VCAP	29	VDD
8	TMS/PWM3H/RP43/RB11	30	Vss
9	TCK/PWM3L/RP44/RB12	31	AN8/CMP4C/PGA2P4/RP49/RC1
10	PWM2H/RP45/RB13	32	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/RP33/RB1
11	PWM2L/RP46/RB14	33	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/RP34/RB2
12	PWM1H/RP20/RA4	34	PGED2/DACOUT1/AN18/INT0/RP35/RB3
13	PWM1L/RP19/RA3	35	PGEC2/ADTRG31/RP36/RB4
14	FLT12/RP48/RC0	36	EXTREF1/AN9/CMP4D/RP50/RC2
15	FLT11/RP61/RC13	37	ASDA1/RP55/RC7
16	AVss	38	ASCL1/RP56/RC8
17	AVDD	39	Vss
18	MCLR	40	VDD
19	AVDD	41	PGED3/SDA2/FLT31/RP40/RB8
20	AN14/PGA2N3/RP60/RC12	42	PGEC3/SCL2/RP47/RB15
21	AN0/CMP1A/PGA1P1/RP16/RA0	43	TDO/AN19/PGA2N2/RP37/RB5
22	AN1/CMP1B/PGA1P2/PGA2P1/RP17/RA1	44	PGED1/TDI/AN20/SCL1/RP38/RB6

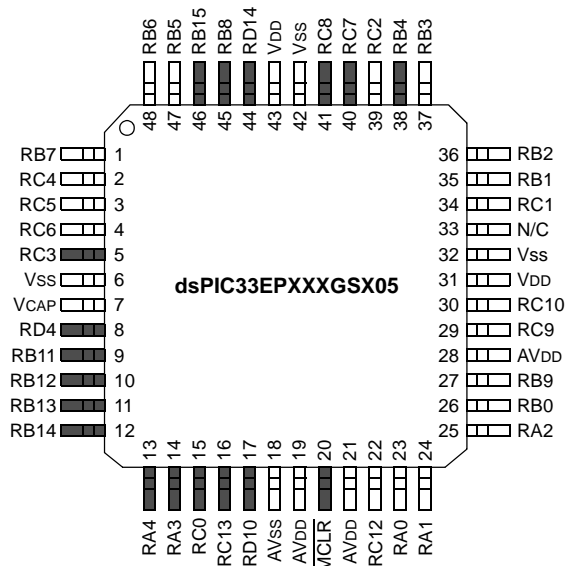
图注: 阴影引脚最高可承受 5 VDC。

RPn 表示可重映射外设功能。有关可重映射源的完整列表, 请参见表 11-12 和表 11-13。

# dsPIC33EPXXGS70X/80X 系列

## 引脚图 (续)

### 48 引脚 TQFP



引脚	引脚功能	引脚	引脚功能
1	PGEC1/AN21/SDA1/RP39/RB7	25	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/RP18/RA2
2	AN1ALT/RP52/RC4	26	AN3/CMP1D/CMP2B/PGA2P3/RP32/RB0
3	AN0ALT/RP53/RC5	27	AN4/CMP2C/CMP3A/ISRC4/RP41/RB9
4	AN17/RP54/RC6	28	AVDD
5	RP51/RC3	29	AN11/PGA1N3/RP57/RC9
6	Vss	30	EXTREF2/AN10/PGA1P4/RP58/RC10
7	VCAP	31	VDD
8	RP68/RD4	32	Vss
9	TMS/PWM3H/RP43/RB11	33	N/C
10	TCK/PWM3L/RP44/RB12	34	AN8/CMP4C/PGA2P4/RP49/RC1
11	PWM2H/RP45/RB13	35	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/RP33/RB1
12	PWM2L/RP46/RB14	36	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/RP34/RB2
13	PWM1H/RP20/RA4	37	PGED2/DACOUT1/AN18/INT0/RP35/RB3
14	PWM1L/RP19/RA3	38	PGEC2/ADTRG31/RP36/RB4
15	FLT12/RP48/RC0	39	EXTREF1/AN9/CMP4D/RP50/RC2
16	FLT11/RP61/RC13	40	ASDA1/RP55/RC7
17	CLC4OUT/FLT10/RP74/RD10	41	ASCL1/RP56/RC8
18	AVss	42	Vss
19	AVDD	43	VDD
20	MCLR	44	CLC3OUT/RD14
21	AVDD	45	PGED3/SDA2/FLT31/RP40/RB8
22	AN14/PGA2N3/RP60/RC12	46	PGEC3/SCL2/RP47/RB15
23	AN0/CMP1A/PGA1P1/RP16/RA0	47	TDO/AN19/PGA2N2/RP37/RB5
24	AN1/CMP1B/PGA1P2/PGA2P1/RP17/RA1	48	PGED1/TDI/AN20/SCL1/RP38/RB6

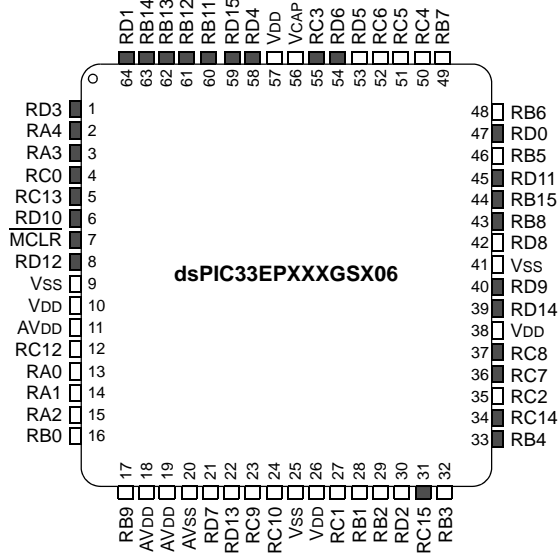
图注：阴影引脚最高可承受 5 VDC。

RPn 表示可重映射外设功能。有关可重映射源的完整列表，请参见表 11-12 和表 11-13。

# dsPIC33EPXXGS70X/80X 系列

## 引脚图 (续)

### 64 引脚 TQFP



引脚	引脚功能	引脚	引脚功能
1	PWM4L/RP67/RD3	33	PGEC2/ADTRG31/RP36/RB4
2	PWM1H/RP20/RA4	34	RP62/RC14
3	PWM1L/RP19/RA3	35	EXTREF1/AN9/CMP4D/RP50/RC2
4	FLT12/RP48/RC0	36	ASDA1/RP55/RC7
5	FLT11/RP61/RC13	37	ASCL1/RP56/RC8
6	CLC4OUT/FLT10/RP74/RD10	38	VDD
7	MCLR	39	CLC3OUT/RD14
8	T5CK/FLT9/RP76/RD12	40	SCK3/RP73/RD9
9	Vss	41	Vss
10	VDD	42	AN5/CMP2D/CMP3B/ISRC3/RP72/RD8
11	AVDD	43	PGED3/SDA2/FLT31/RP40/RB8
12	AN14/PGA2N3/RP60/RC12	44	PGEC3/SCL2/RP47/RB15
13	AN0/CMP1A/PGA1P1/RP16/RA0	45	INT4/RP75/RD11
14	AN1/CMP1B/PGA1P2/PGA2P1/RP17/RA1	46	TDO/AN19/PGA2N2/RP37/RB5
15	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/RP18/RA2	47	T4CK/RP64/RD0
16	AN3/CMP1D/CMP2B/PGA2P3/RP32/RB0	48	PGED1/TDI/AN20/SCL1/RP38/RB6
17	AN4/CMP2C/CMP3A/ISRC4/RP41/RB9	49	PGEC1/AN21/SDA1/RP39/RB7
18	AVDD	50	AN1ALT/RP52/RC4
19	AVDD	51	AN0ALT/RP53/RC5
20	AVss	52	AN17/RP54/RC6
21	AN15/RP71/RD7	53	AN12/ISRC1/RP69/RD5
22	DACOUT2/AN13/RD13	54	PWM5H/RP70/RD6
23	AN11/PGA1N3/RP57/RC9	55	PWM5L/RP51/RC3
24	EXTREF2/AN10/PGA1P4/RP58/RC10	56	VCAP
25	Vss	57	VDD
26	VDD	58	PWM6H/RP68/RD4
27	AN8/CMP4C/PGA2P4/RP49/RC1	59	PWM6L/RD15
28	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/RP33/RB1	60	TMS/PWM3H/RP43/RB11
29	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/RP34/RB2	61	TCK/PWM3L/RP44/RB12
30	AN16/RP66/RD2	62	PWM2H/RP45/RB13
31	ASDA2/RP63/RC15	63	PWM2L/RP46/RB14
32	PGED2/DACOUT1/AN18/ASCL2/INT0/RP35/RB3	64	PWM4H/RP65/RD1

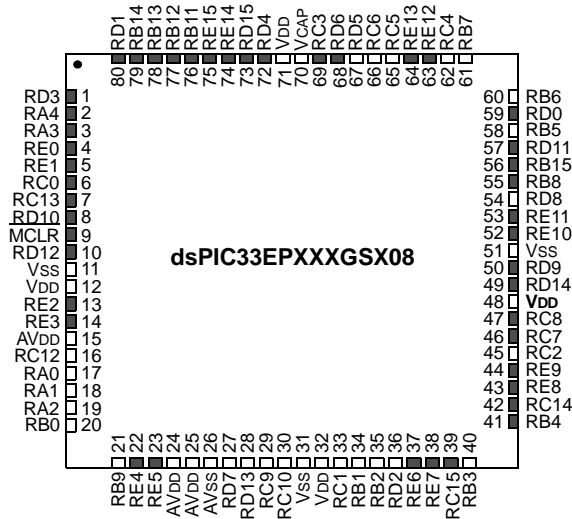
图注: 阴影引脚最高可承受 5 VDC。

RPn 表示可重映射外设功能。有关可重映射源的完整列表, 请参见表 11-12 和表 11-13。

# dsPIC33EPXXGS70X/80X 系列

## 引脚图 (续)

### 80 引脚 TQFP



引脚	引脚功能	引脚	引脚功能
1	PWM4L/RP67/RD3	41	PGEC2/ADTRG31/RP36/RB4
2	PWM1H/RP20/RA4	42	RP62/RC14
3	PWM1L/RP19/RA3	43	RE8
4	PWM8L/RE0	44	RE9
5	PWM8H/RE1	45	EXTREF1/AN9/CMP4D/RP50/RC2
6	FLT12/RP48/RC0	46	ASDA1/RP55/RC7
7	FLT11/RP61/RC13	47	ASCL1/RP56/RC8
8	CLC4OUT/FLT10/RP74/RD10	48	VDD
9	MCLR	49	CLC3OUT/RD14
10	T5CK/FLT9/RP76/RD12	50	SCK3/RP73/RD9
11	VSS	51	VSS
12	VDD	52	FLT21/RE10
13	FLT17/RE2	53	FLT22/RE11
14	FLT18/RE3	54	AN5/CMP2D/CMP3B/ISRC3/RP72/RD8
15	AVDD	55	PGED3/SDA2/FLT31/RP40/RB8
16	AN14/PGA2N3/RP60/RC12	56	PGEC3/SCL2/RP47/RB15
17	AN0/CMP1A/PGA1P1/RP16/RA0	57	INT4/RP75/RD11
18	AN1/CMP1B/PGA1P2/PGA2P1/RP17/RA1	58	TD0/AN19/PGA2N2/RP37/RB5
19	AN2/CMP1C/CMP2A/PGA1P3/PGA2P2/RP18/RA2	59	T4CK/RP64/RD0
20	AN3/CMP1D/CMP2B/PGA2P3/RP32/RB0	60	PGED1/TDI/AN20/SCL1/RP38/RB6
21	AN4/CMP2C/CMP3A/ISRC4/RP41/RB9	61	PGEC1/AN21/SDA1/RP39/RB7
22	RE4	62	AN1ALT/RP52/RC4
23	RE5	63	RE12
24	AVDD	64	RE13
25	AVDD	65	AN0ALT/RP53/RC5
26	AVSS	66	AN17/RP54/RC6
27	AN15/RP71/RD7	67	AN12/ISRC1/RP69/RD5
28	DACOUT2/AN13/RD13	68	PWM5H/RP70/RD6
29	AN11/PGA1N3/RP57/RC9	69	PWM5L/RP51/RC3
30	EXTREF2/AN10/PGA1P4/RP58/RC10	70	VCAP
31	VSS	71	VDD
32	VDD	72	PWM6H/RP68/RD4
33	AN8/CMP4C/PGA2P4/RP49/RC1	73	PWM6L/RD15
34	OSCI/CLKI/AN6/CMP3C/CMP4A/ISRC2/RP33/RB1	74	PWM7L/RE14
35	OSC2/CLKO/AN7/CMP3D/CMP4B/PGA1N2/RP34/RB2	75	PWM7H/RE15
36	AN16/RP66/RD2	76	TMS/PWM3H/RP43/RB11
37	FLT19/RE6	77	TCK/PWM3L/RP44/RB12
38	FLT20/RE7	78	PWM2H/RP45/RB13
39	ASDA2/RP63/RC15	79	PWM2L/RP46/RB14
40	PGED2/DACOUT1/AN18/ASCL2/INT0/RP35/RB3	80	PWM4H/RP65/RD1

图注: 阴影引脚最高可承受 5 VDC。

RPn 表示可重映射外设功能。有关可重映射源的完整列表, 请参见表 11-12 和表 11-13。



# dsPIC33EPXXGS70X/80X 系列

## 目录

1.0	器件概述	11
2.0	16 位数字信号控制器入门指南	15
3.0	CPU	21
4.0	存储器构成	31
5.0	闪存程序存储器	61
6.0	复位	69
7.0	中断控制器	73
8.0	直接存储器访问 (DMA)	89
9.0	振荡器配置	103
10.0	节能特性	115
11.0	I/O 端口	125
12.0	Timer1	169
13.0	Timer2/3 和 Timer4/5	173
14.0	输入捕捉	177
15.0	输出比较	181
16.0	高速 PWM	187
17.0	外设触发信号发生器 (PTG) 模块	213
18.0	串行外设接口 (SPI)	229
19.0	I <sup>2</sup> C	245
20.0	通用异步收发器 (UART)	253
21.0	可配置逻辑单元 (CLC)	259
22.0	高速 12 位模数转换器 (ADC)	273
23.0	控制器局域网 (CAN) 模块 (仅限 dsPIC33EPXXGS80X 器件)	307
24.0	高速模拟比较器	333
25.0	可编程增益放大器 (PGA)	341
26.0	恒流源	345
27.0	特殊功能	347
28.0	指令集汇总	361
29.0	开发支持	371
30.0	电气特性	375
31.0	器件直流和交流特性曲线图	435
32.0	封装信息	439
附录 A:	版本历史	465
	索引	467
	Microchip 网站	475
	变更通知客户服务	475
	客户支持	475
	产品标识体系	477

# dsPIC33EPXXGS70X/80X 系列

---

---

## 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](mailto:CTRC@microchip.com)。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：DS30000000A 是 DS30000000 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](http://www.microchip.com) 上注册。

# dsPIC33EPXXXGS70X/80X 系列

## 1.0 器件概述

**注 1:** 本数据手册总结了 dsPIC33EPXXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的相关章节，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

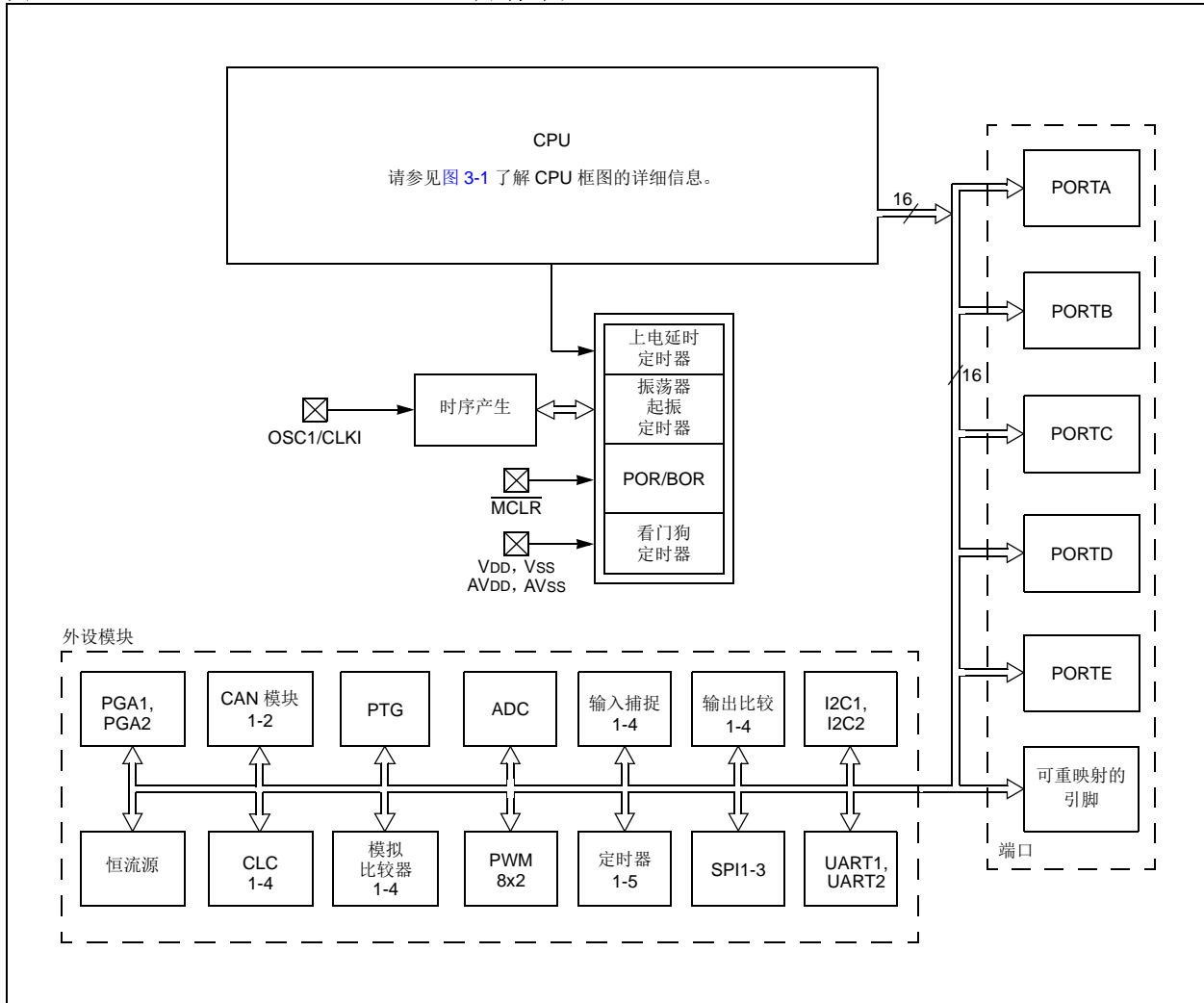
**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

本文档包含 dsPIC33EPXXXGS70X/80X 数字信号控制器 (Digital Signal Controller, DSC) 器件的具体信息。

dsPIC33EPXXXGS70X/80X 器件在高性能 16 位 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 给出了内核和外设模块的一般框图。表 1-1 列出了引脚图中显示的各引脚的功能。

图 1-1: dsPIC33EPXXXGS70X/80X 系列框图



# dsPIC33EPXXGS70X/80X 系列

表 1-1: 引脚 I/O 说明

引脚名称 <sup>(1)</sup>	引脚类型	缓冲器类型	支持 PPS 功能?	说明
AN0-AN21 AN0ALT-AN1ALT	I I	模拟 模拟	否 否	模拟输入通道。 备用模拟输入通道。
C1RXR C2RXR C1TX C2TX	I I O O	ST ST ST ST	是 是 是 是	CAN1 接收。 CAN2 接收。 CAN1 发送。 CAN2 发送。
CLKI CLKO	I O	ST/ CMOS —	否 否	外部时钟源输入。总是与 OSC1 引脚功能相关联。 晶振输出。在晶振模式下，该引脚与晶振或谐振器相连。也可选择在 RC 和 EC 模式下用作 CLKO。总是与 OSC2 引脚功能相关联。
OSC1 OSC2	I I/O	ST/ CMOS —	否 否	晶振输入。配置为 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。 晶振输出。在晶振模式下，该引脚与晶振或谐振器相连。也可选择在 RC 和 EC 模式下用作 CLKO。
CLC1OUT CLC2OUT CLC3OUT CLC4OUT	O O O O	DIG DIG DIG DIG	是 是 否 <sup>(4)</sup> 否 <sup>(4)</sup>	CLC1 输出。 CLC2 输出。 CLC3 输出。 CLC4 输出。
REFCLKO	O	—	是	参考时钟输出。
IC1-IC4	I	ST	是	捕捉输入 1 至 4。
OCFA OC1-OC4	I O	ST —	是 是	比较故障 A 输入（对于比较通道）。 比较输出 1 至 4。
INT0 INT1 INT2 INT4	I I I I	ST ST ST ST	否 是 是 是	外部中断 0。 外部中断 1。 外部中断 2。 外部中断 4。
RA0-RA4	I/O	ST	否	PORTA 是双向 I/O 端口。
RB0-RB15	I/O	ST	否	PORTB 是双向 I/O 端口。
RC0-RC15	I/O	ST	否	PORTC 是双向 I/O 端口。
RD0-RD15	I/O	ST	否	PORTD 是双向 I/O 端口。
RE0-RE15	I/O	ST	否	PORTE 是双向 I/O 端口。
T1CK T2CK T3CK T4CK T5CK	I I I I I	ST ST ST ST ST	是 是 是 否 否	Timer1 外部时钟输入。 Timer2 外部时钟输入。 Timer3 外部时钟输入。 Timer4 外部时钟输入。 Timer5 外部时钟输入。
U1CTS U1RTS U1RX U1TX BCLK1	I O I O O	ST — ST — ST	是 是 是 是 是	UART1 允许发送。 UART1 请求发送。 UART1 接收。 UART1 发送。 UART1 IrDA <sup>®</sup> 波特率时钟输出。

图注: CMOS = CMOS 兼容输入或输出      模拟 = 模拟输入      P = 电源  
 ST = CMOS 电平的施密特触发器输入      O = 输出      I = 输入  
 PPS = 外设引脚选择      TTL = TTL 输入缓冲器

- 并非所有引脚都在所有封装类型上可用。关于引脚可用性，请参见“引脚图”部分。
- 在 dsPIC33EPXXGS708/808 器件上，PWM4H/L 至 PWM8H/L 是固定的。在 dsPIC33EPXXGS706/806 器件上，PWM4H/L 至 PWM6H/L 是固定的。
- 在 dsPIC33EPXXGS706/806 和 dsPIC33EPXXGS708/808 器件上，SCK3 引脚是固定的。
- PPS 仅在 dsPIC33EPXXGS702 器件上可用。

# dsPIC33EPXXGS70X/80X 系列

表 1-1: 引脚 I/O 说明 (续)

引脚名称 (1)	引脚类型	缓冲器类型	支持 PPS 功能?	说明
U2CTS	I	ST	是	UART2 允许发送。
U2RTS	O	—	是	UART2 请求发送。
U2RX	I	ST	是	UART2 接收。
U2TX	O	—	是	UART2 发送。
BCLK2	O	ST	是	UART2 IrDA 波特率时钟输出。
SCK1	I/O	ST	是	SPI1 的同步串行时钟输入 / 输出。
SDI1	I	ST	是	SPI1 数据输入。
SDO1	O	—	是	SPI1 数据输出。
SS1	I/O	ST	是	SPI1 从同步或帧脉冲 I/O。
SCK2	I/O	ST	是	SPI2 的同步串行时钟输入 / 输出。
SDI2	I	ST	是	SPI2 数据输入。
SDO2	O	—	是	SPI2 数据输出。
SS2	I/O	ST	是	SPI2 从同步或帧脉冲 I/O。
SCK3	I/O	ST	是 (3)	SPI3 的同步串行时钟输入 / 输出。
SDI3	I	ST	是	SPI3 数据输入。
SDO3	O	—	是	SPI3 数据输出。
SS3	I/O	ST	是	SPI3 从同步或帧脉冲 I/O。
SCL1	I/O	ST	否	I2C1 的同步串行时钟输入 / 输出。
SDA1	I/O	ST	否	I2C1 的同步串行数据输入 / 输出。
ASCL1	I/O	ST	否	I2C1 的备用同步串行时钟输入 / 输出。
ASDA1	I/O	ST	否	I2C1 的备用同步串行数据输入 / 输出。
SCL2	I/O	ST	否	I2C2 的同步串行时钟输入 / 输出。
SDA2	I/O	ST	否	I2C2 的同步串行数据输入 / 输出。
ASCL2	I/O	ST	否	I2C2 的备用同步串行时钟输入 / 输出。
ASDA2	I/O	ST	否	I2C2 的备用同步串行数据输入 / 输出。
TMS	I	ST	否	JTAG 测试模式选择引脚。
TCK	I	ST	否	JTAG 测试时钟输入引脚。
TDI	I	ST	否	JTAG 测试数据输入引脚。
TDO	O	—	否	JTAG 测试数据输出引脚。
FLT1-FLT8	I	ST	是	PWM 故障输入 1 至 8。
FLT9-FLT12	I	ST	否	PWM 故障输入 9 至 12。
PWM1L-PWM3L	O	—	否	PWM 低端输出 1 至 3。
PWM1H-PWM3H	O	—	否	PWM 高端输出 1 至 3。
PWM4L-PWM8L (2)	O	—	是	PWM 低端输出 4 至 8。
PWM4H-PWM8H (2)	O	—	是	PWM 高端输出 4 至 8。
SYNCI1, SYNCI2	I	ST	是	PWM 同步输入 1 和 2。
SYNCO1, SYNCO2	O	—	是	PWM 同步输出 1 和 2。

图注: CMOS = CMOS 兼容输入或输出      模拟 = 模拟输入      P = 电源  
 ST = CMOS 电平的施密特触发器输入      O = 输出      I = 输入  
 PPS = 外设引脚选择      TTL = TTL 输入缓冲器

- 并非所有引脚都在所有封装类型上可用。关于引脚可用性，请参见“引脚图”部分。
- 在 dsPIC33EPXXXGS708/808 器件上，PWM4H/L 至 PWM8H/L 是固定的。在 dsPIC33EPXXXGS706/806 器件上，PWM4H/L 至 PWM6H/L 是固定的。
- 在 dsPIC33EPXXXGS706/806 和 dsPIC33EPXXXGS708/808 器件上，SCK3 引脚是固定的。
- PPS 仅在 dsPIC33EPXXXGS702 器件上可用。

# dsPIC33EPXXXGS70X/80X 系列

表 1-1: 引脚 I/O 说明 (续)

引脚名称 (1)	引脚类型	缓冲器类型	支持 PPS 功能?	说明
CMP1A-CMP4A CMP1B-CMP4B CMP1C-CMP4C CMP1D-CMP4D	I I I I	模拟 模拟 模拟 模拟	否 否 否 否	比较器通道 1A 至 4A 的输入。 比较器通道 1B 至 4B 的输入。 比较器通道 1C 至 4C 的输入。 比较器通道 1D 至 4D 的输入。
ACMP1-ACMP4	O	—	是	模拟比较器输出 1-4。
DACOUT1, DACOUT2	O	—	否	DAC 输出电压 1 和 2。
EXTREF1, EXTREF2	I	模拟	否	参考 DAC 的外部参考电压输入 1 和 2。
PGA1P1-PGA1P4	I	模拟	否	PGA1 同相输入 1 至 4。
PGA1N1-PGA1N3	I	模拟	否	PGA1 反相输入 1 至 3。
PGA2P1-PGA2P4	I	模拟	否	PGA2 同相输入 1 至 4。
PGA2N1-PGA2N3	I	模拟	否	PGA2 反相输入 1 至 3。
ADTRG31	I	ST	否	外部 ADC 触发电源。
PGED1 PGEC1 PGED2 PGEC2 PGED3 PGEC3	I/O I I/O I I/O I	ST ST ST ST ST ST	否 否 否 否 否 否	编程 / 调试通信通道 1 使用的数据 I/O 引脚。 编程 / 调试通信通道 1 使用的时钟输入引脚。 编程 / 调试通信通道 2 使用的数据 I/O 引脚。 编程 / 调试通信通道 2 使用的时钟输入引脚。 编程 / 调试通信通道 3 使用的数据 I/O 引脚。 编程 / 调试通信通道 3 使用的时钟输入引脚。
MCLR	I/P	ST	否	主复位输入。此引脚为低电平有效的器件复位输入端。
AVDD	P	P	否	模拟模块的正电源。此引脚必须始终连接。
AVSS	P	P	否	模拟模块的参考地。此引脚必须始终连接。
VDD	P	—	否	外设逻辑和 I/O 引脚的正电源。
VCAP	P	—	否	CPU 逻辑滤波电容连接。
VSS	P	—	否	逻辑和 I/O 引脚的参考地。

图注: CMOS = CMOS 兼容输入或输出      模拟 = 模拟输入      P = 电源  
ST = CMOS 电平的施密特触发器输入      O = 输出      I = 输入  
PPS = 外设引脚选择      TTL = TTL 输入缓冲器

- 并非所有引脚都在所有封装类型上可用。关于引脚可用性，请参见“引脚图”部分。
- 在 dsPIC33EPXXXGS708/808 器件上，PWM4H/L 至 PWM8H/L 是固定的。在 dsPIC33EPXXXGS706/806 器件上，PWM4H/L 至 PWM6H/L 是固定的。
- 在 dsPIC33EPXXXGS706/806 和 dsPIC33EPXXXGS708/808 器件上，SCK3 引脚是固定的。
- PPS 仅在 dsPIC33EPXXXGS702 器件上可用。

## 2.0 16 位数字信号控制器入门指南

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的相关章节，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

### 2.1 基本连接要求

在开始使用 dsPIC33EPXXGS70X/80X 系列进行开发之前，需要注意最基本的器件引脚连接要求。下面列出了必须始终连接的引脚名称：

- 所有 VDD 和 VSS 引脚（见第 2.2 节“去耦电容”）
- 所有 AVDD 和 AVSS 引脚（不论是否使用 ADC 模块，见第 2.2 节“去耦电容”）
- VCAP（见第 2.3 节“CPU 逻辑滤波电容连接 (VCAP)”）
- MCLR 引脚（见第 2.4 节“主复位 (MCLR) 引脚”）
- 用于进行在线串行编程 (In-Circuit Serial Programming™, ICSP™) 和调试的 PGECx/PGEDx 引脚（见第 2.5 节“ICSP 引脚”）
- OSC1 和 OSC2 引脚（使用外部振荡器源时，见第 2.6 节“外部振荡器引脚”）

### 2.2 去耦电容

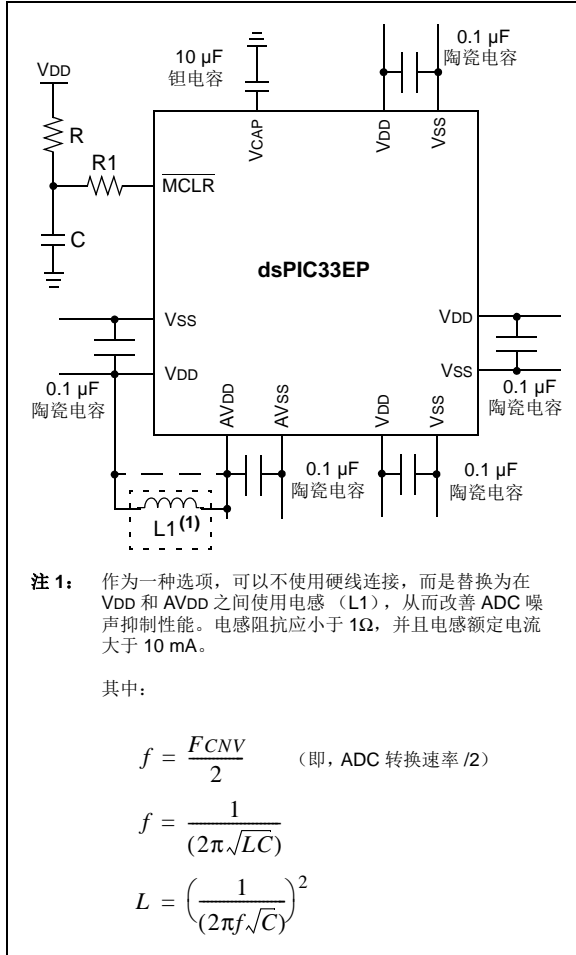
需要在每对电源引脚（例如，VDD/VSS 和 AVDD/AVSS）上使用去耦电容。

使用去耦电容时，需要考虑以下条件：

- **电容的值和类型：**建议使用参数为 0.1  $\mu\text{F}$ （100 nF）、10-20V 的电容。该电容应具有低 ESR，谐振频率为 20 MHz 或更高。建议使用陶瓷电容。
- **印刷电路板上的位置：**去耦电容应尽可能靠近引脚。建议将电容与器件放置在电路板的同一层。如果空间受限，可以使用过孔将电容放置在 PCB 的另一层，但请确保从引脚到电容的走线长度小于 0.25 英寸（6 毫米）。
- **高频噪声处理：**如果电路板会遇到高频噪声（频率高于数十 MHz），则另外添加一个陶瓷电容，与上述去耦电容并联。第二个电容的电容值可以介于 0.001  $\mu\text{F}$  和 0.01  $\mu\text{F}$  之间。请将第二个电容放置在靠近主去耦电容的位置。在高速电路设计中，需要考虑尽可能靠近电源和接地引脚放置这一对电容。例如，0.1  $\mu\text{F}$  电容与 0.001  $\mu\text{F}$  电容并联。
- **性能最大化：**对于从电源电路开始的电路板布线，需要将电源和返回走线先连接到去耦电容，然后再与器件引脚连接。这可以确保去耦电容是电源链中的第一个元件。同等重要的是尽可能减小电容和电源引脚之间的走线长度，从而降低 PCB 走线电感。

# dsPIC33EPXXXGS70X/80X 系列

图 2-1: 建议的最基本连接



## 2.2.1 大容量电容

对于电源走线长度超出 6 英寸的电路板, 建议对集成电路 (包括 DSC) 使用大容量电容来提供本地电源。大容量电容的电容值应根据连接电源与器件的走线电阻和应用中器件消耗的最大电流确定。也就是说, 选择的大容量电容需要满足器件的可接受电压骤降要求。典型值的范围为 4.7 μF 至 47 μF。

## 2.3 CPU 逻辑滤波电容连接 (VCAP)

需要在 Vcap 引脚上使用低 ESR (< 0.5Ω) 电容, 以稳定稳压器的输出电压。VCAP 引脚一定不能与 VDD 连接, 并且必须使用 4.7 μF 以上 (建议使用 10 μF)、16V 的电容接地。可以使用陶瓷电容或钽电容。更多信息, 请参见第 30.0 节 “电气特性”。

该电容的位置应靠近 VCAP 引脚。建议走线长度不要超出 0.25 英寸 (6 毫米)。有关详细信息, 请参见第 27.4 节 “片上稳压器”。

## 2.4 主复位 (MCLR) 引脚

MCLR 引脚提供两种特定的器件功能:

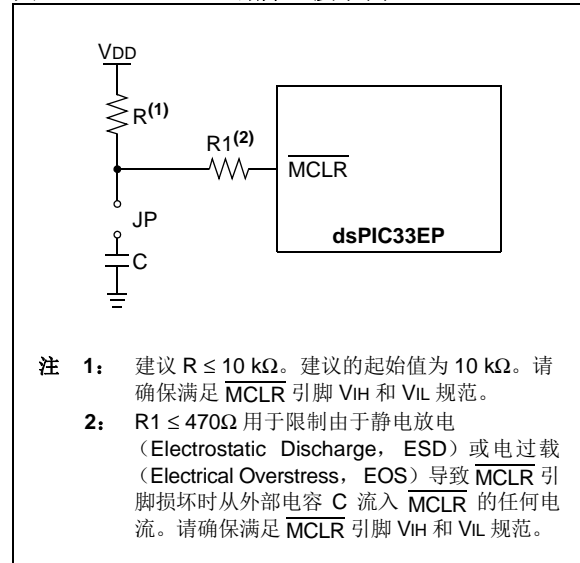
- 器件复位
- 器件编程和调试

在器件编程和调试过程中, 必须考虑到引脚上可能会增加的电阻和电容。器件编程器和调试器会驱动 MCLR 引脚。因此, 特定电压 (VIH 和 VIL) 和快速信号跳变一定不能受到不利影响。所以, 需要根据应用和 PCB 需求来调整 R 和 C 的具体值。

例如, 如图 2-2 所示, 在编程和调试操作期间, 建议将电容 C 与 MCLR 引脚隔离。

将图 2-2 所示的元件放置在距离 MCLR 引脚 0.25 英寸 (6 毫米) 范围内。

图 2-2: MCLR 引脚连接示例





## 2.5 ICSP 引脚

PGECx 和 PGEDx 引脚用于进行 ICSP 和调试。建议尽可能减小 ICSP 连接器与器件 ICSP 引脚之间的走线长度。如果 ICSP 连接器会遇到 ESD 事件，则建议添加一个串联电阻，电阻值为几十欧姆，不要超出 100Ω。

建议不要在 PGECx 和 PGEDx 引脚上连接上拉电阻、串联二极管和电容，因为它们会影响编程器 / 调试器与器件的通信。如果应用需要此类分立元件，则在编程和调试期间应将它们从电路中去掉。或者，请参见相应器件闪存编程规范中的交流 / 直流特性与时序要求信息，了解关于容性负载限制、引脚输入高电压 (V<sub>IH</sub>) 和输入低电压 (V<sub>IL</sub>) 要求的信息。

请确保编程到器件中的“通信通道选择”（即 PGECx/PGEDx 引脚）符合与 MPLAB<sup>®</sup> PICKit™ 3、MPLAB ICD 3 或 MPLAB REAL ICE™ 的 ICSP 物理连接。

关于 MPLAB ICD 2、MPLAB ICD 3 和 REAL ICE 连接要求的更多信息，请参见 Microchip 网站上提供的以下文档。

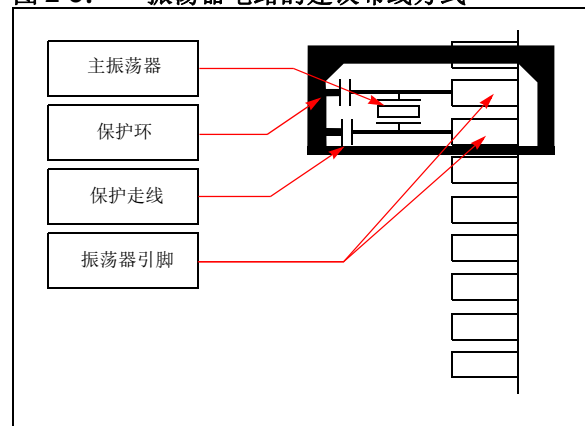
- “Using MPLAB<sup>®</sup> ICD 3 In-Circuit Debugger” (宣传页) (DS51765)
- 《多工具设计忠告》(DS51764C\_CN)
- 《适用于 MPLAB<sup>®</sup> X IDE 的 MPLAB<sup>®</sup> REAL ICE™ 在线仿真器用户指南》(DS50002085E\_CN)
- “Using MPLAB<sup>®</sup> REAL ICE™ In-Circuit Emulator” (宣传页) (DS51749)

## 2.6 外部振荡器引脚

许多 DSC 都有至少两个振荡器可供选择：高频主振荡器和低频辅助振荡器。有关详细信息，请参见第 9.0 节“振荡器配置”。

振荡器电路与器件应放置在电路板的同一侧。此外，请将振荡器电路放置在靠近相应振荡器引脚的位置，它们之间的距离不要超出 0.5 英寸（12 毫米）。负载电容应靠近振荡器本身，位于电路板的同一侧。请在振荡器电路周围使用接地覆铜区，以将其与周围电路隔离。接地覆铜区应与 MCU 地直接连接。不要在接地覆铜区内安排任何信号走线或电源走线。此外，如果使用双面电路板，请避免在电路板上晶振所在位置的背面有任何走线。图 2-3 给出了一个建议的布线图。

图 2-3: 振荡器电路的建议布线方式



# dsPIC33EPXXGS70X/80X 系列

## 2.7 器件启动时的振荡器值条件

如果目标器件的 PLL 被使能且配置为器件启动时使用的振荡器，则振荡器源的最高频率必须限制为  $3 \text{ MHz} < F_{\text{IN}} < 5.5 \text{ MHz}$ ，以符合器件的 PLL 启动条件。这意味着，如果外部振荡器频率超出该范围，应用必须首先在 FRC 模式下启动。如果 POR 之后的默认 PLL 设置的振荡器频率超出该范围，将违反器件工作速度。

器件上电之后，应用固件可以将 PLL SFR、CLKDIV 和 PLLFBD 初始化为适当的值，然后执行时钟切换，切换为振荡器 + PLL 时钟源。注意，必须在器件配置字中使能时钟切换。

## 2.8 未用 I/O

未用 I/O 引脚应配置为输出，并驱动为逻辑低电平状态。或者，在 Vss 和未用引脚之间连接一个 1k 至 10k 的电阻，并将输出驱动为逻辑低电平。

## 2.9 目标应用

- 功率因数校正 (Power Factor Correction, PFC)
  - 交错式 PFC
  - 临界导通 PFC
  - 无桥 PFC
- 直流 / 直流转换器
  - 降压、升压、正激、反激和推挽
  - 半桥 / 全桥
  - 相移全桥
  - 谐振转换器
- 直流 / 交流
  - 半桥 / 全桥逆变器
  - 谐振逆变器

图 2-4 至图 2-6 给出了典型应用连接的示例。

图 2-4: 交错式 PFC

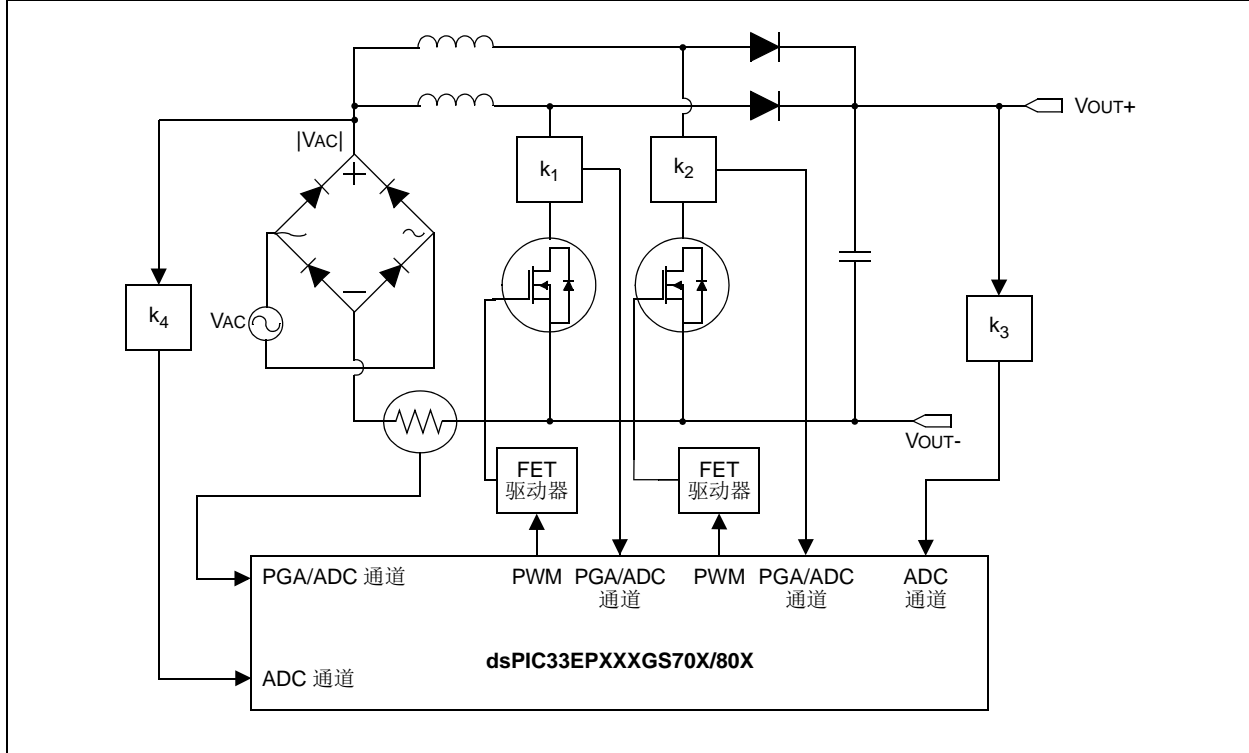
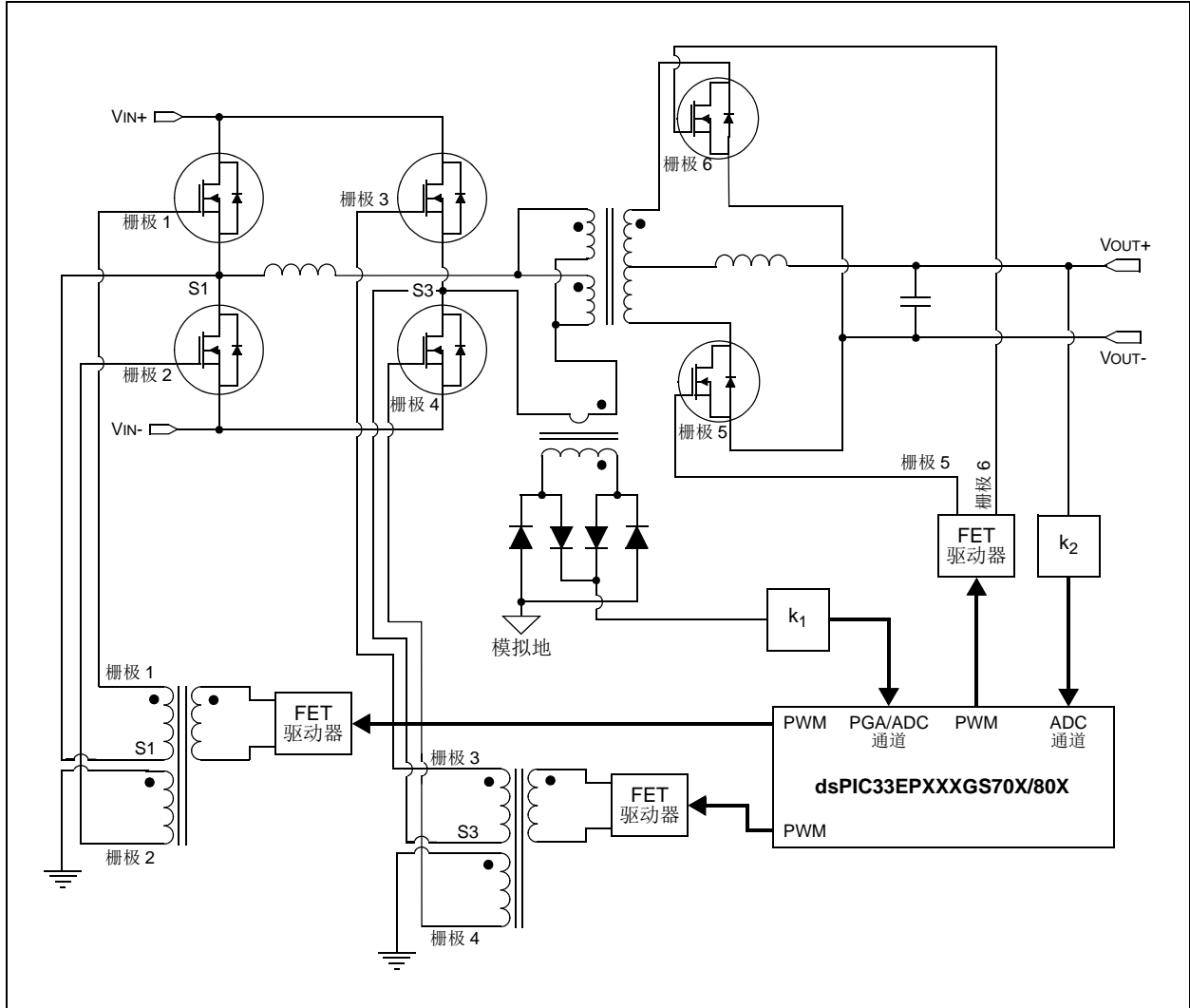
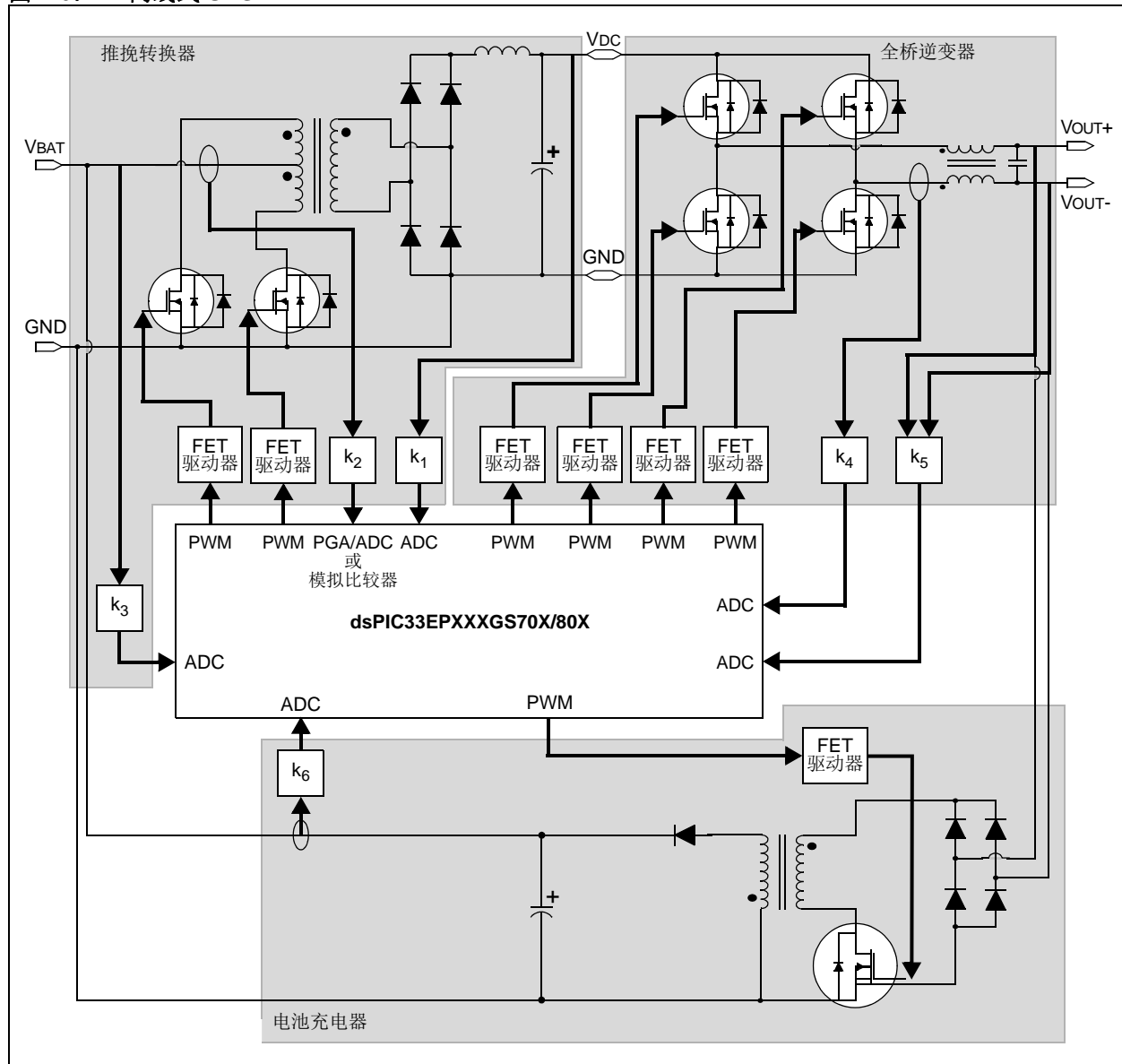


图 2-5: 相移全桥转换器



# dsPIC33EPXXXGS70X/80X 系列

图 2-6: 离线式 UPS



## 3.0 CPU

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“dsPIC33E 增强型 CPU” (DS70005158)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

dsPIC33EPXXGS70X/80X 系列 CPU 采用 16 位数据总线的改进型哈佛架构，具有增强指令集，以及对数字信号处理 (Digital Signal Processing, DSP) 的重要支持。CPU 具有 24 位指令字，指令字包含长度可变的操作码字段。程序计数器 (Program Counter, PC) 为 23 位宽，可以寻址最大 4M x 24 位的用户程序存储空间。

指令预取机制可帮助维持吞吐量，并使指令的执行具有可预测性。除了改变程序流的指令、双字传送 (MOV.D) 指令、PSV 访问和表指令以外，大多数指令都以单个周期有效执行速率执行。使用 DO 和 REPEAT 指令支持无开销的程序循环结构，这两条指令在任何时间都可以被中断。

## 3.1 寄存器

dsPIC33EPXXGS70X/80X 器件在编程模型中有 16 个 16 位工作寄存器。每个工作寄存器都可以充当数据、地址或地址偏移量寄存器。第 16 个工作寄存器 (W15) 作为软件堆栈指针工作，用于中断和调用。

另外，dsPIC33EPXXGS70X/80X 器件还包括 4 个备用工作寄存器组，由 W0 到 W14 组成。备用工作寄存器的内容是可持续保存的，有助于在中断服务程序 (Interrupt Service Routine, ISR) 执行期间减少寄存器内容的保存和恢复开销。可以通过配置 FALTREG 配置寄存器的 CTXTx<2:0> 位，为备用工作寄存器分配特定的中断优先级 (IPL1 到 IPL7)。也可以使用 CTXTSWP 指令手动访问备用工作寄存器。CTXTSTAT 寄存器中的 CCTXI<2:0> 和 MCTXI<2:0> 位可用于识别当前和最近手动选择的工作寄存器组。

## 3.2 指令集

dsPIC33EPXXGS70X/80X 器件的指令集包含两类指令：MCU 类指令和 DSP 类指令。这两类指令无缝地集成到架构中，并从单个执行单元执行。指令集包含多种寻址模式，指令的设计可使 C 编译器的效率达到最优。

## 3.3 数据空间寻址

基本数据空间可以作为最大 4K 字或 8 KB 寻址，并被分成两块，称为 X 和 Y 数据存储区。每个存储块有各自独立的地址发生单元 (Address Generation Unit, AGU)。MCU 类指令只通过 X 存储区 AGU 进行操作，可将整个存储器映射作为一个线性数据空间访问。某些 DSP 指令通过 X 和 Y 的 AGU 进行操作以支持双操作数读操作，这样会将数据地址空间分成两个部分。X 和 Y 数据空间的边界视具体器件而定。

可以选择将数据存储空间的高 32 KB 映射到任何 16K 程序字范围的程序空间 (Program Space, PS)。程序空间到数据空间的映射功能 (称为程序空间可视性 (Program Space Visibility, PSV)) 让任何指令都能像访问数据空间一样访问程序空间。有关 PSV 和表访问的更多详细信息，请参见《dsPIC33/PIC24 系列参考手册》的“数据存储区” (DS70595)。

在 dsPIC33EPXXGS70X/80X 器件上，X 和 Y 地址空间都支持无开销循环缓冲区 (模寻址)。模寻址省去了 DSP 算法的软件边界检查开销。X AGU 的循环寻址可以用于任何 MCU 类指令。X AGU 还支持位反转寻址，这大幅简化了基 2 FFT 算法对输入或输出数据的重新排序。

## 3.4 寻址模式

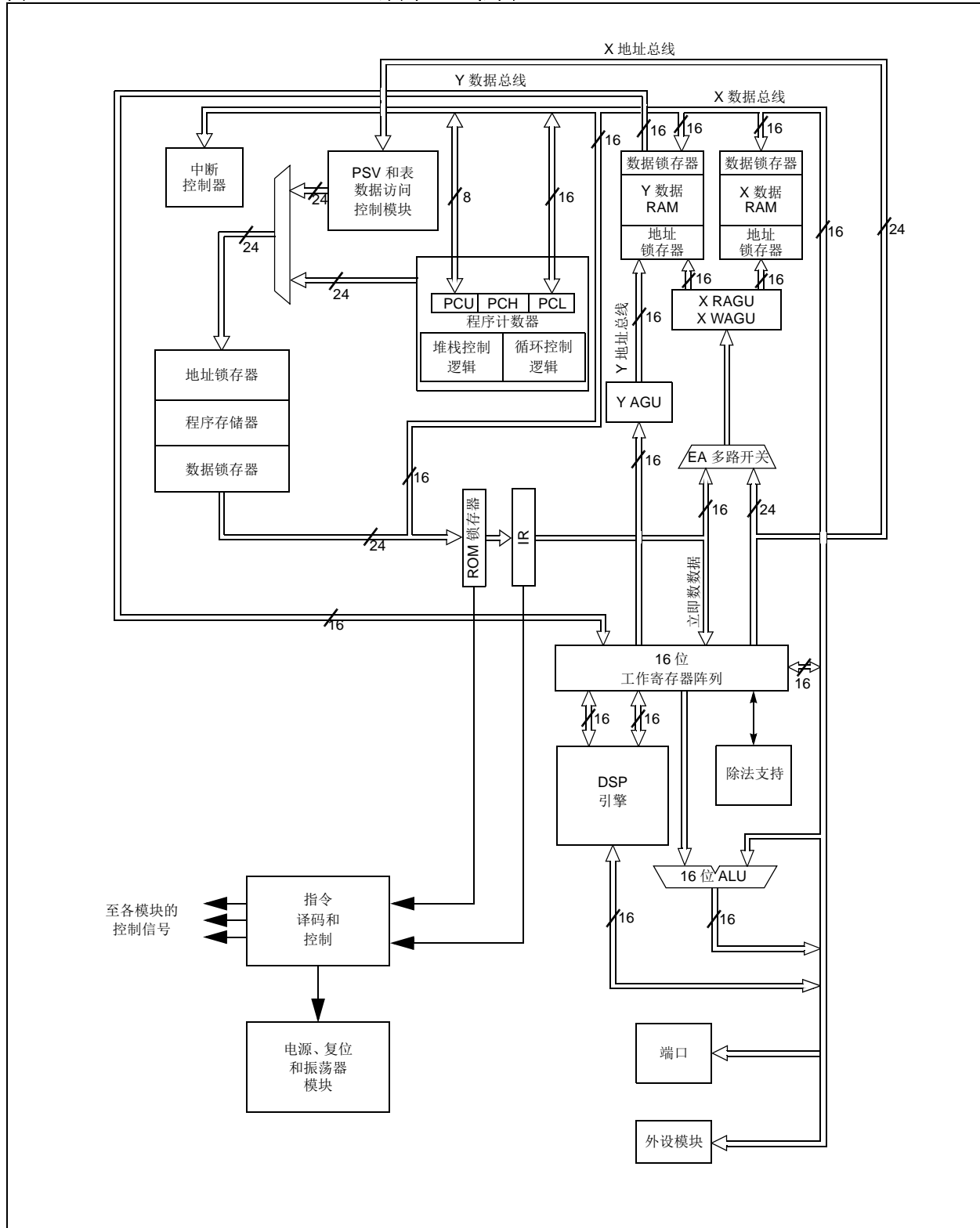
CPU 支持以下寻址模式：

- 固有寻址 (无操作数)
- 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

根据每条指令的功能要求，每条指令与预定义的寻址模式组相关联。每条指令最多支持六种寻址模式。

# dsPIC33EPXXGS70X/80X 系列

图 3-1: dsPIC33EPXXGS70X/80X 系列 CPU 框图



# dsPIC33EPXXGS70X/80X 系列

## 3.5 编程模型

图 3-2 给出了 dsPIC33EPXXGS70X/80X 系列的编程模型。编程模型中的所有寄存器都是存储器映射的，并且可以通过指令直接操作。表 3-1 列出了对每个寄存器的说明。

除了编程模型中包含的寄存器，dsPIC33EPXXGS70X/80X 系列器件还包含用于模寻址、位反转寻址和中断的控制寄存器。这些寄存器将在本文档后续章节中进行说明。

与编程模型相关的所有寄存器都是存储器映射的，如表 3-1 中所示。

表 3-1: 编程模型的寄存器说明

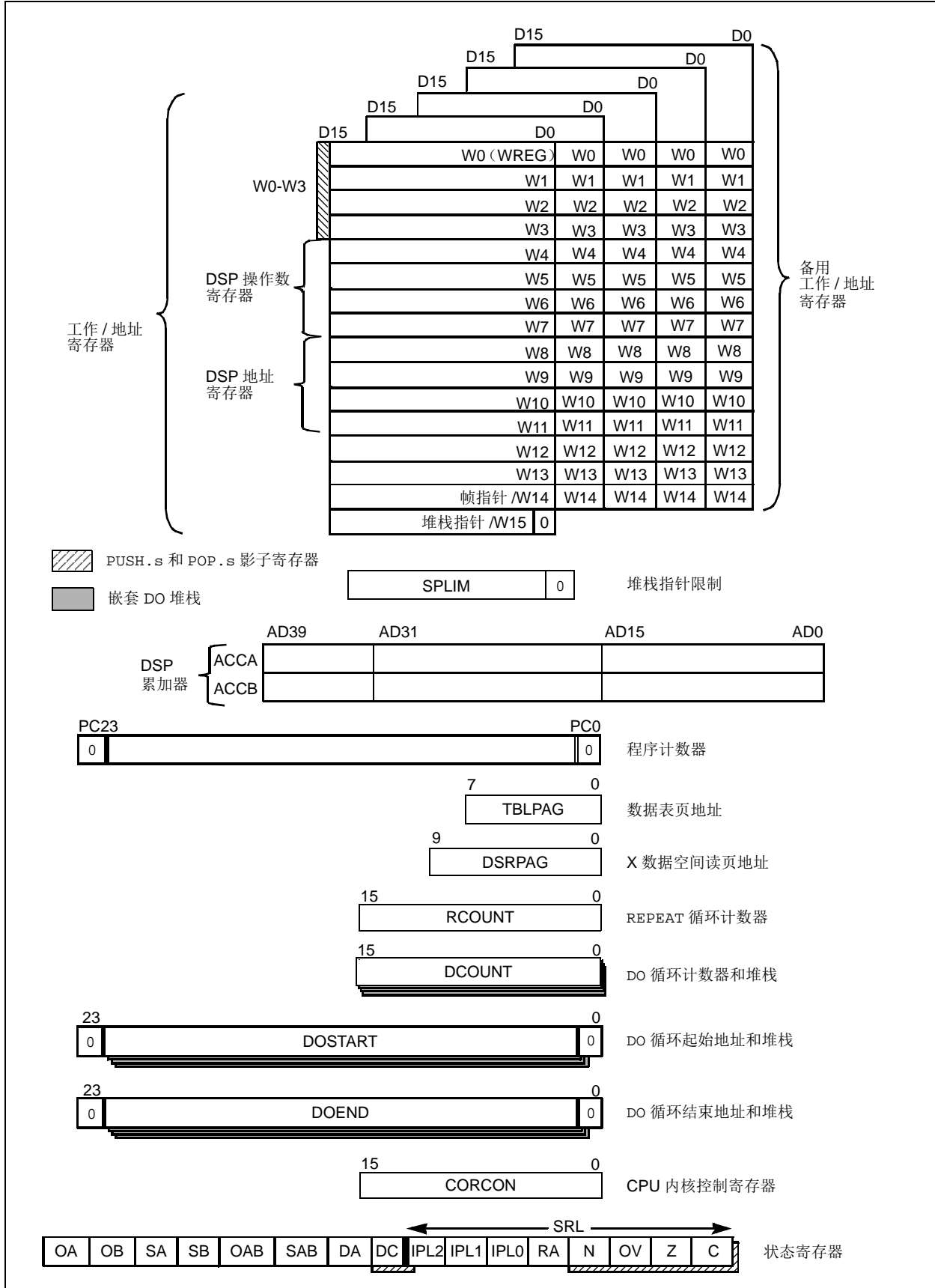
寄存器名称	说明
W0 至 W15 <sup>(1)</sup>	工作寄存器阵列
W0 至 W14 <sup>(1)</sup>	备用工作寄存器阵列 1
W0 至 W14 <sup>(1)</sup>	备用工作寄存器阵列 2
W0 至 W14 <sup>(1)</sup>	备用工作寄存器阵列 3
W0 至 W14 <sup>(1)</sup>	备用工作寄存器阵列 4
ACCA 和 ACCB	40 位 DSP 累加器
PC	23 位程序计数器
SR	ALU 和 DSP 引擎状态寄存器
SPLIM	堆栈指针限制值寄存器
TBLPAG	表存储器页地址寄存器
DSRPAG	扩展数据空间 (Extended Data Space, EDS) 读页寄存器
RCOUNT	REPEAT 循环计数器寄存器
DCOUNT	DO 循环计数器寄存器
DOSTARTH <sup>(2)</sup> 和 DOSTARTL <sup>(2)</sup>	DO 循环起始地址寄存器 (高位字和低位字)
DOENDH 和 DOENDL	DO 循环结束地址寄存器 (高位字和低位字)
CORCON	包含 DSP 引擎、DO 循环控制和陷阱状态位

注 1: 存储器映射的 W0 至 W14 表示当前激活的 CPU 现场中寄存器的值。

注 2: DOSTARTH 和 DOSTARTL 寄存器是只读的。

# dsPIC33EPXXGS70X/80X 系列

图 3-2: 编程模型





## 3.6 CPU 资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 3.6.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“**dsPIC33E 增强型 CPU**” (DS70005158)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

## 3.7 CPU 控制寄存器

### 寄存器 3-1: SR: CPU 状态寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/C-0	R/C-0	R-0	R/W-0
OA	OB	SA <sup>(3)</sup>	SB <sup>(3)</sup>	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0 <sup>(2)</sup>	R/W-0 <sup>(2)</sup>	R/W-0 <sup>(2)</sup>	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 <sup>(1)</sup>	IPL1 <sup>(1)</sup>	IPL0 <sup>(1)</sup>	RA	N	OV	Z	C
bit 7							bit 0

<b>图注:</b>	C = 可清零位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15      **OA:** 累加器 A 溢出状态位  
1 = 累加器 A 已溢出  
0 = 累加器 A 未溢出
- bit 14      **OB:** 累加器 B 溢出状态位  
1 = 累加器 B 已溢出  
0 = 累加器 B 未溢出
- bit 13      **SA:** 累加器 A 饱和 “粘住” 状态位 <sup>(3)</sup>  
1 = 累加器 A 饱和或在过去某时已经饱和  
0 = 累加器 A 未饱和
- bit 12      **SB:** 累加器 B 饱和 “粘住” 状态位 <sup>(3)</sup>  
1 = 累加器 B 饱和或在过去某时已经饱和  
0 = 累加器 B 未饱和
- bit 11      **OAB:** OA 和 OB 组合的累加器溢出状态位  
1 = 累加器 A 或 B 已溢出  
0 = 累加器 A 和 B 都未溢出
- bit 10      **SAB:** SA 和 SB 组合的累加器 “粘住” 状态位  
1 = 累加器 A 或 B 饱和或在过去某时已经饱和  
0 = 累加器 A 和 B 都未饱和
- bit 9        **DA:** DO 循环活动位  
1 = 正在进行 DO 循环  
0 = 不在进行 DO 循环
- bit 8        **DC:** MCU ALU 半进位 / 借位标志位  
1 = 结果的第 4 个低位 (对于字节大小的数据) 或第 8 个低位 (对于字大小的数据) 发生了进位  
0 = 结果的第 4 个低位 (对于字节大小的数据) 或第 8 个低位 (对于字大小的数据) 未发生进位

**注 1:** IPL<2:0> 位与 IPL<3> 位 (CORCON<3>) 组合形成 CPU 中断优先级。如果 IPL<3> = 1, 那么括号中的值表示 IPL。当 IPL<3> = 1 时, 禁止用户中断。

**注 2:** 当 NSTDIS 位 (INTCON1<15>) = 1 时, IPL<2:0> 状态位是只读的。

**注 3:** 对 SR 寄存器的数据写操作可以修改 SA 和 SB 位, 方法是向 SA 和 SB 写入数据或清零 SAB 位。要避免可能出现的 SA 或 SB 位写竞争条件, 不要使用位操作来修改 SA 和 SB 位。

## 寄存器 3-1: SR: CPU 状态寄存器 (续)

bit 7-5	<b>IPL&lt;2:0&gt;</b> : CPU 中断优先级状态位 <sup>(1,2)</sup> 111 = CPU 中断优先级为 7 (15); 禁止用户中断 110 = CPU 中断优先级为 6 (14) 101 = CPU 中断优先级为 5 (13) 100 = CPU 中断优先级为 4 (12) 011 = CPU 中断优先级为 3 (11) 010 = CPU 中断优先级为 2 (10) 001 = CPU 中断优先级为 1 (9) 000 = CPU 中断优先级为 0 (8)
bit 4	<b>RA</b> : REPEAT 循环活动位 1 = 正在进行 REPEAT 循环 0 = 不在进行 REPEAT 循环
bit 3	<b>N</b> : MCU ALU 负标志位 1 = 结果为负 0 = 结果为非负 (零或正值)
bit 2	<b>OV</b> : MCU ALU 溢出标志位 该位用于有符号的算术运算 (以二进制补码方式进行)。它表示量值上的溢出, 这种溢出将导致符号位改变状态。 1 = 有符号算术运算中发生溢出 (本次算术运算) 0 = 未发生溢出
bit 1	<b>Z</b> : MCU ALU 全零标志位 1 = 影响 Z 位的任何运算在过去某时已将该位置 1 0 = 影响 Z 位的最近一次运算已将该位清零 (即运算结果非零)
bit 0	<b>C</b> : MCU ALU 进位 / 借位标志位 1 = 结果的最高有效位发生了进位 0 = 结果的最高有效位未发生进位

**注 1:** IPL<2:0> 位与 IPL<3> 位 (CORCON<3>) 组合形成 CPU 中断优先级。如果 IPL<3> = 1, 那么括号中的值表示 IPL。当 IPL<3> = 1 时, 禁止用户中断。

**2:** 当 NSTDIS 位 (INTCON1<15>) = 1 时, IPL<2:0> 状态位是只读的。

**3:** 对 SR 寄存器的数据写操作可以修改 SA 和 SB 位, 方法是向 SA 和 SB 写入数据或清零 SAB 位。要避免可能出现的 SA 或 SB 位写竞争条件, 不要使用位操作来修改 SA 和 SB 位。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 3-2: CORCON: 内核控制寄存器

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-0
VAR	—	US1	US0	EDT <sup>(1)</sup>	DL2	DL1	DL0
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	IPL3 <sup>(2)</sup>	SFA	RND	IF
bit 7							bit 0

<b>图注:</b>	C = 可清零位
R = 可读位	W = 可写位
-n = POR 时的值	1 = 置 1
	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

- bit 15      **VAR:** 可变异常处理延时控制位  
 1 = 使能可变异常处理延时  
 0 = 使能固定异常处理延时
- bit 14      **未实现:** 读为 0
- bit 13-12   **US<1:0>:** DSP 乘法无符号 / 有符号控制位  
 11 = 保留  
 10 = DSP 引擎执行混合符号乘法运算  
 01 = DSP 引擎执行无符号乘法运算  
 00 = DSP 引擎执行有符号乘法运算
- bit 11      **EDT:** DO 循环提前终止控制位 <sup>(1)</sup>  
 1 = 在当前循环迭代结束时终止执行 DO 循环  
 0 = 无影响
- bit 10-8    **DL<2:0>:** DO 循环嵌套层级状态位  
 111 = 正在进行 7 层 DO 循环嵌套  
 •  
 •  
 •  
 001 = 正在进行 1 层 DO 循环嵌套  
 000 = 正在进行 0 层 DO 循环嵌套
- bit 7        **SATA:** ACCA 饱和使能位  
 1 = 使能累加器 A 饱和  
 0 = 禁止累加器 A 饱和
- bit 6        **SATB:** ACCB 饱和使能位  
 1 = 使能累加器 B 饱和  
 0 = 禁止累加器 B 饱和
- bit 5        **SATDW:** DSP 引擎的数据空间写饱和使能位  
 1 = 使能数据空间写饱和  
 0 = 禁止数据空间写饱和
- bit 4        **ACCSAT:** 累加器饱和模式选择位  
 1 = 9.31 饱和 (超饱和)  
 0 = 1.31 饱和 (正常饱和)
- bit 3        **IPL3:** CPU 中断优先级状态位 <sup>(2)</sup>  
 1 = CPU 中断优先级大于 7  
 0 = CPU 中断优先级小于或等于 7

注 1: 该位将总是读为 0。

注 2: IPL3 位与 IPL<2:0> 位 (SR<7:5>) 组合形成 CPU 中断优先级。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 3-2: CORCON: 内核控制寄存器 (续)

- bit 2      **SFA:** 堆栈帧有效状态位  
 1 = 堆栈帧有效; 无论 DSRPAG 如何, W14 和 W15 都寻址 0x0000 至 0xFFFF  
 0 = 堆栈帧无效; W14 和 W15 寻址基本数据空间
- bit 1      **RND:** 舍入模式选择位  
 1 = 使能有偏 (常规) 舍入  
 0 = 使能无偏 (收敛) 舍入
- bit 0      **IF:** 整数或小数乘法器模式选择位  
 1 = 使能整数模式的 DSP 乘法运算  
 0 = 使能小数模式的 DSP 乘法运算

注 1: 该位将总是读为 0。

2: IPL3 位与 IPL<2:0> 位 (SR<7:5>) 组合形成 CPU 中断优先级。

## 寄存器 3-3: CTXTSTAT: CPU W 寄存器现场状态寄存器

U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
—	—	—	—	—	CCTXI2	CCTXI1	CCTXI0
bit 15					bit 8		

U-0	U-0	U-0	U-0	U-0	R-0	R-0	R-0
—	—	—	—	—	MCTXI2	MCTXI1	MCTXI0
bit 7					bit 0		

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-11      **未实现:** 读为 0

bit 10-8      **CCTXI<2:0>:** 当前 (W 寄存器) 现场标识符位

- 111 = 保留
- 
- 
- 
- 101 = 保留
- 100 = 当前正在使用备用工作寄存器组 4
- 011 = 当前正在使用备用工作寄存器组 3
- 010 = 当前正在使用备用工作寄存器组 2
- 001 = 当前正在使用备用工作寄存器组 1
- 000 = 当前正在使用默认寄存器组

bit 7-3      **未实现:** 读为 0

bit 2-0      **MCTXI<2:0>:** 手动 (W 寄存器) 现场标识符位

- 111 = 保留
- 
- 
- 
- 101 = 保留
- 100 = 最近手动选择了备用工作寄存器组 4
- 011 = 最近手动选择了备用工作寄存器组 3
- 010 = 最近手动选择了备用工作寄存器组 2
- 001 = 最近手动选择了备用工作寄存器组 1
- 000 = 最近手动选择了默认寄存器组

# dsPIC33EPXXGS70X/80X 系列

## 3.8 算术逻辑单元 (ALU)

dsPIC33EPXXGS70X/80X 系列的 ALU 为 16 位宽，能进行加法、减法、移位和逻辑运算。除非另外说明，算术运算一般采用二进制补码方式进行。根据不同的运算，ALU 可能会影响 SR 寄存器中的进位标志位 (C)、全零标志位 (Z)、负标志位 (N)、溢出标志位 (OV) 和半进位标志位 (DC) 的值。在减法运算中，C 和 DC 状态位分别作为借位位和半借位位。

根据所使用的指令模式，ALU 可执行 8 位或 16 位运算。根据指令的寻址模式，ALU 运算的数据可以来自 W 寄存器阵列或数据存储单元。同样，ALU 的输出数据可被写入 W 寄存器阵列或数据存储单元。

关于每条指令所影响的 SR 位的信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E\_CN)。

内核 CPU 融入了对乘法和除法的硬件支持。这包括专门的硬件乘法器以及支持 16 位除数除法的硬件。

### 3.8.1 乘法器

通过使用高速 17 位 x 17 位乘法器，ALU 支持无符号、有符号或混合符号乘法运算的几种 MCU 乘法模式：

- 16 位有符号 x 16 位有符号
- 16 位无符号 x 16 位无符号
- 16 位有符号 x 5 位 (立即数) 无符号
- 16 位有符号 x 16 位无符号
- 16 位无符号 x 5 位 (立即数) 无符号
- 16 位无符号 x 16 位有符号
- 8 位无符号 x 8 位无符号

### 3.8.2 除法器

除法模块支持具有以下数据长度的 32 位 /16 位和 16 位 /16 位有符号和无符号整数除法运算：

- 32 位有符号 /16 位有符号除法
- 32 位无符号 /16 位无符号除法
- 16 位有符号 /16 位有符号除法
- 16 位无符号 /16 位无符号除法

所有除法指令的商都存放在 W0 中，余数放在 W1 中。16 位有符号和无符号 DIV 指令可为 16 位除数指定任一 W 寄存器 (Wn)，为 32 位被除数指定任意两个连续的 W 寄存器 (W(m+1):Wm)。除法运算中处理除数的每一位需要一个周期，因此 32 位 /16 位和 16 位 /16 位指令的执行周期数相同。

## 3.9 DSP 引擎

DSP 引擎由一个高速 17 位 x 17 位乘法器、一个 40 位桶形移位寄存器和一个 40 位加法器 / 减法器 (带两个目标累加器、舍入逻辑和饱和逻辑) 组成。

DSP 引擎还能执行固有、不需要其他数据的累加器 - 累加器操作。这些指令是 ADD、SUB 和 NEG。

通过 CPU 内核控制寄存器 (CORCON) 中的各个位，可以对 DSP 引擎的操作进行多种选择，这些选择如下：

- 小数或整数 DSP 乘法 (IF)
- 有符号、无符号或混合符号 DSP 乘法 (USx)
- 常规或收敛舍入 (RND)
- ACCA 自动饱和和使能 / 禁止 (SATA)
- ACCB 自动饱和和使能 / 禁止 (SATB)
- 对于写数据存储器的自动饱和和使能 / 禁止 (SATDW)
- 累加器饱和模式选择 (ACCSAT)

表 3-2: DSP 指令汇总

指令	代数运算	ACC 回写
CLR	$A = 0$	是
ED	$A = (x - y)^2$	否
EDAC	$A = A + (x - y)^2$	否
MAC	$A = A + (x \cdot y)$	是
MAC	$A = A + x^2$	否
MOVSAC	A 中内容将不发生改变	是
MPY	$A = x \cdot y$	否
MPY	$A = x^2$	否
MPY.N	$A = -x \cdot y$	否
MSC	$A = A - x \cdot y$	是

## 4.0 存储器构成

**注：**本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本数据手册当作无所不包的参考资料来使用。如需了解本手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“**dsPIC33E/PIC24E 程序存储器**”（DS70000613），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。

dsPIC33EPXXGS70X/80X 系列的架构具有独立的程序和数据存储空间以及总线。这一架构同时还允许在代码执行过程中从数据空间（Data Space, DS）直接访问程序存储器。

### 4.1 程序地址空间

dsPIC33EPXXGS70X/80X 系列器件的程序存储空间可存储 4M 个指令字。可通过由程序执行过程中 23 位 PC 或者第 4.9 节“**程序存储空间与数据存储空间的接口**”中所述的表操作或数据空间重映射得到的 24 位值寻址这一空间。

用户应用只能访问程序存储空间的低半地址范围（0x000000 至 0x7FFFFFFF）。使用 TBLRD 指令时，情况有所不同，该指令使用 TBLPAG<7> 以允许访问配置存储空间中的校准数据和器件 ID 部分。

图 4-1 和图 4-2 给出了不在双分区模式下工作的 dsPIC33EPXXGS70X/80X 器件的程序存储器映射。

dsPIC33EPXXGS70X/80X 器件可以在双分区闪存程序存储器模式下工作，其中用户闪存程序存储区分为两个独立的地址空间，每个地址空间对应一个闪存分区。活动分区始终以地址 0x000000 开始，包含可用闪存的一半（64k/128k，取决于器件）。非活动分区始终以地址 0x400000 开始，包含剩余的一半闪存。如图 4-3 和图 4-4 所示，活动分区和非活动分区相同，均包含复位向量、中断向量表（LVT 和 AIVT（如果使能））和闪存配置字的唯一副本。

# dsPIC33EPXXXGS70X/80X 系列

## 4.2 器件唯一标识符 (UDID)

在最终制造过程中，会为所有 dsPIC33EPXXXGS70X/80X 系列器件单独地编码一个器件唯一标识符 (Unique Device Identifier, UDID)。凭借该功能，可以在需要时跟踪应用中 Microchip 器件的制造信息。应用制造商也可将该功能用于任意个可能需要唯一标识的应用场合，例如：

- 跟踪器件
- 唯一序列号
- 唯一安全密钥

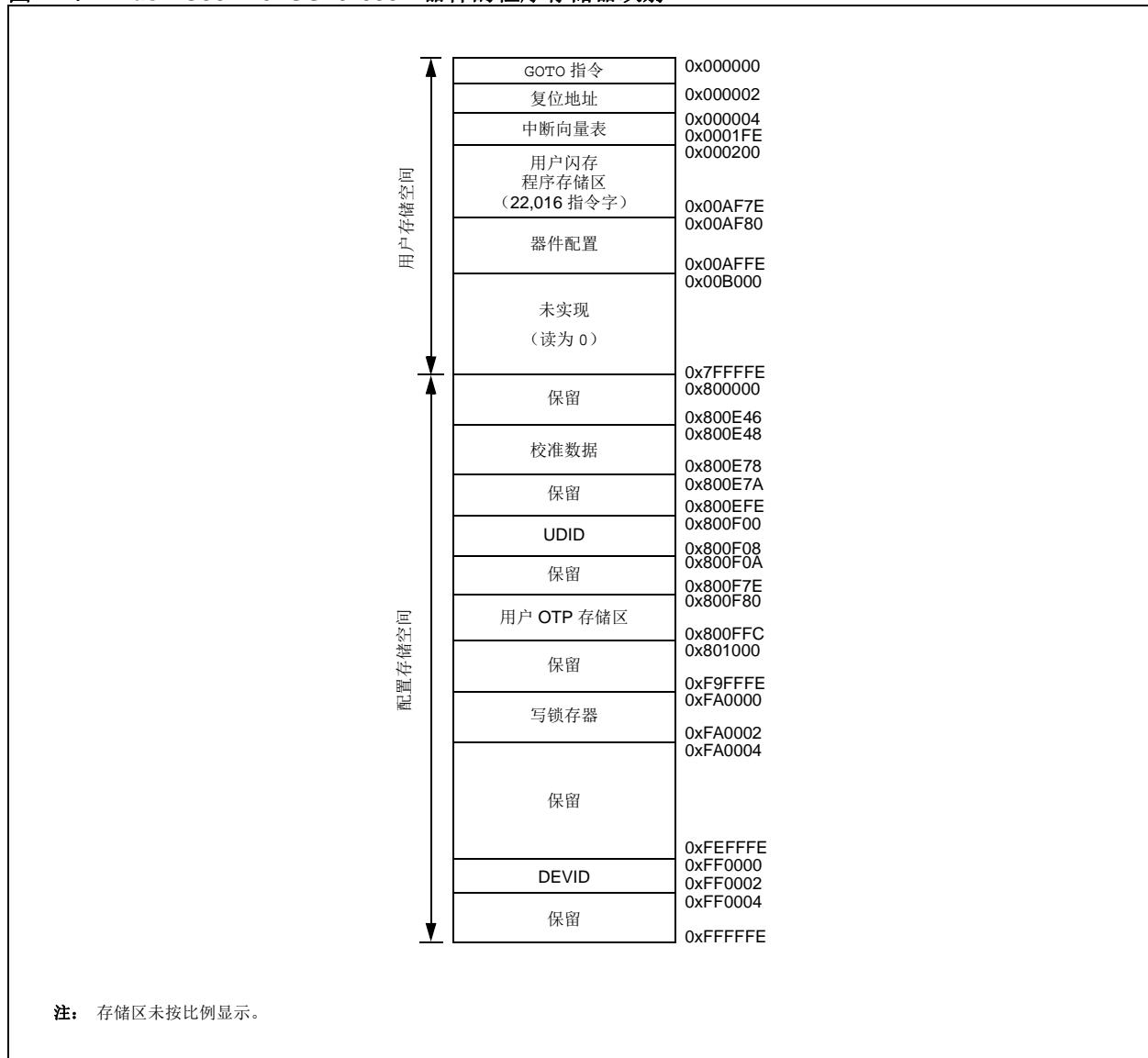
UDID 包含五个 24 位程序字。这些字段组合在一起构成一个 120 位的唯一标识符。

UDID 存储在器件配置空间中 800F00h 和 800F08h 之间的五个只读单元中。表 4-1 列出了标识符字的地址并显示了其中的内容。

表 4-1: UDID 地址

名称	地址	Bit 23:16	Bit 15:8	Bit 7:0
UDID1	800F00	UDID 字 1		
UDID2	800F02	UDID 字 2		
UDID3	800F04	UDID 字 3		
UDID4	800F06	UDID 字 4		
UDID5	800F08	UDID 字 5		

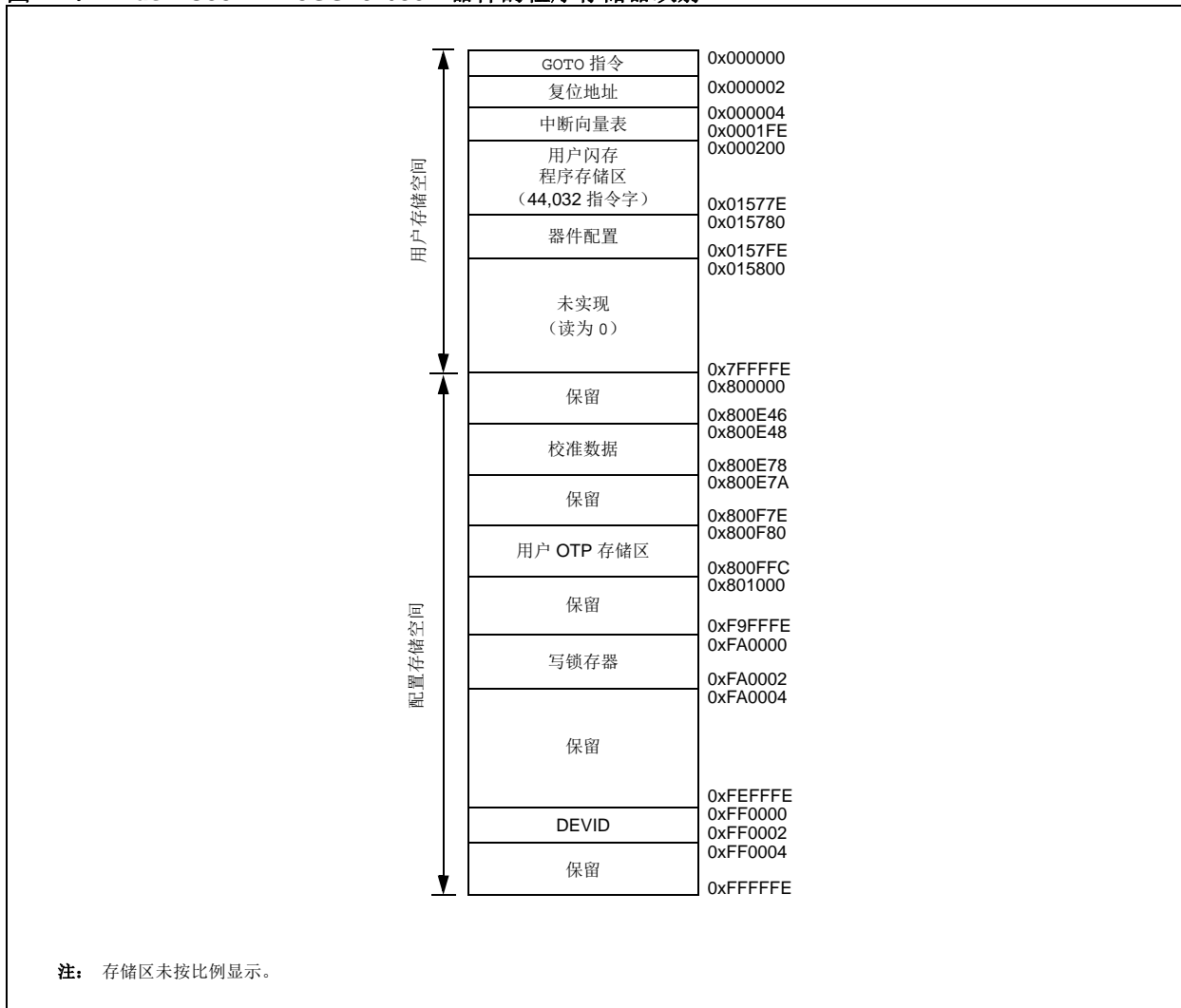
图 4-1: dsPIC33EP64GS70X/80X 器件的程序存储器映射





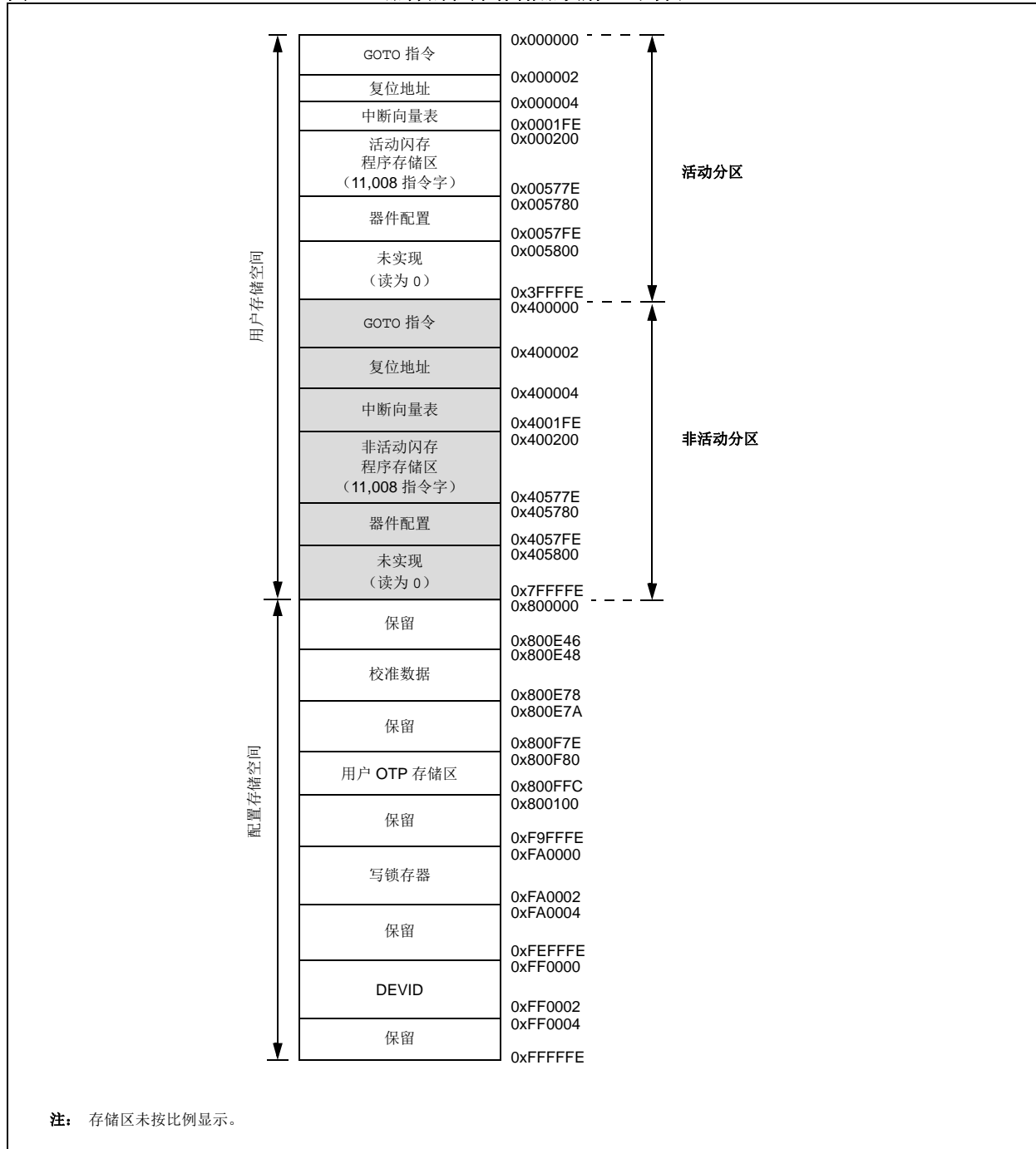
# dsPIC33EPXXGS70X/80X 系列

图 4-2: dsPIC33EP128GS70X/80X 器件的程序存储器映射



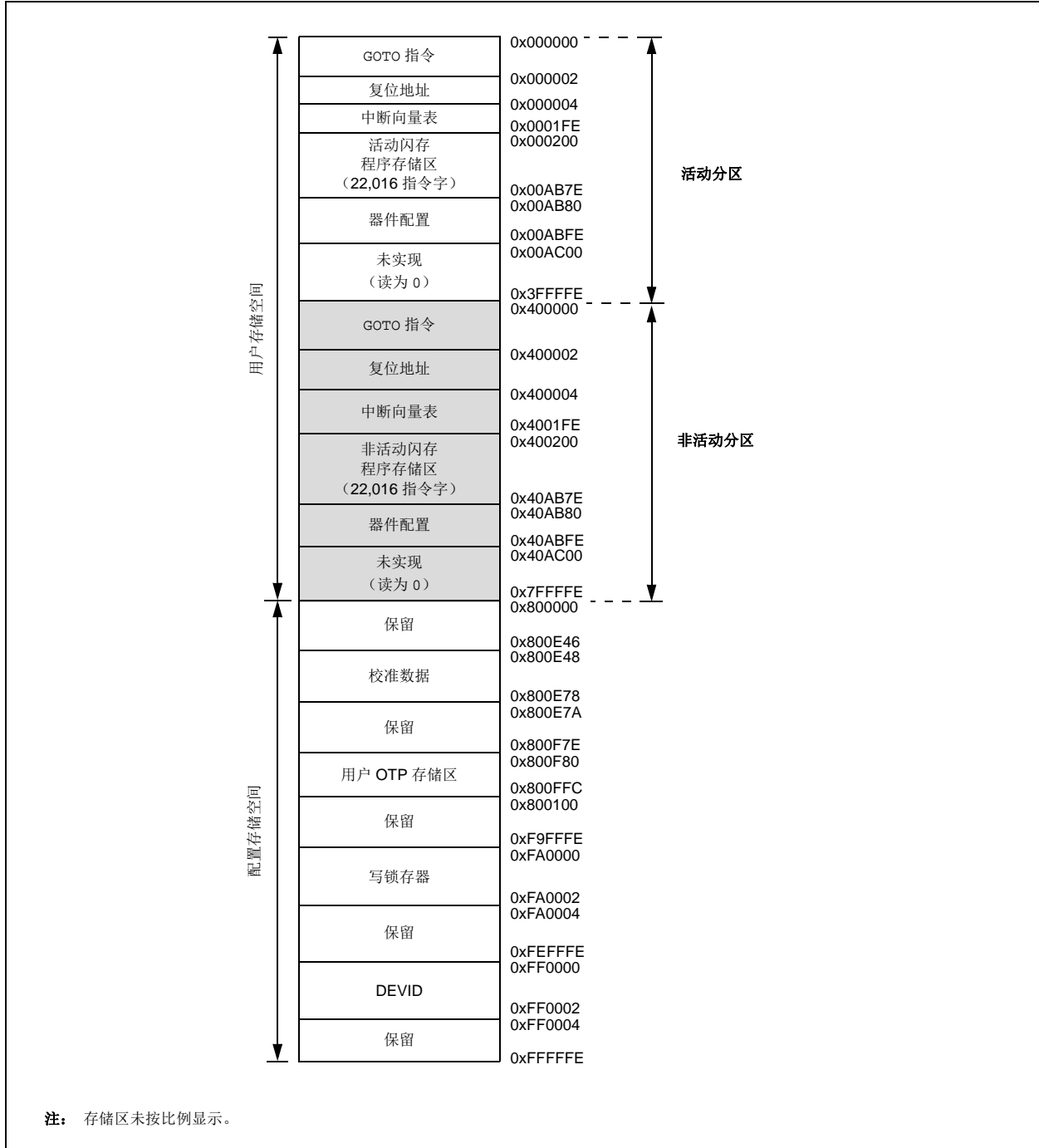
# dsPIC33EPXXGS70X/80X 系列

图 4-3: dsPIC33EP64GS70X/80X 器件的程序存储器映射 (双分区)



# dsPIC33EPXXGS70X/80X 系列

图 4-4: dsPIC33EP128GS70X/80X 器件的程序存储器映射 (双分区)



# dsPIC33EPXXGS70X/80X 系列

## 4.2.1 程序存储器构成

程序存储空间由可寻址的块构成。虽然它被视为24位宽，但将程序存储器的每个地址单元视作一个低位字和一个高位字的组合更加合理，其中高位字的高字节部分未实现。低位字的地址始终为偶数，而高位字的地址为奇数（图 4-5）。

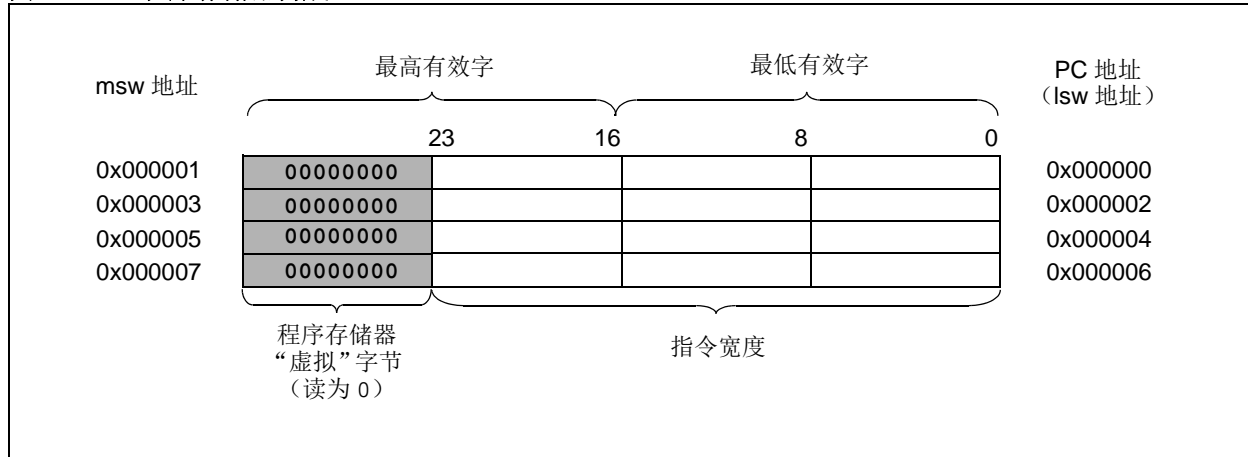
程序存储器地址始终在低位字处按字对齐，并且在代码执行过程中地址将递增或递减 2。这种寻址模式也与数据存储空间寻址兼容，且为访问程序存储空间中的数据提供了可能。

## 4.2.2 中断和陷阱向量

所有 dsPIC33EPXXGS70X/80X 系列器件中从 0x000000 至 0x000200 之间的地址空间都是保留的，用来存储硬编码的程序执行向量。提供了一个硬件复位向量将代码执行从器件复位时 PC 的默认值重新定位到代码实际起始处。用户应用程序可在闪存地址 0x000000 处编写一条 GOTO 指令，以将代码的实际起始地址设置为闪存地址 0x000002。

关于中断向量表（Interrupt Vector Table, IVT）更详细的讨论，请参见第 7.1 节“中断向量表”。

图 4-5: 程序存储器构成



## 4.3 数据地址空间

dsPIC33EPXXXGS70X/80X 系列 CPU 具有独立的 16 位宽数据存储空间。使用独立的地址发生单元 (AGU) 对数据空间执行读写操作。数据存储映射如图 4-6 所示。

数据存储空间中的所有有效地址 (Effective Address, EA) 均为 16 位宽, 并且指向数据空间内的字节。这种构成方式使得基本数据空间地址范围为 64 KB 或 32K 字。数据存储空间的低半地址部分 (即当  $EA_{<15>} = 0$  时) 用作实现的存储单元, 而高半地址部分 ( $EA_{<15>} = 1$ ) 则保留为程序空间可视性 (PSV) 区域。

dsPIC33EPXXXGS70X/80X 系列器件实现了最大 12 KB 的数据存储空间。如果 EA 指向了该区域以外的存储单元, 则将返回一个全零的字或字节。

### 4.3.1 数据空间宽度

数据存储空间组织为可字节寻址的 16 位宽的块。在数据存储器和寄存器中的数据是以 16 位字为单位对齐的, 但所有数据空间 EA 都将解析为字节。每个字的最低有效字节 (Least Significant Byte, LSB) 具有偶地址, 而最高有效字节 (Most Significant Byte, MSB) 则具有奇地址。

### 4.3.2 数据存储器和对齐方式

为维持与 PIC<sup>®</sup>MCU 器件的向后兼容性和提高数据存储空间的使用效率, dsPIC33EPXXXGS70X/80X 系列指令集支持字和字节操作。字节访问会在内部对接字对齐的存储空间的所有有效地址计算进行调整。例如, 对于执行后修改寄存器间接寻址模式 [Ws++], 字节操作时, 内核将其识别为值  $Ws + 1$ ; 而字操作时, 内核将其识别为值  $Ws + 2$ 。

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。选定的字节被放在数据总线的 LSB。这就是说, 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, 但写入线相互独立。数据字节写操作只写入阵列或寄存器中与字节地址匹配的那一侧。

所有字访问必须按偶地址对齐。不支持不对齐的字数据取操作, 所以在混合字节和字操作时, 或者从 8 位 MCU 代码移植时, 必须要小心。如果试图进行不对齐的读或写操作, 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, 都会执行陷阱, 允许系统和 / 或用户应用程序检查地址错误发生之前的机器状态。

所有装入 W 寄存器的字节都将被装入 LSB, 但 MSB 不会更改。

提供了一条符号扩展 (SE) 指令, 允许用户应用程序将 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, 用户应用程序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。

### 4.3.3 SFR 空间

Near 数据空间的前 4 KB 存储单元 (从 0x0000 至 0x0FFF) 主要被特殊功能寄存器 (Special Function Register, SFR) 占用。dsPIC33EPXXXGS70X/80X 系列内核和外设模块使用这些寄存器来控制器件的工作。

SFR 分布在受其控制的模块中, 通常一个模块会使用一组 SFR。大部分 SFR 空间包含未用的地址单元; 它们读为 0。

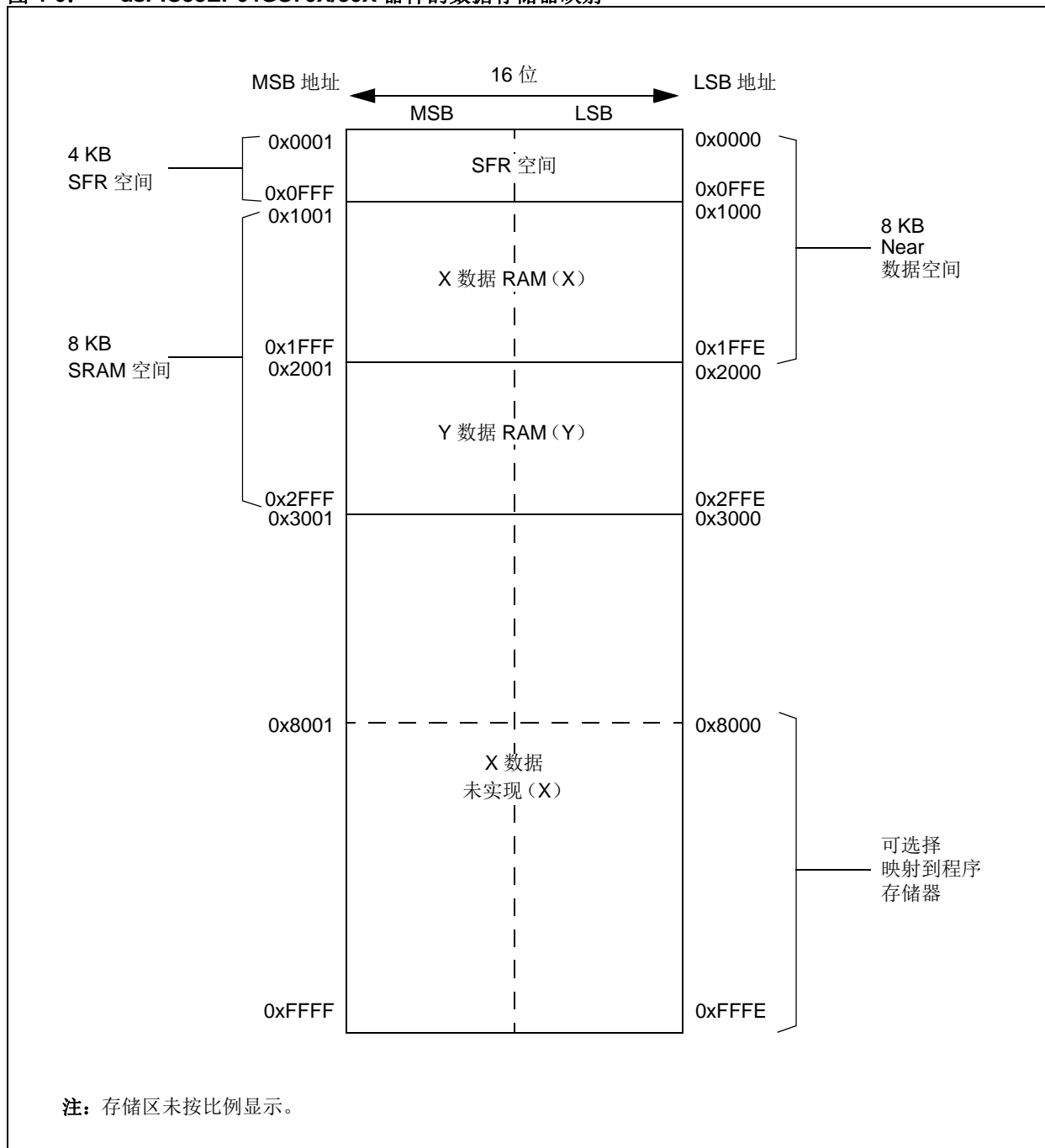
**注:** 不同器件的实际外设功能集和中断也各不相同。关于具体器件的信息, 请参见相应器件的数据表和引脚图。

### 4.3.4 NEAR 数据空间

在 0x0000 和 0x1FFF 之间的 8 KB 的区域被称为 Near 数据空间。可以使用所有存储器直接寻址指令中的 13 位绝对地址字段直接寻址这一空间中的存储单元。此外, 还可以使用 MOV 指令寻址整个数据空间, MOV 指令支持使用 16 位地址字段的存储器直接寻址模式或使用工作寄存器作为地址指针的间接寻址模式。

# dsPIC33EPXXGS70X/80X 系列

图 4-6: dsPIC33EP64GS70X/80X 器件的数据存储器映射



## 4.3.5 X 和 Y 数据空间

dsPIC33EPXXGS70X/80X 内核有两个数据空间 X 和 Y。这两个数据空间可以看作是独立的（对于某些 DSP 指令），或者看作是一个统一的线性地址范围（对于 MCU 指令）。使用两个地址发生单元（AGU）和独立的数据总线来访问这两个数据空间。此特性允许某些指令同时从 RAM 中取两个字，因此提高了 DSP 算法的执行效率，如有限冲激响应（Finite Impulse Response, FIR）滤波器和快速傅立叶变换（Fast Fourier Transform, FFT）。

X 数据空间可用于所有指令，并且支持所有寻址模式。X 数据空间的读数据总线和写数据总线是独立的。所有将数据空间视为组合的 X 和 Y 地址空间的指令均将 X 读数据总线作为读数据路径。X 读数据总线也是双操作数 DSP 指令（MAC 类）的 X 数据预取路径。

MAC 类指令（CLR、ED、EDAC、MAC、MOVSAC、MPY、MPY.N 和 MSC）将同时使用 X 数据空间和 Y 数据空间，从而提供两条可同时对数据进行读操作的路径。

X 和 Y 数据空间都支持所有指令的模寻址，但要受到寻址模式的限制。位反转寻址模式只是在写 X 数据空间时才支持。

所有数据存储寄存器写操作（包括 DSP 指令中的数据存储寄存器写操作）均将数据空间视为组合的 X 和 Y 地址空间。X 和 Y 数据空间的分界取决于具体的器件，且不能由用户编程。

## 4.4 存储器资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 4.4.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“dsPIC33E/PIC24E 程序存储器”（DS70000613）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

## 4.5 特殊功能寄存器映射

表 4-2: SFR 块 000h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
内核			WREG14	01C	0000000000000000	DOSTARTL	03A	xxxxxxxxxxxxxxxx0
WREG0	000	0000000000000000	WREG15	01E	0000100000000000	DOSTARTH	03C	0000000000xxxxxx
WREG1	002	0000000000000000	SPLIM	020	xxxxxxxxxxxxxxxx0	DOENDL	03E	xxxxxxxxxxxxxxxx0
WREG2	004	0000000000000000	ACCAL	022	xxxxxxxxxxxxxxxx	DOENDH	040	0000000000xxxxxx
WREG3	006	0000000000000000	ACCAH	024	xxxxxxxxxxxxxxxx	SR	042	0000000000000000
WREG4	008	0000000000000000	ACCAU	026	00000000xxxxxxxx	CORCON	044	0000000000100000
WREG5	00A	0000000000000000	ACCBH	028	xxxxxxxxxxxxxxxx	MODCON	046	0000000000000000
WREG6	00C	0000000000000000	ACCBH	02A	xxxxxxxxxxxxxxxx	XMODSRT	048	xxxxxxxxxxxxxxxx0
WREG7	00E	0000000000000000	ACCBH	02C	00000000xxxxxxxx	XMODEND	04A	xxxxxxxxxxxxxxxx1
WREG8	010	0000000000000000	PCL	02E	0000000000000000	YMODSRT	04C	xxxxxxxxxxxxxxxx0
WREG9	012	0000000000000000	PCH	030	0000000000000000	YMODEND	04E	xxxxxxxxxxxxxxxx1
WREG10	014	0000000000000000	DSRPAG	032	0000000000000001	XBREV	050	xxxxxxxxxxxxxxxx
WREG11	016	0000000000000000	DSWPAG	034	0000000000000001	DISICNT	052	00xxxxxxxxxxxxxxxx
WREG12	018	0000000000000000	RCOUNT	036	xxxxxxxxxxxxxxxx	TBLPAG	054	00000000xxxxxxxx
WREG13	01A	0000000000000000	DCOUNT	038	xxxxxxxxxxxxxxxx	CTXSTAT	05A	0000000000000000

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

表 4-3: SFR 块 100h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
定时器			TMR5HLD	116	xxxxxxxxxxxxxxxx	IC2CON2	14A	000000000001101
TMR1	100	xxxxxxxxxxxxxxxx	TMR5	118	xxxxxxxxxxxxxxxx	IC2BUF	14C	xxxxxxxxxxxxxxxx
PR1	102	1111111111111111	PR4	11A	1111111111111111	IC2TMR	14E	0000000000000000
T1CON	104	0000000000000000	PR5	11C	1111111111111111	IC3CON1	150	0000000000000000
TMR2	106	xxxxxxxxxxxxxxxx	T4CON	11E	0000000000000000	IC3CON2	152	000000000001101
TMR3HLD	108	xxxxxxxxxxxxxxxx	T5CON	120	0000000000000000	IC3BUF	154	xxxxxxxxxxxxxxxx
TMR3	10A	xxxxxxxxxxxxxxxx	输入捕捉			IC3TMR	156	0000000000000000
PR2	10C	1111111111111111	IC1CON1	140	0000000000000000	IC4CON1	158	0000000000000000
PR3	10E	1111111111111111	IC1CON2	142	000000000001101	IC4CON2	15A	000000000001101
T2CON	110	0000000000000000	IC1BUF	144	xxxxxxxxxxxxxxxx	IC4BUF	15C	xxxxxxxxxxxxxxxx
T3CON	112	0000000000000000	IC1TMR	146	0000000000000000	IC4TMR	15E	0000000000000000
TMR4	114	xxxxxxxxxxxxxxxx	IC2CON1	148	0000000000000000			

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。



# dsPIC33EPXXGS70X/80X 系列

表 4-4: SFR 块 200h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
<b>I2C1 和 I2C2</b>			U1STA	222	0000000010010000	SPI1BRGH	252	0000000000000000
I2C1CONL	200	0001000000000000	U1TXREG	224	0000000xxxxxxx	SPI1MSKL	254	0000000000000000
I2C1CONH	202	0000000000000000	U1RXREG	226	0000000000000000	SPI1MSKH	256	0000000000000000
I2C1STAT	204	0000000000000000	U1BRG	228	0000000000000000	SPI1URDTL	258	0000000000000000
I2C1ADD	206	0000000000000000	U2MODE	230	0000000000000000	SPI1URDTH	25A	0000000000000000
I2C1MSK	208	0000000000000000	U2STA	232	0000000010010000	SPI2CON1L	260	0000000000000000
I2C1BRG	20A	0000000000000000	U2TXREG	234	0000000xxxxxxx	SPI2CON1H	262	0000000000000000
I2C1TRN	20C	0000000011111111	U2RXREG	236	0000000000000000	SPI2CON2L	264	0000000000000000
I2C1RCV	20E	0000000000000000	U2BRG	238	0000000000000000	SPI2CON2H	266	0000000000000000
I2C2CON1	210	0001000000000000	<b>SPI</b>			SPI2STATL	268	000000000101000
I2C2CON2	212	0000000000000000	SPI1CON1L	240	0000000000000000	SPI2STATH	26A	0000000000000000
I2C2STAT	214	0000000000000000	SPI1CON1H	242	0000000000000000	SPI2BUFL	26C	0000000000000000
I2C2ADD	216	0000000000000000	SPI1CON2L	244	0000000000000000	SPI2BUFH	26E	0000000000000000
I2C2MSK	218	0000000000000000	SPI1CON2H	246	0000000000000000	SPI3STAT	270	00xxxxxxx
I2C2BRG	21A	0000000000000000	SPI1STATL	248	000000000101000	SPI2BRGH	272	0000000000000000
I2C2TRN	21C	0000000011111111	SPI1STATH	24A	0000000000000000	SPI2MSKL	274	0000000000000000
I2C2RCV	21E	0000000000000000	SPI1BUFL	24C	0000000000000000	SPI2MSKH	276	0000000000000000
<b>UART1 和 UART2</b>			SPI1BUFH	24E	0000000000000000	SPI2URDTL	278	0000000000000000
U1MODE	220	0000000000000000	SPI1BRGL	250	000xxxxxxx	SPI2URDTH	27A	0000000000000000

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

表 4-5: SFR 块 300h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
<b>ADC</b>			ADCMPOENH	33A	0000000000000000	ADTRIG4L	390	0000000000000000
ADCON1L	300	0000000000000000	ADCMPOLO	33C	0000000000000000	ADTRIG4H	392	0000000000000000
ADCON1H	302	0000000001100000	ADCMPOHI	33E	0000000000000000	ADCMPOCON	3A0	0000000000000000
ADCON2L	304	0000000000000000	ADCMPOENL	340	0000000000000000	ADCMPO1CON	3A4	0000000000000000
ADCON2H	306	0000000000000000	ADCMPOENH	342	0000000000000000	ADBASE	3C0	0000000000000000
ADCON3L	308	0000000000000000	ADCMPOLO	344	0000000000000000	ADLVLTRGL	3D0	0000000000000000
ADCON3H	30A	0000000000000000	ADCMPOHI	346	0000000000000000	ADLVLTRGH	3D2	0000000000000000
ADCON4L	30C	0000000000000000	ADFL0DAT	368	0000000000000000	ADCORE0L	3D4	0000000000000000
ADCON4H	30E	0000000000000000	ADFL0CON	36A	0000000000000000	ADCORE0H	3D6	0000001100000000
ADMOD0L	310	0000000000000000	ADFL1DAT	36C	0000000000000000	ADCORE1L	3D8	0000000000000000
ADMOD0H	312	0000000000000000	ADFL1CON	36E	0000000000000000	ADCORE1H	3DA	0000001100000000
ADMOD1L	314	0000000000000000	ADTRIG0L	380	0000000000000000	ADCORE2L	3DC	0000000000000000
ADIEL	320	0000000000000000	ADTRIG0H	382	0000000000000000	ADCORE2H	3DE	0000001100000000
ADIEH	322	0000000000000000	ADTRIG1L	384	0000000000000000	ADCORE3L	3E0	0000000000000000
ADCSS1L	328	0000000000000000	ADTRIG1H	386	0000000000000000	ADCORE3H	3E2	0000001100000000
ADCSS1H	32A	0000000000000000	ADTRIG2L	388	0000000000000000	ADEIEL	3F0	0000000000000000
ADSTATL	330	0000000000000000	ADTRIG2H	38A	0000000000000000	ADEIEH	3F2	0000000000000000
ADSTATH	332	0000000000000000	ADTRIG3L	38C	0000000000000000	ADEI STATL	3F8	0000000000000000
ADCMPOENL	338	0000000000000000	ADTRIG3H	38E	0000000000000000	ADEI STATH	3FA	0000000000000000

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

# dsPIC33EPXXGS70X/80X 系列

表 4-6: SFR 块 400h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
<b>ADC</b>			C1FCTRL	486	0000000000000000	C1RXM2EID	4BA	xxxxxxxxxxxxxxxxxxxx
ADCON5L	400	0000000000000000	C1FIFO	488	0000000000000000	C1RXF1SID	4C4	xxxxxxxxxxxxxxxxxxxx
ADCON5H	402	0000000000000000	C1INTF	48A	0000000000000000	C1RXF1EID	4C6	xxxxxxxxxxxxxxxxxxxx
ADCAL0L	404	0000000000000000	C1INTE	48C	0000000000000000	C1RXF2SID	4C8	xxxxxxxxxxxxxxxxxxxx
ADCAL0H	406	0000000000000000	C1IEC	48E	0000000000000000	C1RXF2EID	4CA	xxxxxxxxxxxxxxxxxxxx
ADCAL1H	40A	0000000000000000	C1CFG1	490	0000000000000000	C1RXF3SID	4CC	xxxxxxxxxxxxxxxxxxxx
ADCBUF0	40C	0000000000000000	C1CFG2	492	0x000xxxxxxxxxxxxx	C1RXF3EID	4CE	xxxxxxxxxxxxxxxxxxxx
ADCBUF1	40E	0000000000000000	C1FEN1	494	1111111111111111	C1RXF4SID	4D0	xxxxxxxxxxxxxxxxxxxx
ADCBUF2	410	0000000000000000	C1FMSKSEL1	498	0000000000000000	C1RXF4EID	4D2	xxxxxxxxxxxxxxxxxxxx
ADCBUF3	412	0000000000000000	C1FMSKSEL2	49A	0000000000000000	C1RXF5SID	4D4	xxxxxxxxxxxxxxxxxxxx
ADCBUF4	414	0000000000000000	<b>CAN (WIN (C1CTRL&lt;0&gt;) = 0)</b>			C1RXF5EID	4D6	xxxxxxxxxxxxxxxxxxxx
ADCBUF5	416	0000000000000000	C1RXFUL1	4A0	0000000000000000	C1RXF6SID	4D8	xxxxxxxxxxxxxxxxxxxx
ADCBUF6	418	0000000000000000	C1RXFUL2	4A2	0000000000000000	C1RXF6EID	4DA	xxxxxxxxxxxxxxxxxxxx
ADCBUF7	41A	0000000000000000	C1RXOVF1	4A8	0000000000000000	C1RXF7SID	4DC	xxxxxxxxxxxxxxxxxxxx
ADCBUF8	41C	0000000000000000	C1RXOVF2	4AA	0000000000000000	C1RXF7EID	4DE	xxxxxxxxxxxxxxxxxxxx
ADCBUF9	41E	0000000000000000	C1TR01CON	4B0	0000000000000000	C1RXF8SID	4E0	xxxxxxxxxxxxxxxxxxxx
ADCBUF10	420	0000000000000000	C1TR23CON	4B2	0000000000000000	C1RXF8EID	4E2	xxxxxxxxxxxxxxxxxxxx
ADCBUF11	422	0000000000000000	C1TR45CON	4B4	0000000000000000	C1RXF9SID	4E4	xxxxxxxxxxxxxxxxxxxx
ADCBUF12	424	0000000000000000	C1TR67CON	4B6	xxxxxxxxxxxxxxxxxxxx	C1RXF9EID	4E6	xxxxxxxxxxxxxxxxxxxx
ADCBUF13	426	0000000000000000	C1RXD	4C0	xxxxxxxxxxxxxxxxxxxx	C1RXF10SID	4E8	xxxxxxxxxxxxxxxxxxxx
ADCBUF14	428	0000000000000000	C1TXD	4C2	xxxxxxxxxxxxxxxxxxxx	C1RXF10EID	4EA	xxxxxxxxxxxxxxxxxxxx
ADCBUF15	42A	0000000000000000	<b>CAN (WIN (C1CTRL&lt;0&gt;) = 1)</b>			C1RXF11SID	4EC	xxxxxxxxxxxxxxxxxxxx
ADCBUF16	42C	0000000000000000	C1BUFINT1	4A0	0000000000000000	C1RXF11EID	4EE	xxxxxxxxxxxxxxxxxxxx
ADCBUF17	42E	0000000000000000	C1BUFINT2	4A2	0000000000000000	C1RXF12SID	4F0	xxxxxxxxxxxxxxxxxxxx
ADCBUF18	430	0000000000000000	C1BUFINT3	4A4	0000000000000000	C1RXF12EID	4F2	xxxxxxxxxxxxxxxxxxxx
ADCBUF19	432	0000000000000000	C1BUFINT4	4A6	0000000000000000	C1RXF13SID	4F4	xxxxxxxxxxxxxxxxxxxx
ADCBUF20	434	0000000000000000	C1RXM0SID	4B0	xxxxxxxxxxxxxxxxxxxx	C1RXF13EID	4F6	xxxxxxxxxxxxxxxxxxxx
ADCBUF21	436	0000000000000000	C1RXM0EID	4B2	xxxxxxxxxxxxxxxxxxxx	C1RXF14SID	4F8	xxxxxxxxxxxxxxxxxxxx
<b>CAN (WIN (C1CTRL&lt;0&gt;) = 0 或 1)</b>			C1RXM1SID	4B4	xxxxxxxxxxxxxxxxxxxx	C1RXF14EID	4FA	xxxxxxxxxxxxxxxxxxxx
C1CTRL1	480	0000100100000000	C1RXM1EID	4B6	xxxxxxxxxxxxxxxxxxxx	C1RXF15SID	4FC	xxxxxxxxxxxxxxxxxxxx
C1CTRL2	482	0000000000000000	<b>CAN</b>			C1RXF15EID	4FE	xxxxxxxxxxxxxxxxxxxx
C1VEC	484	0000000010000000	C1RXM2SID	4B8	xxxxxxxxxxxxxxxxxxxx			

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

表 4-7: SFR 块 500h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
<b>PGA</b>			PGA2CAL	50A	0000000000000000	CMP2DAC	546	0000000000000000
ISRCCON	500	0000000000000000	<b>比较器</b>			CMP3CON	548	0000000000000000
PGA1CON	504	0000000000000000	CMP1CON	540	0000000000000000	CMP3DAC	54A	0000000000000000
PGA1CAL	506	0000000000000000	CMP1DAC	542	0000000000000000	CMP4CON	54C	0000000000000000
PGA2CON	508	0000000000000000	CMP2CON	544	0000000000000000	CMP4DAC	54E	0000000000000000

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

# dsPIC33EPXXGS70X/80X 系列

表 4-8: SFR 块 600h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
SPI			RPOR8	678	0000000000000000	RPINR7	6AE	0000000000000000
SPI3CON1L	600	0000000000000000	RPOR9	67A	0000000000000000	RPINR8	6B0	0000000000000000
SPI3CON1H	602	0000000000000000	RPOR10	67C	0000000000000000	RPINR11	6B6	0000000000000000
SPI3CON2L	604	0000000000000000	RPOR11	67E	0000000000000000	RPINR12	6B8	0000000000000000
SPI3CON2H	606	0000000000000000	RPOR12	680	0000000000000000	RPINR13	6BA	0000000000000000
SPI3STATL	608	0000000000101000	RPOR13	682	0000000000000000	RPINR18	6C4	0000000000000000
SPI3STATH	60A	0000000000000000	RPOR14	684	0000000000000000	RPINR19	6C6	0000000000000000
SPI3BUFL	60C	0000000000000000	RPOR15	686	0000000000000000	RPINR20	6C8	0000000000000000
SPI3BUFH	60E	0000000000000000	RPOR17	68A	0000000000000000	RPINR21	6CA	0000000000000000
SPI3BRGL	610	000xxxxxxxxxxxxxxx	RPOR18	68C	0000000000000000	RPINR22	6CC	0000000000000000
SPI3BRGH	612	0000000000000000	RPOR19	68E	0000000000000000	RPINR23	6CE	0000000000000000
SPI3IMSKL	614	0000000000000000	RPOR20	690	0000000000000000	RPINR26	6D4	0000000000000000
SPI3IMSKH	616	0000000000000000	RPOR21	692	0000000000000000	RPINR29	6DA	0000000000000000
SPI3URDTL	618	0000000000000000	RPOR22	694	0000000000000000	RPINR30	6DC	0000000000000000
SPI3URDTH	61A	0000000000000000	RPOR23	696	0000000000000000	RPINR37	6EA	0000000000000000
RPOR0	668	0000000000000000	RPOR24	698	0000000000000000	RPINR38	6EC	0000000000000000
RPOR1	66A	0000000000000000	RPOR25	69A	0000000000000000	RPINR42	6F4	0000000000000000
RPOR2	66C	0000000000000000	RPOR26	69C	0000000000000000	RPINR43	6F6	0000000000000000
RPOR3	66E	0000000000000000	RPINR0	6A0	0000000000000000	RPINR45	6FA	0000000000000000
RPOR4	670	0000000000000000	RPINR1	6A2	0000000000000000	RPINR46	6FC	0000000000000000
RPOR5	672	0000000000000000	RPINR2	6A4	0000000000000000			
RPOR6	674	0000000000000000	RPINR3	6A6	0000000000000000			

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

# dsPIC33EPXXGS70X/80X 系列

表 4-9: SFR 块 700h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
<b>NVM</b>			C2INTF	78A	0000000000000000	C2RXF1SID	7C4	xxxxxxxxxxxxxxxx
NVMCON	728	0000000000000000	C2INTE	78C	0000000000000000	C2RXF1EID	7C6	xxxxxxxxxxxxxxxx
NVMADR	72A	0000000000000000	C2EC	78E	0000000000000000	C2RXF2SID	7C8	xxxxxxxxxxxxxxxx
NVMADRU	72C	0000000000000000	C2CFG1	790	0000000000000000	C2RXF2EID	7CA	xxxxxxxxxxxxxxxx
NVMKEY	72E	0000000000000000	C2CFG2	792	0x000xxxxxxxxxxxxx	C2RXF3SID	7CC	xxxxxxxxxxxxxxxx
NVMSRCADR	730	0000000000000000	C2FEN1	794	1111111111111111	C2RXF3EID	7CE	xxxxxxxxxxxxxxxx
NVMSRCADRH	732	0000000000000000	C2FMSKSEL1	798	0000000000000000	C2RXF4SID	7D0	xxxxxxxxxxxxxxxx
<b>系统控制</b>			C2FMSKSEL2	79A	0000000000000000	C2RXF4EID	7D2	xxxxxxxxxxxxxxxx
RCON	740	0x00x0x01x0xxxxx	<b>CAN (WIN (C1CTR1&lt;0&gt;) = 0)</b>			C2RXF5SID	7D4	xxxxxxxxxxxxxxxx
OSCCON	742	0000000000000000	C2RXFUL1	7A0	0000000000000000	C2RXF5EID	7D6	xxxxxxxxxxxxxxxx
CLKDIV	744	0000000000000000	C2RXFUL2	7A2	0000000000000000	C2RXF6SID	7D8	xxxxxxxxxxxxxxxx
PLLFBD	746	0000000000000000	C2RXOVF1	7A8	0000000000000000	C2RXF6EID	7DA	xxxxxxxxxxxxxxxx
OSCTUN	748	0000000000000000	C2RXOVF2	7AA	0000000000000000	C2RXF7SID	7DC	xxxxxxxxxxxxxxxx
LFSR	74C	0000000000000000	C2TR01CON	7B0	0000000000000000	C2RXF7EID	7DE	xxxxxxxxxxxxxxxx
REFOCON	74E	0000000000000000	C2TR23CON	7B2	0000000000000000	C2RXF8SID	7E0	xxxxxxxxxxxxxxxx
ACLKCON	750	0000000000000000	C2TR45CON	7B4	0000000000000000	C2RXF8EID	7E2	xxxxxxxxxxxxxxxx
<b>PMD</b>			C2TR67CON	7B6	xxxxxxxxxxxxxxxx	C2RXF9SID	7E4	xxxxxxxxxxxxxxxx
PMD1	760	0000000000000000	C2RXD	7C0	xxxxxxxxxxxxxxxx	C2RXF9EID	7E6	xxxxxxxxxxxxxxxx
PMD2	762	0000000000000000	C2TXD	7C2	xxxxxxxxxxxxxxxx	C2RXF10SID	7E8	xxxxxxxxxxxxxxxx
PMD3	764	0000000000000000	<b>CAN (WIN (C1CTR1&lt;0&gt;) = 1)</b>			C2RXF10EID	7EA	xxxxxxxxxxxxxxxx
PMD4	766	0000000000000000	C2BUFNT1	7A0	0000000000000000	C2RXF11SID	7EC	xxxxxxxxxxxxxxxx
PMD6	76A	0000000000000000	C2BUFNT2	7A2	0000000000000000	C2RXF11EID	7EE	xxxxxxxxxxxxxxxx
PMD7	76C	0000000000000000	C2BUFNT3	7A4	0000000000000000	C2RXF12SID	7F0	xxxxxxxxxxxxxxxx
PMD8	76E	0000000000000000	C2BUFNT4	7A6	0000000000000000	C2RXF12EID	7F2	xxxxxxxxxxxxxxxx
<b>CAN (WIN (C1CTR1&lt;0&gt;) = 0 或 1)</b>			C2RXM0SID	7B0	xxxxxxxxxxxxxxxx	C2RXF13SID	7F4	xxxxxxxxxxxxxxxx
C2CTRL1	780	0000010010000000	C2RXM0EID	7B2	xxxxxxxxxxxxxxxx	C2RXF13EID	7F6	xxxxxxxxxxxxxxxx
C2CTRL2	782	0000000000000000	C2RXM1SID	7B4	xxxxxxxxxxxxxxxx	C2RXF14SID	7F8	xxxxxxxxxxxxxxxx
C2VEC	784	0000000001000000	C2RXM1EID	7B6	xxxxxxxxxxxxxxxx	C2RXF14EID	7FA	xxxxxxxxxxxxxxxx
C2FCTRL	786	0000000000000000	C2RXM2SID	7B8	xxxxxxxxxxxxxxxx	C2RXF15SID	7FC	xxxxxxxxxxxxxxxx
C2FIFO	788	0000000000000000	C2RXM2EID	7BA	xxxxxxxxxxxxxxxx	C2RXF15EID	7FE	xxxxxxxxxxxxxxxx

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

# dsPIC33EPXXGS70X/80X 系列

表 4-10: SFR 块 800h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
中断控制器			IEC9	832	0000000000000000	IPC26	874	000000001000100
IFS0	800	0000000000000000	IEC10	834	0000000000000000	IPC27	876	0100010000000000
IFS1	802	0000000000000000	IEC11	836	0000000000000000	IPC28	878	0100010001000100
IFS2	804	0000000000000000	IPC0	840	0100010001000100	IPC29	87A	000000001000100
IFS3	806	0000000000000000	IPC1	842	0100010001000000	IPC35	886	0100010000000000
IFS4	808	0000000000000000	IPC2	844	0100010001000100	IPC36	888	0000000000000000
IFS5	80A	0000000000000000	IPC3	846	0100000001000100	IPC37	88A	0100000000000000
IFS6	80C	0000000000000000	IPC4	848	0100010001000100	IPC38	88C	0100010001000100
IFS7	80E	0000000000000000	IPC5	84A	0000000000000100	IPC39	88E	0100010001000100
IFS8	810	0000000000000000	IPC6	84C	0100010001000000	IPC40	890	0100010001000100
IFS9	812	0000000000000000	IPC7	84E	0100010001000100	IPC41	892	0100010001000100
IFS10	814	0000000000000000	IPC8	850	0000000001000100	IPC42	894	0000000001000100
IFS11	816	0000000000000000	IPC9	852	0000010001000000	IPC43	896	0000010001000000
IEC0	820	0000000000000000	IPC11	856	0000000000000000	IPC44	898	0100010001000000
IEC1	822	0000000000000000	IPC12	858	0000010001000000	IPC45	89A	0000000000000100
IEC2	824	0000000000000000	IPC13	85A	0000010000000000	IPC46	89C	0100010000000000
IEC3	826	0000000000000000	IPC14	85C	0000000001000000	IPC47	89E	0000010001000100
IEC4	828	0000000000000000	IPC16	860	0000010001000000	INTCON1	8C0	0000000000000000
IEC5	82A	0000000000000000	IPC18	864	0000000001000000	INTCON2	8C2	0000000000000000
IEC6	82C	0000000000000000	IPC23	86E	0100010000000000	INTCON3	8C4	0000000000000000
IEC7	82E	0000000000000000	IPC24	870	0000010001000100	INTCON4	8C6	0000000000000000
IEC8	830	0000000000000000	IPC25	872	0100000000000000	INTTREG	8C8	0000000000000000

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

表 4-11: SFR 块 900h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
输出比较			OC3R	91A	xxxxxxxxxxxxxxxx	CLC2CONH	9CE	0000000000000000
OC1CON1	900	0000000000000000	OC3TMR	91C	0000000000000000	CLC2SEL	9D0	0000000000000000
OC1CON2	902	0000000000001100	OC4CON1	91E	0000000000000000	CLC2GLSL	9D4	0000000000000000
OC1RS	904	xxxxxxxxxxxxxxxx	OC4CON2	920	0000000000001100	CLC2GLSH	9D6	0000000000000000
OC1R	906	xxxxxxxxxxxxxxxx	OC4RS	922	xxxxxxxxxxxxxxxx	CLC3CONL	9D8	0000000000000000
OC1TMR	908	0000000000000000	OC4R	924	xxxxxxxxxxxxxxxx	CLC3CONH	9DA	0000000000000000
OC2CON1	90A	0000000000000000	OC4TMR	926	0000000000000000	CLC3SEL	9DC	0000000000000000
OC2CON2	90C	0000000000001100	<b>CLC</b>			CLC3GLSL	9E0	0000000000000000
OC2RS	90E	xxxxxxxxxxxxxxxx	CLC1CONL	9C0	0000000000000000	CLC3GLSH	9E2	0000000000000000
OC2R	910	xxxxxxxxxxxxxxxx	CLC1CONH	9C2	0000000000000000	CLC4CONL	9E4	0000000000000000
OC2TMR	912	0000000000000000	CLC1SEL	9C4	0000000000000000	CLC4CONH	9E6	0000000000000000
OC3CON1	914	0000000000000000	CLC1GLSL	9C8	0000000000000000	CLC4SEL	9E8	0000000000000000
OC3CON2	916	0000000000001100	CLC1GLSH	9CA	0000000000000000	CLC4GLSL	9EC	0000000000000000
OC3RS	918	xxxxxxxxxxxxxxxx	CLC2CONL	9CC	0000000000000000	CLC4GLSH	9EE	0000000000000000

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

# dsPIC33EPXXGS70X/80X 系列

表 4-12: SFR 块 A00h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
PTG			PTGADJ	AD2	0000000000000000	PTGQUE7	AE6	xxxxxxxxxxxxxxxx
PTGCST	AC0	0000000000000000	PTGL0	AD4	0000000000000000	PTGQUE8	AE8	xxxxxxxxxxxxxxxx
PTGCON	AC2	0000000000000000	PTGQPTR	AD6	0000000000000000	PTGQUE9	AEA	xxxxxxxxxxxxxxxx
PTGBTE	AC4	0000000000000000	PTGQUE0	AD8	xxxxxxxxxxxxxxxx	PTGQUE10	AEC	xxxxxxxxxxxxxxxx
PTGHOLD	AC6	0000000000000000	PTGQUE1	ADA	xxxxxxxxxxxxxxxx	PTGQUE11	AEE	xxxxxxxxxxxxxxxx
PTGTOLIM	AC8	0000000000000000	PTGQUE2	ADC	xxxxxxxxxxxxxxxx	PTGQUE12	AF0	xxxxxxxxxxxxxxxx
PTGT1LIM	ACA	0000000000000000	PTGQUE3	ADE	xxxxxxxxxxxxxxxx	PTGQUE13	AF2	xxxxxxxxxxxxxxxx
PTGSDLIM	ACC	0000000000000000	PTGQUE4	AE0	xxxxxxxxxxxxxxxx	PTGQUE14	AF4	xxxxxxxxxxxxxxxx
PTGC0LIM	ACE	0000000000000000	PTGQUE5	AE2	xxxxxxxxxxxxxxxx	PTGQUE15	AF6	xxxxxxxxxxxxxxxx
PTGC1LIM	AD0	0000000000000000	PTGQUE6	AE4	xxxxxxxxxxxxxxxx			

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

表 4-13: SFR 块 B00h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
DMA			DMA1STBL	B18	0000000000000000	DMA3REQ	B32	0000000000000000
DMA0CON	B00	0000000000000000	DMA1STBH	B1A	0000000000000000	DMA3STAL	B34	0000000000000000
DMA0REQ	B02	0000000000000000	DMA1PAD	B1C	0000000000000000	DMA3STAH	B36	0000000000000000
DMA0STAL	B04	0000000000000000	DMA1CNT	B1E	0000000000000000	DMA3STBL	B38	0000000000000000
DMA0STAH	B06	0000000000000000	DMA2CON	B20	0000000000000000	DMA3STBH	B3A	0000000000000000
DMA0STBL	B08	0000000000000000	DMA2REQ	B22	0000000000000000	DMA3PAD	B3C	0000000000000000
DMA0STBH	B0A	0000000000000000	DMA2STAL	B24	0000000000000000	DMA3CNT	B3E	0000000000000000
DMA0PAD	B0C	0000000000000000	DMA2STAH	B26	0000000000000000	DMA3PWC	BF0	0000000000000000
DMA0CNT	B0E	0000000000000000	DMA2STBL	B28	0000000000000000	DMA3RQC	BF2	0000000000000000
DMA1CON	B10	0000000000000000	DMA2STBH	B2A	0000000000000000	DMA3PPS	BF4	0000000000000000
DMA1REQ	B12	0000000000000000	DMA2PAD	B2C	0000000000000000	DMA3LCA	BF6	000000000001111
DMA1STAL	B14	0000000000000000	DMA2CNT	B2E	0000000000000000	DSADRL	BF8	0000000000000000
DMA1STAH	B16	0000000000000000	DMA3CON	B30	0000000000000000	DSADRH	BFA	0000000000000000

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

# dsPIC33EPXXGS70X/80X 系列

**表 4-14: SFR 块 C00h-D00h**

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
<b>PWM</b>			FCLCON3	C64	0000000000000000	IOCON6	CC2	1100000000000000
PTCON	C00	0000000000000000	PDC3	C66	0000000000000000	FCLCON6	CC4	0000000000000000
PTCON2	C02	0000000000000000	PHASE3	C68	0000000000000000	PDC6	CC6	0000000000000000
PTPER	C04	1111111111111000	DTR3	C6A	0000000000000000	PHASE6	CC8	0000000000000000
SEVTCMP	C06	0000000000000000	ALTDTR3	C6C	0000000000000000	DTR6	CCA	0000000000000000
MDC	C0A	0000000000000000	SDC3	C6E	0000000000000000	ALTDTR6	CCC	0000000000000000
STCON	C0E	0000000000000000	SPHASE3	C70	0000000000000000	SDC6	CCE	0000000000000000
STCON2	C10	0000000000000000	TRIG3	C72	0000000000000000	SPHASE6	CD0	0000000000000000
STPER	C12	1111111111111000	TRGCON3	C74	0000000000000000	TRIG6	CD2	0000000000000000
SSEVTCMP	C14	0000000000000000	STRIG3	C76	0000000000000000	TRGCON6	CD4	0000000000000000
CHOP	C1A	0000000000000000	PWMCAP3	C78	0000000000000000	STRIG6	CD6	0000000000000000
PWMKEY	C1E	xxxxxxxxxxxxxxxx	LEBCON3	C7A	0000000000000000	PWMCAP6	CD8	0000000000000000
<b>PWM 发生器</b>			LEBDLY3	C7C	0000000000000000	LEBCON6	CDA	0000000000000000
PWMCON1	C20	0000000000000000	AUXCON3	C7E	0000000000000000	LEBDLY6	CD6	0000000000000000
IOCON1	C22	1100000000000000	PWMCON4	C80	0000000000000000	AUXCON6	CDE	0000000000000000
FCLCON1	C24	0000000000000000	IOCON4	C82	1100000000000000	PWMCON7	CE0	0000000000000000
PDC1	C26	0000000000000000	FCLCON4	C84	0000000000000000	IOCON7	CE2	1100000000000000
PHASE1	C28	0000000000000000	PDC4	C86	0000000000000000	FCLCON7	CE4	0000000000000000
DTR1	C2A	0000000000000000	PHASE4	C88	0000000000000000	PDC7	CE6	0000000000000000
ALTDTR1	C2C	0000000000000000	DTR4	C8A	0000000000000000	PHASE7	CE8	0000000000000000
SDC1	C2E	0000000000000000	ALTDTR4	C8C	0000000000000000	DTR7	CEA	0000000000000000
SPHASE1	C30	0000000000000000	SDC4	C8E	0000000000000000	ALTDTR7	CEC	0000000000000000
TRIG1	C32	0000000000000000	SPHASE4	C90	0000000000000000	SDC7	CEE	0000000000000000
TRGCON1	C34	0000000000000000	TRIG4	C92	0000000000000000	SPHASE7	CF0	0000000000000000
STRIG1	C36	0000000000000000	TRGCON4	C94	0000000000000000	TRIG7	CF2	0000000000000000
PWMCAP1	C38	0000000000000000	STRIG4	C96	0000000000000000	TRGCON7	CF4	0000000000000000
LEBCON1	C3A	0000000000000000	PWMCAP4	C98	0000000000000000	STRIG7	CF6	0000000000000000
LEBDLY1	C3C	0000000000000000	LEBCON4	C9A	0000000000000000	PWMCAP7	CF8	0000000000000000
AUXCON1	C3E	0000000000000000	LEBDLY4	C9C	0000000000000000	LEBCON7	CFA	0000000000000000
PWMCON2	C40	0000000000000000	AUXCON4	C9E	0000000000000000	LEBDLY7	CFC	0000000000000000
IOCON2	C42	1100000000000000	PWMCON5	CA0	0000000000000000	AUXCON7	CFE	0000000000000000
FCLCON2	C44	0000000000000000	IOCON5	CA2	1100000000000000	PWMCON8	D00	0000000000000000
PDC2	C46	0000000000000000	FCLCON5	CA4	0000000000000000	IOCON8	D02	1100000000000000
PHASE2	C48	0000000000000000	PDC5	CA6	0000000000000000	FCLCON8	D04	0000000000000000
DTR2	C4A	0000000000000000	PHASE5	CA8	0000000000000000	PDC8	D06	0000000000000000
ALTDTR2	C4C	0000000000000000	DTR5	CAA	0000000000000000	PHASE8	D08	0000000000000000
SDC2	C4E	0000000000000000	ALTDTR5	CAC	0000000000000000	ALTDTR8	D0C	0000000000000000
SPHASE2	C50	0000000000000000	SDC5	CAE	0000000000000000	SDC8	D0E	0000000000000000
TRIG2	C52	0000000000000000	SPHASE5	CB0	0000000000000000	SPHASE8	D10	0000000000000000
TRGCON2	C54	0000000000000000	TRIG5	CB2	0000000000000000	TRIG8	D12	0000000000000000
STRIG2	C56	0000000000000000	TRGCON5	CB4	0000000000000000	TRGCON8	D14	0000000000000000
PWMCAP2	C58	0000000000000000	STRIG5	CB6	0000000000000000	STRIG8	D16	0000000000000000
LEBCON2	C5A	0000000000000000	PWMCAP5	CB8	0000000000000000	PWMCAP8	D18	0000000000000000
LEBDLY2	C5C	0000000000000000	LEBCON5	CBA	0000000000000000	LEBCON8	D1A	0000000000000000
AUXCON2	C5E	0000000000000000	LEBDLY5	CBC	0000000000000000	LEBDLY8	D1C	0000000000000000
PWMCON3	C60	0000000000000000	AUXCON5	CBE	0000000000000000	AUXCON8	D1E	0000000000000000
IOCON3	C62	1100000000000000	PWMCON6	CC0	0000000000000000			

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。

# dsPIC33EPXXGS70X/80X 系列

表 4-15: SFR 块 E00h-F00h

寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态	寄存器	地址	所有复位时的状态
<b>PORTA</b>			ANSELB	E1E	0000001011101111	CNPDD	E3C	0000000000000000
TRISA	E00	0000000000011111	<b>PORTC</b>			ANSELB	E3E	0110000110100000
PORTA	E02	0000000000000000	TRISC	E20	0111011111111111	<b>PORTE</b>		
LATA	E04	0000000000000000	PORTC	E22	0000000000000000	TRISE	E40	1111111111111111
ODCA	E06	0000000000000000	LATC	E24	0000000000000000	PORTE	E42	0000000000000000
CNENA	E08	0000000000000000	ODCC	E26	0000000000000000	LATE	E44	0000000000000000
CNPUA	E0A	0000000000000000	CNENC	E28	0000000000000000	ODCE	E46	0000000000000000
CNPDA	E0C	0000000000000000	CNPUC	E2A	0000000000000000	CNENE	E48	0000000000000000
ANSELA	E0E	0000000000001111	CNPDC	E2C	0000000000000000	CNPUE	E4A	0000000000000000
<b>PORTB</b>			ANSELB	E2E	0001011001110111	CNPDE	E4C	0000000000000000
TRISB	E10	0111101111111111	<b>PORTD</b>			ANSELE	E4E	1100000100000000
PORTB	E12	0000000000000000	TRISD	E30	1111111111111111	<b>CPU</b>		
LATB	E14	0000000000000000	PORTD	E32	0000000000000000	VISI	F88	0000000000000000
ODCB	E16	0000000000000000	LATD	E34	0000000000000000	<b>JTAG</b>		
CNENB	E18	0000000000000000	ODCD	E36	0000000000000000	JDATAH	FF0	0000000000000000
CNPUB	E1A	0000000000000000	CNEND	E38	0000000000000000	JDATAL	FF2	0000000000000000
CNPDB	E1C	0000000000000000	CNPUD	E3A	0000000000000000			

图注: x = 值未知或不确定。地址值使用十六进制表示。复位值使用二进制表示。



## 4.5.1 分页存储器方案

dsPIC33EPXXGS70X/80X 系列架构通过分页方案扩展可用数据空间，分页方案支持使用 MOV 指令以线性方式（对有效地址（EA）执行前修改或后修改）来访问可用数据空间。基本数据空间的高半地址部分与数据空间读页（DSRPAG）寄存器配合使用，构成程序空间可视性（PSV）地址。

数据空间读页（DSRPAG）寄存器位于 SFR 空间中。图 4-7 给出了 PSV 地址的构成。当  $DSRPAG\langle 9 \rangle = 1$  且基址位  $EA\langle 15 \rangle = 1$  时， $DSRPAG\langle 8:0 \rangle$  位与  $EA\langle 14:0 \rangle$  相连构成 24 位 PSV 读地址。

通过分页存储器方案可以访问 PSV 存储区中的多个 32 KB 窗口。数据空间读页（DSRPAG）寄存器与数据空间地址高半部分组合使用时，最多可以提供 8 MB 的 PSV 地址空间。分页数据存储空间如图 4-8 所示。

DSRPAG 为 0x200 或更大时，可以访问程序空间（PS）。仅支持使用 DSRPAG 寄存器对 PS 执行读操作。

图 4-7: 程序空间可视性（PSV）的读地址生成方式

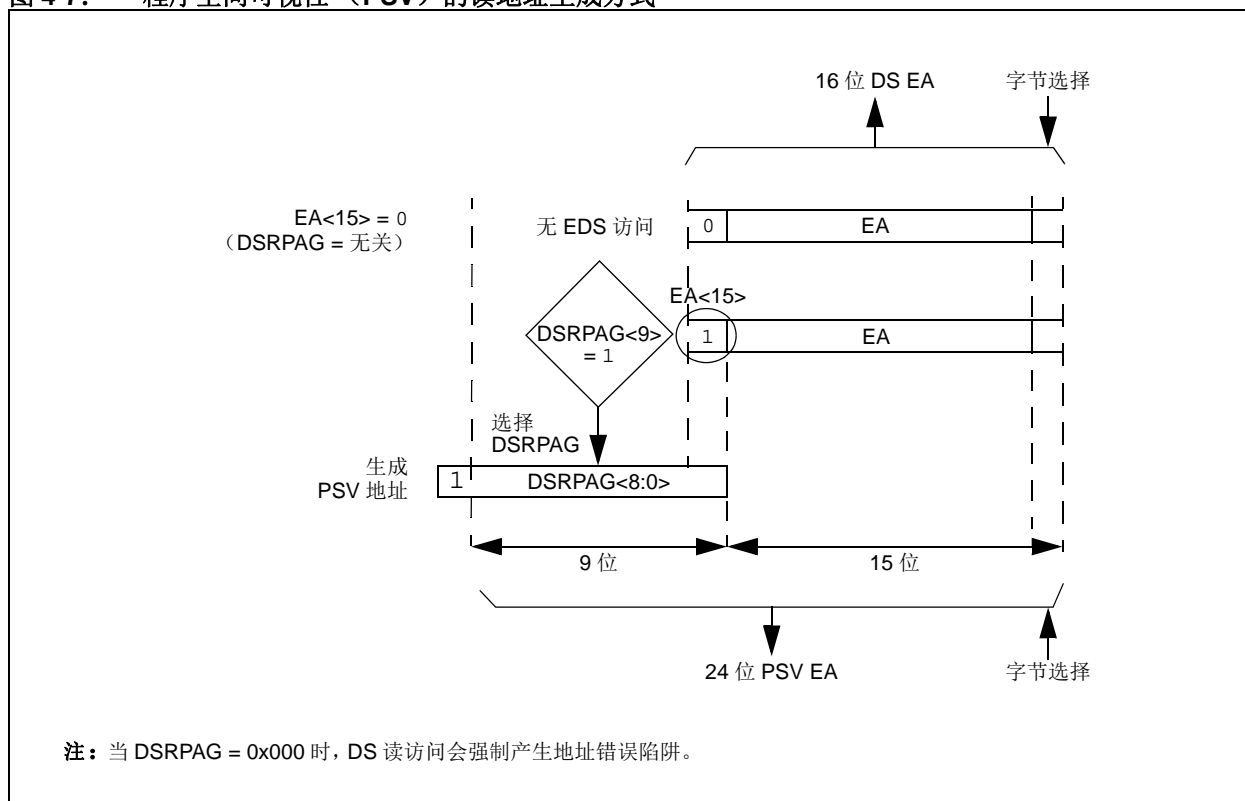
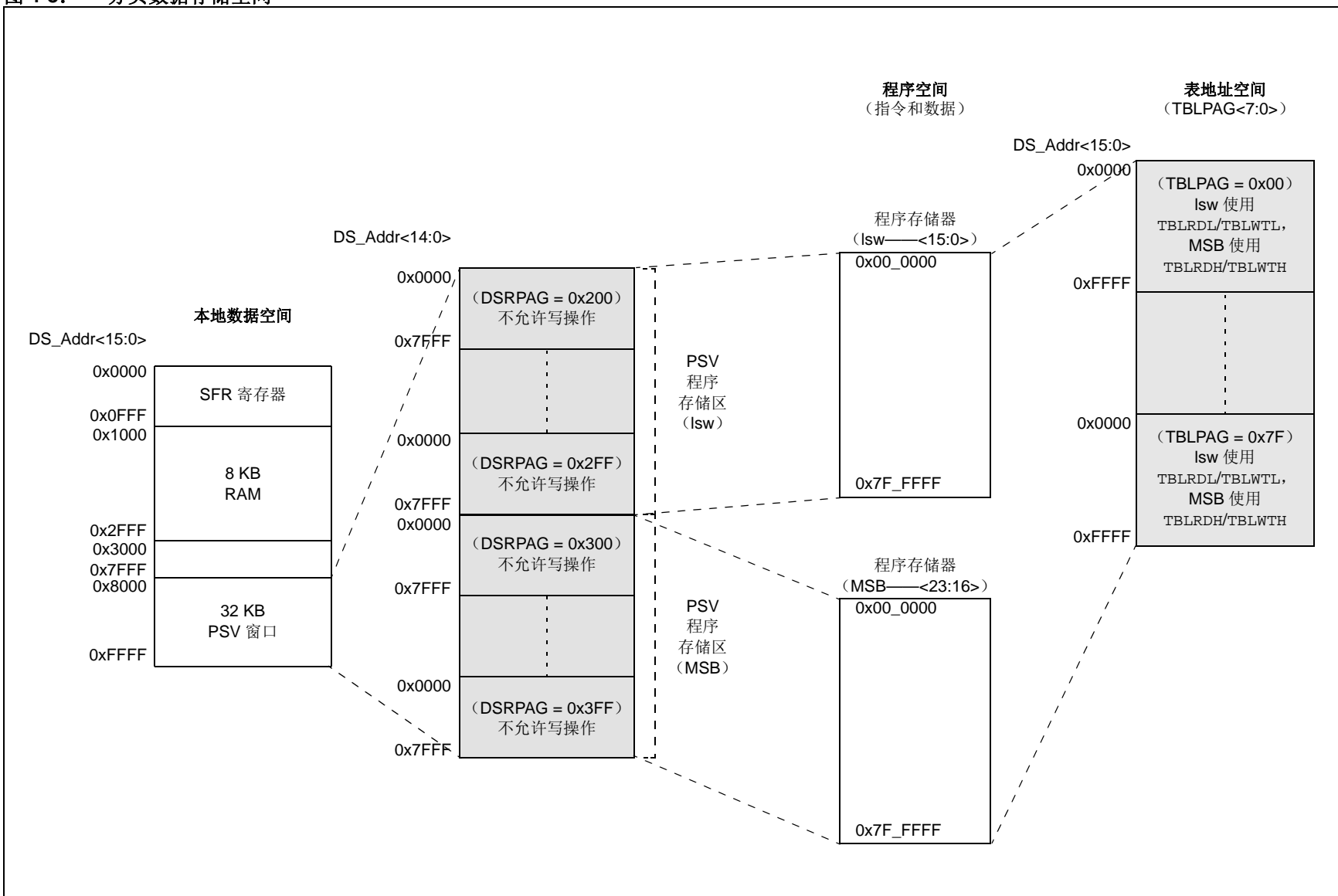


图 4-8: 分页数据存储空间



# dsPIC33EPXXGS70X/80X 系列

发生 PSV 页上溢或下溢时，EA<15> 会由于寄存器间接 EA 计算而清零。在以下情况下，PSV 页中的 EA 会在页边界处发生上溢或下溢：

- 初始地址（在修改之前）寻址 PSV 页
- EA 计算使用执行前修改或执行后修改的寄存器间接寻址模式；但是，这不包括寄存器偏移量寻址

通常在检测到上溢时，DSRPAG 寄存器会递增，EA<15> 位会置 1，以使基址保持在 PSV 窗口内。当检测到下溢时，DSRPAG 寄存器会递减，EA<15> 位会置 1，以使基址保持在 PSV 窗口内。这可以产生线性的 PSV 地址空间，但仅限使用寄存器间接寻址模式时。

在进入和退出 Page 0 和 PSV 空间的边界时，会出现上述操作的例外情形。表 4-16 列出了不同边界处上溢和下溢情形的影响。

在以下情形中，发生上溢或下溢时，EA<15> 位会置 1，DSRPAG 不会发生修改；因此，EA 会折回到当前页的起始位置：

- 带寄存器偏移量的寄存器间接寻址
- 模寻址
- 位反转寻址

表 4-16: PAGE 0 和 PSV 空间边界处的上溢和下溢情形 (2,3,4)

O/U, R/W	操作	之前			之后		
		DSxPAG	DS EA<15>	页说明	DSxPAG	DS EA<15>	页说明
O, 读	[++Wn] 或 [Wn++]	DSRPAG = 0x2FF	1	PSV: 最后一个 lsw 页	DSRPAG = 0x300	1	PSV: 第一个 MSB 页
O, 读	[Wn++]	DSRPAG = 0x3FF	1	PSV: 最后一个 MSB 页	DSRPAG = 0x3FF	0	见注 1
U, 读	[--Wn] 或 [Wn--]	DSRPAG = 0x001	1	PSV 页	DSRPAG = 0x001	0	见注 1
U, 读	[--Wn] 或 [Wn--]	DSRPAG = 0x200	1	PSV: 第一个 lsw 页	DSRPAG = 0x200	0	见注 1
U, 读	[Wn--]	DSRPAG = 0x300	1	PSV: 第一个 MSB 页	DSRPAG = 0x2FF	1	PSV: 最后一个 lsw 页

图注：O = 上溢，U = 下溢，R = 读，W = 写

注 1：寄存器间接寻址现在寻址基本数据空间（0x0000-0x7FFFF）中的某个存储单元。

- 2: DSRPAG = 0x000 时的 EDS 访问会产生地址错误陷阱。
- 3: 仅支持使用 DSRPAG 对 PS 执行读操作。
- 4: 偏移量很大时，不支持伪线性寻址。

# dsPIC33EPXXGS70X/80X 系列

## 4.5.2 扩展 X 数据空间

无论数据空间读页寄存器的内容如何，总是可以访问介于 0x0000 和 0x7FFF 之间的基本地址空间的低地址单元。它可以通过寄存器间接寻址指令进行间接寻址。它可以视为位于默认的 EDS Page 0 中（即，EDS 地址范围 0x000000 至 0x007FFF，并且对于该地址范围，基址位 EA<15> = 0）。但是，Page 0 不能通过基本数据空间高 32 KB（0x8000 至 0xFFFF）结合 DSRPAG = 0x000 来进行访问。因此在复位时，DSRPAG 会被初始化为 0x001。

- 注 1:** DSRPAG 不应用于访问 Page 0。DSRPAG 设置为 0x000 时进行 EDS 访问会产生地址错误陷阱。
- 2:** 用软件清零 DSRPAG 没有任何作用。

其余 PSV 页只能使用 DSRPAG 寄存器结合基本地址空间高 32 KB（0x8000 至 0xFFFF）进行访问，此时基址位 EA<15> = 1。

## 4.5.3 软件堆栈

W15 寄存器用作专用软件堆栈指针（Software Stack Pointer, SSP），由异常处理、子程序调用和返回自动修改；但是，W15 可以被任何指令以与所有其他 W 寄存器相同的方式引用。这样就简化了堆栈指针的读、写和操作（例如，创建堆栈帧）。

- 注:** 为了防止不对齐的堆栈访问，W15<0>被硬件固定设置为 0。

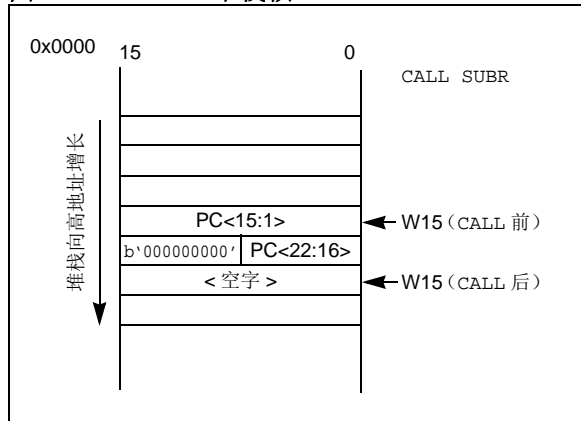
所有复位均将 W15 初始化为 0x1000。该地址确保在所有 dsPIC33EPXXGS70X/80X 器件中 SSP 将指向有效的 RAM，并允许不可屏蔽陷阱异常使用堆栈。这些可能在 SSP 被用户软件初始化之前发生。在初始化期间，可以将 SSP 再编程以指向数据空间内的任何存储单元。

软件堆栈指针总是指向第一个可用的空字并从低地址到高地址填充软件堆栈。图 4-9 显示了它如何在弹出堆栈（读）时预递减，以及在压入堆栈（写）时后递增。

当 PC 压入堆栈时，PC<15:0> 被压入第一个可用的堆栈字，然后 PC<22:16> 被压入第二个可用的堆栈单元。对于任何 CALL 指令执行期间的 PC 压栈，压栈前 PC 的 MSB 被零扩展，如图 4-9 所示。异常处理期间，PC 的 MSB 与 CPU 状态寄存器 SR 的低 8 位组合在一起。这样就能在中断处理期间自动保存 SRL 的内容。

- 注 1:** 为了维持软件堆栈指针（W15）一致性，W15 永远不会受（EDS）分页影响，因此被限制在地址范围 0x0000 至 0xFFFF。当 W14 用作堆栈帧指针时（SFA = 1），这同样适用于 W14。
- 2:** 由于堆栈可置于 X 和 Y 空间中并能访问这两个空间，因此在使用堆栈、特别是涉及到 C 语言开发环境中的局部自动变量时必须小心。

图 4-9: CALL 堆栈帧



## 4.6 指令寻址模式

表 4-17 给出了基本的寻址模式，这些寻址模式经过优化可以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有所不同。

### 4.6.1 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单元中的前 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，W0 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，它将结果写入寄存器或寄存器对)。使用 MOV 指令能够获得更大的灵活性，并可以访问整个数据空间。

### 4.6.2 MCU 指令

三操作数 MCU 指令的形式是：

操作数 3 = 操作数 1 < 功能 > 操作数 2

其中，操作数 1 始终是工作寄存器 (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，取自数据存储单元，或为一个 5 位立即数。结果可以保存在 W 寄存器或数据存储单元中。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部分模式。

表 4-17: 支持的基本寻址模式

寻址模式	说明
文件寄存器直接寻址	明确指定文件寄存器的地址。
寄存器直接寻址	直接访问寄存器的内容。
寄存器间接寻址	Wn 的内容形成有效地址 (EA)。
执行后修改的寄存器间接寻址	Wn 的内容形成 EA。然后用一个常量值来修改 Wn (递增或递减)。
执行前修改的寄存器间接寻址	先用一个有符号常量值修改 Wn (递增或递减)，再由此时的 Wn 内容形成 EA。
带寄存器偏移量的寄存器间接寻址 (寄存器变址寻址)	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。

# dsPIC33EPXXGS70X/80X 系列

## 4.6.3 传送指令和累加器指令

与其他指令相比，传送指令和 DSP 累加器类指令提供了更为灵活的寻址模式。除了大多数 MCU 指令支持的寻址模式以外，传送指令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，传送指令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部分模式。

## 4.6.4 MAC 指令

双源操作数 DSP 指令（CLR、ED、EDAC、MAC、MPY、MPY.N、MOVSAC 和 MSC），也称为 MAC 指令，它们使用一组简化的寻址模式，允许用户应用程序通过寄存器间接寻址表有效地对数据指针进行操作。

双源操作数预取寄存器必须是集合 {W8, W9, W10, W11} 的成员。对于数据读取操作，W8 和 W9 始终用于 X RAGU，而 W10 和 W11 始终用于 Y AGU。从而，产生的有效地址（无论是在修改之前还是之后），对于 W8 和 W9 必须是 X 数据空间中的有效地址，对于 W10 和 W11 则必须是 Y 数据空间中的有效地址。

**注：** 带寄存器偏移量的寄存器间接寻址模式仅可用于 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）

## 4.6.5 其他指令

除了上述的各种寻址模式之外，一些指令使用各种长度的立即数常量。例如，BRA（跳转）指令使用 16 位有符号立即数来直接指定跳转的目标，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 ULNK，操作数或结果的存储位置已经暗含在操作码中。某些操作，例如 NOP，没有任何操作数。

## 4.7 模寻址

模寻址模式是一种使用硬件来自动支持循环数据缓冲区的方法。目的是在执行紧密循环代码时（这在许多 DSP 算法中很典型），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）。X 数据空间（也提供指向程序空间的指针）和 Y 数据空间中都可支持一个循环缓冲区。模寻址可以对任何 W 寄存器指针进行操作。然而，最好不要将 W14 或 W15 用于模寻址，因为这两个寄存器分别用作堆栈帧指针和堆栈指针。

一般来说，任何特定的循环缓冲区只能配置为单向操作，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）有一定限制。

使用限制的唯一例外是那些长度为 2 的幂的缓冲区。由于这些缓冲区满足起始和结束地址判据，它们可以双向操作（即，在低地址边界和高地址边界上都将进行地址边界检查）。

### 4.7.1 起始地址和结束地址

模寻址机制要求指定起始地址和结束地址，并将它们装入 16 位模缓冲区地址寄存器：XMODSRT、XMODEND、YMODSRT 和 YMODEND（见表 4-2）。

**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）。

循环缓冲区的长度不直接指定。它由相应的起始地址和结束地址之差决定。循环缓冲区的最大可能长度为 32K 字（64 KB）。

### 4.7.2 W 地址寄存器选择

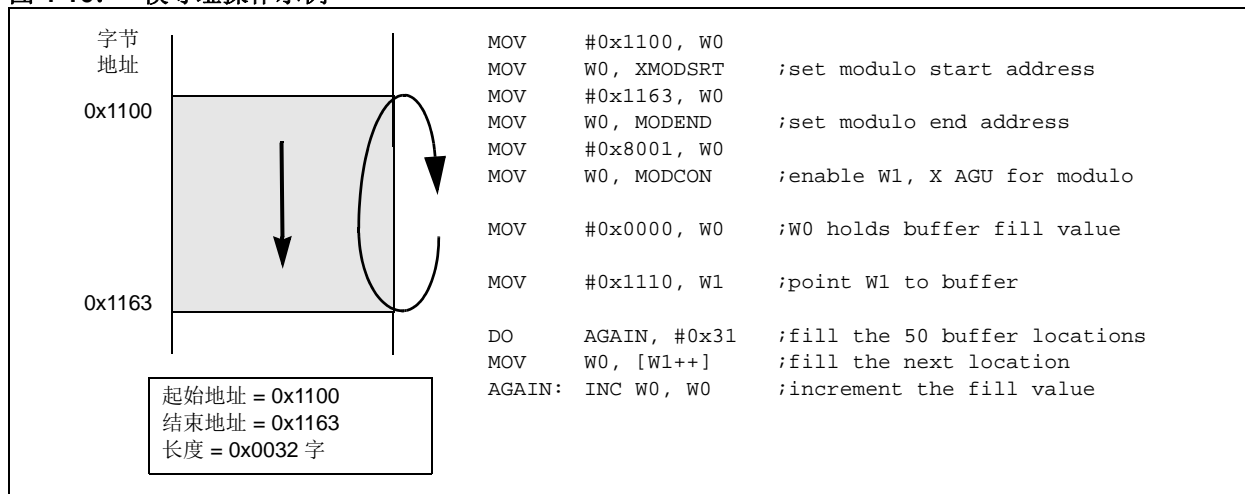
模寻址和位反转寻址控制寄存器 MODCON<15:0> 包含使能标志以及指定 W 地址寄存器的 W 寄存器字段。XWM 和 YWM 字段选择对哪些寄存器进行模寻址：

- 如果 XWM = 1111，则禁止 X RAGU 和 X WAGU 模寻址
- 如果 YWM = 1111，则禁止 Y AGU 模寻址

要进行模寻址的 X 地址空间指针 W（XWM）寄存器存储在 MODCON<3:0> 中（见表 4-2）。当 XWM 设置为除 1111 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。

要进行模寻址的 Y 地址空间指针 W（YWM）寄存器存储在 MODCON<7:4> 中。当 YWM 设置为除 1111 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 4-10: 模寻址操作示例



# dsPIC33EPXXGS70X/80X 系列

## 4.7.3 模寻址的应用

模寻址可以应用于与任何 W 寄存器相关的有效地址 (EA) 计算中。地址边界检查功能检查地址是否等于：

- 上边界地址 (对于递增缓冲区)
- 下边界地址 (对于递减缓冲区)

重要的是要意识到，地址边界检查功能不仅会检查地址是否正好在地址边界上，而且会检查地址是否大于上边界地址 (对于递增缓冲区)、或小于下边界地址 (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。

## 4.8 位反转寻址

位反转寻址模式用来简化基 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量或寄存器的内容，可视为将其位顺序反转。源地址和目标地址仍然是正常的顺序。于是，唯一需要反转的操作数就是地址修改量。

## 4.8.1 位反转寻址的实现

只有满足下列所有条件，才能使能位反转寻址模式：

- MODCON 寄存器中 BWMx 位 (W 寄存器选择) 的值是除 1111 以外的任何值 (不能使用位反转寻址访问堆栈)
- XBREV 寄存器中的 BREN 位置 1
- 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“轴心点”(pivot point)，通常是一个常量。对于 FFT 计算，其值等于 FFT 数据缓冲区长度的一半。

**注：** 所有位反转 EA 的计算都使用字长度数据 (每个 EA 的 LSb 始终清零)。为了产生兼容 (字节) 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、且仅对字长度数据写，才会进行位反转寻址。对于任何其他寻址模式或对于字节长度数据，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 地址指针将始终加上地址修改量 (XB)，与寄存器间接寻址模式相关的偏移量将被忽略。此外，由于要求是字长度数据，EA 的 LSb 被忽略 (且始终清零)。

**注：** 可以使用同一个 W 寄存器同时使能模寻址和位反转寻址，但当同时使能时，对于数据写操作位反转寻址总是优先。

如果已通过将 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。



# dsPIC33EPXXGS70X/80X 系列

图 4-11: 位反转寻址示例

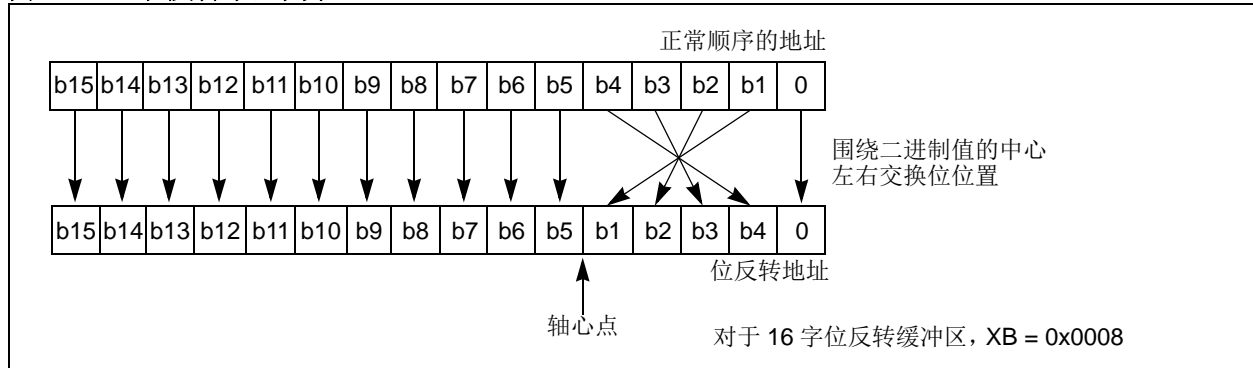


表 4-18: 位反转寻址序列 (16 项)

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

# dsPIC33EPXXGS70X/80X 系列

## 4.9 程序存储空间与数据存储空间的接口

dsPIC33EPXXGS70X/80X 系列架构采用 24 位宽的程序空间（PS）和 16 位宽的数据空间（DS）。该架构也是一种改进型哈佛结构，这意味着数据也能存放在程序空间内。要成功使用该数据，在访问数据时必须确保这两种存储空间中的信息是对齐的。

除了正常执行外，dsPIC33EPXXGS70X/80X 系列器件的架构还提供了两种可在操作过程中访问程序空间的方法：

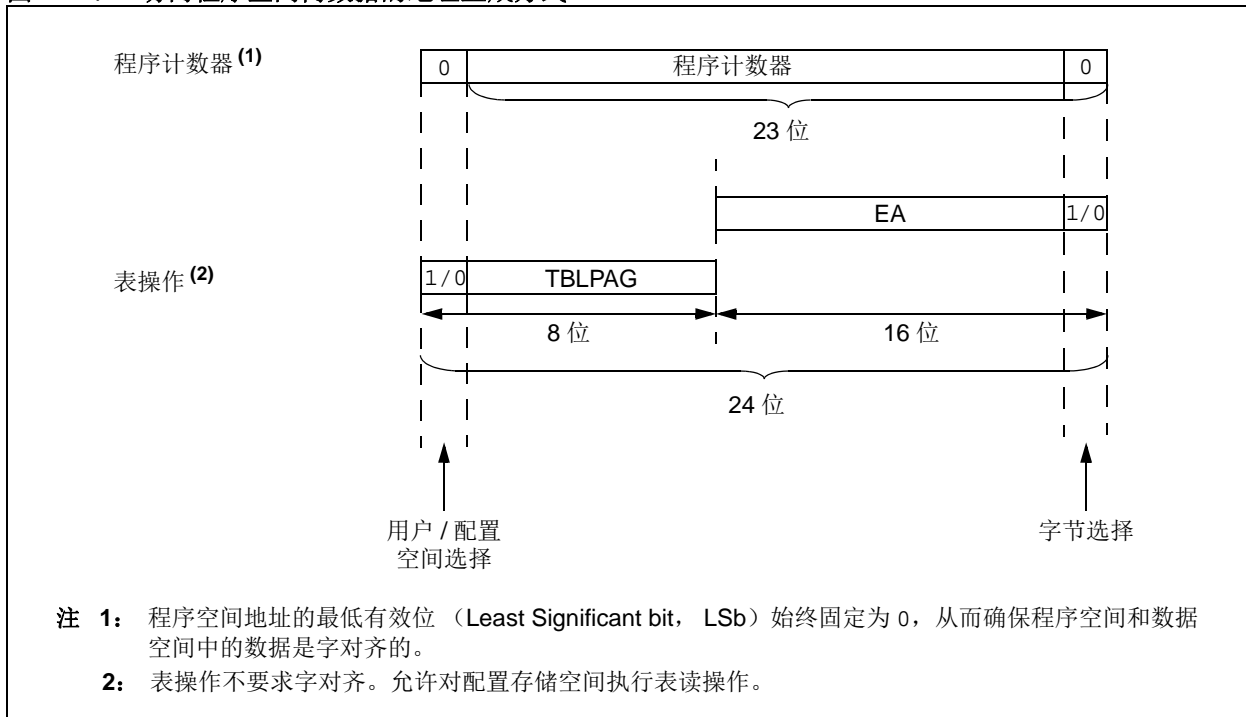
- 使用表指令访问程序空间中任意位置的各个字节或字
- 将程序空间的一部分重映射到数据空间（程序空间可视性）

表指令允许应用程序读写程序存储器的一小块区域。这一功能对于访问需要定期更新的数据表来说非常理想。也可通过表操作访问一个程序字的所有字节。重映射方法允许应用程序访问一大块数据，但只限于读操作，它非常适合于在一个大的静态数据表中查找。通过这一方法应用程序只能访问程序字的低位字。

表 4-19: 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 (代码执行)	用户	0	PC<22:1>			0
		0xxx xxxx xxxxx xxxxx xxxxx xxx0				
TBLRD/TBLWT (读 / 写字节或字)	用户	TBLPAG<7:0>		数据 EA<15:0>		
		0xxx xxxxx		xxxx xxxxx xxxxx xxxxx		
	配置	TBLPAG<7:0>		数据 EA<15:0>		
		1xxx xxxxx		xxxx xxxxx xxxxx xxxxx		

图 4-12: 访问程序空间内数据的地址生成方式



## 4.9.1 使用表指令访问程序存储器中的数据

TBLRDL 和 TBLWTL 指令提供了直接读或写程序空间内任何地址的低位字的方法，无需通过数据空间。TBLRDH 和 TBLWTH 指令是可以将一个程序空间字的最高 8 位作为数据读写的唯一方法。

对于每个连续的 24 位程序字，PC 的递增量为 2。这使得程序存储器地址能够直接映射到数据空间地址。于是，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 访问存有最低有效数据字的的空间，而 TBLRDH 和 TBLWTH 则访问存有最高数据字节的的空间。

提供了两条表指令来对程序空间执行字节或字（16 位）长度的数据读写。读和写都可以采用字节或字操作的形式。

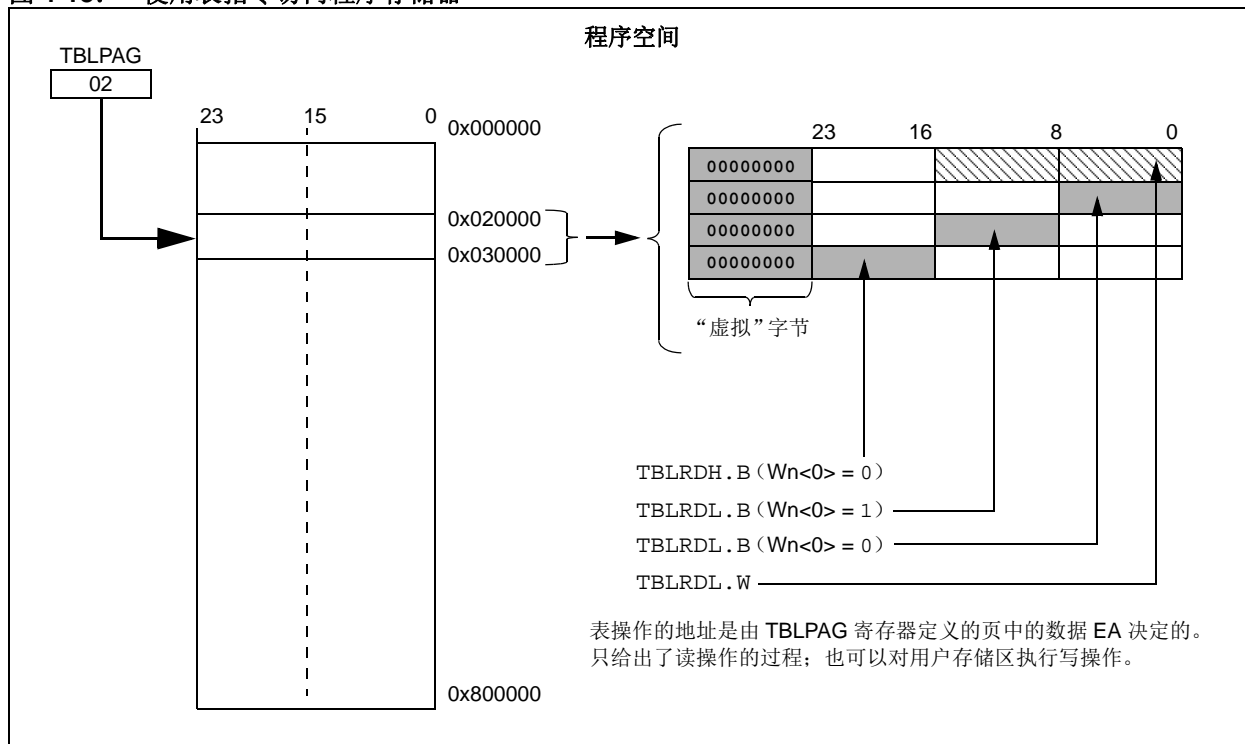
- TBLRDL（表读低位字）：
  - 在字模式下，该指令将程序空间地址中的低位字（P<15:0>）映射到数据地址（D<15:0>）中
  - 在字节模式下，低位程序字的高字节或低字节被映射到数据地址的低字节中。当字节选择位为 1 时映射高字节；当字节选择位为 0 时映射低字节。

- TBLRDH（表读高位字）：
  - 在字模式下，该指令将程序地址中的整个高位字（P<23:16>）映射到数据地址中。“虚拟”字节（D<15:8>）始终为 0。
  - 在字节模式下，该指令将程序字的高字节或低字节映射到数据地址的 D<7:0> 中，就如同 TBLRDL 指令。当选择最高“虚拟”字节（字节选择位 = 1）时，数据将始终为全 0。

表指令 TBLWTH 和 TBLWTL 以类似的方式向程序空间地址写入各字节或字。第 5.0 节“闪存程序存储器”对这两条指令的详细操作给出了说明。

对于所有的表操作，要访问程序存储空间的哪个区域是由表页寄存器（TBLPAG）决定的。TBLPAG 可寻址器件的整个程序存储空间，包括用户存储空间和配置空间。当 TBLPAG<7> = 0 时，表页位于用户存储空间中。当 TBLPAG<7> = 1 时，表页位于配置存储空间中。

图 4-13： 使用表指令访问程序存储器



# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 5.0 闪存程序存储器

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“双分区闪存程序存储器” (DS70005156)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

dsPIC33EPXXGS70X/80X 系列器件包含用于存储和执行应用代码的内部闪存程序存储器。在整个 VDD 范围内，正常操作期间，该存储器都是可读写、可擦除的。

可采用三种方式对闪存进行编程：

- 在线串行编程 (ICSP) 功能
- 增强型在线串行编程 (增强型 ICSP)
- 运行时自编程 (Run-Time Self-Programming, RTSP)

ICSP 允许在最终的应用电路中对

dsPIC33EPXXGS70X/80X 系列器件进行串行编程。只需要使用 5 根线就可以完成编程，它们分别是编程时钟线 (PGECx)、编程数据线 (PGEDx)、电源线 (VDD)、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，而仅在产品交付之前才对器件进行编程，从而可以使用最新版本的固件或者定制固件进行编程。

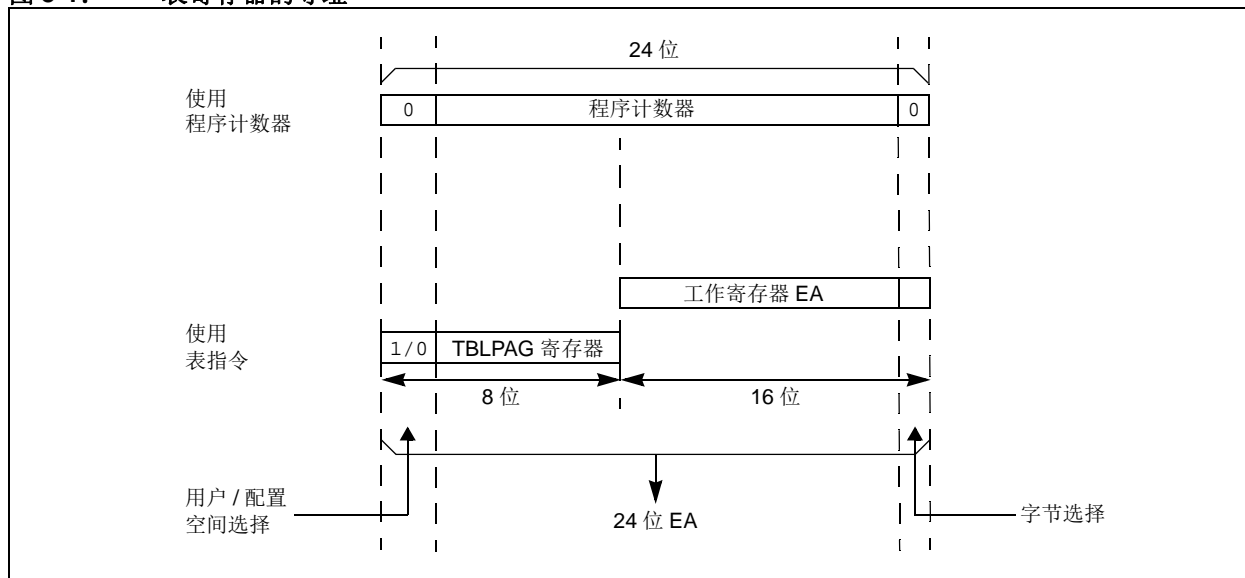
增强型在线串行编程使用称为编程执行程序的片内自举程序来管理编程过程。编程执行程序可以使用 SPI 数据帧格式擦除、编程和校验程序存储器。有关增强型 ICSP 的更多信息，请参见器件编程规范。

使用 TBLRD (表读) 和 TBLWT (表写) 指令来实现 RTSP。通过 RTSP，用户应用程序可以按一次写入单个程序存储字的方式写入程序存储数据，以及按块或“页” (512 条指令，1536 字节) 的方式擦除程序存储器。

### 5.1 表指令和闪存编程

闪存的所有编程都是通过表读和表写指令完成的，与使用的编程方法无关。这两条指令允许器件在正常工作模式下从数据存储直接读写程序存储空间。程序存储器中 24 位目标地址由 TBLPAG 寄存器的 bit<7:0> 和表指令中指定 W 寄存器中的有效地址 (EA) 组成，如图 5-1 所示。TBLRDL 和 TBLWTL 指令用于读或写程序存储器的 bit<15:0>。TBLRDL 和 TBLWTL 能以字或字节模式访问程序存储器。TBLRDH 和 TBLWTH 指令用于读或写程序存储器的 bit<23:16>。TBLRDH 和 TBLWTH 同样能以字或字节模式访问程序存储器。

图 5-1: 表寄存器的寻址



# dsPIC33EPXXGS70X/80X 系列

## 5.2 RTSP 工作原理

dsPIC33EPXXGS70X/80X 系列闪存程序存储器阵列是由 64 条指令或 192 字节的行组成的。RTSP 允许用户应用程序一次擦除一个存储器页（8 行或 512 条指令），以及一次编程一行。也可以一次编程两条指令。

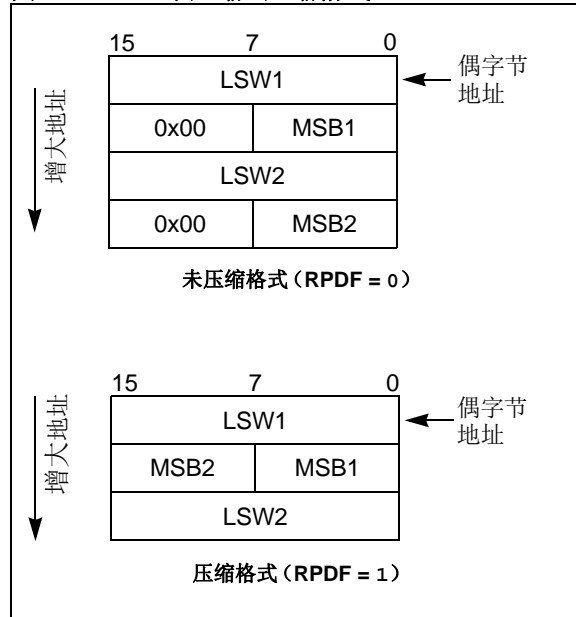
页擦除和单行写入块都是边沿对齐的，从程序存储器起始地址开始，分别到 1536 字节边界和 192 字节边界。第 30.0 节“电气特性”中的图 30-14 列出了典型的擦除和编程时间。

通过将 192 个字节装入数据存储器，然后将该行第一个字节的地址装入 NVMSRCADR 寄存器来执行编程。写操作启动后，器件将自动装入写锁寄存器并递增 NVMSRCADR 和 NVMADR(U) 寄存器，直到编程完所有字节。RPDF 位（NVMCON<9>）用于选择 RAM 中已存储数据的格式为压缩还是未压缩。关于数据格式，请参见图 5-2。压缩数据会将第二个字的高字节用于第二条指令的 MSB，这样有助于减少所需 RAM 的量。

RTSP 字编程的基本过程是使用 TBLWTL 和 TBLWTH 指令将两条 24 位指令装入位于配置存储器空间的写锁寄存器。关于写锁寄存器的地址，请参见图 4-1 至图 4-4。通过解锁和设置 NVMCON 寄存器中的控制位来执行编程。

所有擦除和编程操作均可选择使用 NVM 中断来发出已成功完成操作的信号。例如，当在双分区模式下对非活动分区执行闪存写操作时（此时 CPU 保持运行），需要等待至接收到 NVM 中断信号之后才能编程闪存程序存储器的下一个块。

图 5-2: 未压缩 / 压缩格式



## 5.3 编程操作

在 RTSP 模式下，对内部闪存进行编程或擦除需要执行完整的编程序列。处理器暂停（等待）直到编程操作完成。将 WR 位（NVMCON<15>）置 1 启动操作，当操作完成时 WR 位会自动清零。

### 5.3.1 闪存程序存储器的编程算法

编程器每次可以对闪存程序存储器的两个相邻字（24 位 x 2）进行编程，即每隔一个字地址边界（0x000000、0x000004 和 0x000008 等）。为此，需要先擦除包含用户希望更改存储单元的地址的页。为防止意外操作，必须向 NVMKEY 写入写启动序列，以允许执行任何擦除或编程操作。在执行编程命令后，用户应用程序必须等待一段编程时间，直到编程操作完成。编程序列启动后紧跟的两条指令应该为 NOP 指令。

## 5.4 双分区闪存配置

对于在双分区闪存程序存储器模式下工作的 dsPIC33EPXXGS70X/80X 器件，无需暂停处理器即可对非活动分区进行擦除和编程。可以使用第 5.2 节“RTSP 工作原理”所述的编程算法对非活动分区闪存进行编程和擦除。除页擦除选项外，还可通过配置 NVMCON 寄存器中的 NVMOP<3:0> 位来擦除整个非活动分区闪存。

**注 1:** 待装入非活动分区的应用软件将采用活动分区的地址。自举程序固件需要使地址偏移 0x400000，才能对非活动分区执行写操作。

### 5.4.1 闪存分区交换

引导序列号用于在启动时确定活动分区，该序列号在 FBTSEQ 配置寄存器中进行编码。与大多数仅利用程序存储器低 16 位的配置寄存器不同，FBTSEQ 是 24 位配置字。引导序列号 (BSEQ) 是 12 位值，在 FBTSEQ 中存储两次。位 FBTSEQ<11:0> 中存储其真值，而位 FBTSEQ<23:12> 中存储其反码值。器件复位时，将读取序列号，其中序列号较小的分区将成为活动分区。如果其中一个引导序列号无效，则器件将选择引导序列号有效的分区，如果两个序列号均无效，则会默认选择分区 1。更多信息，请参见第 27.0 节“特殊功能”。

BOOTSWP 指令提供了一种备选的分​​区交换方法（软交换），无需器件复位即可交换活动分区与非活动分区。BOOTSWP 指令必须后跟一条 GOTO 指令。BOOTSWP 指令用于交换活动分区与非活动分区，PC 指向当前活动分区中的 GOTO 指令所指定的存储单元。

请务必注意：执行软交换序列时应暂时禁止中断；分区交换后，所有已使能的外设将保持使能状态，所有已允许的中断将保持允许状态。此外，RAM 和堆栈在分区交换后也将保持先前的状态。因此，建议让使用软交换的应用程序跳转到将重新初始化器件的程序中，以确保固件按预期运行。配置寄存器在软交换期间不起作用。

为确保操作的可靠性，执行 BOOTSWP 指令时，必须按如下步骤执行 NVM 解锁序列：

1. 向 NVMKEY 写入 0x55。
2. 向 NVMKEY 写入 0xAA。
3. 执行 BOOTSWP 指令。

如果未执行解锁序列，BOOTSWP 指令将作为强制 NOP 执行，接着将执行 BOOTSWP 指令后的 GOTO 指令，以便使 PC 跳转到当前活动分区中的相应存储单元。

NVMCON 寄存器中的 SFTSWP 和 P2ACTIV 位用于确定活动分区和非活动分区是否已成功交换以及当前的活动分区。在 BOOTSWP 和 GOTO 指令后，应轮询 SFTSWP 位以验证是否已发生分区交换，之后应将该位清零，以便执行下一次分区交换事件。

### 5.4.2 双分区模式

在双分区模式下工作时，dsPIC33EPXXGS70X/80X 系列器件可以选择使两个分区具有自定义的安全段，如图 27-4 所示。器件也可在受保护双分区模式下工作，其中分区 1 将受到永久擦除 / 写保护。受保护双分区模式可实现“出厂默认”模式，该模式在分区 1 中存储了故障保护备份映像。

dsPIC33EPXXGS70X/80X 系列器件也可在特权双分区模式下工作，该模式下可实施附加安全保护，从而为使用多方软件的器件提供知识产权保护。在特权双分区模式下，两个分区均将对 FBSLIM 寄存器施加额外的限制。这样可防止更改自举段和通用段的大小，从而确保两个段均不会被更改。

## 5.5 闪存资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 5.5.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“双分区闪存程序存储器” (DS70005156)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

---

## 5.6 控制寄存器

有五个 SFR 用于写入和擦除闪存程序存储器：

NVMCON、NVMKEY、NVMADR、NVMADRU 和 NVMSRCADR/H。

NVMCON 寄存器（寄存器 5-1）用于选择要执行的操作（页擦除、字 / 行编程和非活动分区擦除）、启动编程或擦除周期以及确定双分区模式下的活动分区。

NVMKEY（寄存器 5-4）是一个只写寄存器，用于写保护。要启动编程或擦除序列，用户应用程序必须将 0x55 和 0xAA 连续写入 NVMKEY 寄存器。

有两个 NVM 地址寄存器：NVMADRU 和 NVMADR。这两个寄存器组合在一起时，构成要进行编程操作的选定字 / 行的 24 位有效地址（EA），或者要进行擦除操作的选定页的 24 位有效地址（EA）。NVMADRU 寄存器用于保存 EA 的高 8 位，而 NVMADR 寄存器用于保存 EA 的低 16 位。

对于行编程操作，要写入闪存程序存储器的数据将写入位于由 NVMSRCADR 寄存器定义的地址（行编程数据中第一个元素的位置）处的数据存储空间（RAM）。



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 5-1: NVMCON: 非易失性存储器 (NVM) 控制寄存器

R/SO-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0	R/C-0	R-0	R/W-0	R/C-0
WR	WREN	WRERR	NVMSIDL <sup>(2)</sup>	SFTSWP <sup>(6)</sup>	P2ACTIV <sup>(6)</sup>	RPDF	URERR
bit 15						bit 8	
U-0	U-0	U-0	U-0	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>
—	—	—	—	NVMOP3 <sup>(3,4)</sup>	NVMOP2 <sup>(3,4)</sup>	NVMOP1 <sup>(3,4)</sup>	NVMOP0 <sup>(3,4)</sup>
bit 7						bit 0	

<b>图注:</b>	C = 可清零位	SO = 只可置 1 位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15      **WR:** 写控制位 <sup>(1)</sup>  
 1 = 启动闪存程序存储器编程或擦除操作; 该操作是自定时的, 一旦操作完成, 该位即由硬件清零  
 0 = 编程或擦除操作完成, 并处于停止状态
- bit 14      **WREN:** 写使能位 <sup>(1)</sup>  
 1 = 使能闪存编程 / 擦除操作  
 0 = 禁止闪存编程 / 擦除操作
- bit 13      **WRERR:** 写序列错误标志位 <sup>(1)</sup>  
 1 = 试图执行不合法的编程或擦除序列, 或者发生终止 (试图将 WR 位置 1 时自动置 1 该位)  
 0 = 编程或擦除操作正常完成
- bit 12      **NVMSIDL:** NVM 空闲模式停止控制位 <sup>(2)</sup>  
 1 = 在空闲模式下闪存稳压器进入待机模式  
 0 = 在空闲模式下闪存稳压器继续工作
- bit 11      **SFTSWP:** 分区软交换状态位 <sup>(6)</sup>  
 1 = 已通过 BOOTSWP 指令 (软交换) 成功完成分区交换  
 0 = 使用 BOOTSWP 指令后等待分区成功交换, 或器件复位后根据 FBTSEQ 寄存器确定活动分区
- bit 10      **P2ACTIV:** 分区 2 活动状态位 <sup>(6)</sup>  
 1 = 分区 2 闪存映射到活动区域  
 0 = 分区 1 闪存映射到活动区域
- bit 9        **RPDF:** 行编程数据格式位  
 1 = 行数据以压缩格式存储在 RAM 中  
 0 = 行数据以未压缩格式存储在 RAM 中
- bit 8        **URERR:** 行编程数据不足错误位  
 1 = 指示行编程操作已终止  
 0 = 未检测到数据不足错误
- bit 7-4      **未实现:** 读为 0

- 注 1:** 这些位只能在 POR 时复位。  
**2:** 如果该位置 1, 会进一步降低功耗 (IDLE), 并且当退出空闲模式后、闪存开始工作之前会有一段延时 (TVREG)。  
**3:** NVMOP<3:0> 的所有其他组合均未实现。  
**4:** 在进行任意 NVM 操作时, PWRSAV 指令的执行会被忽略。  
**5:** 执行该操作期间, 会编程 4 字边界上的两个相邻字。  
**6:** 仅适用于在双分区模式下工作时。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 5-1: NVMCON: 非易失性存储器 (NVM) 控制寄存器 (续)

bit 3-0      **NVMOP<3:0>**: NVM 操作选择位 (1,3,4)

- 1111 = 保留
- 
- 
- 
- 0101 = 保留
- 0100 = 非活动分区存储器擦除操作
- 0011 = 存储器页擦除操作
- 0010 = 存储器行编程操作
- 0001 = 存储器双字编程操作 (5)
- 0000 = 保留

- 注 1:** 这些位只能在 POR 时复位。
- 2:** 如果该位置 1, 会进一步降低功耗 (IDLE), 并且当退出空闲模式后、闪存开始工作之前会有一段延时 (TVREG)。
- 3:** NVMOP<3:0> 的所有其他组合均未实现。
- 4:** 在进行任意 NVM 操作时, PWRSAV 指令的执行会被忽略。
- 5:** 执行该操作期间, 会编程 4 字边界上的两个相邻字。
- 6:** 仅适用于在双分区模式下工作时。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 5-2: NVMADR: 非易失性存储器低位字地址寄存器

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
NVMADR<15:8>							
bit 15				bit 8			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
NVMADR<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-0      **NVMADR<15:0>**: 非易失性存储器低位字写地址位  
 选择闪存程序存储器中要进行编程或擦除的存储单元的低 16 位。用户应用程序可以读写该寄存器。

## 寄存器 5-3: NVMADRU: 非易失性存储器高位字地址寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15				bit 8			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
NVMADRU<23:16>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8      **未实现**: 读为 0  
 bit 7-0      **NVMADRU<23:16>**: 非易失性存储器高字节写地址位  
 选择闪存程序存储器中要进行编程或擦除的存储单元的高 8 位。用户应用程序可以读写该寄存器。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 5-4: NVMKEY: 非易失性存储器密钥寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

W-0	W-0	W-0	W-0	W-0	W-0	W-0	W-0
NVMKEY<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-8            未实现: 读为 0  
 bit 7-0            **NVMKEY<7:0>**: NVM 密钥寄存器位 (只写)

## 寄存器 5-5: NVMSRCADR: NVM 源数据地址寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADR<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMSRCADR<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-0            **NVMSRCADR<15:0>**: NVM 源数据地址位  
 待编程到闪存中的数据 RAM 地址 (将 NVMOP<3:0> 位设置为行编程时)。

## 6.0 复位

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“复位” (DS70602)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

复位模块结合了所有复位源并控制器件的主复位信号  $\overline{\text{SYSRST}}$ 。下面列出了器件的复位源：

- POR: 上电复位
- BOR: 欠压复位
- $\overline{\text{MCLR}}$ : 主复位引脚复位
- SWR: RESET 指令
- WDTO: 看门狗定时器超时复位
- CM: 配置不匹配复位
- TRAPR: 陷阱冲突复位
- IOPUWR: 非法条件器件复位
  - 非法操作码复位
  - 未初始化的 W 寄存器复位
  - 安全性复位

图 6-1 给出了复位模块的简化框图。

任何有效的复位源都将使  $\overline{\text{SYSRST}}$  信号有效。在系统复位时，一些与 CPU 和外设相关的寄存器被强制为已知的复位状态，而有一些寄存器不受影响。

**注:** 如需了解寄存器复位状态的信息，请参见本数据手册中的特定外设章节或第 4.0 节“存储器构成”。

任何类型的器件复位都会将 RCON 寄存器中相应的状态位置 1，以指示复位类型（见寄存器 6-1）。

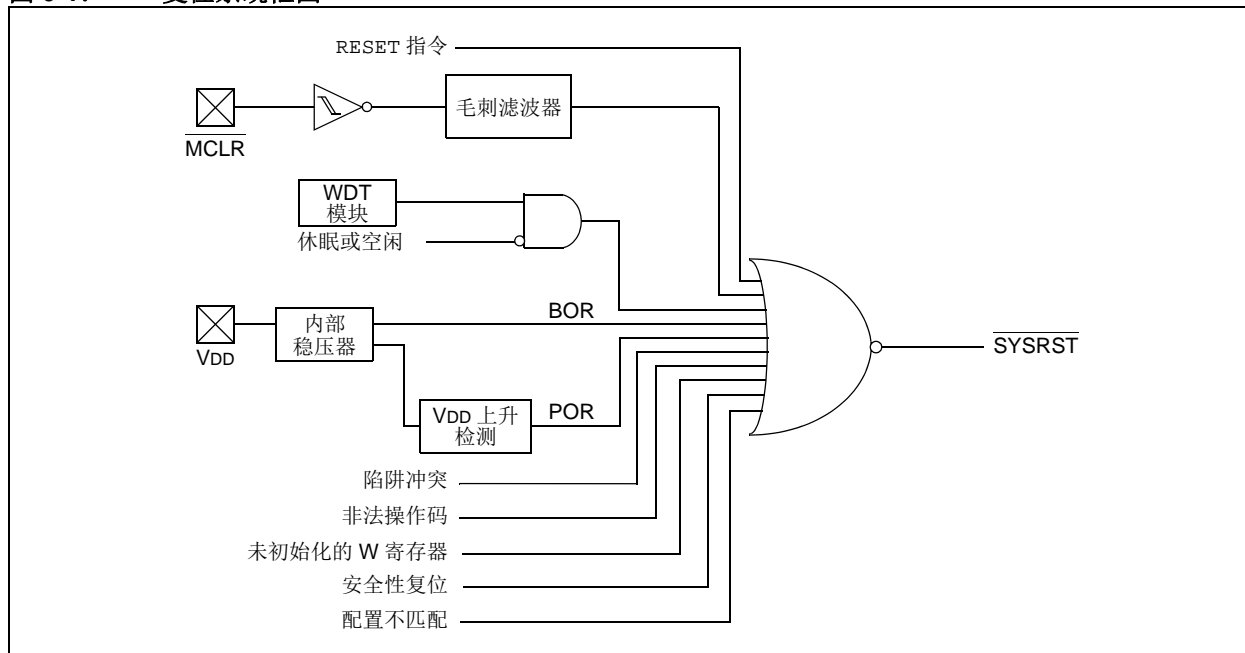
POR 将清零除 BOR 和 POR 位 (RCON<1:0>) 之外的所有位，BOR 和 POR 位在 POR 时被置 1。用户应用程序可在代码执行过程中的任何时间置 1 或清零任何位。RCON 寄存器中的位仅用作状态位。用软件将某个复位状态位置 1 不会导致器件发生复位。

RCON 寄存器还包含与看门狗定时器和器件节能状态相关的其他位。本手册的其他章节中将讨论这些位的功能。

**注:** RCON 寄存器中的状态位应该在被读取后清零，这样在器件复位后 RCON 寄存器的下一个值才有意义。

在所有类型的复位中，默认时钟源由 FOSCSEL 配置寄存器中的 FNOSC<2:0> 位决定。复位时，FNOSC<sub>x</sub> 位的值装入 NOSC<2:0> (OSCCON<10:8>) 位，从而初始化系统时钟。

图 6-1: 复位系统框图



# dsPIC33EPXXGS70X/80X 系列

---

## 6.1 复位资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 6.1.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“复位” (DS70602)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 6-1: RCON: 复位控制寄存器<sup>(1)</sup>

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	R/W-0	R/W-0
TRAPR	IOPUWR	—	—	VREGSF	—	CM	VREGS
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
EXTR	SWR	SWDTEN <sup>(2)</sup>	WDTO	SLEEP	IDLE	BOR	POR
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15      **TRAPR:** 陷阱复位标志位  
1 = 发生了陷阱冲突复位  
0 = 未发生陷阱冲突复位
- bit 14      **IOPUWR:** 非法操作码或访问未初始化的 W 寄存器复位标志位  
1 = 检测到非法操作码、非法地址模式或将未初始化的 W 寄存器用作地址指针而导致复位  
0 = 未发生非法操作码或未初始化的 W 寄存器复位
- bit 13-12    **未实现:** 读为 0
- bit 11      **VREGSF:** 休眠模式下闪存稳压器待机位  
1 = 在休眠模式下闪存稳压器继续工作  
0 = 在休眠模式下闪存稳压器进入待机模式
- bit 10      **未实现:** 读为 0
- bit 9        **CM:** 配置不匹配标志位  
1 = 发生了配置不匹配复位  
0 = 未发生配置不匹配复位
- bit 8        **VREGS:** 休眠模式下稳压器待机位  
1 = 在休眠模式下稳压器继续工作  
0 = 在休眠模式下稳压器进入待机模式
- bit 7        **EXTR:** 外部复位 ( $\overline{\text{MCLR}}$ ) 引脚位  
1 = 发生了主复位 (引脚) 复位  
0 = 未发生主复位 (引脚) 复位
- bit 6        **SWR:** 软件 RESET (指令) 标志位  
1 = 执行了 RESET 指令  
0 = 未执行 RESET 指令
- bit 5        **SWDTEN:** 软件使能 / 禁止 WDT 位<sup>(2)</sup>  
1 = 使能 WDT  
0 = 禁止 WDT
- bit 4        **WDTO:** 看门狗定时器超时标志位  
1 = 发生了 WDT 超时  
0 = 未发生 WDT 超时

**注 1:** 所有复位状态位都可以用软件置 1 或清零。用软件将这些位中的某一位置 1 不会导致器件复位。

**注 2:** 如果 WDTEN<1:0> 配置位为 11 (未编程), 则 WDT 总是使能, 而与 SWDTEN 位的设置无关。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 6-1: RCON: 复位控制寄存器<sup>(1)</sup> (续)

bit 3	<b>SLEEP:</b> 从休眠模式唤醒标志位 1 = 器件处于休眠模式 0 = 器件不处于休眠模式
bit 2	<b>IDLE:</b> 从空闲模式唤醒标志位 1 = 器件处于空闲模式 0 = 器件不处于空闲模式
bit 1	<b>BOR:</b> 欠压复位标志位 1 = 发生了欠压复位 0 = 未发生欠压复位
bit 0	<b>POR:</b> 上电复位标志位 1 = 发生了上电复位 0 = 未发生上电复位

**注 1:** 所有复位状态位都可以用软件置 1 或清零。用软件将这些位中的某一位置 1 不会导致器件复位。

**2:** 如果 WDTEN<1:0> 配置位为 11 (未编程), 则 WDT 总是使能, 而与 SWDTEN 位的设置无关。



## 7.0 中断控制器

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“中断”（DS70000600），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

dsPIC33EPXXGS70X/80X 系列中断控制器将诸多外置中断请求信号缩减为一个送往 dsPIC33EPXXGS70X/80X 系列 CPU 的中断请求信号。

中断控制器具有以下特性：

- 6 个处理器异常和软件陷阱
- 7 个可由用户选择的优先级
- 中断向量表（Interrupt Vector Table, IVT），其中每个中断或异常源对应唯一的向量
- 在指定的用户优先级内具有固定的优先级
- 固定的中断进入和返回延时
- 用于支持调试功能的备用中断向量表（Alternate Interrupt Vector Table, AIVT）

### 7.1 中断向量表

dsPIC33EPXXGS70X/80X 系列的中断向量表（IVT）（如图 7-1 所示）位于程序存储器中，起始单元地址是 000004h。IVT 包含 6 个不可屏蔽陷阱向量和最多 246 个中断源。一般来说，每个中断源都有自己的中断向量。每个中断向量都包含一个 24 位宽的地址。每个中断向量单元中编程的值是其相关的中断服务程序（Interrupt Service Routine, ISR）的起始地址。

中断向量根据其自然优先级区分优先顺序。自然优先级与中断向量在向量表中的位置有关。一般来说，较低地址的中断向量具有较高的自然优先级。例如，与向量 0 相关的中断比任何其他向量地址的中断具有更高的自然优先级。

#### 7.1.1 备用中断向量表

只有定义了引导段并已使能 AIVT 时，备用中断向量表（AIVT）（如图 7-2 所示）才可用。要使能备用中断向量表，必须编程 FSEC 寄存器中的配置位 AIVTDIS 并将 AIVTEN 位置 1（INTCON2<8> = 1）。使能 AIVT 时，所有中断和异常处理都将使用备用向量，而非默认向量。AIVT 从引导段最后一页的起始处（由 BSLIM<12:0> 定义）开始。该页的另一半为不再可用的空间。引导段必须至少具有 2 页以使能 AIVT。

**注：** 尽管必须使能引导段才能使能 AIVT，但应用代码无需位于引导段内部。AIVT（和 IVT）将禁止引导段代码保护。

AIVT 通过提供一种不需要将中断向量再编程就可以在应用程序和支持环境之间切换的方法，来支持调试功能。此特性也支持运行时在不同应用程序之间切换以便评估各种不同的软件算法。

### 7.2 复位过程

器件复位不是真正的异常，因为复位过程中并不涉及到中断控制器。作为对复位的响应，dsPIC33EPXXGS70X/80X 系列器件清零其寄存器，强制 PC 为零。然后器件从地址 0x000000 处开始执行程序。用户可以在复位地址处编程一条 GOTO 指令，将程序执行重定向到相应的启动程序。

**注：** 应使用包含 RESET 指令的默认中断处理程序的地址编程 IVT 中所有未实现或未使用的向量单元。

# dsPIC33EPXXXGS70X/80X 系列

图 7-1: dsPIC33EPXXXGS70X/80X 系列中断向量表

自然顺序优先级降序排列  ↓  IVT  ↓	复位 —— GOTO 指令	0x000000	有关中断向量的详细信息，请参见表 7-1
	复位 —— GOTO 地址	0x000002	
	振荡器故障陷阱向量	0x000004	
	地址错误陷阱向量	0x000006	
	通用硬陷阱向量	0x000008	
	堆栈错误陷阱向量	0x00000A	
	数学错误陷阱向量	0x00000C	
	保留	0x00000E	
	通用软陷阱向量	0x000010	
	保留	0x000012	
	中断向量 0	0x000014	
	中断向量 1	0x000016	
	⋮	⋮	
	⋮	⋮	
	⋮	⋮	
	中断向量 52	0x00007C	
	中断向量 53	0x00007E	
	中断向量 54	0x000080	
	⋮	⋮	
	⋮	⋮	
	⋮	⋮	
	中断向量 116	0x0000FC	
	中断向量 117	0x0000FE	
	中断向量 118	0x000100	
	中断向量 119	0x000102	
	中断向量 120	0x000104	
	⋮	⋮	
	⋮	⋮	
⋮	⋮		
中断向量 244	0x0001FC		
中断向量 245	0x0001FE		
代码起始单元	0x000200		

注：在双分区闪存模式下，每个分区都有专用的中断向量表。

图 7-2: dsPIC33EPXXGS70X/80X 备用中断向量表 (2)

自然顺序优先级降序排列  AIVT	保留	$BSLIM<12:0>^{(1)} + 0x000000$
	保留	$BSLIM<12:0>^{(1)} + 0x000002$
	振荡器故障陷阱向量	$BSLIM<12:0>^{(1)} + 0x000004$
	地址错误陷阱向量	$BSLIM<12:0>^{(1)} + 0x000006$
	通用硬陷阱向量	$BSLIM<12:0>^{(1)} + 0x000008$
	堆栈错误陷阱向量	$BSLIM<12:0>^{(1)} + 0x00000A$
	数学错误陷阱向量	$BSLIM<12:0>^{(1)} + 0x00000C$
	保留	$BSLIM<12:0>^{(1)} + 0x00000E$
	通用软陷阱向量	$BSLIM<12:0>^{(1)} + 0x000010$
	保留	$BSLIM<12:0>^{(1)} + 0x000012$
	中断向量 0	$BSLIM<12:0>^{(1)} + 0x000014$
	中断向量 1	$BSLIM<12:0>^{(1)} + 0x000016$
	:	:
	:	:
	:	:
	中断向量 52	$BSLIM<12:0>^{(1)} + 0x00007C$
	中断向量 53	$BSLIM<12:0>^{(1)} + 0x00007E$
	中断向量 54	$BSLIM<12:0>^{(1)} + 0x000080$
	:	:
	:	:
	:	:
	中断向量 116	$BSLIM<12:0>^{(1)} + 0x0000FC$
	中断向量 117	$BSLIM<12:0>^{(1)} + 0x0000FE$
	中断向量 118	$BSLIM<12:0>^{(1)} + 0x000100$
	中断向量 119	$BSLIM<12:0>^{(1)} + 0x000102$
	中断向量 120	$BSLIM<12:0>^{(1)} + 0x000104$
	:	:
:	:	
:	:	
中断向量 244	$BSLIM<12:0>^{(1)} + 0x0001FC$	
中断向量 245	$BSLIM<12:0>^{(1)} + 0x0001FE$	

有关中断向量的详细信息，请参见表 7-1

注 1: 地址取决于 BSLIM<12:0> 定义的引导段的大小。  $[(BSLIM<12:0> - 1) \times 0x400] + \text{偏移量}$ 。

注 2: 在双分区闪存模式下，每个分区都有专用的备用中断向量表（如果使能）。

# dsPIC33EPXXGS70X/80X 系列

表 7-1: 中断向量详细信息

中断源	向量编号	IRQ 编号	IVT 地址	中断位的位置		
				标志	允许	优先级
最高自然顺序优先级						
INT0——外部中断 0	8	0	0x000014	IFS0<0> INT0IF	IEC0<0> INT0IE	IPC0<2:0> INT0IP<2:0>
IC1——输入捕捉 1	9	1	0x000016	IFS0<1> IC1IF	IEC0<1> IC1IE	IPC0<6:4> IC1IP<2:0>
OC1——输出比较 1	10	2	0x000018	IFS0<2> OC1IF	IEC0<2> OC1IE	IPC0<10:8> OC1IP<2:0>
T1——Timer1	11	3	0x00001A	IFS0<3> T1IF	IEC0<3> T1IE	IPC0<14:12> T1IP<2:0>
DMA0——DMA 通道 0	12	4	0x00001C	IFS0<4> DMA0IF	IEC0<4> DMA0IE	IPC1<2:0> DMA0IP<2:0>
IC2——输入捕捉 2	13	5	0x00001E	IFS0<5> IC2IF	IEC0<5> IC2IE	IPC1<6:4> IC2IP<2:0>
OC2——输出比较 2	14	6	0x000020	IFS0<6> OC2IF	IEC0<6> OC2IE	IPC1<10:8> OC2IP<2:0>
T2——Timer2	15	7	0x000022	IFS0<7> T2IF	IEC0<7> T2IE	IPC1<14:12> T2IP<2:0>
T3——Timer3	16	8	0x000024	IFS0<8> T3IF	IEC0<8> T3IE	IPC2<2:0> T3IP<2:0>
SPI1TX——SPI1 传输完成	17	9	0x000026	IFS0<9> SPI1TXIF	IEC0<9> SPI1TXIE	IPC2<6:4> SPI1TXIP<2:0>
SPI1RX——SPI1 接收完成	18	10	0x000028	IFS0<10> SPI1RXIF	IEC0<10> SPI1RXIE	IPC2<10:8> SPI1RXIP<2:0>
U1RX——UART1 接收器	19	11	0x00002A	IFS0<11> U1RXIF	IEC0<11> U1RXIE	IPC2<14:12> U1RXIP<2:0>
U1TX——UART1 发送器	20	12	0x00002C	IFS0<12> U1TXIF	IEC0<12> U1TXIE	IPC3<2:0> U1TXIP<2:0>
ADC——ADC 全局转换完成	21	13	0x00002E	IFS0<13> ADCIF	IEC0<13> ADCIE	IPC3<6:4> ADCIP<2:0>
DMA1——DMA 通道 1	22	14	0x000030	IFS0<14> DMA1IF	IEC0<14> DMA1IE	IPC3<10:8> DMA1IP<2:0>
NVM——NVM 写完成	23	15	0x000032	IFS0<15> NVMIF	IEC0<15> NVMIE	IPC3<14:12> NVMIP<2:0>
SI2C1——I2C1 从事件	24	16	0x000034	IFS1<0> SI2C1IF	IEC1<0> SI2C1IE	IPC4<2:0> SI2C1IP<2:0>
MI2C1——I2C1 主事件	25	17	0x000036	IFS1<1> MI2C1IF	IEC1<1> MI2C1IE	IPC4<6:4> MI2C1IP<2:0>
AC1——模拟比较器 1 中断	26	18	0x000038	IFS1<2> AC1IF	IEC1<2> AC1IE	IPC4<10:8> AC1IP<2:0>
CN——输入电平变化中断	27	19	0x00003A	IFS1<3> CNIF	IEC1<3> CNIE	IPC4<14:12> CNIP<2:0>
INT1——外部中断 1	28	20	0x00003C	IFS1<4> INT1IF	IEC1<4> INT1IE	IPC5<2:0> INT1IP<2:0>
保留	29-31	21-23	0x00003E-0x000043	—	—	—
DMA2——DMA 通道 2	32	24	0x000044	IFS1<8> DMA2IF	IEC1<8> DMA2IE	IPC6<2:0> DMA2IP<2:0>
OC3——输出比较 3	33	25	0x000046	IFS1<9> OC3IF	IEC1<9> OC3IE	IPC6<6:4> OC3IP<2:0>
OC4——输出比较 4	34	26	0x000048	IFS1<10> OC4IF	IEC1<10> OC4IE	IPC6<10:8> OC4IP<2:0>

# dsPIC33EPXXGS70X/80X 系列

**表 7-1: 中断向量详细信息 (续)**

中断源	向量编号	IRQ 编号	IVT 地址	中断位的位置		
				标志	允许	优先级
T4——Timer4	35	27	0x00004A	IFS1<11> T4IF	IEC1<11> T4IE	IPC6<14:12> T4IP<2:0>
T5——Timer5	36	28	0x00004C	IFS1<12> T5IF	IEC1<12> T5IE	IPC7<2:0> T5IP<2:0>
INT2——外部中断 2	37	29	0x00004E	IFS1<13> INT2IF	IEC1<13> INT2IE	IPC7<6:4> INT2IP<2:0>
U2RX——UART2 接收器	38	30	0x000050	IFS1<14> U2RXIF	IEC1<14> U2RXIE	IPC7<10:8> U2RXIP<2:0>
U2TX——UART2 发送器	39	31	0x000052	IFS1<15> U2TXIF	IEC1<15> U2TXIE	IPC7<14:12> U2TXIP<2:0>
SPI2TX——SPI2 传输完成	40	32	0x000054	IFS2<0> SPI2TXIF	IEC2<0> SPI2TXIE	IPC8<2:0> SPI2TXIP<2:0>
SPI2RX——SPI2 接收完成	41	33	0x000056	IFS2<1> SPI2RXIF	IEC2<1> SPI2RXIE	IPC8<6:4> SPI2RXIP<2:0>
C1RX——CAN1 接收数据就绪	42	34	0x000058	IFS2<2> C1RXIF	IEC2<2> C1RXIE	IPC8<10:8> C1RXIP<2:0>
C1——CAN1 组合错误	43	35	0x000059	IFS2<3> C1IF	IEC2<3> C1IE	IPC8<14:12> C1IP<2:0>
DMA3——DMA 通道 3	44	36	0x00005A	IFS2<4> DMA3IF	IEC2<4> DMA3IE	IPC9<2:0> DMA3IP<2:0>
IC3——输入捕捉 3	45	37	0x00005E	IFS2<5> IC3IF	IEC2<5> IC3IE	IPC9<6:4> IC3IP<2:0>
IC4——输入捕捉 4	46	38	0x000060	IFS2<6> IC4IF	IEC2<6> IC4IE	IPC9<10:8> IC4IP<2:0>
保留	47-56	39-48	0x000062-0x000074	—	—	—
SI2C2——I2C2 从事件	57	49	0x000076	IFS3<1> SI2C2IF	IEC3<1> SI2C2IE	IPC12<6:4> SI2C2IP<2:0>
MI2C2——I2C2 主事件	58	50	0x000078	IFS3<2> MI2C2IF	IEC3<2> MI2C2IE	IPC12<10:8> MI2C2IP<2:0>
保留	59-61	51-53	0x00007A-0x00007E	—	—	—
INT4——外部中断 4	62	54	0x000080	IFS3<6> INT4IF	IEC3<6> INT4IE	IPC13<10:8> INT4IP<2:0>
C2RX——CAN2 接收数据就绪	63	55	0x000082	IFS3<7> C2RXIF	IEC3<7> C2RXIE	IPC13<14:12> C2RXIP<2:0>
C2——CAN2 组合错误	64	56	0x000083	IFS3<8> C2IF	IEC3<8> C2IE	IPC14<2:0> C2IP<2:0>
PSEM——PWM 特殊事件匹配	65	57	0x000086	IFS3<9> PSEMIF	IEC3<9> PSEMIE	IPC14<6:4> PSEMIP<2:0>
保留	66-72	58-64	0x000088-0x000094	—	—	—
U1E——UART1 错误中断	73	65	0x000096	IFS4<1> U1EIF	IEC4<1> U1EIE	IPC16<6:4> U1EIP<2:0>
U2E——UART2 错误中断	74	66	0x000098	IFS4<2> U2EIF	IEC4<2> U2EIE	IPC16<10:8> U2EIP<2:0>
保留	75-77	67-69	0x00009A-0x0000A2	—	—	—
C1TX——CAN1 发送数据请求	78	70	0x0000A0	IFS4<6> C1TXIF	IEC4<6> C1TXIE	IPC17<10:8> C1TXIP<2:0>
C2TX——CAN2 发送数据请求	79	71	0x0000A	IFS4<7> C2TXIF	IEC4<7> C2TXIE	IPC17<14:12> C2TXIP<2:0>
保留	80	72	0x0000A4	—	—	—

# dsPIC33EPXXGS70X/80X 系列

表 7-1: 中断向量详细信息 (续)

中断源	向量编号	IRQ 编号	IVT 地址	中断位的位置		
				标志	允许	优先级
PSES——PWM 辅助特殊事件匹配	81	73	0x0000A6	IFS4<9> PSESIF	IEC4<9> PSESIE	IPC18<6:4> PSESIP<2:0>
保留	82-97	74-89	0x0000A8-0x0000C6	—	—	—
SPI3TX——SPI3 传输完成	98	90	0x0000C8	IFS5<10> SPI3TXIF	IEC5<10> SPI3TXIE	IPC22<10:8> SPI3TXIP<2:0>
SPI3RX——SPI3 接收完成	99	91	0x0000CA	IFS5<10> SPI3RXIF	IEC5<11> SPI3RXIE	IPC22<14:12> SPI3RXIP<2:0>
保留	100-101	92-93	0x0000CC-0x0000CE	—	—	—
PWM1——PWM1 中断	102	94	0x0000D0	IFS5<14> PWM1IF	IEC5<14> PWM1IE	IPC23<10:8> PWM1IP<2:0>
PWM2——PWM2 中断	103	95	0x0000D2	IFS5<15> PWM2IF	IEC5<15> PWM2IE	IPC23<14:12> PWM2IP<2:0>
PWM3——PWM3 中断	104	96	0x0000D4	IFS6<0> PWM3IF	IEC6<0> PWM3IE	IPC24<2:0> PWM3IP<2:0>
PWM4——PWM4 中断	105	97	0x0000D6	IFS6<1> PWM4IF	IEC6<1> PWM4IE	IPC24<6:4> PWM4IP<2:0>
PWM5——PWM5 中断	106	98	0x0000D8	IFS6<2> PWM5IF	IEC6<2> PWM5IE	IPC24<10:8> PWM5IP<2:0>
PWM6——PWM6 中断	107	99	0x0000DA	IFS6<3> PWM6IF	IEC6<3> PWM6IE	IPC24<14:12> PWM6IP<2:0>
PWM7——PWM7 中断	108	100	0x0000DC	IFS6<4> PWM7IF	IEC6<4> PWM7IE	IPC25<2:0> PWM7IP<2:0>
PWM8——PWM8 中断	109	101	0x0000DE	IFS6<5> PWM8IF	IEC6<5> PWM8IE	IPC25<6:4> PWM8IP<2:0>
保留	110	102	0x0000E0	—	—	—
AC2——模拟比较器 2 中断	111	103	0x0000E2	IFS6<7> AC2IF	IEC6<7> AC2IE	IPC25<14:12> AC2IP<2:0>
AC3——模拟比较器 3 中断	112	104	0x0000E4	IFS6<8> AC3IF	IEC6<8> AC3IE	IPC26<2:0> AC3IP<2:0>
AC4——模拟比较器 4 中断	113	105	0x0000E6	IFS6<9> AC4IF	IEC6<9> AC4IE	IPC26<6:4> AC4IP<2:0>
保留	114-117	106-109	0x0000E8-0x0000EE	—	—	—
AN0 转换完成	118	110	0x0000F0	IFS6<14> AN0IF	IEC6<14> AN0IE	IPC27<10:8> AN0IP<2:0>
AN1 转换完成	119	111	0x0000F2	IFS6<15> AN1IF	IEC6<15> AN1IE	IPC27<14:12> AN1IP<2:0>
AN2 转换完成	120	112	0x0000F4	IFS7<0> AN2IF	IEC7<0> AN2IE	IPC28<2:0> AN2IP<2:0>
AN3 转换完成	121	113	0x0000F6	IFS7<1> AN3IF	IEC7<1> AN3IE	IPC28<6:4> AN3IP<2:0>
AN4 转换完成	122	114	0x0000F8	IFS7<2> AN4IF	IEC7<2> AN4IE	IPC28<10:8> AN4IP<2:0>
AN5 转换完成	123	115	0x0000FA	IFS7<3> AN5IF	IEC7<3> AN5IE	IPC28<14:12> AN5IP<2:0>
AN6 转换完成	124	116	0x0000FC	IFS7<4> AN6IF	IEC7<4> AN6IE	IPC29<2:0> AN6IP<2:0>
AN7 转换完成	125	117	0x0000FE	IFS7<5> AN7IF	IEC7<5> AN7IE	IPC29<6:4> AN7IP<2:0>
保留	126-131	118-123	0x000100-0x00010A	—	—	—

# dsPIC33EPXXGS70X/80X 系列

表 7-1: 中断向量详细信息 (续)

中断源	向量编号	IRQ 编号	IVT 地址	中断位的位置		
				标志	允许	优先级
SPI1 错误中断	132	124	0x00010C	IFS7<12> SPI1IF	IEC7<12> SPI1IE	IPC31<2:0> SPI1IP<2:0>
SPI2 错误中断	133	125	0x00010E	IFS7<13> SPI2IF	IEC7<13> SPI2IE	IPC31<6:4> SPI2IP<2:0>
SPI3 错误中断	134	126	0x000110	IFS7<13> SPI3IF	IEC7<13> SPI3IE	IPC31<10:8> SPI3IP<2:0>
保留	135-145	127-137	0x000112-0x000126	—	—	—
CLC1 中断	146	138	0x000128	IFS8<10> CLC1IF	IEC8<10> CLC1IE	IPC34<10:8> CLC1IP<2:0>
CLC2 中断	147	139	0x00012A	IFS8<11> CLC2IF	IEC8<11> CLC2IE	IPC34<14:12> CLC2IP<2:0>
CLC3 中断	148	140	0x00012C	IFS8<12> CLC3IF	IEC8<12> CLC3IE	IPC35<2:0> CLC3IP<2:0>
CLC4 中断	149	141	0x00012E	IFS8<13> CLC4IF	IEC8<13> CLC4IE	IPC35<6:4> CLC4IP<2:0>
ICD——ICD 应用	150	142	0x000130	IFS8<14> ICDIF	IEC8<14> ICDIE	IPC35<10:8> ICDIP<2:0>
JTAG——JTAG 编程	151	143	0x000132	IFS8<15> JTAGIF	IEC8<15> JTAGIE	IPC35<14:12> JTAGIP<2:0>
保留	152	144	0x000134	—	—	—
PTGSTEP——PTG 步阶	153	145	0x000136	IFS9<1> PTGSTEPIF	IEC9<1> PTGSTEPIE	IPC36<6:4> PTGSTEP<2:0>
PTGWDT——PTG WDT 超时	154	146	0x000138	IFS9<2> PTGWDTIF	IEC9<2> PTGWDTIE	IPC36<10:8> PTGWDT<2:0>
PTG0——PTG 中断触发 0	155	147	0x00013A	IFS9<3> PTG0IF	IEC9<3> PTG0IE	IPC36<14:12> PTG0IP<2:0>
PTG1——PTG 中断触发 1	156	148	0x00013C	IFS9<4> PTG1IF	IEC9<4> PTG1IE	IPC37<2:0> PTG1IP<2:0>
PTG2——PTG 中断触发 2	157	149	0x00013E	IFS9<5> PTG2IF	IEC9<5> PTG2IE	IPC37<6:4> PTG2IP<2:0>
PTG3——PTG 中断触发 3	158	150	0x000140	IFS9<6> PTG3IF	IEC9<6> PTG3IE	IPC37<10:8> PTG3IP<2:0>
AN8 转换完成	159	151	0x000142	IFS9<7> AN8IF	IEC9<7> AN8IE	IPC37<14:12> AN8IP<2:0>
AN9 转换完成	160	152	0x000144	IFS9<8> AN9IF	IEC9<8> AN9IE	IPC38<2:0> AN9IP<2:0>
AN10 转换完成	161	153	0x000146	IFS9<9> AN10IF	IEC9<9> AN10IE	IPC38<6:4> AN10IP<2:0>
AN11 转换完成	162	154	0x000148	IFS9<10> AN11IF	IEC9<10> AN11IE	IPC38<10:8> AN11IP<2:0>
AN12 转换完成	163	155	0x00014A	IFS9<11> AN12IF	IEC9<11> AN12IE	IPC38<14:12> AN12IP<2:0>
AN13 转换完成	164	156	0x00014C	IFS9<12> AN13IF	IEC9<12> AN13IE	IPC39<2:0> AN13IP<2:0>
AN14 转换完成	165	157	0x00014E	IFS9<13> AN14IF	IEC9<13> AN14IE	IPC39<6:4> AN14IP<2:0>
AN15 转换完成	166	158	0x000150	IFS9<14> AN15IF	IEC9<14> AN15IE	IPC39<10:8> AN15IP<2:0>
AN16 转换完成	167	159	0x000152	IFS9<15> AN16IF	IEC9<15> AN16IE	IPC39<14:12> AN16IP<2:0>

# dsPIC33EPXXGS70X/80X 系列

表 7-1: 中断向量详细信息 (续)

中断源	向量编号	IRQ 编号	IVT 地址	中断位的位置		
				标志	允许	优先级
AN17 转换完成	168	160	0x000154	IFS10<0> AN17IF	IEC10<0> AN17IE	IPC40<2:0> AN17IP<2:0>
AN18 转换完成	169	161	0x000156	IFS10<1> AN18IF	IEC10<1> AN18IE	IPC40<6:4> AN18IP<2:0>
AN19 转换完成	170	162	0x000158	IFS10<2> AN19IF	IEC10<2> AN19IE	IPC40<10:8> AN19IP<2:0>
AN20 转换完成	171	163	0x00015A	IFS10<3> AN20IF	IEC10<3> AN20IE	IPC40<14:12> AN20IP<2:0>
AN21 转换完成	172	164	0x00015C	IFS10<4> AN21IF	IEC10<4> AN21IE	IPC41<2:0> AN21IP<2:0>
保留	173-180	165-172	0x00015C-0x00016C	—	—	—
I2C1——I2C1 总线冲突	181	173	0x00016E	IFS10<13> I2C1IF	IEC10<13> I2C1IE	IPC43<6:4> I2C1IP<2:0>
I2C2——I2C2 总线冲突	182	174	0x000170	IFS10<14> I2C2IF	IEC10<14> I2C2IE	IPC43<10:8> I2C2IP<2:0>
保留	183-184	175-176	0x000172-0x000174	—	—	—
ADCMP0——ADC 数字比较器 0	185	177	0x000176	IFS11<1> ADCMP0IF	IEC11<1> ADCMP0IE	IPC44<6:4> ADCMP0IP<2:0>
ADCMP1——ADC 数字比较器 1	186	178	0x000178	IFS11<2> ADCMP1IF	IEC11<2> ADCMP1IE	IPC44<10:8> ADCMP1IP<2:0>
ADFLTR0——ADC 滤波器 0	187	179	0x00017A	IFS11<3> ADFLTR0IF	IEC11<3> ADFLTR0IE	IPC44<14:12> ADFLTR0IP<2:0>
ADFLTR1——ADC 滤波器 1	188	180	0x00017C	IFS11<4> ADFLTR1IF	IEC11<4> ADFLTR1IE	IPC45<2:0> ADFLTR1IP<2:0>
保留	189-253	181-245	0x00017E-0x000192	—	—	—



## 7.3 中断资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 7.3.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“中断” (DS70000600)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

## 7.4 中断控制和状态寄存器

dsPIC33EPXXGS70X/80X 系列器件实现了以下用于中断控制器的寄存器：

- INTCON1
- INTCON2
- INTCON3
- INTCON4
- INTTREG

### 7.4.1 INTCON1 至 INTCON4

INTCON1、INTCON2、INTCON3 和 INTCON4 控制全局中断控制功能。

INTCON1 包含中断嵌套禁止位 (NSTDIS) 以及处理器陷阱源的控制和状态标志。

INTCON2 寄存器控制外部中断请求信号行为，还包含全局中断允许 (Global Interrupt Enable, GIE) 位和备用中断向量表使能 (Alternate Interrupt Vector Table Enable, AIVTEN) 位。

INTCON3 包含附属 PLL 和 DO 堆栈溢出状态陷阱源的状态标志。

INTCON4 寄存器包含软件生成的硬陷阱 (Software Generated Hard Trap, SGHT) 状态位。

### 7.4.2 IFSx

IFSx 寄存器维护所有中断请求标志。每个中断源都具有一个中断标志状态位，该状态位由相应的外设中断或外部中断信号置 1，通过软件进行清零。

### 7.4.3 IECx

IECx 寄存器维护所有中断允许位。这些控制位用于单独允许外设中断或外部中断信号。

### 7.4.4 IPCx

IPCx 寄存器用于设置每个中断源的中断优先级 (Interrupt Priority Level, IPL)。可以为每个用户中断源分配 7 个优先级之一。

### 7.4.5 INTTREG

INTTREG 寄存器包含相关的中断向量编号和新的 CPU 中断优先级，分别锁存在 INTTREG 寄存器中的待处理中断向量编号 (VECNUM<7:0>) 和新 CPU 中断优先级 (ILR<3:0>) 位域中。新的中断优先级是等待处理中断的优先级。

中断源按表 7-1 中的顺序分配给 IFSx、IECx 和 IPCx 寄存器。例如，INT0 (外部中断 0) 向量编号为 8、自然顺序优先级为 0。所以，INT0IF 位在 IFS0<0> 中，INT0IE 位在 IEC0<0> 中，INT0IP<2:0> 位在 IPC0 的第一个位置 (IPC0<2:0>) 中。

### 7.4.6 状态 / 控制寄存器

尽管这些寄存器不是中断控制硬件的特定组成部分，但其中两个 CPU 控制寄存器包含控制中断功能的位。关于这些寄存器的详细信息，请参见《dsPIC33/PIC24 系列参考手册》中的“dsPIC33E 增强型 CPU”一章 (DS70005158)。

- CPU 状态寄存器 SR 包含 IPL<2:0> 位 (SR<7:5>)。这些位指示当前 CPU 中断优先级。用户软件可以通过写 IPLx 位来更改当前 CPU 中断优先级。
- CORCON 寄存器包含 IPL3 位，这个位与 IPL<2:0> 位一起指示当前 CPU 优先级。IPL3 是只读位，所以用户软件不能屏蔽陷阱事件。

下面各页中的寄存器 7-3 至寄存器 7-7 介绍了所有中断寄存器。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 7-1: SR: CPU 状态寄存器 <sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/C-0	R/C-0	R-0	R/W-0
OA	OB	SA	SB	OAB	SAB	DA	DC
bit 15							bit 8

R/W-0 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 <sup>(2)</sup>	IPL1 <sup>(2)</sup>	IPL0 <sup>(2)</sup>	RA	N	OV	Z	C
bit 7							bit 0

<b>图注:</b>	C = 可清零位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 7-5      **IPL<2:0>**: CPU 中断优先级状态位 <sup>(2,3)</sup>
- 111 = CPU 中断优先级为 7 (15); 禁止用户中断
  - 110 = CPU 中断优先级为 6 (14)
  - 101 = CPU 中断优先级为 5 (13)
  - 100 = CPU 中断优先级为 4 (12)
  - 011 = CPU 中断优先级为 3 (11)
  - 010 = CPU 中断优先级为 2 (10)
  - 001 = CPU 中断优先级为 1 (9)
  - 000 = CPU 中断优先级为 0 (8)

- 注 1:** 如需了解整个寄存器的详细信息, 请参见 [寄存器 3-1](#)。
- 2:** IPL<2:0> 位与 IPL<3> 位 (CORCON<3>) 组合形成 CPU 中断优先级。如果 IPL<3> = 1, 那么括号中的值表示 IPL。当 IPL<3> = 1 时, 禁止用户中断。
- 3:** 当 NSTDIS 位 (INTCON1<15>) = 1 时, IPL<2:0> 状态位是只读的。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 7-2: CORCON: 内核控制寄存器<sup>(1)</sup>

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-0
VAR	—	US1	US0	EDT	DL2	DL1	DL0
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-0	R/C-0	R-0	R/W-0	R/W-0
SATA	SATB	SATDW	ACCSAT	IPL3 <sup>(2)</sup>	SFA	RND	IF
bit 7							bit 0

### 图注:

C = 可清零位	U = 未实现位, 读为 0
R = 可读位	W = 可写位
-n = POR 时的值	1 = 置 1
	0 = 清零
	x = 未知

- bit 15      **VAR:** 可变异常处理延时控制位  
 1 = 使能可变异常处理延时  
 0 = 使能固定异常处理延时
- bit 3      **IPL3:** CPU 中断优先级状态位 3<sup>(2)</sup>  
 1 = CPU 中断优先级大于 7  
 0 = CPU 中断优先级小于或等于 7

注 1: 如需了解整个寄存器的详细信息, 请参见寄存器 3-2。  
 2: IPL3 位与 IPL<2:0> 位 (SR<7:5>) 组合形成 CPU 中断优先级。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 7-3: INTCON1: 中断控制寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NSTDIS	OVAERR	OVBERR	COVAERR	COVBERR	OVATE	OVBTE	COVTE
bit 15						bit 8	

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
SFTACERR	DIV0ERR	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
bit 7						bit 0	

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15      **NSTDIS:** 中断嵌套禁止位  
1 = 禁止中断嵌套  
0 = 使能中断嵌套
- bit 14      **OVAERR:** 累加器 A 溢出陷阱标志位  
1 = 陷阱由累加器 A 溢出引起  
0 = 陷阱不是由累加器 A 溢出引起
- bit 13      **OVBERR:** 累加器 B 溢出陷阱标志位  
1 = 陷阱由累加器 B 溢出引起  
0 = 陷阱不是由累加器 B 溢出引起
- bit 12      **COVAERR:** 累加器 A 灾难性溢出陷阱标志位  
1 = 陷阱由累加器 A 灾难性溢出引起  
0 = 陷阱不是由累加器 A 灾难性溢出引起
- bit 11      **COVBERR:** 累加器 B 灾难性溢出陷阱标志位  
1 = 陷阱由累加器 B 灾难性溢出引起  
0 = 陷阱不是由累加器 B 灾难性溢出引起
- bit 10      **OVATE:** 累加器 A 溢出陷阱允许位  
1 = 允许累加器 A 溢出陷阱  
0 = 禁止陷阱
- bit 9        **OVBTE:** 累加器 B 溢出陷阱允许位  
1 = 允许累加器 B 溢出陷阱  
0 = 禁止陷阱
- bit 8        **COVTE:** 灾难性溢出陷阱允许位  
1 = 允许累加器 A 或 B 的灾难性溢出陷阱  
0 = 禁止陷阱
- bit 7        **SFTACERR:** 累加器移位错误状态位  
1 = 数学错误陷阱由非法累加器移位引起  
0 = 数学错误陷阱不是由非法累加器移位引起
- bit 6        **DIV0ERR:** 被零除错误状态位  
1 = 数学错误陷阱由被零除引起  
0 = 数学错误陷阱不是由被零除引起
- bit 5        **未实现:** 读为 0
- bit 4        **MATHERR:** 数学错误状态位  
1 = 发生了数学错误陷阱  
0 = 未发生数学错误陷阱
- bit 3        **ADDRERR:** 地址错误陷阱状态位  
1 = 发生了地址错误陷阱  
0 = 未发生地址错误陷阱

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 7-3: INTCON1: 中断控制寄存器 1 (续)

bit 2	<b>STKERR:</b> 堆栈错误陷阱状态位 1 = 发生了堆栈错误陷阱 0 = 未发生堆栈错误陷阱
bit 1	<b>OSCFAIL:</b> 振荡器故障陷阱状态位 1 = 发生了振荡器故障陷阱 0 = 未发生振荡器故障陷阱
bit 0	<b>未实现:</b> 读为 0

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 7-4: INTCON2: 中断控制寄存器 2

R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
GIE	DISI	SWTRAP	—	—	—	—	AIVTEN
bit 15							bit 8

U-0	U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	INT4EP	—	INT2EP	INT1EP	INT0EP
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15      **GIE:** 全局中断允许位  
1 = 允许中断并将相关的 IE 位置 1  
0 = 禁止中断, 但仍然允许陷阱
- bit 14      **DISI:** DISI 指令状态位  
1 = DISI 指令有效  
0 = DISI 指令无效
- bit 13      **SWTRAP:** 软件陷阱状态位  
1 = 允许软件陷阱  
0 = 禁止软件陷阱
- bit 12-9    **未实现:** 读为 0
- bit 8        **AIVTEN:** 备用中断向量表使能位  
1 = 使用备用中断向量表  
0 = 使用标准中断向量表
- bit 7-5     **未实现:** 读为 0
- bit 4        **INT4EP:** 外部中断 4 边沿检测极性选择位  
1 = 下降沿中断  
0 = 上升沿中断
- bit 3        **未实现:** 读为 0
- bit 2        **INT2EP:** 外部中断 2 边沿检测极性选择位  
1 = 下降沿中断  
0 = 上升沿中断
- bit 1        **INT1EP:** 外部中断 1 边沿检测极性选择位  
1 = 下降沿中断  
0 = 上升沿中断
- bit 0        **INT0EP:** 外部中断 0 边沿检测极性选择位  
1 = 下降沿中断  
0 = 上升沿中断

# dsPIC33EPXXGS70X/80X 系列

**寄存器 7-5: INTCON3: 中断控制寄存器 3**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	NAE
bit 15							bit 8
U-0	U-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0
—	—	—	DOOVR	—	—	—	APLL
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-9      **未实现:** 读为 0
- bit 8        **NAE:** NVM 地址错误软陷阱状态位  
               1 = 发生了 NVM 地址错误软陷阱  
               0 = 未发生 NVM 地址错误软陷阱
- bit 7-5      **未实现:** 读为 0
- bit 4        **DOOVR:** DO 堆栈溢出软陷阱状态位  
               1 = 发生了 DO 堆栈溢出软陷阱  
               0 = 未发生 DO 堆栈溢出软陷阱
- bit 3-1      **未实现:** 读为 0
- bit 0        **APLL:** 附属 PLL 丢失锁定软陷阱状态位  
               1 = 发生了 APLL 锁定软陷阱  
               0 = 未发生 APLL 锁定软陷阱

**寄存器 7-6: INTCON4: 中断控制寄存器 4**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	SGHT
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-1     **未实现:** 读为 0
- bit 0        **SGHT:** 软件生成的硬陷阱状态位  
               1 = 发生了软件生成的硬陷阱  
               0 = 未发生软件生成的硬陷阱

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 7-7: INTTREG: 中断控制和状态寄存器

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ILR3	ILR2	ILR1	ILR0
bit 15				bit 8			

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
VECNUM7	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-12      **未实现:** 读为 0
- bit 11-8      **ILR<3:0>:** 新的 CPU 中断优先级位
  - 1111 = CPU 中断优先级为 15
  - 
  - 
  - 
  - 0001 = CPU 中断优先级为 1
  - 0000 = CPU 中断优先级为 0
- bit 7-0      **VECNUM<7:0>:** 待处理中断向量编号位
  - 11111111 = 255, 保留; 不要使用
  - 
  - 
  - 
  - 00001001 = 9, IC1—— 输入捕捉 1
  - 00001000 = 8, INTO—— 外部中断 0
  - 00000111 = 7, 保留; 不要使用
  - 00000110 = 6, 通用软陷阱
  - 00000101 = 5, 保留; 不要使用
  - 00000100 = 4, 数学错误陷阱
  - 00000011 = 3, 堆栈错误陷阱
  - 00000010 = 2, 通用硬陷阱
  - 00000001 = 1, 地址错误陷阱
  - 00000000 = 0, 振荡器故障陷阱



## 8.0 直接存储器访问 (DMA)

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息,请参见《dsPIC33/PIC24 系列参考手册》的“直接存储器访问 (DMA)” (DS70348), 该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息,请参见本数据手册中的第 4.0 节“存储器构成”。

DMA 控制器在外设数据寄存器和数据空间 SRAM 之间传输数据。

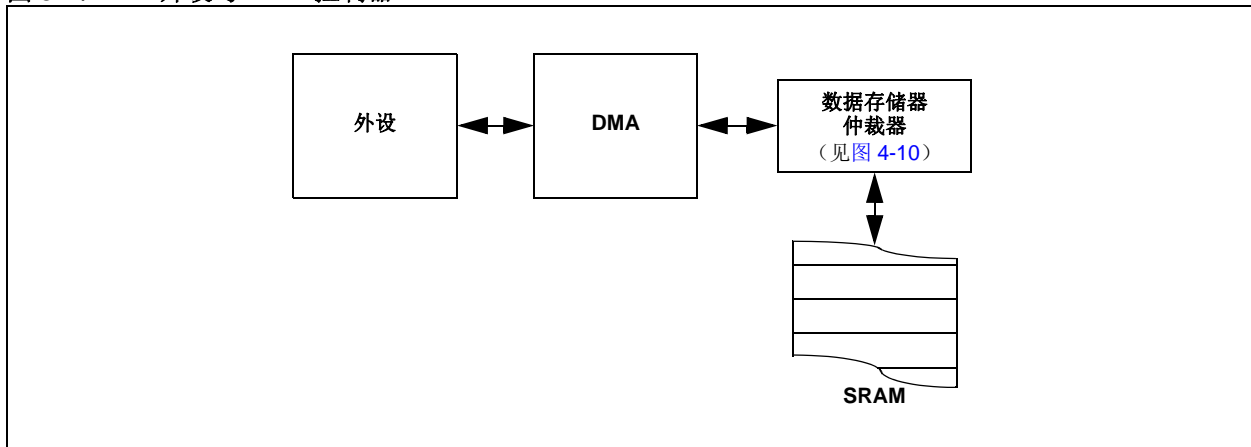
此外, DMA 可以访问整个数据存储空间。当 CPU 或 DMA 尝试访问 SRAM 时, 将会使用数据存储器总线仲裁器, 这可能导致 DMA 或 CPU 暂停。

DMA 控制器支持 4 个独立通道。每个通道都可以配置为向选定外设发送数据或从选定外设接收数据。DMA 控制器支持的外设包括:

- CAN
- UART
- 输入捕捉
- 输出比较
- 定时器

关于受支持外设的完整列表, 请参见表 8-1。

图 8-1: 外设与 DMA 控制器



# dsPIC33EPXXGS70X/80X 系列

此外，DMA 传输可以通过定时器和外部中断进行触发。每个 DMA 通道都是单向的。要对某个外设进行读和写操作，必须分配两个 DMA 通道。如果有多个通道接收到数据传输请求，则基于通道编号的简单固定优先级机制会指定哪个通道完成传输，哪个或哪些通道保持等待状态。每个 DMA 通道会传送数据块，之后向 CPU 发出中断，指示数据块已可进行处理。

DMA 控制器具有以下功能：

- 4 个 DMA 通道
- 带后递增的寄存器间接寻址模式
- 不带后递增的寄存器间接寻址模式

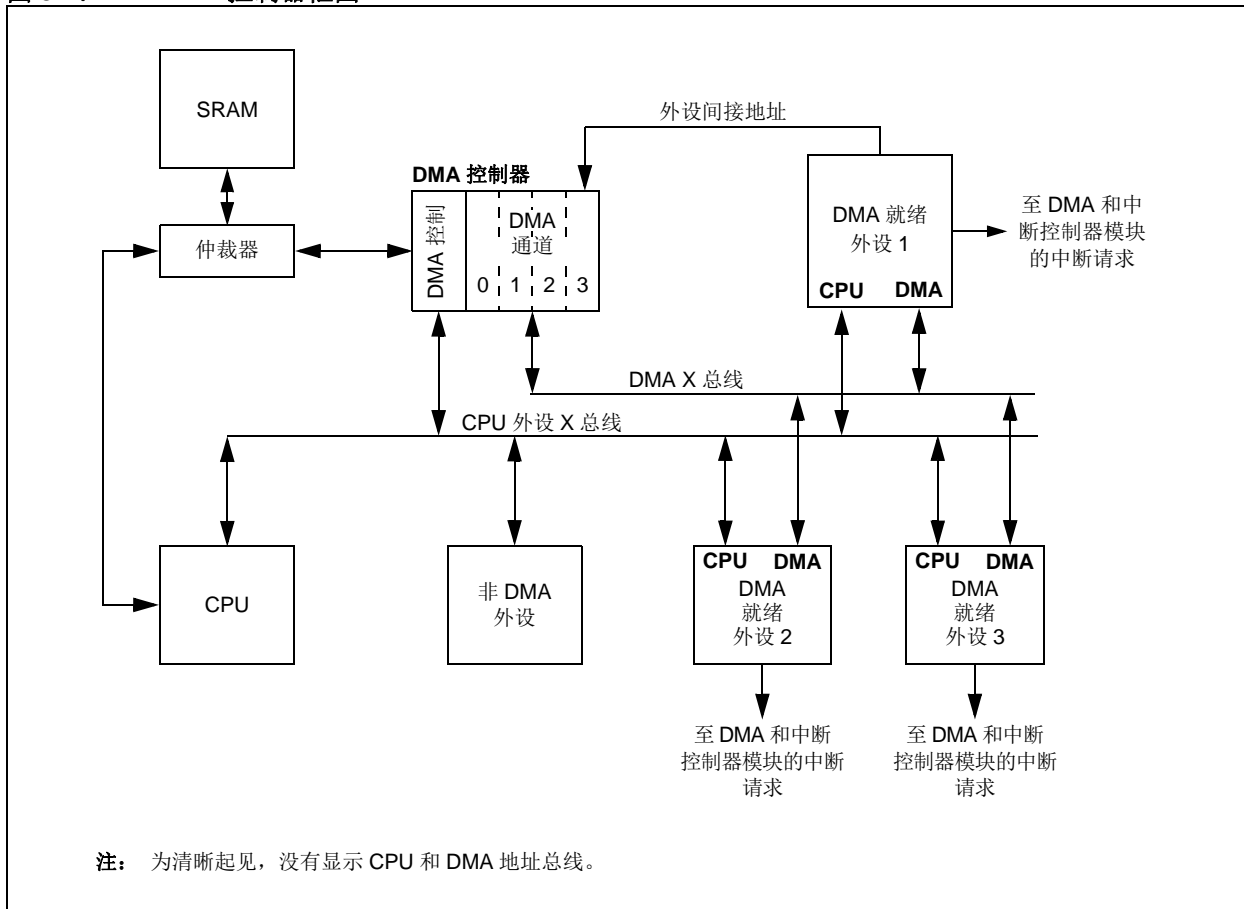
- 外设间接寻址模式（外设生成目标地址）
- 在传输完一半或整个数据块后发出中断给 CPU
- 字节或字传输
- 固定优先级通道仲裁
- 手动（软件）或自动（外设 DMA 请求）启动传输
- 单数据块或自动重复数据块传输模式
- “乒乓”（Ping-Pong）模式（每个数据块传输完成后，在两个 SRAM 起始地址之间进行自动切换）
- 每个通道的 DMA 请求可以从任何支持的中断源选择
- 调试支持功能

表 8-1 中列出了可以使用 DMA 的外设。

表 8-1: DMA 通道与外设的关联

外设与 DMA 的关联	DMAxREQ 寄存器 IRQSEL<7:0> 位	DMAxPAD 寄存器（要向外设读取的值）	DMAxPAD 寄存器（要向外设写入的值）
INT0——外部中断 0	00000000	—	—
IC1——输入捕捉 1	00000001	0x0144（IC1BUF）	—
IC2——输入捕捉 2	00000101	0x014C（IC2BUF）	—
IC3——输入捕捉 3	00100101	0x0154（IC3BUF）	—
IC4——输入捕捉 4	00100110	0x015C（IC4BUF）	—
OC1——输出比较 1	00000010	—	0x0906（OC1R） 0x0904（OC1RS）
OC2——输出比较 2	00000110	—	0x0910（OC2R） 0x090E（OC2RS）
OC3——输出比较 3	00011001	—	0x091A（OC3R） 0x0918（OC3RS）
OC4——输出比较 4	00011010	—	0x0924（OC4R） 0x0922（OC4RS）
TMR2——Timer2	00000111	—	—
TMR3——Timer3	00001000	—	—
TMR4——Timer4	00011011	—	—
TMR5——Timer5	00011100	—	—
UART1RX——UART1 接收器	00001011	0x0226（U1RXREG）	—
UART1TX——UART1 发送器	00001100	—	0x0224（U1TXREG）
UART2RX——UART2 接收器	00011110	0x0236（U2RXREG）	—
UART2TX——UART2 发送器	00011111	—	0x0234（U2TXREG）
CAN1——接收数据就绪	00100010	0x0440（C1RXD）	—
CAN1——发送数据请求	01000110	—	0x0442（C1TXD）
CAN2——接收数据就绪	00110111	0x0540（C2RXD）	—
CAN2——发送数据请求	01000111	—	0x0542（C2TXD）

图 8-2: DMA 控制器框图



# dsPIC33EPXXGS70X/80X 系列

## 8.1 DMA 控制器寄存器

每个 DMA 控制器通道 x（其中，x = 0 至 3）均包含以下寄存器：

- 16 位 DMA 通道 x 控制寄存器（DMAxCON）
- 16 位 DMA 通道 x IRQ 选择寄存器（DMAxREQ）
- 32 位 DMA 通道 x 起始地址寄存器 A（DMAxSTAL/H）
- 32 位 DMA 通道 x 起始地址寄存器 B（DMAxSTBL/H）
- 16 位 DMA 通道 x 外设地址寄存器（DMAxPAD）
- 14 位 DMA 通道 x 传输计数寄存器（DMAxCNT）

还有一些状态寄存器（DMAPWC、DMARQC、DMAPPS、DMALCA 和 DSADRL/H）是所有 DMA 控制器通道共用的。这些状态寄存器提供关于写冲突和请求冲突的信息，以及最近地址和通道访问信息。

中断标志（DMAxIF）位于中断控制器的 IFSx 寄存器中。对应的中断允许控制位（DMAxIE）位于中断控制器的 IECx 寄存器中，对应的中断优先级控制位（DMAxIP）位于中断控制器的 IPCx 寄存器中。

**寄存器 8-1: DMAxCON: DMA 通道 x 控制寄存器**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHEN	SIZE	DIR	HALF	NULLW	—	—	—
bit 15					bit 8		

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	AMODE1	AMODE0	—	—	MODE1	MODE0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位，读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

- bit 15        **CHEN:** DMA 通道使能位  
 1 = 使能通道  
 0 = 禁止通道
- bit 14        **SIZE:** DMA 数据传输长度位  
 1 = 字节  
 0 = 字
- bit 13        **DIR:** 传输方向位（源 / 目标总线选择）  
 1 = 从 RAM 地址读取，写入外设地址  
 0 = 从外设地址读取，写入 RAM 地址
- bit 12        **HALF:** 数据块传输中断选择位  
 1 = 当传送了一半数据时，发出中断  
 0 = 当传送了所有数据时，发出中断
- bit 11        **NULLW:** 空数据外设写模式选择位  
 1 = 除将外设地址中的数据写入 RAM 外，还将空数据写入外设地址（DIR 位也必须清零）  
 0 = 正常工作
- bit 10-6      **未实现:** 读为 0
- bit 5-4        **AMODE<1:0>:** DMA 通道寻址模式选择位  
 11 = 保留  
 10 = 外设间接寻址模式  
 01 = 不带后递增的寄存器间接寻址模式  
 00 = 带后递增的寄存器间接寻址模式
- bit 3-2        **未实现:** 读为 0
- bit 1-0        **MODE<1:0>:** DMA 通道工作模式选择位  
 11 = 单数据块，使能乒乓模式（与每个 DMA 缓冲区之间传输一块数据）  
 10 = 连续数据块，使能乒乓模式  
 01 = 单数据块，禁止乒乓模式  
 00 = 连续数据块，禁止乒乓模式

# dsPIC33EPXXGS70X/80X 系列

寄存器 8-2: DMAxREQ: DMA 通道 x IRQ 选择寄存器

R/S-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
FORCE <sup>(1)</sup>	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IRQSEL7	IRQSEL6	IRQSEL5	IRQSEL4	IRQSEL3	IRQSEL2	IRQSEL1	IRQSEL0
bit 7							bit 0

<b>图注:</b>	S = 可置 1 位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15      **FORCE:** 强制 DMA 传输位 <sup>(1)</sup>  
 1 = 强制进行单次 DMA 传输 (手动模式)  
 0 = 按照 DMA 请求自动启动 DMA 传输
- bit 14-8    **未实现:** 读为 0
- bit 7-0     **IRQSEL<7:0>:** DMA 外设 IRQ 编号选择位  
 01000111 = CAN2—— 发送数据请求  
 01000110 = CAN1—— 发送数据请求  
 00110111 = CAN2—— 接收数据就绪  
 00100110 = IC4—— 输入捕捉 4  
 00100101 = IC3—— 输入捕捉 3  
 00100010 = CAN1—— 接收数据就绪  
 00011111 = UART2TX——UART2 发送器  
 00011110 = UART2RX——UART2 接收器  
 00011100 = TMR5——Timer5  
 00011011 = TMR4——Timer4  
 00011010 = OC4—— 输出比较 4  
 00011001 = OC3—— 输出比较 3  
 00001100 = UART1TX——UART1 发送器  
 00001011 = UART1RX——UART1 接收器  
 00001000 = TMR3——Timer3  
 00000111 = TMR2——Timer2  
 00000110 = OC2—— 输出比较 2  
 00000101 = IC2—— 输入捕捉 2  
 00000010 = OC1—— 输出比较 1  
 00000001 = IC1—— 输入捕捉 1  
 00000000 = INTO—— 外部中断 0

**注 1:** FORCE 位不能被用户软件清零。当强制的 DMA 传输完成或通道被禁止 (CHEN = 0) 时, FORCE 位由硬件清零。

# dsPIC33EPXXXGS70X/80X 系列

## 寄存器 8-3: DMAxSTAH: DMA 通道 x 起始地址寄存器 A (高位字)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<23:16>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-8            未实现: 读为 0  
 bit 7-0            **STA<23:16>**: DMA 主起始地址位 (源地址或目标地址)

## 寄存器 8-4: DMAxSTAL: DMA 通道 x 起始地址寄存器 A (低位字)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-0            **STA<15:0>**: DMA 主起始地址位 (源地址或目标地址)

# dsPIC33EPXXGS70X/80X 系列

寄存器 8-5: **DMAxSTBH: DMA 通道 x 起始地址寄存器 B (高位字)**

U-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<23:16>							
bit 7							bit 0

图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            未实现: 读为 0  
 bit 7-0            **STB<23:16>**: DMA 辅助起始地址位 (源地址或目标地址)

寄存器 8-6: **DMAxSTBL: DMA 通道 x 起始地址寄存器 B (低位字)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<15:8>							
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<7:0>							
bit 7							bit 0

图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0            **STB<15:0>**: DMA 辅助起始地址位 (源地址或目标地址)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 8-7: DMAxPAD: DMA 通道 x 外设地址寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PAD<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PAD<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0            **PAD<15:0>**: DMA 外设地址寄存器位

**注 1:** 如果使能了通道 (即通道处于工作状态), 写入该寄存器可能导致 DMA 通道的行为不可预测, 应该避免。

## 寄存器 8-8: DMAxCNT: DMA 通道 x 传输计数寄存器<sup>(1)</sup>

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—		CNT<13:8> <sup>(2)</sup>					
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNT<7:0> <sup>(2)</sup>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-14            **未实现:** 读为 0

bit 13-0            **CNT<13:0>**: DMA 传输计数寄存器位<sup>(2)</sup>

**注 1:** 如果使能了通道 (即通道处于工作状态), 写入该寄存器可能导致 DMA 通道的行为不可预测, 应该避免。

**2:** DMA 传输的次数 = CNT<13:0> + 1。



# dsPIC33EPXXGS70X/80X 系列

寄存器 8-9: **DSADRH: DMA 最近访问的 RAM 地址高位字寄存器**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<23:16>							
bit 7							bit 0

图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8              未实现: 读为 0  
 bit 7-0              **DSADR<23:16>**: DMA 最近访问的 DMA 地址位

寄存器 8-10: **DSADRL: DMA 最近访问的 RAM 地址低位字寄存器**

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<15:8>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<7:0>							
bit 7							bit 0

图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-0              **DSADR<15:0>**: DMA 最近访问的 DMA 地址位

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 8-11: DMAPWC: DMA 外设写冲突状态寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	PWCOL3	PWCOL2	PWCOL1	PWCOL0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-4      **未实现:** 读为 0
- bit 3        **PWCOL3:** 通道 3 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 2        **PWCOL2:** 通道 2 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 1        **PWCOL1:** 通道 1 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 0        **PWCOL0:** 通道 0 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突

# dsPIC33EPXXGS70X/80X 系列

**寄存器 8-12: DMARQC: DMA 请求冲突状态寄存器**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	RQCOL3	RQCOL2	RQCOL1	RQCOL0
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15-4      **未实现:** 读为 0
- bit 3        **RQCOL3:** 通道 3 传输请求冲突标志位  
1 = 检测到用户强制传输请求与基于中断传输请求的冲突  
0 = 未检测到请求冲突
- bit 2        **RQCOL2:** 通道 2 传输请求冲突标志位  
1 = 检测到用户强制传输请求与基于中断传输请求的冲突  
0 = 未检测到请求冲突
- bit 1        **RQCOL1:** 通道 1 传输请求冲突标志位  
1 = 检测到用户强制传输请求与基于中断传输请求的冲突  
0 = 未检测到请求冲突
- bit 0        **RQCOL0:** 通道 0 传输请求冲突标志位  
1 = 检测到用户强制传输请求与基于中断传输请求的冲突  
0 = 未检测到请求冲突

# dsPIC33EPXXGS70X/80X 系列

寄存器 8-13: **DMALCA**: 上一次工作的 DMA 通道状态寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	R-1	R-1	R-1	R-1
—	—	—	—	LSTCH<3:0>			
bit 7				bit 0			

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-4        **未实现:** 读为 0  
 bit 3-0        **LSTCH<3:0>**: 上一次工作的 DMA 控制器通道状态位  
               1111 = 自系统复位以来没有发生 DMA 传输  
               1110 = 保留  
               •  
               •  
               •  
               0100 = 保留  
               0011 = 上次数据传输是通过通道 3 进行处理的  
               0010 = 上次数据传输是通过通道 2 进行处理的  
               0001 = 上次数据传输是通过通道 1 进行处理的  
               0000 = 上次数据传输是通过通道 0 进行处理的

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 8-14: DMAPPS: DMA 乒乓状态寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	PPST3	PPST2	PPST1	PPST0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-4      **未实现:** 读为 0
- bit 3      **PPST3:** 通道 3 乒乓模式状态标志位  
 1 = 选择 DMA3STB 寄存器  
 0 = 选择 DMA3STA 寄存器
- bit 2      **PPST2:** 通道 2 乒乓模式状态标志位  
 1 = 选择 DMA2STB 寄存器  
 0 = 选择 DMA2STA 寄存器
- bit 1      **PPST1:** 通道 1 乒乓模式状态标志位  
 1 = 选择 DMA1STB 寄存器  
 0 = 选择 DMA1STA 寄存器
- bit 0      **PPST0:** 通道 0 乒乓模式状态标志位  
 1 = 选择 DMA0STB 寄存器  
 0 = 选择 DMA0STA 寄存器

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 9.0 振荡器配置

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册中的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“**振荡器模块**”（DS70005131），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

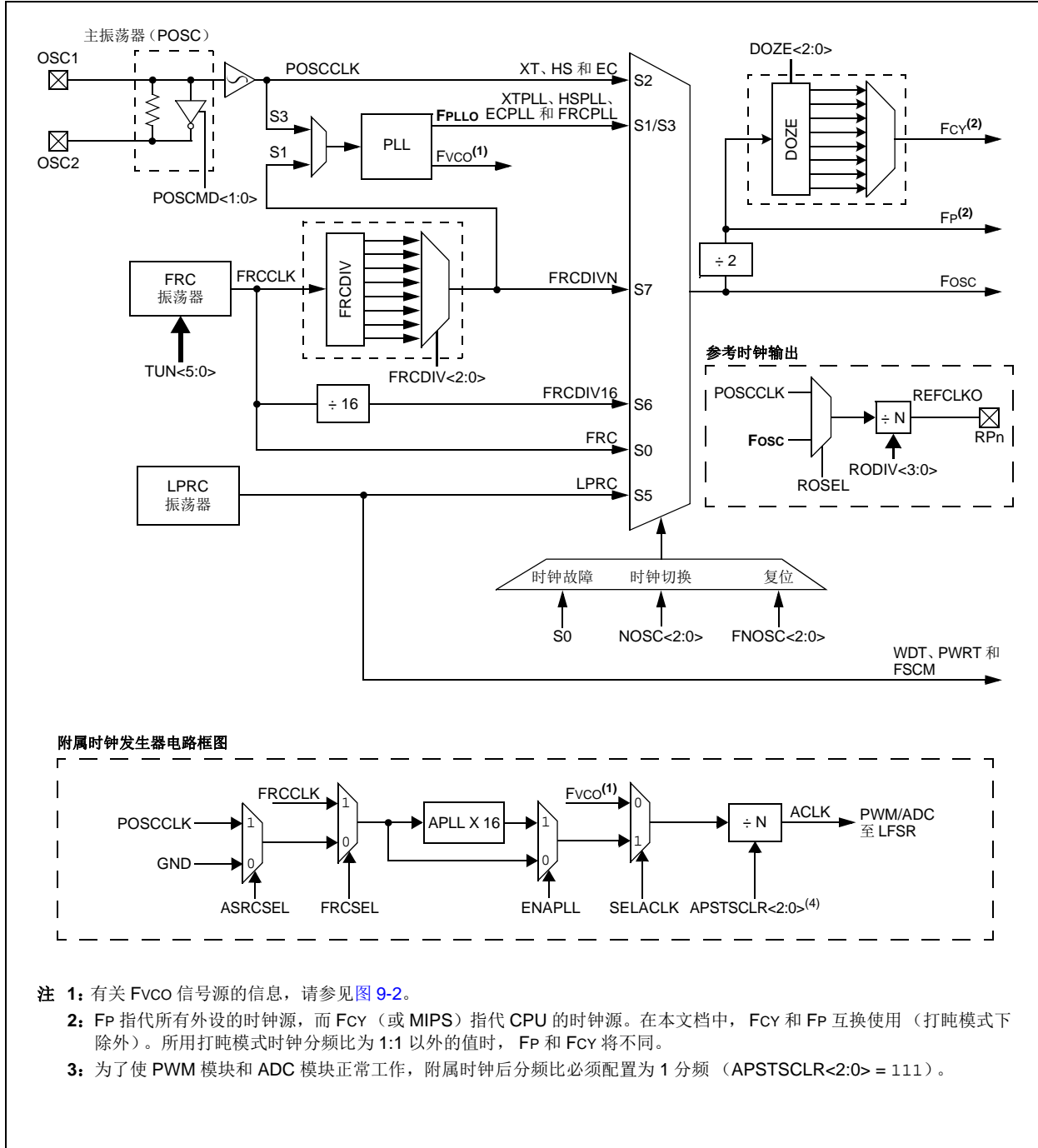
dsPIC33EPXXGS70X/80X 系列振荡器系统提供：

- 片上锁相环（Phase-Locked Loop, PLL），可基于选定的内部和外部振荡器源升高内部工作频率
- 在各种时钟源之间进行动态时钟切换
- 用于节省系统功耗的打盹模式
- 故障保护时钟监视器（Fail-Safe Clock Monitor, FSCM），可检测时钟故障，并允许安全地恢复或关闭应用
- 用于时钟源选择的配置位
- 用于 ADC 和 PWM 的附属 PLL

图 9-1 给出了振荡器系统的简化框图。

# dsPIC33EPXXGS70X/80X 系列

图 9-1: 振荡器系统框图



- 注 1: 有关 Fvco 信号源的信息, 请参见图 9-2。
- 2: Fp 指代所有外设的时钟源, 而 Fcy (或 MIPS) 指代 CPU 的时钟源。在本文中, Fcy 和 Fp 互换使用 (打盹模式下除外)。所用打盹模式时钟分频比为 1:1 以外的值时, Fp 和 Fcy 将不同。
- 3: 为了使 PWM 模块和 ADC 模块正常工作, 附属时钟后分频比必须配置为 1 分频 (APSTSCLR<2:0> = 111)。



# dsPIC33EPXXXGS70X/80X 系列

## 9.1 CPU 时钟系统

dsPIC33EPXXXGS70X/80X 系列器件提供 6 种系统时钟选项：

- 快速 RC (FRC) 振荡器
- 带锁相环的 FRC 振荡器 (FRCPLL)
- 带后分频器的 FRC 振荡器
- 主 (XT、HS 或 EC) 振荡器
- 带 PLL 的主振荡器 (XTPLL、HSPLL 和 ECPLL)
- 低功耗 RC (LPRC) 振荡器

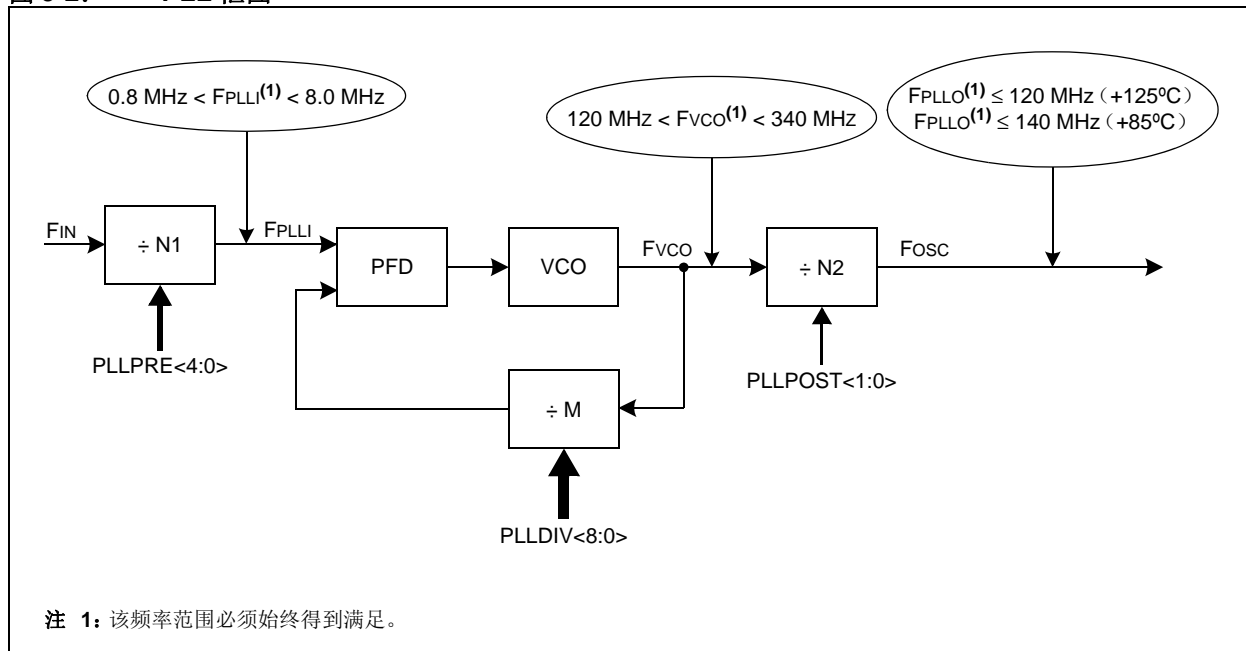
指令执行速度或器件工作频率  $F_{CY}$  由公式 9-1 计算。

公式 9-1: 器件工作频率

$$F_{CY} = F_{OSC}/2$$

图 9-2 给出了 PLL 模块的框图。公式 9-2 给出了输入频率 ( $F_{IN}$ ) 和输出频率 ( $F_{PULO}$ ) 之间的关系。公式 9-3 给出了输入频率 ( $F_{IN}$ ) 和 VCO 频率 ( $F_{VCO}$ ) 之间的关系。

图 9-2: PLL 框图



公式 9-2:  $F_{PULO}$  计算

$$F_{PULO} = F_{IN} \times \left( \frac{M}{N1 \times N2} \right) = F_{IN} \times \left( \frac{PLLDIV<8:0> + 2}{(PLLPRE<4:0> + 2) \times 2(PLLPOST<1:0> + 1)} \right)$$

其中:

$$N1 = PLLPRE<4:0> + 2$$

$$N2 = 2 \times (PLLPOST<1:0> + 1)$$

$$M = PLLDIV<8:0> + 2$$

公式 9-3:  $F_{VCO}$  计算

$$F_{VCO} = F_{IN} \times \left( \frac{M}{N1} \right) = F_{IN} \times \left( \frac{PLLDIV<8:0> + 2}{(PLLPRE<4:0> + 2)} \right)$$

# dsPIC33EPXXGS70X/80X 系列

表 9-1: 用于时钟选择的配置位值

振荡器模式	振荡器源	POSCMD<1:0>	FNOSC<2:0>	参见注
n 分频快速 RC 振荡器 (FRCDIVN)	内部	xx	111	1, 2
16 分频快速 RC 振荡器 (FRCDIV16)	内部	xx	110	1
低功耗 RC 振荡器 (LPRC)	内部	xx	101	1
带 PLL 的主振荡器 (HS) (HSPLL)	主	10	011	
带 PLL 的主振荡器 (XT) (XTPLL)	主	01	011	
带 PLL 的主振荡器 (EC) (ECPLL)	主	00	011	1
主振荡器 (HS)	主	10	010	
主振荡器 (XT)	主	01	010	
主振荡器 (EC)	主	00	010	1
带 PLL 的 N 分频快速 RC 振荡器 (FRC) (FRCPLL)	内部	xx	001	1
快速 RC 振荡器 (FRC)	内部	xx	000	1

注 1: OSC2 引脚功能由 OSCIOFNC 配置位决定。

注 2: 这是未编程 (已擦除) 器件的默认振荡器模式。

## 9.2 附属时钟产生

附属时钟产生用于需要以与系统时钟无关的频率工作的外设, 例如 PWM 或 ADC。

可将主振荡器和内部 FRC 振荡器源与附属 PLL (APLL) 配合使用, 以获得附属时钟。附属 PLL 具有固定的 16x 倍频因子。

附属时钟具有以下配置限制:

- 为了使 PWM 正常工作, 必须将产生的附属时钟频率配置为 120 MHz (见第 30.0 节“电气特性”中的参数 OS56)。如果需要较低的频率, 则应使用 PWM 输入时钟预分频比选择位 (PCLKDIV<2:0>)。
- 要实现 1.04 ns 的 PWM 分辨率, 附属时钟必须使用 16x 附属 PLL (APLL)。所有其他时钟源的最小 PWM 分辨率为 8 ns。
- 如果使用主 PLL 作为附属时钟的来源, 则主 PLL 的最大工作速率应配置为 30 MIPS 或更低。

## 9.3 参考时钟产生

通过参考时钟输出逻辑, 用户能够基于系统时钟或器件引脚上的晶振来输出时钟信号。用户应用程序可以指定范围较宽的时钟频率缩放比例来产生参考时钟输出。

## 9.4 振荡器资源

对于本数据手册中列出的器件, Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 9.4.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“振荡器模块” (DS70005131)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

## 9.5 振荡器控制寄存器

寄存器 9-1: **OSCCON**: 振荡器控制寄存器 <sup>(1)</sup>

U-0	R-0	R-0	R-0	U-0	R/W-y	R/W-y	R/W-y
—	COSC2	COSC1	COSC0	—	NOSC2 <sup>(2)</sup>	NOSC1 <sup>(2)</sup>	NOSC0 <sup>(2)</sup>
bit 15							bit 8
R/W-0	R/W-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
CLKLOCK	IOLOCK	LOCK	—	CF <sup>(3)</sup>	—	—	OSWEN
bit 7							bit 0

**图注:** y = 在 POR 时由配置位设置的值  
R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15            **未实现:** 读为 0

bit 14-12        **COSC<2:0>:** 当前振荡器选择位 (只读)  
111 = n 分频的快速 RC 振荡器 (FRC)  
110 = 16 分频的快速 RC 振荡器 (FRC)  
101 = 低功耗 RC 振荡器 (LPRC)  
100 = 保留  
011 = 带 PLL 的主振荡器 (XT、HS 和 EC)  
010 = 主振荡器 (XT、HS 和 EC)  
001 = 带 PLL 的 N 分频快速 RC 振荡器 (FRC) (FRCPLL)  
000 = 快速 RC 振荡器 (FRC)

bit 11            **未实现:** 读为 0

bit 10-8        **NOSC<2:0>:** 新振荡器选择位 <sup>(2)</sup>  
111 = n 分频的快速 RC 振荡器 (FRC)  
110 = 16 分频的快速 RC 振荡器 (FRC)  
101 = 低功耗 RC 振荡器 (LPRC)  
100 = 保留  
011 = 带 PLL 的主振荡器 (XT、HS 和 EC)  
010 = 主振荡器 (XT、HS 和 EC)  
001 = 带 PLL 的 N 分频快速 RC 振荡器 (FRC) (FRCPLL)  
000 = 快速 RC 振荡器 (FRC)

bit 7            **CLKLOCK:** 时钟锁定使能位  
1 = 如果 FCKSM0 = 1, 则锁定时钟和 PLL 配置; 如果 FCKSM0 = 0, 则可以修改时钟和 PLL 配置  
0 = 不锁定时钟和 PPL 选择, 可以修改配置

bit 6            **IOLOCK:** I/O 锁定使能位  
1 = 锁定 I/O  
0 = 不锁定 I/O

bit 5            **LOCK:** PLL 锁定状态位 (只读)  
1 = 指示 PLL 处于锁定状态, 或 PLL 起振定时器延时结束  
0 = 指示 PLL 处于失锁状态, 起振定时器在进行延时或 PLL 被禁止

- 注 1:** 对该寄存器进行写操作需要解锁序列。  
**2:** 不允许直接在使能 PLL 的任何主振荡器模式和 FRCPLL 模式之间进行时钟切换 (这指两者之间任意方向的时钟切换)。在这些情况下, 应用必须首先切换到 FRC 模式将其作为两个 PLL 模式之间的过渡时钟源。  
**3:** 该位只能用软件清零。用软件将该位置 1 (= 1) 的效果与实际振荡器故障相同, 将触发振荡器故障陷阱。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 9-1: OSCCON: 振荡器控制寄存器<sup>(1)</sup> (续)

bit 4	<b>未实现:</b> 读为 0
bit 3	<b>CF:</b> 时钟故障检测位 <sup>(3)</sup> 1 = FSCM 已检测到时钟故障 0 = FSCM 未检测到时钟故障
bit 2-1	<b>未实现:</b> 读为 0
bit 0	<b>OSWEN:</b> 振荡器切换使能位 1 = 请求切换为由 NOSC<2:0> 位指定的振荡器 0 = 振荡器切换完成

**注 1:** 对该寄存器进行写操作需要解锁序列。

- 2:** 不允许直接在使能 PLL 的任何主振荡器模式和 FRCPLL 模式之间进行时钟切换（这指两者之间任意方向的时钟切换）。在这些情况下，应用必须首先切换到 FRC 模式将其作为两个 PLL 模式之间的过渡时钟源。
- 3:** 该位只能用软件清零。用软件将该位置 1 (= 1) 的效果与实际振荡器故障相同，将触发振荡器故障陷阱。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 9-2: CLKDIV: 时钟分频比寄存器

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE2 <sup>(1)</sup>	DOZE1 <sup>(1)</sup>	DOZE0 <sup>(1)</sup>	DOZEN <sup>(2,3)</sup>	FRCDIV2	FRCDIV1	FRCDIV0
bit 15						bit 8	

R/W-0	R/W-1	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PLLPOST1	PLLPOST0	—	PLLPRE4	PLLPRE3	PLLPRE2	PLLPRE1	PLLPRE0
bit 7						bit 0	

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15        **ROI:** 中断恢复位  
 1 = 中断将清零 DOZEN 位, 并且处理器时钟与外设时钟之间的频率比被设置为 1:1  
 0 = 中断对 DOZEN 位没有影响

bit 14-12    **DOZE<2:0>:** 处理器时钟分频比选择位 <sup>(1)</sup>  
 111 = Fcy 被 128 分频  
 110 = Fcy 被 64 分频  
 101 = Fcy 被 32 分频  
 100 = Fcy 被 16 分频  
 011 = Fcy 被 8 分频 (默认)  
 010 = Fcy 被 4 分频  
 001 = Fcy 被 2 分频  
 000 = Fcy 被 1 分频

bit 11        **DOZEN:** 打吨模式使能位 <sup>(2,3)</sup>  
 1 = DOZE<2:0> 位域指定外设时钟与处理器时钟之间的频率比  
 0 = 处理器时钟与外设时钟之间的频率比被强制为 1:1

bit 10-8     **FRCDIV<2:0>:** 内部快速 RC 振荡器后分频比位  
 111 = FRC 被 256 分频  
 110 = FRC 被 64 分频  
 101 = FRC 被 32 分频  
 100 = FRC 被 16 分频  
 011 = FRC 被 8 分频  
 010 = FRC 被 4 分频  
 001 = FRC 被 2 分频  
 000 = FRC 被 1 分频 (默认)

bit 7-6      **PLLPOST<1:0>:** PLL VCO 输出分频比选择位 (也表示为 “N2”, PLL 后分频比)  
 11 = 输出被 8 分频  
 10 = 保留  
 01 = 输出被 4 分频 (默认)  
 00 = 输出被 2 分频

bit 5        **未实现:** 读为 0

**注 1:** 只有 DOZEN 位清零时, 才能写入 DOZE<2:0> 位。如果 DOZEN = 1, 则对 DOZE<2:0> 的任何写操作都会被忽略。

**2:** 该位在 ROI 位置 1 和产生中断时清零。

**3:** 如果 DOZE<2:0> = 000, 则 DOZEN 位不能置 1。如果 DOZE<2:0> = 000, 则用户软件将 DOZEN 位置 1 的任何尝试都会被忽略。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 9-2: CLKDIV: 时钟分频比寄存器 (续)

bit 4-0      **PLLPRE<4:0>**: PLL 相位检测器输入分频比选择位 (也表示为 “N1”, PLL 预分频比)  
 11111 = 输入被 33 分频  
 •  
 •  
 •  
 00001 = 输入被 3 分频  
 00000 = 输入被 2 分频 (默认)

**注 1:** 只有 DOZEN 位清零时, 才能写入 DOZE<2:0> 位。如果 DOZEN = 1, 则对 DOZE<2:0> 的任何写操作都会被忽略。

**2:** 该位在 ROI 位置 1 和产生中断时清零。

**3:** 如果 DOZE<2:0> = 000, 则 DOZEN 位不能置 1。如果 DOZE<2:0> = 000, 则用户软件将 DOZEN 位置 1 的任何尝试都会被忽略。

## 寄存器 9-3: PLLFBD: PLL 反馈分频比寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PLLDIV8
bit 15							bit 8

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
PLLDIV<7:0>							
bit 7							bit 0

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-9      **未实现:** 读为 0

bit 8-0      **PLLDIV<8:0>**: PLL 反馈分频比位 (也表示为 “M”, PLL 倍频比)

111111111 = 513  
 •  
 •  
 •  
 000110000 = 50 (默认)  
 •  
 •  
 •  
 000000010 = 4  
 000000001 = 3  
 000000000 = 2

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 9-4: OSCTUN: FRC 振荡器调节寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN<5:0>					
bit 7						bit 0	

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-6      **未实现:** 读为 0

bit 5-0      **TUN<5:0>:** FRC 振荡器调节位

0111111 = 最大频率偏差为 1.457% (7.477 MHz)

0111110 = 中心频率 + 1.41% (7.474 MHz)

•

•

•

000001 = 中心频率 + 0.047% (7.373 MHz)

000000 = 中心频率 (标称值 7.37 MHz)

1111111 = 中心频率 - 0.047% (7.367 MHz)

•

•

•

100001 = 中心频率 - 1.457% (7.263 MHz)

100000 = 最小频率偏差为 -1.5% (7.259 MHz)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 9-5: ACLKCON: 附属时钟分频比控制寄存器

R/W-0	R-0	R/W-1	U-0	U-0	R/W-1	R/W-1	R/W-1
ENAPLL	APLLCK	SELACLK	—	—	APSTSCLR2	APSTSCLR1	APSTSCLR0
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
ASRCSEL	FRCSEL	—	—	—	—	—	—
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15        **ENAPLL:** 附属 PLL 使能位  
               1 = 使能 APLL  
               0 = 禁止 APLL
- bit 14        **APLLCK:** APLL 锁定状态位 (只读)  
               1 = 指示附属 PLL 处于锁定状态  
               0 = 指示附属 PLL 不处于锁定状态
- bit 13        **SELACLK:** 附属时钟分频器的附属时钟源选择位  
               1 = 附属振荡器为附属时钟分频器提供时钟源  
               0 = 主 PLL (Fvco) 为附属时钟分频器提供时钟源
- bit 12-11    **未实现:** 读为 0
- bit 10-8     **APSTSCLR<2:0>:** 附属时钟输出分频比位  
               111 = 1 分频  
               110 = 2 分频  
               101 = 4 分频  
               100 = 8 分频  
               011 = 16 分频  
               010 = 32 分频  
               001 = 64 分频  
               000 = 256 分频
- bit 7        **ASRCSEL:** 附属时钟的参考时钟源选择位  
               1 = 主振荡器是时钟源  
               0 = 不选择任何时钟输入
- bit 6        **FRCSEL:** 附属 PLL 的参考时钟源选择位  
               1 = 为附属 PLL 选择 FRC 时钟  
               0 = 输入时钟源由 ASRCSEL 位的设置决定
- bit 5-0     **未实现:** 读为 0



# dsPIC33EPXXGS70X/80X 系列

**寄存器 9-6: REFOCON: 参考振荡器控制寄存器**

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROON	—	ROSSLP	ROSEL	RODIV3 <sup>(1)</sup>	RODIV2 <sup>(1)</sup>	RODIV1 <sup>(1)</sup>	RODIV0 <sup>(1)</sup>
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15      **ROON:** 参考振荡器输出使能位  
1 = 在 RPn 引脚上使能参考振荡器输出 <sup>(2)</sup>  
0 = 禁止参考振荡器输出
- bit 14      **未实现:** 读为 0
- bit 13      **ROSSLP:** 参考振荡器在休眠模式下运行位  
1 = 参考振荡器输出在休眠模式下继续运行  
0 = 参考振荡器输出在休眠模式下被禁止
- bit 12      **ROSEL:** 参考振荡器源选择位  
1 = 晶振用作参考时钟  
0 = 系统时钟用作参考时钟
- bit 11-8    **RODIV<3:0>:** 参考振荡器分频比位 <sup>(1)</sup>  
1111 = 参考时钟被 32,768 分频  
1110 = 参考时钟被 16,384 分频  
1101 = 参考时钟被 8,192 分频  
1100 = 参考时钟被 4,096 分频  
1011 = 参考时钟被 2,048 分频  
1010 = 参考时钟被 1,024 分频  
1001 = 参考时钟被 512 分频  
1000 = 参考时钟被 256 分频  
0111 = 参考时钟被 128 分频  
0110 = 参考时钟被 64 分频  
0101 = 参考时钟被 32 分频  
0100 = 参考时钟被 16 分频  
0011 = 参考时钟被 8 分频  
0010 = 参考时钟被 4 分频  
0001 = 参考时钟被 2 分频  
0000 = 参考时钟
- bit 7-0     **未实现:** 读为 0

**注 1:** 在写入这些位之前, 必须先禁止参考振荡器输出 (ROON = 0)。

**注 2:** 该引脚是可重映射的。更多信息, 请参见第 11.6 节 “外设引脚选择 (PPS)”。

# dsPIC33EPXXGS70X/80X 系列

寄存器 9-7: LFSR: 线性反馈移位寄存器

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	LFSR<14:8>						
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LFSR<7:0>							
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15      未实现: 读为 0

bit 14-0    LFSR<14:0>: 伪随机数据位

## 10.0 节能特性

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“看门狗定时器 and 节能模式” (DS70615)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

dsPIC33EPXXGS70X/80X 系列器件提供了管理功耗的功能，该功能是通过有选择地管理 CPU 和外设的时钟来实现的。一般来说，较低的时钟频率和减少时钟所驱动外设的数目可降低功耗。

dsPIC33EPXXGS70X/80X 系列器件可通过以下 4 种方式来管理功耗：

- 时钟频率
- 基于指令的休眠模式和空闲模式
- 软件控制的打盹模式
- 用软件有选择地进行外设控制

可以组合使用这些方法，从而在保证关键应用特性（如对于时序敏感的通信）的情况下有选择地调节应用的功耗。

### 例 10-1: PWRSAV 指令语法

```
PWRSAV #SLEEP_MODE ; Put the device into Sleep mode
PWRSAV #IDLE_MODE ; Put the device into Idle mode
```

## 10.1 时钟频率和时钟切换

dsPIC33EPXXGS70X/80X 系列器件的时钟频率范围较宽，用户可根据应用需要进行选择。如果未锁定系统时钟配置，用户只需更改 NOSC<sub>x</sub> 位 (OSCCON<10:8>) 即可选择低功耗或高精度振荡器。在工作期间更改系统时钟的过程以及相应的限制，已在第 9.0 节“振荡器配置”中进行了详细的讨论。

## 10.2 基于指令的节能模式

dsPIC33EPXXGS70X/80X 系列器件有两种特殊的节能模式，通过执行特殊的 PWRSAV 指令可以进入这两种模式。休眠模式下时钟停止工作并暂停所有代码执行。空闲模式下 CPU 暂停工作并暂停代码执行，但是允许外设模块继续工作。PWRSAV 指令的汇编语法如例 10-1 所示。

**注:** SLEEP\_MODE 和 IDLE\_MODE 是在所选器件的汇编包含文件中定义的常量。

在被允许的中断产生、WDT 超时或器件复位时，器件会退出休眠和空闲模式。器件退出这两种模式的过程称为“唤醒”。

# dsPIC33EPXXGS70X/80X 系列

## 10.2.1 休眠模式

休眠模式具有以下特征：

- 系统时钟源关闭。如果使用了片上振荡器，也要关闭它。
- 如果没有 I/O 引脚拉出电流，则器件电流消耗将降至最低。
- 因为系统时钟源被禁止，所以故障保护时钟监视器在休眠模式下不工作。
- 如果 WDT 被使能，则 LPRC 时钟将在休眠模式下继续运行。
- 如果 WDT 被使能，则它在进入休眠模式之前被自动清零。
- 有些器件功能或外设能在休眠模式下继续工作，包括 I/O 端口上的输入电平变化通知功能或使用外部时钟输入的外设等。
- 任何需要使用系统时钟源工作的外设将在休眠模式下将被禁止。

当发生以下任何事件时，器件将从休眠模式唤醒：

- 产生任何已被单独允许的中断
- 任何形式的器件复位
- WDT 超时

从休眠模式唤醒时，处理器将使用在进入休眠模式时处于工作状态的时钟源重新开始工作。

为了实现最佳的节能效果，可以通过清零 VREGS (RCON<8>) 和 VREGSF (RCON<11>) 位，将内部稳压器和闪存稳压器配置为在休眠模式下进入待机状态（默认配置）。

如果应用需要更快的唤醒速度，并且可以接受更高的电流要求，可以将 VREGS (RCON<8>) 和 VREGSF (RCON<11>) 位置 1，使内部稳压器和闪存稳压器在休眠模式下保持工作状态。

## 10.2.2 空闲模式

空闲模式具有以下特征：

- CPU 停止执行指令。
- WDT 被自动清零。
- 系统时钟源保持工作状态。默认情况下，所有外设模块将继续使用系统时钟源正常工作，也可以有选择地禁止它们（见第 10.4 节“外设模块禁止”）。
- 如果 WDT 或 FSCM 被使能，则 LPRC 也将保持工作状态。

当发生以下任何事件时，器件将从空闲模式唤醒：

- 产生任何已被单独允许的中断
- 任何器件复位
- WDT 超时

从空闲模式唤醒时，重新为 CPU 提供时钟并开始执行指令（2-4 个时钟周期后），从 PWRSAV 指令之后的下一条指令或 ISR 中的第一条指令开始执行。

所有外设还都具有在进入空闲模式时停止工作的选项，以提高节能效果。该选项可在每个外设的控制寄存器中进行选择；例如，Timer1 控制寄存器中的 TSIDL 位 (T1CON<13>)。

## 10.2.3 在节能指令执行期间的中断

在执行 PWRSAV 指令时产生的任何中断都将延迟到进入休眠或空闲模式后才起作用，并导致器件从休眠或空闲模式唤醒。

## 10.3 打盹模式

更改时钟速度和使用某种节能模式是降低功耗的首选策略。在某些情况下可能不可行。例如，某些应用可能必须保持不间断的同步通信，即便在它不执行任何其他操作时也不例外。降低系统时钟速度可能会带来通信错误，而使用节能模式可能会完全停止通信。

打盹模式是另一种简单有效的节能方法，它可以在器件仍然执行代码的情况下降低功耗。在此模式下，系统时钟以相同的时钟源和相同的速度继续工作。外设模块时钟速度保持不变，但 CPU 时钟速度降低了。保持这两个时钟域同步，可以保持外设访问 SFR 的能力，同时 CPU 以较慢的速度执行代码。

通过将 DOZEN 位 (CLKDIV<11>) 置 1 使能打盹模式。外设与内核的时钟速度之比是由 DOZE<2:0> 位 (CLKDIV<14:12>) 决定的。有 8 种可能的配置，从 1:1 至 1:128，其中 1:1 是默认设置。

在事件驱动的应用中，程序可以使用打盹模式有选择地降低功耗。这样就可以实现不间断地运行对时序敏感的功能（如同步通信），而 CPU 保持空闲等待事件调用中断服务程序。通过将 ROI 位 (CLKDIV<15>) 置 1，可以使器件在产生中断时自动返回到全速 CPU 工作模式。默认情况下，中断事件对打盹模式工作没有影响。

## 10.4 外设模块禁止

外设模块禁止 (Peripheral Module Disable, PMD) 寄存器通过停止提供给外设模块的所有时钟源提供一种禁止外设模块的方法。当通过相应的 PMD 控制位禁止外设时，外设就进入了功耗最低的状态。与外设相关的控制和状态寄存器也会被禁止，因此写入这些寄存器不起作用，且读取值无效。

只有在 PMD 寄存器中的相应位被清零且特定的 dsPIC® DSC 器件支持某个外设时，才会使能相应的外设模块。如果外设存在于器件中，则默认情况下，它是通过 PMD 寄存器使能的。

**注：** 如果 PMD 位置 1，则对应的模块将在一个指令周期的延时后被禁止。类似地，如果 PMD 位清零，则对应的模块将在一个指令周期的延时后被使能（假设已将模块控制寄存器配置为使能模块的工作）。

## 10.5 节能资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 10.5.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“看门狗定时器 and 节能模式” (DS70615)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

寄存器 10-1: PMD1: 外设模块禁止控制寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
T5MD	T4MD	T3MD	T2MD	T1MD	—	PWMMD	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	C2MD	C1MD	ADCMD
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15        **T5MD:** Timer5 模块禁止位  
 1 = 禁止 Timer5 模块  
 0 = 使能 Timer5 模块

bit 14        **T4MD:** Timer4 模块禁止位  
 1 = 禁止 Timer4 模块  
 0 = 使能 Timer4 模块

bit 13        **T3MD:** Timer3 模块禁止位  
 1 = 禁止 Timer3 模块  
 0 = 使能 Timer3 模块

bit 12        **T2MD:** Timer2 模块禁止位  
 1 = 禁止 Timer2 模块  
 0 = 使能 Timer2 模块

bit 11        **T1MD:** Timer1 模块禁止位  
 1 = 禁止 Timer1 模块  
 0 = 使能 Timer1 模块

bit 10        **未实现:** 读为 0

bit 9         **PWMMD:** PWM 模块禁止位  
 1 = 禁止 PWM 模块  
 0 = 使能 PWM 模块

bit 8         **未实现:** 读为 0

bit 7         **I2C1MD:** I2C1 模块禁止位  
 1 = 禁止 I2C1 模块  
 0 = 使能 I2C1 模块

bit 6         **U2MD:** UART2 模块禁止位  
 1 = 禁止 UART2 模块  
 0 = 使能 UART2 模块

bit 5         **U1MD:** UART1 模块禁止位  
 1 = 禁止 UART1 模块  
 0 = 使能 UART1 模块

bit 4         **SPI2MD:** SPI2 模块禁止位  
 1 = 禁止 SPI2 模块  
 0 = 使能 SPI2 模块

bit 3         **SPI1MD:** SPI1 模块禁止位  
 1 = 禁止 SPI1 模块  
 0 = 使能 SPI1 模块

bit 2         **C2MD:** CAN2 模块禁止位  
 1 = 禁止 CAN2 模块  
 0 = 使能 CAN2 模块

## 寄存器 10-1: PMD1: 外设模块禁止控制寄存器 1 (续)

- bit 1      **C1MD:** CAN1 模块禁止位  
            1 = 禁止 CAN1 模块  
            0 = 使能 CAN1 模块
- bit 0      **ADCMD:** ADC 模块禁止位  
            1 = 禁止 ADC 模块  
            0 = 使能 ADC 模块

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 10-2: PMD2: 外设模块禁止控制寄存器 2

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD
bit 15						bit 8	

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	OC4MD	OC3MD	OC2MD	OC1MD
bit 7						bit 0	

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-12     **未实现:** 读为 0
- bit 11       **IC4MD:** 输入捕捉 4 模块禁止位  
1 = 禁止输入捕捉 4 模块  
0 = 使能输入捕捉 4 模块
- bit 10       **IC3MD:** 输入捕捉 3 模块禁止位  
1 = 禁止输入捕捉 3 模块  
0 = 使能输入捕捉 3 模块
- bit 9         **IC2MD:** 输入捕捉 2 模块禁止位  
1 = 禁止输入捕捉 2 模块  
0 = 使能输入捕捉 2 模块
- bit 8         **IC1MD:** 输入捕捉 1 模块禁止位  
1 = 禁止输入捕捉 1 模块  
0 = 使能输入捕捉 1 模块
- bit 7-4       **未实现:** 读为 0
- bit 3         **OC4MD:** 输出比较 4 模块禁止位  
1 = 禁止输出比较 4 模块  
0 = 使能输出比较 4 模块
- bit 2         **OC3MD:** 输出比较 3 模块禁止位  
1 = 禁止输出比较 3 模块  
0 = 使能输出比较 3 模块
- bit 1         **OC2MD:** 输出比较 2 模块禁止位  
1 = 禁止输出比较 2 模块  
0 = 使能输出比较 2 模块
- bit 0         **OC1MD:** 输出比较 1 模块禁止位  
1 = 禁止输出比较 1 模块  
0 = 使能输出比较 1 模块



# dsPIC33EPXXGS70X/80X 系列

**寄存器 10-3: PMD3: 外设模块禁止控制寄存器 3**

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
—	—	—	—	—	CMPMD	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	I2C2MD	—
bit 7						bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-11    **未实现:** 读为 0
- bit 10        **CMPMD:** 比较器模块禁止位  
               1 = 禁止比较器模块  
               0 = 使能比较器模块
- bit 9-2      **未实现:** 读为 0
- bit 1        **I2C2MD:** I2C2 模块禁止位  
               1 = 禁止 I2C2 模块  
               0 = 使能 I2C2 模块
- bit 0        **未实现:** 读为 0

**寄存器 10-4: PMD4: 外设模块禁止控制寄存器 4**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
—	—	—	—	REFOMD	—	—	—
bit 7						bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-4     **未实现:** 读为 0
- bit 3        **REFOMD:** 参考时钟模块禁止位  
               1 = 禁止参考时钟模块  
               0 = 使能参考时钟模块
- bit 2-0      **未实现:** 读为 0

# dsPIC33EPXXGS70X/80X 系列

寄存器 10-5: **PMD6: 外设模块禁止控制寄存器 6**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	SPI3MD
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15      **PWM8MD:** PWM8 模块禁止位  
1 = 禁止 PWM8 模块  
0 = 使能 PWM8 模块
- bit 14      **PWM7MD:** PWM7 模块禁止位  
1 = 禁止 PWM7 模块  
0 = 使能 PWM7 模块
- bit 13      **PWM6MD:** PWM6 模块禁止位  
1 = 禁止 PWM6 模块  
0 = 使能 PWM6 模块
- bit 12      **PWM5MD:** PWM5 模块禁止位  
1 = 禁止 PWM5 模块  
0 = 使能 PWM5 模块
- bit 11      **PWM4MD:** PWM4 模块禁止位  
1 = 禁止 PWM4 模块  
0 = 使能 PWM4 模块
- bit 10      **PWM3MD:** PWM3 模块禁止位  
1 = 禁止 PWM3 模块  
0 = 使能 PWM3 模块
- bit 9        **PWM2MD:** PWM2 模块禁止位  
1 = 禁止 PWM2 模块  
0 = 使能 PWM2 模块
- bit 8        **PWM1MD:** PWM1 模块禁止位  
1 = 禁止 PWM1 模块  
0 = 使能 PWM1 模块
- bit 7-1     **未实现:** 读为 0
- bit 0        **SPI3MD:** SPI3 模块禁止位  
1 = 禁止 SPI3 模块  
0 = 使能 SPI3 模块

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 10-6: PMD7: 外设模块禁止控制寄存器 7

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	—	—	DMAMD	PTGMD	—	PGA1MD	—
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-12    **未实现:** 读为 0
- bit 11       **CMP4MD:** CMP4 模块禁止位  
1 = 禁止 CMP4 模块  
0 = 使能 CMP4 模块
- bit 10       **CMP3MD:** CMP3 模块禁止位  
1 = 禁止 CMP3 模块  
0 = 使能 CMP3 模块
- bit 9         **CMP2MD:** CMP2 模块禁止位  
1 = 禁止 CMP2 模块  
0 = 使能 CMP2 模块
- bit 8         **CMP1MD:** CMP1 模块禁止位  
1 = 禁止 CMP1 模块  
0 = 使能 CMP1 模块
- bit 7-5      **未实现:** 读为 0
- bit 4         **DMAMD:** DMA 模块禁止位  
1 = 禁止 DMA 模块  
0 = 使能 DMA 模块
- bit 3         **PTGMD:** PTG 模块禁止位  
1 = 禁止 PTG 模块  
0 = 使能 PTG 模块
- bit 2         **未实现:** 读为 0
- bit 1         **PGA1MD:** PGA1 模块禁止位  
1 = 禁止 PGA1 模块  
0 = 使能 PGA1 模块
- bit 0         **未实现:** 读为 0

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 10-7: PMD8: 外设模块禁止控制寄存器 8

U-0	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0
—	—	—	—	—	PGA2MD	—	—
bit 15						bit 8	

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	—	CLC4MD	CLC3MD	CLC2MD	CLC1MD	CCSMD	—
bit 7						bit 0	

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-11 **未实现:** 读为 0

bit 10 **PGA2MD:** PGA2 模块禁止位

1 = 禁止 PGA2 模块

0 = 使能 PGA2 模块

bit 9-6 **未实现:** 读为 0

bit 5 **CLC4MD:** CLC4 模块禁止位

1 = 禁止 CLC4 模块

0 = 使能 CLC4 模块

bit 4 **CLC3MD:** CLC3 模块禁止位

1 = 禁止 CLC3 模块

0 = 使能 CLC3 模块

bit 3 **CLC2MD:** CLC2 模块禁止位

1 = 禁止 CLC2 模块

0 = 使能 CLC2 模块

bit 2 **CLC1MD:** CLC1 模块禁止位

1 = 禁止 CLC1 模块

0 = 使能 CLC1 模块

bit 1 **CCSMD:** 恒流源模块禁止位

1 = 禁止恒流源模块

0 = 使能恒流源模块

bit 0 **未实现:** 读为 0

## 11.0 I/O 端口

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“[I/O 端口](#)”（DS70000598），该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的[第 4.0 节“存储器构成”](#)。

许多器件引脚由外设和并行 I/O 端口所共用。所有 I/O 输入端口都为施密特触发器输入，提高了抗噪声能力。

### 11.1 并行 I/O (PIO) 端口

与某个外设共用一个引脚的并行 I/O 端口通常服从于该外设。外设的输出缓冲器数据和控制信号提供给一对多路开关。这对多路开关用于选择 I/O 引脚的输出数据和控制信号是属于外设还是相应的端口。该逻辑电路同时会阻止“环通”（loop through，即一个端口的数字输

出可以驱动共用同一个引脚的外设的输入）。[图 11-1](#) 显示了端口是如何与其他外设复用的以及与这些外设连接的相关 I/O 引脚。

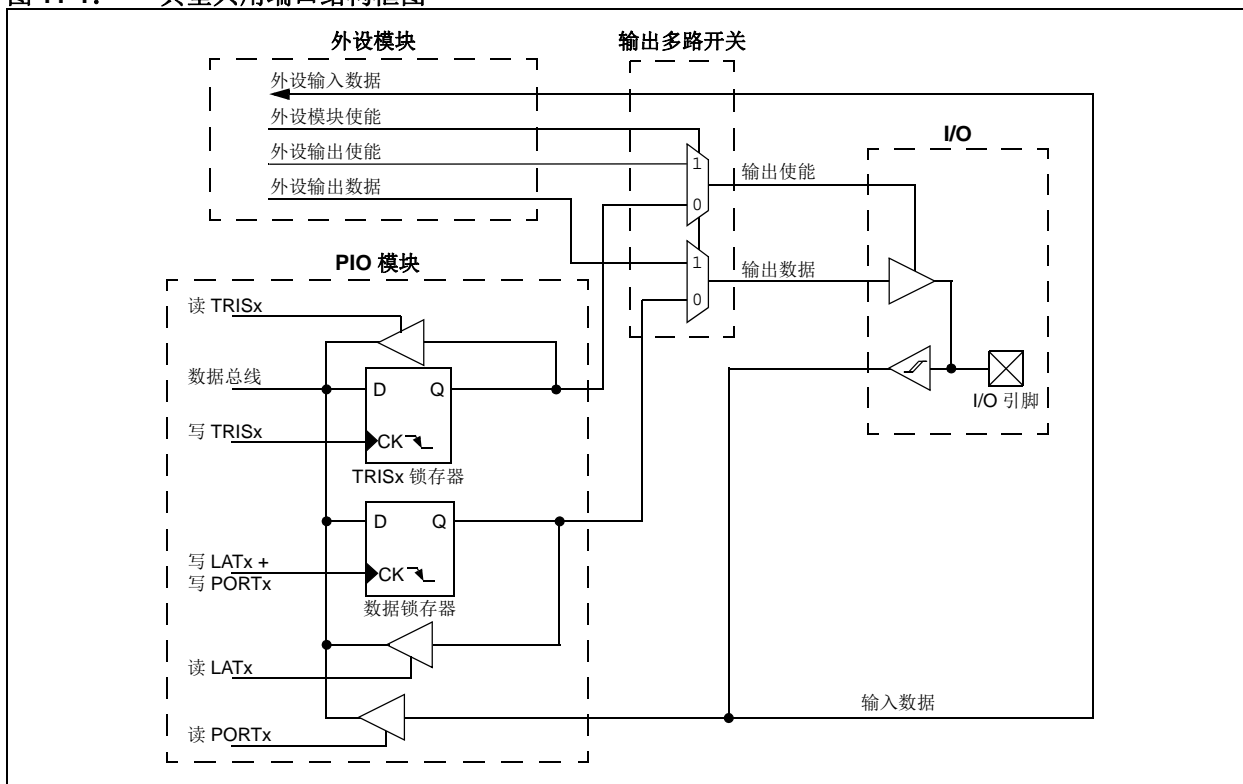
当外设使能，并且外设正在驱动相关引脚时，禁止将该引脚作为通用输出引脚。可以读该 I/O 引脚，但并行端口引脚的输出驱动器将被禁止。如果使能某外设但该外设没有驱动相应的引脚，则该引脚可由端口驱动。

所有端口引脚都有 8 个寄存器与其作为数字 I/O 时的操作直接相关。数据方向寄存器（TRISx）决定引脚是输入还是输出。如果数据方向位为 1，则引脚为输入。复位后，所有端口引脚均定义为输入。读锁存器（LATx）时，读到的是锁存器中的值；写锁存器时，写入的是锁存器。读端口（PORTx）时，读到的是端口引脚的值；而写端口引脚时，写入的是锁存器。

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。[表 11-1](#) 至 [表 11-5](#) 说明了对于每种器件型号，ANSELx 位的可用性。

当端口引脚与另一个外设共用或与定义为仅输入的功能共用时，它将被视为专用端口，因为没有任何其他的竞争输出源。

**图 11-1: 典型共用端口结构框图**



# dsPIC33EPXXGS70X/80X 系列

表 11-1: PORTA 引脚和 ANSELA 的可用性

器件	PORTA I/O 引脚															
	RA15	RA14	RA13	RA12	RA11	RA10	RA9	RA8	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0
dsPIC33EPXXGSX08	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXGSX06	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXGSX05	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXGSX04	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
dsPIC33EPXXGS702	—	—	—	—	—	—	—	—	—	—	—	X	X	X	X	X
ANSELA 位存在	—	—	—	—	—	—	—	—	—	—	—	—	—	X	X	X

表 11-2: PORTB 引脚和 ANSELB 的可用性

器件	PORTB I/O 引脚															
	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
dsPIC33EPXXGSX08	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX06	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX05	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX04	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGS702	X	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X
ANSELB 位存在	—	—	—	—	—	—	X	—	X	X	X	—	X	X	X	X

表 11-3: PORTC 引脚和 ANSEL C 的可用性

器件	PORTC I/O 引脚															
	RC15	RC14	RC13	RC12	RC11	RC10	RC9	RC8	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
dsPIC33EPXXGSX08	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX06	X	X	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX05	—	—	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX04	—	—	X	X	—	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGS702	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ANSEL C 位存在	—	—	—	X	—	X	X	—	—	X	X	X	—	X	X	—

表 11-4: PORTD 引脚和 ANSEL D 的可用性

器件	PORTD I/O 引脚															
	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
dsPIC33EPXXGSX08	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX06	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX05	—	X	—	—	—	X	—	—	—	—	—	X	—	—	—	—
dsPIC33EPXXGSX04	—	X	—	—	—	X	—	—	—	—	—	X	—	—	—	—
dsPIC33EPXXGS702	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ANSEL D 位存在	—	—	X	—	—	—	—	X	X	—	X	—	—	X	—	—

表 11-5: PORTE 引脚和 ANSELE 的可用性

器件	PORTE I/O 引脚															
	RE15	RE14	RE13	RE12	RE11	RE10	RE9	RE8	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0
dsPIC33EPXXGSX08	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
dsPIC33EPXXGSX06	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
dsPIC33EPXXGSX05	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
dsPIC33EPXXGSX04	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
dsPIC33EPXXGS702	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ANSELE 位存在	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

## 11.2 I/O 端口控制寄存器映射

表 11-6: PORTA 寄存器映射<sup>(1)</sup>

寄存器名称	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
TRISA	—	—	—	—	—	—	—	—	—	—	—	TRISA<4:0>				
PORTA	—	—	—	—	—	—	—	—	—	—	—	RA<4:0>				
LATA	—	—	—	—	—	—	—	—	—	—	—	LATA<4:0>				
ODCA	—	—	—	—	—	—	—	—	—	—	—	ODCA<4:0>				
CNENA	—	—	—	—	—	—	—	—	—	—	—	CNIEA<4:0>				
CNPUA	—	—	—	—	—	—	—	—	—	—	—	CNPUA<4:0>				
CNPDA	—	—	—	—	—	—	—	—	—	—	—	CNPDA<4:0>				
ANSELA	—	—	—	—	—	—	—	—	—	—	—	—	—	ANSA<2:0>		

图注: — = 未实现, 读为 0。

注 1: 关于位在每种引脚数的器件型号上的可用性, 请参见表 11-1。

表 11-7: PORTB 寄存器映射<sup>(1)</sup>

寄存器名称	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
TRISB	TRISB<15:11>					—	TRISB<9:0>									
PORTB	RB<15:11>					—	RB<9:0>									
LATB	LATB<15:11>					—	LATB<9:0>									
ODCB	ODCB<15:11>					—	ODCB<9:0>									
CNENB	CNIEB<15:11>					—	CNIEB<9:0>									
CNPUB	CNPUB<15:11>					—	CNPUB<9:0>									
CNPDB	CNPDB<15:11>					—	CNPDB<9:0>									
ANSELB	—	—	—	—	—	—	ANSB9	—	ANSB<7:5>			—	ANSB<3:0>			

图注: — = 未实现, 读为 0。

注 1: 关于位在每种引脚数的器件型号上的可用性, 请参见表 11-2。

表 11-8: PORTC 寄存器映射<sup>(1)</sup>

寄存器名称	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
TRISC	TRISC<15:12>				—	TRISC<10:0>										
PORTC	RC<15:12>				—	RC<10:0>										
LATC	LATC<15:12>				—	LATC<10:0>										
ODCC	ODCC<15:12>				—	ODCC<10:0>										
CNENC	CNIEC<15:12>				—	CNIEC<10:0>										
CNPUC	CNPUC<15:12>				—	CNPUC<10:0>										
CNPDC	CNPDC<15:12>				—	CNPDC<10:0>										
ANSEL	—	—	—	ANSC12	—	ANSC<10:9>		—	—	ANSC<6:4>			—	ANSC<2:1>		—

图注: — = 未实现, 读为 0。

注 1: 关于位在每种引脚数的器件型号上的可用性, 请参见表 11-3。

表 11-9: PORTD 寄存器映射<sup>(1)</sup>

寄存器名称	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
TRISD	TRISD<15:0>															
PORTD	RD<15:0>															
LATD	LATD<15:0>															
ODCD	ODCD<15:0>															
CNEND	CNIED<15:0>															
CNPUD	CNPUD<15:0>															
CNPDD	CNPDD<15:0>															
ANSEL	—	—	ANSD13	—	—	—	—	ANSD<8:7>		—	ANSD5	—	—	ANSD2	—	—

图注: — = 未实现, 读为 0。

注 1: 关于位在每种引脚数的器件型号上的可用性, 请参见表 11-4。



表 11-10: PORTE 寄存器映射<sup>(1)</sup>

寄存器名称	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
TRISE	TRISE<15:0>															
PORTE	RE<15:0>															
LATE	LATE<15:0>															
ODCE	ODCE<15:0>															
CNENE	CNIEE<15:0>															
CNPUE	CNPUE<15:0>															
CNPDE	CNPDE<15:0>															
ANSELE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

图注: — = 未实现, 读为 0。

注 1: 关于位在每种引脚数的器件型号上的可用性, 请参见表 11-5。

# dsPIC33EPXXGS70X/80X 系列

## 11.2.1 漏极开路配置

除 PORTx、LATx 和 TRISx 寄存器用于数据控制外，端口引脚也可被单独地配置为数字输出或漏极开路输出。这是由与每个端口相关的漏极开路控制寄存器 ODCx 控制的。将其中的任何位置 1 即可将相应的引脚配置为漏极开路输出。

漏极开路特性允许通过使用外部上拉电阻，产生除 VDD 之外的输出。任何引脚上允许的最大漏极开路电压与该特定引脚的最大 VIH 规范相同。关于可用的 5V 耐压引脚，请参见“引脚图”部分；关于每个引脚的最大 VIH 规范，请参见表 30-11。

## 11.3 配置模拟和数字端口引脚

ANSELx 寄存器用于控制模拟端口引脚的操作。如果要将端口引脚用作模拟输入或输出，则相应的 ANSELx 和 TRISx 位必须置 1。要将端口引脚用于数字模块（例如，定时器和 UART 等）的 I/O 功能，相应的 ANSELx 位必须清零。

ANSELx 寄存器的默认值为 0xFFFF。因此，在默认情况下，所有共用模拟功能的引脚都是模拟（非数字）引脚。

引脚 I/O 说明（见表 1-1）中列出了模拟功能受 ANSELx 寄存器影响的引脚，其缓冲器类型为模拟。表 11-1 至表 11-5 说明了对于每种器件型号，ANSELx 位的可用性。

如果 TRISx 位清零（输出），而 ANSELx 位置 1，则会通过一个模拟外设（例如，ADC 模块或比较器模块）转换数字输出电平（VOH 或 VOL）。

当读取 PORTx 寄存器时，所有配置为模拟输入通道的引脚均读为零（低电平）。

配置为数字输入的引脚将不会对模拟输入进行转换。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。

## 11.3.1 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间隔一个指令周期。通常该指令是 NOP 指令，如例 11-1 所示。

## 11.4 输入电平变化通知（ICN）

I/O 端口的输入电平变化通知功能允许器件在选定输入引脚的状态变化（Change-Of-State, COS）时，向处理器发出中断请求。该特性可检测输入状态变化，即使在休眠模式下、禁止时钟时。每个 I/O 端口引脚都可以选择（使能）为在发生状态变化时产生中断请求。

有 3 个控制寄存器与每个 I/O 端口的 ICN 功能关联。CNENx 寄存器包含每个输入引脚的 ICN 中断允许控制位。将其中任一位置 1 将允许相应引脚的 ICN 中断。

每个 I/O 引脚都有与之相连的一个弱上拉和一个弱下拉。上拉和下拉充当连接到引脚的拉电流或灌电流源，当连接按钮或键盘设备时，不再需要使用外部电阻。可使用包含每个引脚控制位的 CNPux 和 CNPDx 寄存器分别使能各个上拉和下拉。将任一控制位置 1 均可使能相应引脚的弱上拉和 / 或弱下拉功能。

**注：**当端口引脚被配置为数字输出时，输入电平变化通知引脚上的上拉和下拉应始终被禁止。

### 例 11-1： 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
```

# dsPIC33EPXXGS70X/80X 系列

## 11.5 I/O 端口控制寄存器

### 寄存器 11-1: TRISx: PORTx 数据方向控制寄存器 <sup>(1)</sup>

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISx<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISx<7:0>							
bit 7							bit 0

#### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        **TRISx<15:0>**: PORTx 数据方向控制位  
                  1 = 引脚为输入  
                  0 = 引脚为输出

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

### 寄存器 11-2: PORTx: I/O PORTx 寄存器 <sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PORTx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PORTx<7:0>							
bit 7							bit 0

#### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        **PORTx<15:0>**: I/O PORTx 位  
                  1 = 引脚数据为 1  
                  0 = 引脚数据为 0

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-3: LATx: PORTx 数据锁存器寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LATx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LATx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        **LATx<15:0>**: PORTx 数据锁存器位  
1 = 锁存器内容为 1  
0 = 锁存器内容为 0

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

## 寄存器 11-4: ODCx: PORTx 漏极开路控制寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ODCx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ODCx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        **PORTx<15:0>**: PORTx 漏极开路控制位  
1 = 如果 TRISx 为 0, 则引脚用作漏极开路输出引脚  
0 = 引脚用作常规引脚

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-5: CNENx: 输入电平变化通知中断允许 x 寄存器 <sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNIE <sub>x&lt;15:8&gt;</sub>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNIE <sub>x&lt;7:0&gt;</sub>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8      **CNIE<sub>x<15:0></sub>**: 输入电平变化通知中断允许 x 位  
 1 = 允许输入电平变化中断  
 0 = 禁止输入电平变化中断

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

## 寄存器 11-6: CNPUx: 输入电平变化通知上拉使能 x 寄存器 <sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPU <sub>x&lt;15:8&gt;</sub>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPU <sub>x&lt;7:0&gt;</sub>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8      **CNPU<sub>x<15:0></sub>**: 输入电平变化通知上拉使能位  
 1 = 使能 PORT<sub>x</sub> 引脚上的上拉  
 0 = 禁止 PORT<sub>x</sub> 引脚上的上拉

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-7: CNPDx: 输入电平变化通知下拉使能 x 寄存器 <sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPDx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNPDx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8      **CNPDx<15:0>**: 输入电平变化通知下拉使能 x 位  
1 = 使能 PORTx 引脚上的下拉  
0 = 禁止 PORTx 引脚上的下拉

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

## 寄存器 11-8: ANSELx: 模拟选择控制 x 寄存器 <sup>(1)</sup>

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANSx<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANSx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8      **ANSx<15:0>**: 模拟 PORTx 使能位  
1 = 使能模拟 PORTx 引脚  
0 = 使能数字 PORTx 引脚

注 1: 关于该寄存器中各个位的可用性, 请参见表 11-1、表 11-2、表 11-3、表 11-4 和表 11-5。

## 11.6 外设引脚选择 (PPS)

通用器件的主要挑战是提供尽可能多的外设功能，同时将其与 I/O 引脚功能的冲突降到最小。在低引脚数器件上，这一挑战更为严峻。在需要多个外设复用同一个引脚的应用中，要在应用程序代码中进行变通比较困难，换句话说彻底重新设计可能是唯一的选择。

外设引脚选择配置提供了这些选择的替代方法，使得用户可以在较宽的 I/O 引脚范围内选择和配置外设功能。通过增加特定器件上可用的引脚配置选项，用户可以让他们更好地适合他们的整个应用，而不必通过修改应用来适应器件。

外设引脚选择配置功能对固定的一部分数字 I/O 引脚进行操作。用户可以将大多数数字外设的输入和 / 或输出独立地映射到这些 I/O 引脚中的任何一个。一旦建立外设映射，就同时包含了硬件保护，以防止对外设映射的意外或错误更改。

### 11.6.1 可用的引脚

可用引脚的数目取决于特定器件及其引脚数。支持外设引脚选择功能的引脚在它们的引脚全称中包含标记“RPn”，其中“n”是可重映射的引脚编号。“RP”用于表示支持可重映射输入和输出功能的引脚。

### 11.6.2 可用的外设

外设引脚选择管理的外设都是仅用作数字功能的外设。这些外设包括一般串行通信 (UART 和 SPI)、通用定时器时钟输入、与定时器相关的外设 (输入捕捉和输出比较) 以及电平变化中断输入。

相比较而言，一些仅数字功能的外设模块决不能使用外设引脚选择功能。这是因为这类外设功能需要特定端口上的特殊 I/O 电路，且不能很容易地连接到多个引脚。其中一个示例是 I<sup>2</sup>C 模块。类似的要求排除了所有带模拟输入的模块，例如 ADC 转换器。

可重映射和不可重映射外设之间的主要差异在于可重映射外设与默认的 I/O 引脚无关。必须始终在使用外设前将其分配给特定的 I/O 引脚。相反，不可重映射外设始终可使用默认引脚 (假设该外设有效且与其他外设没有冲突)。

当给定 I/O 引脚上的可重映射外设有效时，它的优先级高于与该引脚相关的所有其他数字 I/O 和数字通信外设。优先级与被映射外设的类型无关。可重映射外设的优先级永远不会高于与该引脚相关的任何模拟功能。

### 11.6.3 控制外设引脚选择

外设引脚选择功能通过两组 SFR 控制：一组映射外设输入，另一组映射输出。由于输入和输出是单独控制的，因此可以不受限制地将特定外设的输入和输出 (如果外设同时具有输入和输出) 配置在任何可选择的功能引脚上。

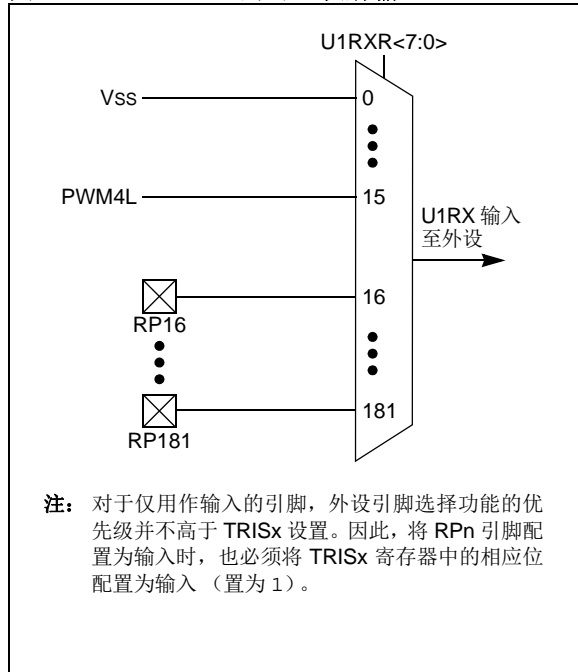
外设与外设可选择引脚之间的关系用两种不同的方式进行处理，这取决于被映射的是输入还是输出。

# dsPIC33EPXXGS70X/80X 系列

## 11.6.4 输入映射

外设引脚选择选项的输入在外设基础上进行映射。即，与外设相关的控制寄存器指示要映射到的引脚。RPINRx 寄存器用于配置外设输入映射（见寄存器 11-9 至寄存器 11-32）。每个寄存器包含 8 位位域组，每组都与可重映射外设之一相关。用适当的 8 位索引值编程给定外设的位域，会将具有对应值或内部信号的 RPn 引脚映射到该外设。关于可用输入的列表，请参见表 11-11。例如，图 11-2 给出了 U1RX 输入的可重映射引脚选择。

图 11-2: U1RX 的可重映射输入



## 11.6.4.1 虚拟连接

dsPIC33EPXXGS70X/80X 器件支持 6 个虚拟 RPn 引脚（RP176-RP181），除引脚分配外，它们与所有其他 RPn 引脚的功能相同。这 6 个引脚位于器件内部，它们不与物理器件引脚连接。

这些引脚提供了一种简单方式来进行外设间的连接，而无需使用物理引脚。例如，模拟比较器的输出可以连接到 RP176，同时可以将 PWM 故障输入配置到 RP176。该配置使模拟比较器无需使用器件上的实际物理引脚就可以触发 PWM 故障。

表 11-11: 可重映射源

重映射索引	输出功能
0	Vss
1	CMP1
2	CMP2
3	CMP3
4	CMP4
5	PWM4H
6	PTGO30
7	PTGO31
8-11	保留
12	REFO
13	SYNCO1
14	SYNCO2
15	PWM4L
16-20	RP16-RP20
21-31	保留
32-41	RP32-RP41
42	保留
43-58	RP43-RP58
59	保留
60-76	RP60-RP76
77-175	保留
176-181	RP176-RP181



# dsPIC33EPXXGS70X/80X 系列

表 11-12: 可选的输入源 (将输入映射到功能)

输入名称 <sup>(1)</sup>	功能名称	寄存器	配置位
外部中断 1	INT1	RPINR0	INT1R<7:0>
外部中断 2	INT2	RPINR1	INT2R<7:0>
Timer1 外部时钟	T1CK	RPINR2	T1CKR<7:0>
Timer2 外部时钟	T2CK	RPINR3	T2CKR<7:0>
Timer3 外部时钟	T3CK	RPINR3	T3CKR<7:0>
输入捕捉 1	IC1	RPINR7	IC1R<7:0>
输入捕捉 2	IC2	RPINR7	IC2R<7:0>
输入捕捉 3	IC3	RPINR8	IC3R<7:0>
输入捕捉 4	IC4	RPINR8	IC4R<7:0>
输出比较故障 A	OCFA	RPINR11	OCFAR<7:0>
PWM 故障 1	FLT1	RPINR12	FLT1R<7:0>
PWM 故障 2	FLT2	RPINR12	FLT2R<7:0>
PWM 故障 3	FLT3	RPINR13	FLT3R<7:0>
PWM 故障 4	FLT4	RPINR13	FLT4R<7:0>
UART1 接收	U1RX	RPINR18	U1RXR<7:0>
UART1 允许发送	U1CTS	RPINR18	U1CTSR<7:0>
UART2 接收	U2RX	RPINR19	U2RXR<7:0>
UART2 允许发送	U2CTS	RPINR19	U2CTSR<7:0>
SPI1 数据输入	SDI1	RPINR20	SDI1R<7:0>
SPI1 时钟输入	SCK1	RPINR20	SCK1R<7:0>
SPI1 从选择	SS1	RPINR21	SS1R<7:0>
CAN1 接收	C1RX	RPINR26	C1RXR<7:0>
CAN2 接收	C2RX	RPINR26	C2RXR<7:0>
SPI3 数据输入	SDI3	RPINR29	SDI3R<7:0>
SPI3 时钟输入	SCK3	RPINR29	SCK3R<7:0>
SPI3 从选择	SS3	RPINR30	SS3R<7:0>
SPI2 数据输入	SDI2	RPINR22	SDI2R<7:0>
SPI2 时钟输入	SCK2	RPINR22	SCK2R<7:0>
SPI2 从选择	SS2	RPINR23	SS2R<7:0>
PWM 同步输入 1	SYNCI1	RPINR37	SYNCI1R<7:0>
PWM 同步输入 2	SYNCI2	RPINR38	SYNCI2R<7:0>
PWM 故障 5	FLT5	RPINR42	FLT5R<7:0>
PWM 故障 6	FLT6	RPINR42	FLT6R<7:0>
PWM 故障 7	FLT7	RPINR43	FLT7R<7:0>
PWM 故障 8	FLT8	RPINR43	FLT8R<7:0>
CLC 输入 A	CLCINA	RPINR45	CLCINAR<7:0>
CLC 输入 B	CLCINB	RPINR46	CLCINBR<7:0>

注 1: 除非另外声明, 否则所有输入均使用施密特触发器输入缓冲器。

# dsPIC33EPXXGS70X/80X 系列

## 11.6.5 输出映射

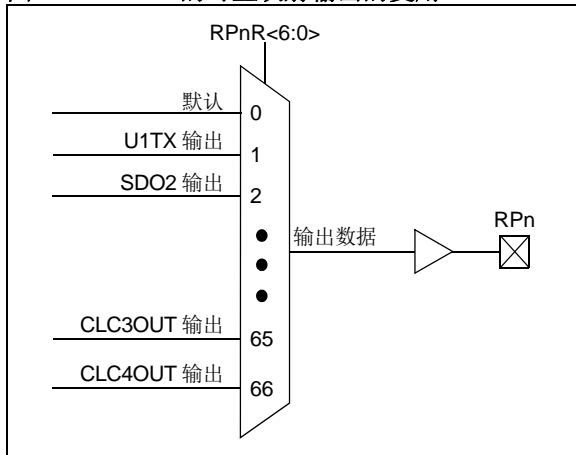
与输入不同，外设引脚选择选项的输出在引脚基础上进行映射。在这种情况下，与特定引脚相关的控制寄存器指示要映射的外设输出。RPORx 寄存器用于控制输出映射。每个寄存器包含 6 位位域组，每组都与一个 RPn 引脚相关（见寄存器 11-33 至寄存器 11-56）。位域的值与外设之一相对应，并且该外设的输出被映射到该引脚（见表 11-13 和图 11-3）。

空输出与输出寄存器的复位值 0 相关。这样做可确保在默认情况下，可重映射输出保持与所有输出引脚之间的断开状态。

### 11.6.5.1 映射限制

外设选择引脚的控制机制不局限于固定外配置的小范围内。在任何外设映射 SFR 之间没有互锁或硬件强制的锁定。也就是说，任何或所有 RPn 引脚上的外设映射的任何组合都是可能的。这包括外设输入和输出到引脚的多对一或一对多映射。从配置角度来看，这种映射在技术上是可行的，但从电气角度来看可能不受支持。

图 11-3: RPn 的可重映射输出的复用



# dsPIC33EPXXGS70X/80X 系列

表 11-13: 可重映射引脚 (RPn) 的输出选择

功能	RPnR<6:0>	输出名称
默认端口	0000000	RPn 连接到默认引脚
U1TX	0000001	RPn 连接到 UART1 发送
U1RTS	0000010	RPn 连接到 UART1 请求发送
U2TX	0000011	RPn 连接到 UART2 发送
U2RTS	0000100	RPn 连接到 UART2 请求发送
SDO1	0000101	RPn 连接到 SPI1 数据输出
SCK1	0000110	RPn 连接到 SPI1 时钟输出
SS1	0000111	RPn 连接到 SPI1 从选择
SDO2	0001000	RPn 连接到 SPI2 数据输出
SCK2	0001001	RPn 连接到 SPI2 时钟输出
SS2	0001010	RPn 连接到 SPI2 从选择
C1TX	0001110	RPn 连接到 CAN1 发送
C2TX	0001111	RPn 连接到 CAN2 发送
OC1	0010000	RPn 连接到输出比较 1 输出
OC2	0010001	RPn 连接到输出比较 2 输出
OC3	0010010	RPn 连接到输出比较 3 输出
OC4	0010011	RPn 连接到输出比较 4 输出
ACMP1	0011000	RPn 连接到模拟比较器 1 输出
ACMP2	0011001	RPn 连接到模拟比较器 2 输出
ACMP3	0011010	RPn 连接到模拟比较器 3 输出
SDO3	0011111	RPn 连接到 SPI3 数据输出
SCK3	0100000	RPn 连接到 SPI3 时钟输出
SS3	0100001	RPn 连接到 SPI3 从选择
SYNCO1	0101101	RPn 连接到 PWM 主主控时基同步输出
SYNCO2	0101110	RPn 连接到 PWM 辅助主控时基同步输出
REFCLKO	0110001	RPn 连接到参考时钟输出
ACMP4	0110010	RPn 连接到模拟比较器 4 输出
PWM4H	0110011	RPn 连接到与 PWM 发生器 4 相关的 PWM 输出引脚
PWM4L	0110100	RPn 连接到与 PWM 发生器 4 相关的 PWM 输出引脚
PWM5H	0110101	RPn 连接到与 PWM 发生器 5 相关的 PWM 输出引脚
PWM5L	0110110	RPn 连接到与 PWM 发生器 5 相关的 PWM 输出引脚
PWM6H	0111001	RPn 连接到与 PWM 发生器 6 相关的 PWM 输出引脚
PWM6L	0111010	RPn 连接到与 PWM 发生器 6 相关的 PWM 输出引脚
PWM7H	0111011	RPn 连接到与 PWM 发生器 7 相关的 PWM 输出引脚
PWM7L	0111100	RPn 连接到与 PWM 发生器 7 相关的 PWM 输出引脚
PWM8H	0111101	RPn 连接到与 PWM 发生器 8 相关的 PWM 输出引脚
PWM8L	0111110	RPn 连接到与 PWM 发生器 8 相关的 PWM 输出引脚
CLC1OUT	0111111	RPn 连接到 CLC1 输出
CLC2OUT	1000000	RPn 连接到 CLC2 输出
CLC3OUT <sup>(1)</sup>	1000001	RPn 连接到 CLC3 输出
CLC4OUT <sup>(1)</sup>	1000010	RPn 连接到 CLC4 输出

注 1: PPS 输出仅在 dsPIC33EPXXGS702 (28 引脚) 器件上可用。

## 11.7 I/O 有用技巧

1. 在某些情况下，一些引脚（如表 30-11 的“注入电流”下所定义）与 VDD 和 VSS 之间存在内部保护二极管。“注入电流”一词也可称为“钳位电流”。在指定引脚上，在用户采取了足够外部限流措施的情况下，允许 I/O 引脚输入电压大于或小于数据手册中的绝对最大值（相对于电源 VSS 和 VDD）。请注意，当用户应用对高端或低端内部输入钳位二极管进行正向偏置时，所产生的电流会注入内部钳位到 VDD 和 VSS 电源轨的器件，该电流可能影响 ADC 精度（4 至 6 个计数）。
2. 在发生任何复位之后，与任意模拟输入引脚复用的 I/O 引脚（即 ANx）默认情况下总是模拟引脚。因此，将某个引脚配置为模拟输入引脚时，将会自动禁止数字输入引脚缓冲器，如果尝试通过读取 PORTx 或 LATx 来读取数字输入电平，无论引脚上的数字逻辑电平为何，将总是返回 0。要在共用 ANx 引脚上使用引脚作为数字 I/O 引脚，用户应用程序需要配置 I/O 端口模块中的模拟引脚配置寄存器（即 ANSELx），方法是将对应于该 I/O 端口引脚的相应位设置为 0。

**注：** 虽然在使能数字输入引脚模拟功能时无法使用该数字输入引脚，但在使能模拟功能的情况下可以使用数字 I/O 输出功能（TRISx = 0x0）。但建议不要这样做，特别是如果模拟输入连接到外部模拟电压源，这会在模拟信号和输出引脚驱动器之间产生信号争用。

3. 大多数 I/O 引脚具有多种功能。请参见本数据手册中的器件引脚图，分配给任意引脚的功能的优先级按照由从左到右阅读引脚名称的方式来指示。在命名约定中，最左侧功能名称的优先级高于其右侧的所有功能。例如：AN16/T2CK/T7CK/RC1；这表明在该示例中 AN16 的优先级最高，将优先于列表中其右侧的所有其他功能。即使已使其右侧的那些其他功能，只要其左侧的任何功能已使能，右侧功能就不会起作用。该规则适用于对于给定引脚列出的所有功能。
4. 每个引脚都具有内部弱上拉电阻和弱下拉电阻，可分别使用 CNPUx 和 CNPDx 寄存器进行配置。由于具有这些电阻，在一些应用中可以不需要外部电阻。内部上拉可将电压最高拉至  $\sim(V_{DD} - 0.8)$ ，而不是 VDD。该值仍然高于 CMOS 和 TTL 器件的最小 V<sub>IH</sub>。
5. 直接驱动 LED 时，I/O 引脚的拉电流或灌电流可以高于 V<sub>OH</sub>/I<sub>OH</sub> 和 V<sub>OL</sub>/I<sub>OL</sub> 直流特性规范中规定的值。相应的 I<sub>OH</sub> 和 I<sub>OL</sub> 电流额定值只是为了使相应输出保持大于等于 V<sub>OH</sub> 和小于等于 V<sub>OL</sub> 电压。但对于 LED，不同于外部连接器件的数字输入，它们不受相同的最小 V<sub>IH</sub>/V<sub>IL</sub> 电压限制。I/O 引脚输出可以安全地灌入或拉出小于本数据手册第 30.0 节“电气特性”中的绝对最大值部分所列值的任何电流。例如：

$I_{OH} = -8 \text{ mA}$  且  $V_{DD} = 3.3\text{V}$  时  $V_{OH} = 2.4\text{V}$   
任意 8 mA I/O 引脚的最大输出拉电流 = 12 mA。  
从技术上说，允许 LED 拉电流 < 12 mA。更多信息，请参见第 31.0 节“器件直流和交流特性曲线图”中的 V<sub>OH</sub>/I<sub>OH</sub> 图。

## 6. 外设引脚选择（PPS）引脚映射规则如下：

- a) 在任意时刻，给定引脚上只能有一个“输出”功能处于活动状态，无论它是专用还是可重映射功能（一个引脚，一个输出）。
- b) 可以将某个“可重映射输出”功能分配给多个引脚，并在外部将它们短接或连接在一起，以提高电流驱动能力。
- c) 如果在某个引脚上使能了任何“专用输出”功能，它将优先于任何可重映射“输出”功能。
- d) 如果在某个引脚上使能了任何“专用数字”（输入或输出）功能，则可以将任意数量的“输入”可重映射功能映射到同一引脚。
- e) 如果在某个给定引脚上使能了任何“专用模拟”功能，则将禁止任何一种“数字输入”，但用户可以审慎地使能单个“数字输出”并使之处于活动状态，前提是它不会与外部模拟输入信号发生信号争用。例如，可以使用 ADC 转换数字输出逻辑电平，或翻转比较器上的数字输出或 ADC 输入，前提是没有类似用于内置自检的外部模拟输入。
- f) 可以同时将任意数量的“输入”可重映射功能映射到相同引脚，包括映射到具有来自专用或可重映射“输出”的单个输出的任意引脚。
- g) TRISx 寄存器仅控制数字 I/O 输出缓冲器。任何其他专用或可重映射的活动“输出”将自动改写 TRISx 设置。TRISx 寄存器不控制数字逻辑“输入”缓冲器。可重映射数字“输入”不会自动改写 TRISx 设置，这意味着对于仅分配有可重映射输入功能的引脚，必须将 TRISx 位设置为输入。
- h) 发生任意复位之后，默认情况下将使能所有模拟引脚，并且引脚上相应的数字输入缓冲器会被禁止。只有模拟引脚选择（ANSELx）寄存器会控制数字输入缓冲器，TRISx 寄存器不会。为了使用某个引脚上的任何“数字输入”，用户必须使用模拟引脚选择寄存器禁止相应引脚上的模拟功能，没有例外。

## 11.8 I/O 端口资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 11.8.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“**I/O 端口**”（DS70000598）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

## 11.9 外设引脚选择寄存器

### 寄存器 11-9: RPINR0: 外设引脚选择输入寄存器 0

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INT1R7	INT1R6	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

#### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8      **INT1R<7:0>**: 将外部中断 1 (INT1) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0      **未实现**: 读为 0

### 寄存器 11-10: RPINR1: 外设引脚选择输入寄存器 1

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INT2R7	INT2R6	INT2R5	INT2R4	INT2R3	INT2R2	INT2R1	INT2R0
bit 7							bit 0

#### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8      **未实现**: 读为 0

bit 7-0      **INT2R<7:0>**: 将外部中断 2 (INT2) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-11: RPINR2: 外设引脚选择输入寄存器 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T1CKR7	T1CKR6	T1CKR5	T1CKR4	T1CKR3	T1CKR2	T1CKR1	T1CKR0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8      **T1CKR<7:0>**: 将 Timer1 外部时钟 (T1CK) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0      **未实现**: 读为 0

## 寄存器 11-12: RPINR3: 外设引脚选择输入寄存器 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T3CKR7	T3CKR6	T3CKR5	T3CKR4	T3CKR3	T3CKR2	T3CKR1	T3CKR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T2CKR7	T2CKR6	T2CKR5	T2CKR4	T2CKR3	T2CKR2	T2CKR1	T2CKR0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8      **T3CKR<7:0>**: 将 Timer3 外部时钟 (T3CK) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0      **T2CKR<7:0>**: 将 Timer2 外部时钟 (T2CK) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-13: RPINR7: 外设引脚选择输入寄存器 7

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC2R7	IC2R6	IC2R5	IC2R4	IC2R3	IC2R2	IC2R1	IC2R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC1R7	IC1R6	IC1R5	IC1R4	IC1R3	IC1R2	IC1R1	IC1R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-8      **IC2R<7:0>**: 将输入捕捉 2 (IC2) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0        **IC1R<7:0>**: 将输入捕捉 1 (IC1) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-14: RPINR8: 外设引脚选择输入寄存器 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC4R7	IC4R6	IC4R5	IC4R4	IC4R3	IC4R2	IC4R1	IC4R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC3R7	IC3R6	IC3R5	IC3R4	IC3R3	IC3R2	IC3R1	IC3R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-8      **IC4R<7:0>**: 将输入捕捉 4 (IC4) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0        **IC3R<7:0>**: 将输入捕捉 3 (IC3) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-15: RPINR11: 外设引脚选择输入寄存器 11

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OCFAR7	OCFAR6	OCFAR5	OCFAR4	OCFAR3	OCFAR2	OCFAR1	OCFAR0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            **未实现:** 读为 0  
 bit 7-0            **OCFAR<7:0>:** 将输出比较故障 A (OCFA) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-16: RPINR12: 外设引脚选择输入寄存器 12

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT2R7	FLT2R6	FLT2R5	FLT2R4	FLT2R3	FLT2R2	FLT2R1	FLT2R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT1R7	FLT1R6	FLT1R5	FLT1R4	FLT1R3	FLT1R2	FLT1R1	FLT1R0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            **FLT2R<7:0>:** 将 PWM 故障 2 (FLT2) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。  
 bit 7-0            **FLT1R<7:0>:** 将 PWM 故障 1 (FLT1) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-17: RPINR13: 外设引脚选择输入寄存器 13

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT4R7	FLT4R6	FLT4R5	FLT4R4	FLT4R3	FLT4R2	FLT4R1	FLT4R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT3R7	FLT3R6	FLT3R5	FLT3R4	FLT3R3	FLT3R2	FLT3R1	FLT3R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                            0 = 清零                            x = 未知

- bit 15-8      **FLT4R<7:0>**: 将 PWM 故障 4 (FLT4) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。
- bit 7-0      **FLT3R<7:0>**: 将 PWM 故障 3 (FLT3) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-18: RPINR18: 外设引脚选择输入寄存器 18

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U1CTSR7	U1CTSR6	U1CTSR5	U1CTSR4	U1CTSR3	U1CTSR2	U1CTSR1	U1CTSR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U1RXR7	U1RXR6	U1RXR5	U1RXR4	U1RXR3	U1RXR2	U1RXR1	U1RXR0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                            0 = 清零                            x = 未知

- bit 15-8      **U1CTSR<7:0>**: 将 UART1 允许发送 ( $\overline{U1CTS}$ ) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。
- bit 7-0      **U1RXR<7:0>**: 将 UART1 接收 (U1RX) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-19: RPINR19: 外设引脚选择输入寄存器 19

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2CTSR7	U2CTSR6	U2CTSR5	U2CTSR4	U2CTSR3	U2CTSR2	U2CTSR1	U2CTSR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2RXR7	U2RXR6	U2RXR5	U2RXR4	U2RXR3	U2RXR2	U2RXR1	U2RXR0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8      **U2CTSR<7:0>**: 将 UART2 允许发送 ( $\overline{U2CTS}$ ) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0      **U2RXR<7:0>**: 将 UART2 接收 (U2RX) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-20: RPINR20: 外设引脚选择输入寄存器 20

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SCK1INR7	SCK1INR6	SCK1INR5	SCK1INR4	SCK1INR3	SCK1INR2	SCK1INR1	SCK1INR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDI1R7	SDI1R6	SDI1R5	SDI1R4	SDI1R3	SDI1R2	SDI1R1	SDI1R0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8      **SCK1INR<7:0>**: 将 SPI1 时钟输入 (SCK1) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0      **SDI1R<7:0>**: 将 SPI1 数据输入 (SDI1) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-21: RPINR21: 外设引脚选择输入寄存器 21

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SS1R7	SS1R6	SS1R5	SS1R4	SS1R3	SS1R2	SS1R1	SS1R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-8        **未实现:** 读为 0

bit 7-0        **SS1R<7:0>:** 将 SPI1 从选择 ( $\overline{SS1}$ ) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-22: RPINR22: 外设引脚选择输入寄存器 22

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SCK2INR7	SCK2INR6	SCK2INR5	SCK2INR4	SCK2INR3	SCK2INR2	SCK2INR1	SCK2INR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDI2R7	SDI2R6	SDI2R5	SDI2R4	SDI2R3	SDI2R2	SDI2R1	SDI2R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-8        **SCK2INR<7:0>:** 将 SPI2 时钟输入 (SCK2) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0        **SDI2R<7:0>:** 将 SPI2 数据输入 (SDI2) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-23: RPINR23: 外设引脚选择输入寄存器 23

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SS2R7	SS2R6	SS2R5	SS2R4	SS2R3	SS2R2	SS2R1	SS2R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            **未实现:** 读为 0  
 bit 7-0            **SS2R<7:0>:** 将 SPI2 从选择 ( $\overline{SS2}$ ) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-24: RPINR26: 外设引脚选择输入寄存器 26

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C2RXR7	C2RXR6	C2RXR5	C2RXR4	C2RXR3	C2RXR2	C2RXR1	C2RXR0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C1RXR7	C1RXR6	C1RXR5	C1RXR4	C1RXR3	C1RXR2	C1RXR1	C1RXR0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            **C2RXR<7:0>:** 将 CAN2 接收 (C2RX) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。  
 bit 7-0            **C1RXR<7:0>:** 将 CAN1 接收 (C1RX) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-25: RPINR29: 外设引脚选择输入寄存器 29

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SCK3R7	SCK3R6	SCK3R5	SCK3R4	SCK3R3	SCK3R2	SCK3R1	SCK3R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDI3R7	SDI3R6	SDI3R5	SDI3R4	SDI3R3	SDI3R2	SDI3R1	SDI3R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        **SCK3R<7:0>**: 将 SPI3 时钟输入 (SCK3) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0         **SDI3R<7:0>**: 将 SPI3 数据输入 (SDI3) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-26: RPINR30: 外设引脚选择输入寄存器 30

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SS3R7	SS3R6	SS3R5	SS3R4	SS3R3	SS3R2	SS3R1	SS3R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        未实现: 读为 0

bit 7-0         **SS3R<7:0>**: 将 SPI3 从选择 ( $\overline{SS3}$ ) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

**寄存器 11-27: RPINR37: 外设引脚选择输入寄存器 37**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCI1R7	SYNCI1R6	SYNCI1R5	SYNCI1R4	SYNCI1R3	SYNCI1R2	SYNCI1R1	SYNCI1R0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        **SYNCI1R<7:0>**: 将 PWM 同步输入 1 (SYNCI1) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0        **未实现**: 读为 0

**寄存器 11-28: RPINR38: 外设引脚选择输入寄存器 38**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCI2R7	SYNCI2R6	SYNCI2R5	SYNCI2R4	SYNCI2R3	SYNCI2R2	SYNCI2R1	SYNCI2R0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8        **未实现**: 读为 0

bit 7-0        **SYNCI2R<7:0>**: 将 PWM 同步输入 2 (SYNCI2) 分配给对应 RPn 引脚的位  
 索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-29: RPINR42: 外设引脚选择输入寄存器 42

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT6R7	FLT6R6	FLT6R5	FLT6R4	FLT6R3	FLT6R2	FLT6R1	FLT6R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT5R7	FLT5R6	FLT5R5	FLT5R4	FLT5R3	FLT5R2	FLT5R1	FLT5R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                            0 = 清零                            x = 未知

- bit 15-8      **FLT6R<7:0>**: 将 PWM 故障 6 (FLT6) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。
- bit 7-0      **FLT5R<7:0>**: 将 PWM 故障 5 (FLT5) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。

## 寄存器 11-30: RPINR43: 外设引脚选择输入寄存器 43

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT8R7	FLT8R6	FLT8R5	FLT8R4	FLT8R3	FLT8R2	FLT8R1	FLT8R0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLT7R7	FLT7R6	FLT7R5	FLT7R4	FLT7R3	FLT7R2	FLT7R1	FLT7R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                            0 = 清零                            x = 未知

- bit 15-8      **FLT8R<7:0>**: 将 PWM 故障 8 (FLT8) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。
- bit 7-0      **FLT7R<7:0>**: 将 PWM 故障 7 (FLT7) 分配给对应 RPn 引脚的位索引值请参见表 11-11, 其中包含了可重映射输入的列表。



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-31: RPINR45: 外设引脚选择输入寄存器 45

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CLCINAR7	CLCINAR6	CLCINAR5	CLCINAR4	CLCINAR3	CLCINAR2	CLCINAR1	CLCINAR0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8      **CLCINAR<7:0>**: 将 CLC 输入 A (CLCINA) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

bit 7-0      **未实现**: 读为 0

## 寄存器 11-32: RPINR46: 外设引脚选择输入寄存器 46

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CLCINBR7	CLCINBR6	CLCINBR5	CLCINBR4	CLCINBR3	CLCINBR2	CLCINBR1	CLCINBR0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8      **未实现**: 读为 0

bit 7-0      **CLCINBR<7:0>**: 将 CLC 输入 B (CLCINB) 分配给对应 RPn 引脚的位  
索引值请参见表 11-11, 其中包含了可重映射输入的列表。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-33: RPOR0: 外设引脚选择输出寄存器 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP17R6	RP17R5	RP17R4	RP17R3	RP17R2	RP17R1	RP17R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP16R6	RP16R5	RP16R4	RP16R3	RP16R2	RP16R1	RP16R0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 未实现: 读为 0

bit 14-8 **RP17R<6:0>**: 将外设输出功能分配给 RP17 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

bit 7 未实现: 读为 0

bit 6-0 **RP16R<6:0>**: 将外设输出功能分配给 RP16 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-34: RPOR1: 外设引脚选择输出寄存器 1

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP19R6	RP19R5	RP19R4	RP19R3	RP19R2	RP19R1	RP19R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP18R6	RP18R5	RP18R4	RP18R3	RP18R2	RP18R1	RP18R0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 未实现: 读为 0

bit 14-8 **RP19R<6:0>**: 将外设输出功能分配给 RP19 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

bit 7 未实现: 读为 0

bit 6-0 **RP18R<6:0>**: 将外设输出功能分配给 RP18 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-35: RPOR2: 外设引脚选择输出寄存器 2

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP32R6	RP32R5	RP32R4	RP32R3	RP32R2	RP32R1	RP32R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP20R6	RP20R5	RP20R4	RP20R3	RP20R2	RP20R1	RP20R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP32R<6:0>:** 将外设输出功能分配给 RP32 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP20R<6:0>:** 将外设输出功能分配给 RP20 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-36: RPOR3: 外设引脚选择输出寄存器 3

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP34R6	RP34R5	RP34R4	RP34R3	RP34R2	RP34R1	RP34R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP33R6	RP33R5	RP33R4	RP33R3	RP33R2	RP33R1	RP33R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP34R<6:0>:** 将外设输出功能分配给 RP34 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP33R<6:0>:** 将外设输出功能分配给 RP33 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-37: RPOR4: 外设引脚选择输出寄存器 4

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP36R6	RP36R5	RP36R4	RP36R3	RP36R2	RP36R1	RP36R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP35R6	RP35R5	RP35R4	RP35R3	RP35R2	RP35R1	RP35R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP36R<6:0>:** 将外设输出功能分配给 RP36 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP35R<6:0>:** 将外设输出功能分配给 RP35 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-38: RPOR5: 外设引脚选择输出寄存器 5

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP38R6	RP38R5	RP38R4	RP38R3	RP38R2	RP38R1	RP38R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP37R6	RP37R5	RP37R4	RP37R3	RP37R2	RP37R1	RP37R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP38R<6:0>:** 将外设输出功能分配给 RP38 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP37R<6:0>:** 将外设输出功能分配给 RP37 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-39: RPOR6: 外设引脚选择输出寄存器 6

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP40R6	RP40R5	RP40R4	RP40R3	RP40R2	RP40R1	RP40R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP39R6	RP39R5	RP39R4	RP39R3	RP39R2	RP39R1	RP39R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP40R<6:0>:** 将外设输出功能分配给 RP40 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP39R<6:0>:** 将外设输出功能分配给 RP39 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-40: RPOR7: 外设引脚选择输出寄存器 7

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP43R6	RP43R5	RP43R4	RP43R3	RP43R2	RP43R1	RP43R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP41R6	RP41R5	RP41R4	RP41R3	RP41R2	RP41R1	RP41R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP43R<6:0>:** 将外设输出功能分配给 RP43 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP41R<6:0>:** 将外设输出功能分配给 RP41 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-41: RPOR8: 外设引脚选择输出寄存器 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP45R6	RP45R5	RP45R4	RP45R3	RP45R2	RP45R1	RP45R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP44R6	RP44R5	RP44R4	RP44R3	RP44R2	RP44R1	RP44R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                            0 = 清零                            x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP45R<6:0>:** 将外设输出功能分配给 RP45 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP44R<6:0>:** 将外设输出功能分配给 RP44 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-42: RPOR9: 外设引脚选择输出寄存器 9

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP47R6	RP47R5	RP47R4	RP47R3	RP47R2	RP47R1	RP47R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP46R6	RP46R5	RP46R4	RP46R3	RP46R2	RP46R1	RP46R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                            0 = 清零                            x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP47R<6:0>:** 将外设输出功能分配给 RP47 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP46R<6:0>:** 将外设输出功能分配给 RP46 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-43: RPOR10: 外设引脚选择输出寄存器 10

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP49R6	RP49R5	RP49R4	RP49R3	RP49R2	RP49R1	RP49R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP48R6	RP48R5	RP48R4	RP48R3	RP48R2	RP48R1	RP48R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP49R<6:0>:** 将外设输出功能分配给 RP49 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP48R<6:0>:** 将外设输出功能分配给 RP48 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-44: RPOR11: 外设引脚选择输出寄存器 11

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP51R6	RP51R5	RP51R4	RP51R3	RP51R2	RP51R1	RP51R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP50R6	RP50R5	RP50R4	RP50R3	RP50R2	RP50R1	RP50R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP51R<6:0>:** 将外设输出功能分配给 RP51 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP50R<6:0>:** 将外设输出功能分配给 RP50 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-45: RPOR12: 外设引脚选择输出寄存器 12

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP53R6	RP53R5	RP53R4	RP53R3	RP53R2	RP53R1	RP53R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP52R6	RP52R5	RP52R4	RP52R3	RP52R2	RP52R1	RP52R0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 未实现: 读为 0

bit 14-8 **RP53R<6:0>**: 将外设输出功能分配给 RP53 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

bit 7 未实现: 读为 0

bit 6-0 **RP52R<6:0>**: 将外设输出功能分配给 RP52 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-46: RPOR13: 外设引脚选择输出寄存器 13

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP55R6	RP55R5	RP55R4	RP55R3	RP55R2	RP55R1	RP55R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP54R6	RP54R5	RP54R4	RP54R3	RP54R2	RP54R1	RP54R0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 未实现: 读为 0

bit 14-8 **RP55R<6:0>**: 将外设输出功能分配给 RP55 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

bit 7 未实现: 读为 0

bit 6-0 **RP54R<6:0>**: 将外设输出功能分配给 RP54 输出引脚的位 (关于外设功能编号, 请参见表 11-13)



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-47: RPOR14: 外设引脚选择输出寄存器 14

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP57R6	RP57R5	RP57R4	RP57R3	RP57R2	RP57R1	RP57R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP56R6	RP56R5	RP56R4	RP56R3	RP56R2	RP56R1	RP56R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP57R<6:0>:** 将外设输出功能分配给 RP57 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP56R<6:0>:** 将外设输出功能分配给 RP56 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-48: RPOR15: 外设引脚选择输出寄存器 15

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP60R6	RP60R5	RP60R4	RP60R3	RP60R2	RP60R1	RP60R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP58R6	RP58R5	RP58R4	RP58R3	RP58R2	RP58R1	RP58R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP60R<6:0>:** 将外设输出功能分配给 RP60 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP58R<6:0>:** 将外设输出功能分配给 RP58 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-49: RPOR16: 外设引脚选择输出寄存器 16

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP62R6	RP62R5	RP62R4	RP62R3	RP62R2	RP62R1	RP62R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP61R6	RP61R5	RP61R4	RP61R3	RP61R2	RP61R1	RP61R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                        0 = 清零                        x = 未知

- bit 15            **未实现:** 读为 0  
bit 14-8        **RP62R<6:0>:** 将外设输出功能分配给 RP62 输出引脚的位 (关于外设功能编号, 请参见表 11-13)  
bit 7            **未实现:** 读为 0  
bit 6-0         **RP61R<6:0>:** 将外设输出功能分配给 RP61 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-50: RPOR17: 外设引脚选择输出寄存器 17

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP64R6	RP64R5	RP64R4	RP64R3	RP64R2	RP64R1	RP64R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP63R6	RP63R5	RP63R4	RP63R3	RP63R2	RP63R1	RP63R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                        0 = 清零                        x = 未知

- bit 15            **未实现:** 读为 0  
bit 14-8        **RP64R<6:0>:** 将外设输出功能分配给 RP64 输出引脚的位 (关于外设功能编号, 请参见表 11-13)  
bit 7            **未实现:** 读为 0  
bit 6-0         **RP63R<6:0>:** 将外设输出功能分配给 RP63 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-51: RPOR18: 外设引脚选择输出寄存器 18

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP66R6	RP66R5	RP66R4	RP66R3	RP66R2	RP66R1	RP66R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP65R6	RP65R5	RP65R4	RP65R3	RP65R2	RP65R1	RP65R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP66R<6:0>:** 将外设输出功能分配给 RP66 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP65R<6:0>:** 将外设输出功能分配给 RP65 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-52: RPOR19: 外设引脚选择输出寄存器 19

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP68R6	RP68R5	RP68R4	RP68R3	RP68R2	RP68R1	RP68R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP67R6	RP67R5	RP67R4	RP67R3	RP67R2	RP67R1	RP67R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP68R<6:0>:** 将外设输出功能分配给 RP68 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP67R<6:0>:** 将外设输出功能分配给 RP67 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

**寄存器 11-53: RPOR20: 外设引脚选择输出寄存器 20**

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP70R6	RP70R5	RP70R4	RP70R3	RP70R2	RP70R1	RP70R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP69R6	RP69R5	RP69R4	RP69R3	RP69R2	RP69R1	RP69R0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP70R<6:0>:** 将外设输出功能分配给 RP70 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP69R<6:0>:** 将外设输出功能分配给 RP69 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

**寄存器 11-54: RPOR21: 外设引脚选择输出寄存器 21**

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP72R6	RP72R5	RP72R4	RP72R3	RP72R2	RP72R1	RP72R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP71R6	RP71R5	RP71R4	RP71R3	RP71R2	RP71R1	RP71R0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP72R<6:0>:** 将外设输出功能分配给 RP72 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP71R<6:0>:** 将外设输出功能分配给 RP71 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-55: RPOR22: 外设引脚选择输出寄存器 22

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP74R6	RP74R5	RP74R4	RP74R3	RP74R2	RP74R1	RP74R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP73R6	RP73R5	RP73R4	RP73R3	RP73R2	RP73R1	RP73R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP74R<6:0>:** 将外设输出功能分配给 RP74 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP73R<6:0>:** 将外设输出功能分配给 RP73 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

## 寄存器 11-56: RPOR23: 外设引脚选择输出寄存器 23

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP76R6	RP76R5	RP76R4	RP76R3	RP76R2	RP76R1	RP76R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP75R6	RP75R5	RP75R4	RP75R3	RP75R2	RP75R1	RP75R0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP76R<6:0>:** 将外设输出功能分配给 RP76 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP75R<6:0>:** 将外设输出功能分配给 RP75 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

寄存器 11-57: RPOR24: 外设引脚选择输出寄存器 24

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP177R6	RP177R5	RP177R4	RP177R3	RP177R2	RP177R1	RP177R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP176R6	RP176R5	RP176R4	RP176R3	RP176R2	RP176R1	RP176R0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                        0 = 清零                        x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP177R<6:0>:** 将外设输出功能分配给 RP177 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP176R<6:0>:** 将外设输出功能分配给 RP176 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

寄存器 11-58: RPOR25: 外设引脚选择输出寄存器 25

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP179R6	RP179R5	RP179R4	RP179R3	RP179R2	RP179R1	RP179R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP178R6	RP178R5	RP178R4	RP178R3	RP178R2	RP178R1	RP178R0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                        0 = 清零                        x = 未知

- bit 15            **未实现:** 读为 0
- bit 14-8        **RP179R<6:0>:** 将外设输出功能分配给 RP179 输出引脚的位 (关于外设功能编号, 请参见表 11-13)
- bit 7            **未实现:** 读为 0
- bit 6-0         **RP178R<6:0>:** 将外设输出功能分配给 RP178 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 11-59: RPOR26: 外设引脚选择输出寄存器 26

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP181R6	RP181R5	RP181R4	RP181R3	RP181R2	RP181R1	RP181R0
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	RP180R6	RP180R5	RP180R4	RP180R3	RP180R2	RP180R1	RP180R0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15

**未实现:** 读为 0

bit 14-8

**RP181R<6:0>:** 将外设输出功能分配给 RP181 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

bit 7

**未实现:** 读为 0

bit 6-0

**RP180R<6:0>:** 将外设输出功能分配给 RP180 输出引脚的位 (关于外设功能编号, 请参见表 11-13)

# dsPIC33EPXXXGS70X/80X 系列

---

注:



## 12.0 Timer1

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“定时器” (DS70362)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

Timer1 模块是一个 16 位定时器，可作为自由运行的间隔定时器 / 计数器。

Timer1 模块具有不同于其他定时器的独特特性，如下：

- 可以依靠外部时钟源以异步计数器模式工作
- 可选择将外部时钟输入 (T1CK) 与内部器件时钟同步，时钟同步在预分频之后执行

图 12-1 给出了 Timer1 的框图。

Timer1 模块可以工作于以下模式之一：

- 定时器模式
- 门控定时器模式
- 同步计数器模式
- 异步计数器模式

在定时器和门控定时器模式下，输入时钟来自于内部指令周期时钟 (FCY)。在同步和异步计数器模式下，输入时钟来自于 T1CK 引脚上的外部时钟输入。

定时器模式由以下位决定：

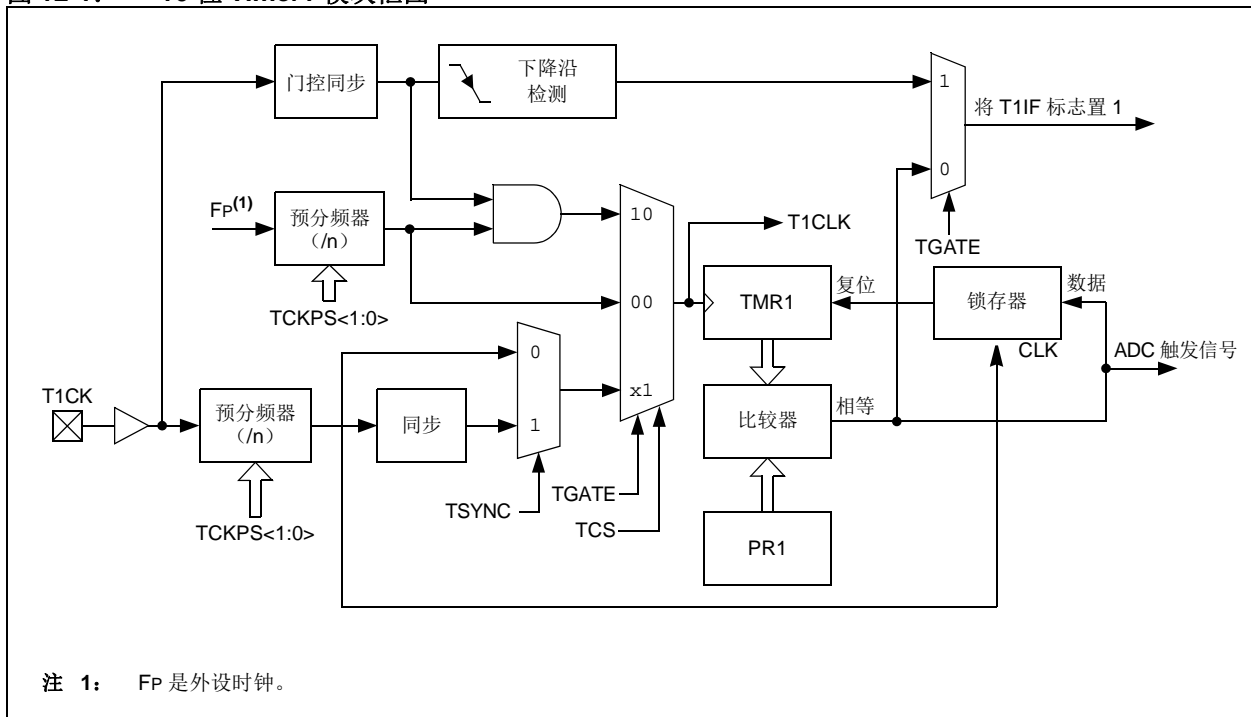
- Timer1 时钟源选择位 (TCS): T1CON<1>
- Timer1 外部时钟输入同步选择位 (TSYNC): T1CON<2>
- Timer1 门控时间累加使能位 (TGATE): T1CON<6>

表 12-1 给出了不同工作模式的定时器控制位的设置。

表 12-1: Timer1 模式设置

模式	TCS	TGATE	TSYNC
定时器	0	0	x
门控定时器	0	1	x
同步计数器	1	x	1
异步计数器	1	x	0

图 12-1: 16 位 Timer1 模块框图



# dsPIC33EPXXGS70X/80X 系列

---

## 12.1 Timer1 资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 12.1.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“定时器” (DS70362)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

## 12.2 Timer1 控制寄存器

寄存器 12-1: T1CON: Timer1 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON <sup>(1)</sup>	—	TSIDL	—	—	—	—	—
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	—	TSYNC <sup>(1)</sup>	TCS <sup>(1)</sup>	—
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15        **TON:** Timer1 使能位 <sup>(1)</sup>  
 1 = 启动 16 位 Timer1  
 0 = 停止 16 位 Timer1
  - bit 14        **未实现:** 读为 0
  - bit 13        **TSIDL:** Timer1 空闲模式停止位  
 1 = 当器件进入空闲模式时, 模块停止工作  
 0 = 在空闲模式下模块继续工作
  - bit 12-7     **未实现:** 读为 0
  - bit 6        **TGATE:** Timer1 门控时间累加使能位  
当 TCS = 1 时:  
 该位被忽略。  
当 TCS = 0 时:  
 1 = 使能门控时间累加  
 0 = 禁止门控时间累加
  - bit 5-4     **TCKPS<1:0>:** Timer1 输入时钟预分频比选择位  
 11 = 1:256  
 10 = 1:64  
 01 = 1:8  
 00 = 1:1
  - bit 3        **未实现:** 读为 0
  - bit 2        **TSYNC:** Timer1 外部时钟输入同步选择位 <sup>(1)</sup>  
当 TCS = 1 时:  
 1 = 同步外部时钟输入  
 0 = 不同步外部时钟输入  
当 TCS = 0 时:  
 该位被忽略。
  - bit 1        **TCS:** Timer1 时钟源选择位 <sup>(1)</sup>  
 1 = 来自 T1CK 引脚的外部时钟 (上升沿触发计数)  
 0 = 内部时钟 (FP)
  - bit 0        **未实现:** 读为 0
- 注 1:** 将 Timer1 使能为外部同步计数器模式 (TCS = 1, TSYNC = 1, TON = 1) 时, 用户软件试图对 TMR1 寄存器进行的任何写操作都会被忽略。

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 13.0 Timer2/3 和 Timer4/5

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本数据手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“定时器”（DS70362），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

Timer2/3 和 Timer4/5 模块为 32 位定时器，也可被配置为 4 个具有可选工作模式的独立 16 位定时器。

作为 32 位定时器，Timer2/3 和 Timer4/5 具有以下三种工作模式：

- 具有所有 16 位工作模式（异步计数器模式除外）的两个独立 16 位定时器（例如，Timer2 和 Timer3）
- 单个 32 位定时器
- 单个 32 位同步计数器

这些定时器还支持以下功能：

- 定时器门控操作
- 可选择的预分频比设置
- 空闲和休眠模式期间的定时器工作
- 在 32 位周期寄存器匹配时产生中断
- 输入捕捉和输出比较模块的时基（仅限 Timer2 和 Timer3）

所有 4 个 16 位定时器都能单独用作同步定时器或计数器。它们也提供上文所列的功能，但事件触发功能除外，它仅由 Timer2/3 实现。通过设置 T2CON、T3CON、T4CON 和 T5CON 寄存器中的相应位来确定工作模式和使能特性。T2CON 和 T4CON 的通用形式如寄存器 13-1 所示。T3CON 和 T5CON 如寄存器 13-2 所示。

对于 32 位定时器/计数器操作，Timer2 和 Timer4 是 32 位定时器的最低有效字（lsw）；Timer3 和 Timer5 则是 32 位定时器的最高有效字（msw）。

**注:** 对于 32 位操作，T3CON 和 T5CON 中的控制位将被忽略。设置和控制只使用 T2CON 和 T4CON 中的控制位。32 位定时器模块采用 Timer2 和 Timer4 的时钟和门控输入，但中断由 Timer3 和 Timer5 中断标志产生。

图 13-2 给出了示例 32 位定时器对（Timer2/3 和 Timer4/5）的框图。

## 13.1 定时器资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 13.1.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“定时器”（DS70362）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

图 13-1: TIMERx 框图 (x = 2 至 5)

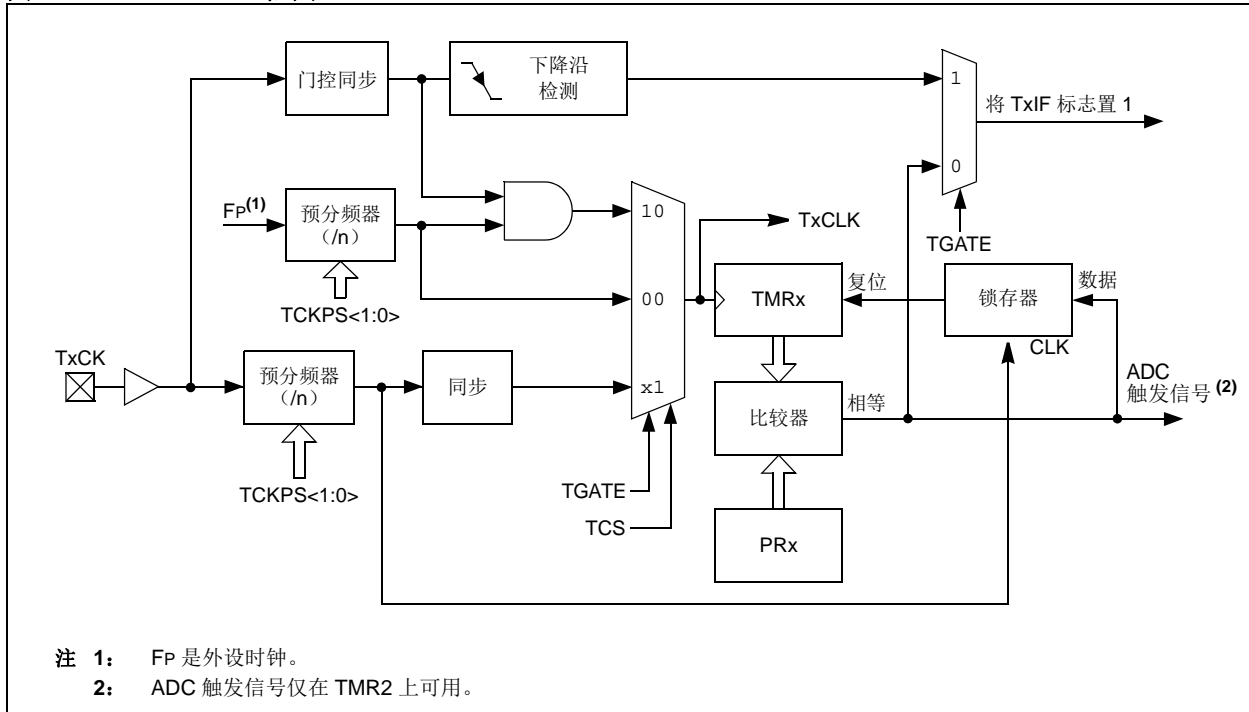
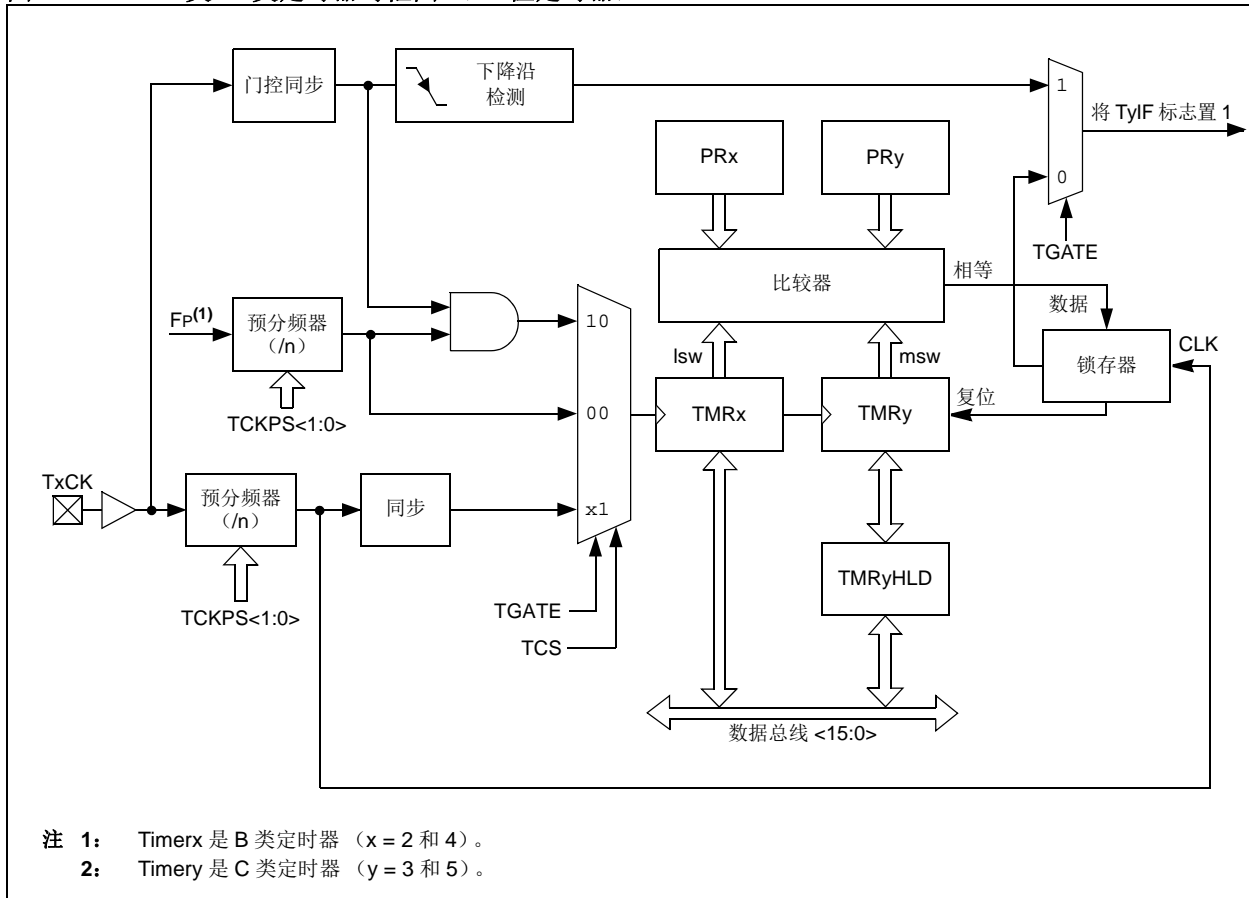


图 13-2: B 类 /C 类定时器对框图 (32 位定时器)



# dsPIC33EPXXGS70X/80X 系列

## 13.2 Timer2/3 和 Timer4/5 控制寄存器

寄存器 13-1: TxCON: (Timer2 和 Timer4) 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	T32	—	TCS <sup>(1)</sup>	—
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15     **TON:** Timerx 使能位  
           当 T32 = 1 时:  
           1 = 启动 32 位 Timerx/y  
           0 = 停止 32 位 Timerx/y  
           当 T32 = 0 时:  
           1 = 启动 16 位 Timerx  
           0 = 停止 16 位 Timerx
- bit 14     **未实现:** 读为 0
- bit 13     **TSIDL:** Timerx 空闲模式停止位  
           1 = 当器件进入空闲模式时, 模块停止工作  
           0 = 在空闲模式下模块继续工作
- bit 12-7   **未实现:** 读为 0
- bit 6     **TGATE:** Timerx 门控时间累加使能位  
           当 TCS = 1 时:  
           该位被忽略。  
           当 TCS = 0 时:  
           1 = 使能门控时间累加  
           0 = 禁止门控时间累加
- bit 5-4   **TCKPS<1:0>:** Timerx 输入时钟预分频比选择位  
           11 = 1:256  
           10 = 1:64  
           01 = 1:8  
           00 = 1:1
- bit 3     **T32:** 32 位定时器模式选择位  
           1 = Timerx 和 Timery 形成一个 32 位定时器  
           0 = Timerx 和 Timery 用作两个 16 位定时器
- bit 2     **未实现:** 读为 0
- bit 1     **TCS:** Timerx 时钟源选择位 <sup>(1)</sup>  
           1 = 来自 TxCK 引脚的外部时钟 (上升沿触发计数)  
           0 = 内部时钟 (FP)
- bit 0     **未实现:** 读为 0

**注 1:** TxCK 引脚并非在所有器件上都可用。关于可用引脚, 请参见 “[引脚图](#)” 部分。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 13-2: TyCON: (Timer3 和 Timer5) 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON <sup>(1)</sup>	—	TSIDL <sup>(2)</sup>	—	—	—	—	—
bit 15						bit 8	

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE <sup>(1)</sup>	TCKPS1 <sup>(1)</sup>	TCKPS0 <sup>(1)</sup>	—	—	TCS <sup>(1,3)</sup>	—
bit 7						bit 0	

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **TON:** Timery 使能位 <sup>(1)</sup>  
 1 = 启动 16 位 Timery  
 0 = 停止 16 位 Timery
- bit 14            **未实现:** 读为 0
- bit 13            **TSIDL:** Timery 空闲模式停止位 <sup>(2)</sup>  
 1 = 当器件进入空闲模式时, 模块停止工作  
 0 = 在空闲模式下模块继续工作
- bit 12-7        **未实现:** 读为 0
- bit 6            **TGATE:** Timery 门控时间累加使能位 <sup>(1)</sup>  
当 TCS = 1 时:  
 该位被忽略。  
当 TCS = 0 时:  
 1 = 使能门控时间累加  
 0 = 禁止门控时间累加
- bit 5-4        **TCKPS<1:0>:** Timery 输入时钟预分频比选择位 <sup>(1)</sup>  
 11 = 1:256  
 10 = 1:64  
 01 = 1:8  
 00 = 1:1
- bit 3-2        **未实现:** 读为 0
- bit 1            **TCS:** Timery 时钟源选择位 <sup>(1,3)</sup>  
 1 = 来自 TyCK 引脚的外部时钟 (上升沿触发计数)  
 0 = 内部时钟 (FP)
- bit 0            **未实现:** 读为 0

- 注 1:** 当使能 32 位操作 (TxCON<3> = 1) 时, 这些位对 Timery 的操作没有影响; 所有定时器功能都通过 TxCON 进行设置。
- 2:** 当在 Timerx 控制寄存器中使能 32 位定时器操作 (T32 (TxCON<3>) = 1) 时, TSIDL 位必须清零以在空闲模式下运行 32 位定时器。
- 3:** TyCK 引脚并非在所有器件上都可用。关于可用引脚, 请参见 “[引脚图](#)” 部分。



## 14.0 输入捕捉

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“使用专用定时器的输入捕捉”（DS70000352），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

输入捕捉模块在需要频率（周期）和脉冲测量的应用中很有用。dsPIC33EPXXGS70X/80X 器件支持 4 个输入捕捉通道。

输入捕捉模块的主要特性包括：

- 硬件可配置，可通过级联两个相邻的模块，配置为所有 32 位工作模式
- 输出比较操作有同步和触发两种模式，最多有 21 个用户可选择的触发 / 同步源可供使用
- 用于捕捉和保持几个事件的定时器值的 4 级 FIFO 缓冲区
- 可配置中断产生
- 每个模块最多有 6 个时钟源可供使用，可驱动一个单独的内部 16 位计数器

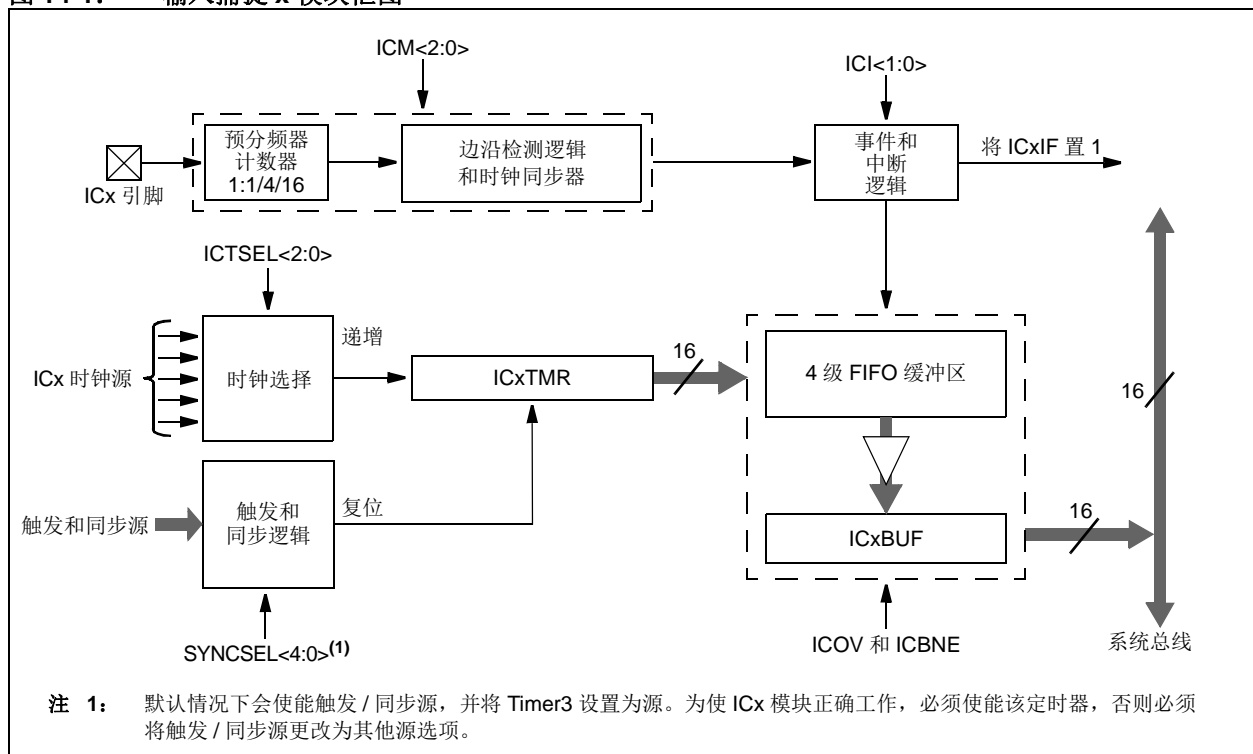
### 14.1 输入捕捉资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

#### 14.1.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“使用专用定时器的输入捕捉”（DS70000352）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

图 14-1: 输入捕捉 x 模块框图



# dsPIC33EPXXGS70X/80X 系列

## 14.2 输入捕捉寄存器

寄存器 14-1: ICxCON1: 输入捕捉 x 控制寄存器 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—
bit 15						bit 8	

U-0	R/W-0	R/W-0	R-0, HC, HS	R-0, HC, HS	R/W-0	R/W-0	R/W-0
—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0
bit 7						bit 0	

<b>图注:</b>	HC = 硬件清零位	HS = 硬件置 1 位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-14 **未实现:** 读为 0

bit 13 **ICSIDL:** 输入捕捉 x 空闲模式停止控制位  
 1 = 在 CPU 空闲模式下输入捕捉将停止工作  
 0 = 在 CPU 空闲模式下输入捕捉将继续工作

bit 12-10 **ICTSEL<2:0>:** 输入捕捉 x 定时器选择位  
 111 = 外设时钟 (FP) 是 ICx 的时钟源  
 110 = 保留  
 101 = 保留  
 100 = T1CLK 是 ICx 的时钟源 (仅支持同步时钟)  
 011 = T5CLK 是 ICx 的时钟源  
 010 = T4CLK 是 ICx 的时钟源  
 001 = T2CLK 是 ICx 的时钟源  
 000 = T3CLK 是 ICx 的时钟源

bit 9-7 **未实现:** 读为 0

bit 6-5 **ICI<1:0>:** 每次中断的捕捉次数选择位 (如果 ICM<2:0> = 001 或 111, 则不使用该位域)  
 11 = 每 4 次捕捉事件产生一次中断  
 10 = 每 3 次捕捉事件产生一次中断  
 01 = 每 2 次捕捉事件产生一次中断  
 00 = 每次捕捉事件产生一次中断

bit 4 **ICOV:** 输入捕捉 x 溢出状态标志位 (只读)  
 1 = 发生了输入捕捉缓冲区溢出  
 0 = 未发生输入捕捉缓冲区溢出

bit 3 **ICBNE:** 输入捕捉 x 缓冲区非空状态位 (只读)  
 1 = 输入捕捉缓冲区非空, 至少可以再读一个捕捉值  
 0 = 输入捕捉缓冲区为空

bit 2-0 **ICM<2:0>:** 输入捕捉 x 模式选择位  
 111 = 处于 CPU 休眠和空闲模式时, 输入捕捉 x 仅用作中断引脚 (只检测上升沿, 所有其他控制位都不适用)  
 110 = 未使用 (禁止模块)  
 101 = 捕捉模式, 每 16 个上升沿捕捉一次 (预分频捕捉模式)  
 100 = 捕捉模式, 每 4 个上升沿捕捉一次 (预分频捕捉模式)  
 011 = 捕捉模式, 每个上升沿捕捉一次 (简单捕捉模式)  
 010 = 捕捉模式, 每个下降沿捕捉一次 (简单捕捉模式)  
 001 = 捕捉模式, 每个边沿 (上升沿和下降沿) 捕捉一次 (边沿检测模式, 在该模式下不使用 ICI<1:0>)  
 000 = 输入捕捉 x 关闭

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 14-2: ICxCON2: 输入捕捉 x 控制寄存器 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	IC32
bit 15							bit 8

R/W-0	R/W-0, HS	U-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-1
ICTRIG <sup>(2)</sup>	TRIGSTAT <sup>(3)</sup>	—	SYNCSEL4 <sup>(4)</sup>	SYNCSEL3 <sup>(4)</sup>	SYNCSEL2 <sup>(4)</sup>	SYNCSEL1 <sup>(4)</sup>	SYNCSEL0 <sup>(4)</sup>
bit 7							bit 0

<b>图注:</b>	HS = 硬件置 1 位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-9 **未实现:** 读为 0

bit 8 **IC32:** 输入捕捉 x 32 位定时器模式选择位 (级联模式)  
 1 = 奇编号 ICx 和偶编号 ICx 构成一个 32 位输入捕捉模块<sup>(1)</sup>  
 0 = 禁止级联模块操作

bit 7 **ICTRIG:** 输入捕捉 x 触发操作选择位<sup>(2)</sup>  
 1 = 输入源用于触发输入捕捉定时器 (触发模式)  
 0 = 输入源用于将输入捕捉定时器与另一个模块的定时器同步 (同步模式)

bit 6 **TRIGSTAT:** 定时器触发状态位<sup>(3)</sup>  
 1 = ICxTMR 已触发并正在运行  
 0 = ICxTMR 未触发并保持清零

bit 5 **未实现:** 读为 0

- 注 1:** 必须将奇编号和偶编号 ICx 中的 IC32 位都置 1 以使能级联模式。  
**2:** 通过 ICxCON2 寄存器的 SYNCSEL<4:0> 位选择输入源。  
**3:** 该位由选定的输入源 (通过 SYNCSEL<4:0> 位选择) 置 1; 可通过软件对该位进行读取、置 1 和清零。  
**4:** 不要将 ICx 模块用作它自己的同步源或触发源。  
**5:** 该选项只能选作触发源, 不能选作同步源。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 14-2: ICxCON2: 输入捕捉 x 控制寄存器 2 (续)

bit 4-0      **SYNCSEL<4:0>**: 同步和触发操作的输入源选择位 <sup>(4)</sup>

11111	= ICx 没有同步源或触发源
11110	= 保留
11101	= 保留
11100	= 保留
11011	= CMP4 模块同步或触发 ICx <sup>(5)</sup>
11010	= CMP3 模块同步或触发 ICx <sup>(5)</sup>
11001	= CMP2 模块同步或触发 ICx <sup>(5)</sup>
11000	= CMP1 模块同步或触发 ICx <sup>(5)</sup>
10111	= 保留
10110	= 保留
10101	= 保留
10100	= 保留
10011	= IC4 模块中断同步或触发 ICx
10010	= IC3 模块中断同步或触发 ICx
10001	= IC2 模块中断同步或触发 ICx
10000	= IC1 模块中断同步或触发 ICx
01111	= Timer5 同步或触发 ICx
01110	= Timer4 同步或触发 ICx
01101	= Timer3 同步或触发 ICx (默认值)
01100	= Timer2 同步或触发 ICx
01011	= Timer1 同步或触发 ICx
01010	= 保留
01001	= 保留
01000	= IC4 模块同步或触发 ICx
00111	= IC3 模块同步或触发 ICx
00110	= IC2 模块同步或触发 ICx
00101	= IC1 模块同步或触发 ICx
00100	= OC4 模块同步或触发 ICx
00011	= OC3 模块同步或触发 ICx
00010	= OC2 模块同步或触发 ICx
00001	= OC1 模块同步或触发 ICx
00000	= ICx 没有同步源或触发源

- 注 1:** 必须将奇编号和偶编号 ICx 中的 IC32 位都置 1 以使能级联模式。
- 2:** 通过 ICxCON2 寄存器的 SYNCSEL<4:0> 位选择输入源。
- 3:** 该位由选定的输入源 (通过 SYNCSEL<4:0> 位选择) 置 1; 可通过软件对该位进行读取、置 1 和清零。
- 4:** 不要将 ICx 模块用作它自己的同步源或触发源。
- 5:** 该选项只能选作触发源, 不能选作同步源。

## 15.0 输出比较

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“带专用定时器的输出比较”（DS70005159），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

输出比较模块可以从 6 个可用时钟源中选择一个作为其时基。模块将定时器的值与一个或两个比较寄存器的值（取决于所选的工作模式）作比较。当定时器值与比较寄存器值匹配时，输出引脚的状态发生改变。输出比较

模块通过在发生比较匹配事件时改变输出引脚的状态，产生单个输出脉冲或连续输出脉冲。输出比较模块还能在发生比较匹配事件时产生中断。

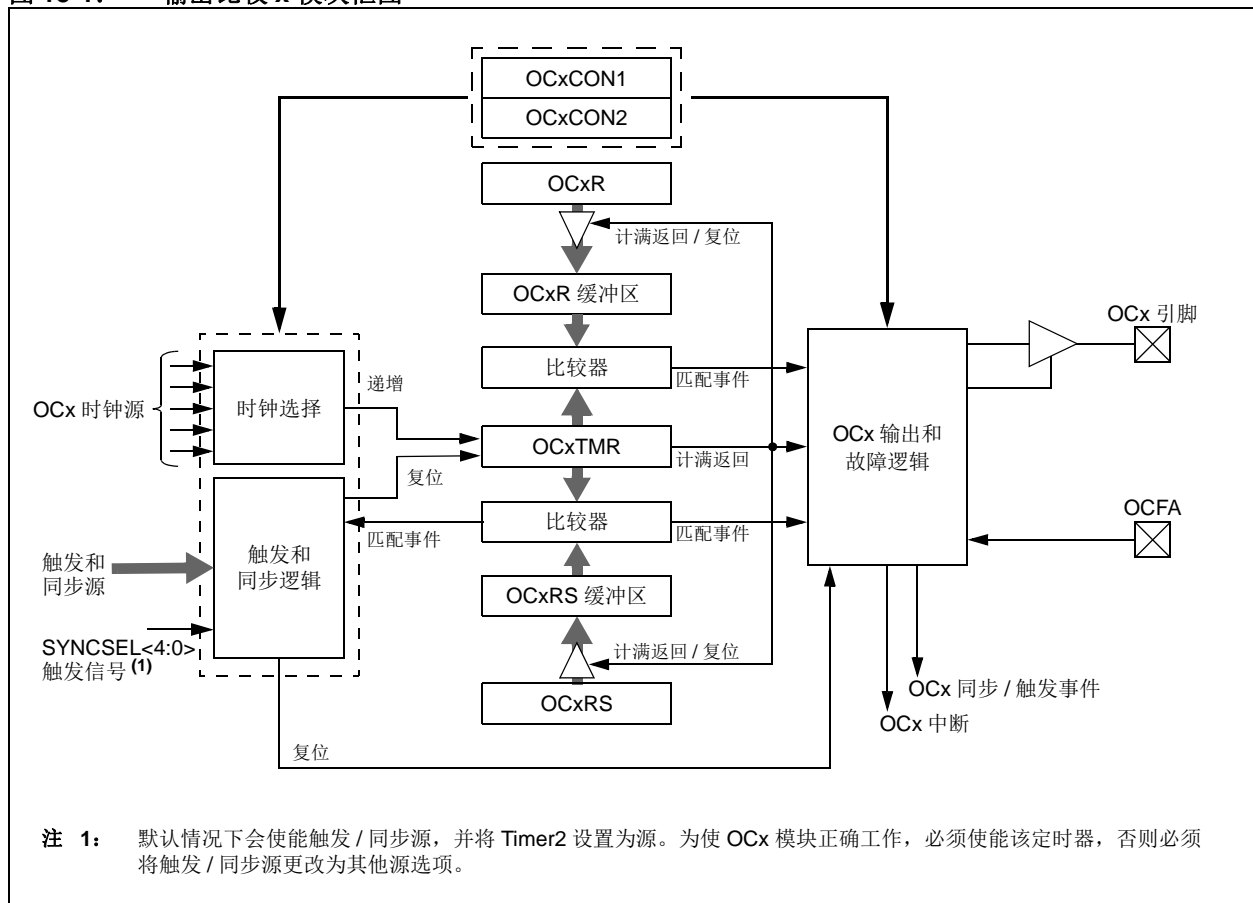
### 15.1 输出比较资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

#### 15.1.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“带专用定时器的输出比较”（DS70005159）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

图 15-1: 输出比较 x 模块框图



**注 1:** 默认情况下会使能触发 / 同步源，并将 Timer2 设置为源。为使 OCx 模块正确工作，必须使能该定时器，否则必须将触发 / 同步源更改为其他源选项。

# dsPIC33EPXXGS70X/80X 系列

## 15.2 输出比较控制寄存器

寄存器 15-1: **OCxCON1: 输出比较 x 控制寄存器 1**

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	—	—
bit 15						bit 8	

R/W-0	U-0	U-0	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
ENFLTA	—	—	OCFLTA	TRIGMODE	OCM2	OCM1	OCM0
bit 7						bit 0	

<b>图注:</b>	HSC = 硬件置 1/ 清零位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14 **未实现:** 读为 0

bit 13 **OCSIDL:** 输出比较 x 空闲模式停止控制位  
 1 = 输出比较 x 在 CPU 空闲模式下停止工作  
 0 = 输出比较 x 在 CPU 空闲模式下继续工作

bit 12-10 **OCTSEL<2:0>:** 输出比较 x 时钟选择位  
 111 = 外设时钟 (FP)  
 110 = 保留  
 101 = 保留  
 100 = T1CLK 是 OCx 的时钟源 (仅支持同步时钟)  
 011 = T5CLK 是 OCx 的时钟源  
 010 = T4CLK 是 OCx 的时钟源  
 001 = T3CLK 是 OCx 的时钟源  
 000 = T2CLK 是 OCx 的时钟源

bit 9-8 **未实现:** 读为 0

bit 7 **ENFLTA:** 故障 A 输入使能位  
 1 = 使能输出比较故障 A 输入 (OCFA)  
 0 = 禁止输出比较故障 A 输入 (OCFA)

bit 6-5 **未实现:** 读为 0

bit 4 **OCFLTA:** PWM 故障 A 条件状态位  
 1 = OCFA 引脚上产生了 PWM 故障 A 条件  
 0 = OCFA 引脚上未产生 PWM 故障 A 条件

bit 3 **TRIGMODE:** 触发状态模式选择位  
 1 = TRIGSTAT (OCxCON2<6>) 在 OCxRS = OCxTMR 时清零或者用软件清零  
 0 = TRIGSTAT 只能用软件清零

**注 1:** OCxR 和 OCxRS 仅在 PWM 模式下是双重缓冲的。

## 寄存器 15-1: OCxCON1: 输出比较 x 控制寄存器 1 (续)

bit 2-0 **OCM<2:0>**: 输出比较 x 模式选择位

111 = 中心对齐 PWM 模式: 当 OCxTMR = OCxR 时输出设置为高电平, 当 OCxTMR = OCxRS 时输出设置为低电平<sup>(1)</sup>

110 = 边沿对齐 PWM 模式: 当 OCxTMR = 0 时输出设置为高电平, 当 OCxTMR = OCxR 时输出设置为低电平<sup>(1)</sup>

101 = 双比较连续脉冲模式: 将 OCx 引脚初始化为低电平, 在 OCxTMR 与 OCxR 和 OCxRS 交替匹配时连续翻转 OCx 状态

100 = 双比较单脉冲模式: 将 OCx 引脚初始化为低电平, 在一个周期内 OCxTMR 分别与 OCxR 和 OCxRS 匹配时翻转 OCx 状态

011 = 单比较模式: 与 OCxR 匹配的比较事件使 OCx 引脚的电平连续翻转

010 = 单比较单脉冲模式: 将 OCx 引脚初始化为高电平, 与 OCxR 匹配的比较事件强制 OCx 引脚为低电平

001 = 单比较单脉冲模式: 将 OCx 引脚初始化为低电平, 与 OCxR 匹配的比较事件强制 OCx 引脚为高电平

000 = 禁止输出比较通道

注 1: OCxR 和 OCxRS 仅在 PWM 模式下是双重缓冲的。

# dsPIC33EPXXGS70X/80X 系列

寄存器 15-2: OCxCON2: 输出比较 x 控制寄存器 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	R/W-0
FLTMD	FLTOUT	FLTTRIEN	OCINV	—	—	—	OC32
bit 15							bit 8

R/W-0	R/W-0, HS	R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0
OCTRIG	TRIGSTAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0
bit 7							bit 0

**图注:** HS = 硬件置 1 位  
R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15            **FLTMD:** 故障模式选择位  
1 = 故障模式将保持到故障源消除; 相应的 OCFLTA 位用软件清零并且新的 PWM 周期开始  
0 = 故障模式将保持到故障源消除, 并且新的 PWM 周期开始
- bit 14            **FLTOUT:** 故障输出位  
1 = PWM 输出在发生故障时被驱动为高电平  
0 = PWM 输出在发生故障时被驱动为低电平
- bit 13            **FLTTRIEN:** 故障输出状态选择位  
1 = 发生故障时 OCx 引脚为三态  
0 = 发生故障时 OCx 引脚的 I/O 状态由 FLTOUT 位定义
- bit 12            **OCINV:** 输出比较 x 反相位  
1 = OCx 输出反相  
0 = OCx 输出不反相
- bit 11-9        **未实现:** 读为 0
- bit 8            **OC32:** 级联两个 OCx 模块使能位 (32 位操作)  
1 = 使能级联模块操作  
0 = 禁止级联模块操作
- bit 7            **OCTRIG:** 输出比较 x 触发 / 同步选择位  
1 = OCx 由 SYNCSELx 位指定的触发源触发  
0 = 将 OCx 与 SYNCSELx 位指定的同步源同步
- bit 6            **TRIGSTAT:** 定时器触发状态位  
1 = 定时器源已触发并正在运行  
0 = 定时器源未触发并保持清零
- bit 5            **OCTRIS:** 输出比较 x 输出引脚方向选择位  
1 = OCx 为三态  
0 = OCx 模块驱动 OCx 引脚

- 注 1:** 不要将 OCx 模块用作它自己的同步源或触发源。
- 2:** 当 OCy 模块关闭时, 它会发送一个触发输出信号。如果 OCx 模块使用 OCy 模块作为触发源, 则在禁止 OCy 模块之前, 必须先取消选择它作为触发源。
- 3:** 对于每个 OCMPx 实例, 使用不同的 PTG 触发输出:  
OCMP1——PTG 触发输出 [0]  
OCMP2——PTG 触发输出 [1]  
OCMP3——PTG 触发输出 [2]  
OCMP4——PTG 触发输出 [3]



## 寄存器 15-2: OCxCON2: 输出比较 x 控制寄存器 2 (续)

bit 4-0      **SYNCSEL<4:0>**: 触发 / 同步源选择位

- 11111 = OCxRS 比较事件用于同步
- 11110 = INT2 引脚同步或触发 OCx
- 11101 = INT1 引脚同步或触发 OCx
- 11100 = 保留
- 11011 = CMP4 模块同步或触发 OCx
- 11010 = CMP3 模块同步或触发 OCx
- 11001 = CMP2 模块同步或触发 OCx
- 11000 = CMP1 模块同步或触发 OCx
- 10111 = 保留
- 10110 = 保留
- 10101 = 保留
- 10100 = 保留
- 10011 = IC4 输入捕捉中断事件同步或触发 OCx
- 10010 = IC3 输入捕捉中断事件同步或触发 OCx
- 10001 = IC2 输入捕捉中断事件同步或触发 OCx
- 10000 = IC1 输入捕捉中断事件同步或触发 OCx
- 01111 = Timer5 同步或触发 OCx
- 01110 = Timer4 同步或触发 OCx
- 01101 = Timer3 同步或触发 OCx
- 01100 = Timer2 同步或触发 OCx (默认值)
- 01011 = Timer1 同步或触发 OCx
- 01010 = PTG 触发输出 x<sup>(3)</sup>
- 01001 = 保留
- 01000 = IC4 输入捕捉事件同步或触发 OCx
- 00111 = IC3 输入捕捉事件同步或触发 OCx
- 00110 = IC2 输入捕捉事件同步或触发 OCx
- 00101 = IC1 输入捕捉事件同步或触发 OCx
- 00100 = OC4 模块同步或触发 OCx<sup>(1,2)</sup>
- 00011 = OC3 模块同步或触发 OCx<sup>(1,2)</sup>
- 00010 = OC2 模块同步或触发 OCx<sup>(1,2)</sup>
- 00001 = OC1 模块同步或触发 OCx<sup>(1,2)</sup>
- 00000 = OCx 没有同步源或触发源

- 注 1:** 不要将 OCx 模块用作它自己的同步源或触发源。
- 2:** 当 OCy 模块关闭时, 它会发送一个触发输出信号。如果 OCx 模块使用 OCy 模块作为触发源, 则在禁止 OCy 模块之前, 必须先取消选择它作为触发源。
- 3:** 对于每个 OCMPx 实例, 使用不同的 PTG 触发输出:
- OCMP1——PTG 触发输出 [0]
  - OCMP2——PTG 触发输出 [1]
  - OCMP3——PTG 触发输出 [2]
  - OCMP4——PTG 触发输出 [3]

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 16.0 高速 PWM

**注：** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“**高速 PWM 模块**” (DS70000323)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

dsPIC33EPXXGS70X/80X 器件上的高速 PWM 模块支持多种 PWM 模式和输出格式。该 PWM 模块是如下电源转换应用的理想选择：

- 交流 / 直流转换器
- 直流 / 直流转换器
- 功率因数校正
- 不间断电源 (Uninterruptible Power Supply, UPS)
- 逆变器
- 电池充电器
- 数字照明

### 16.1 特性概述

高速 PWM 模块具有以下特性：

- 8 个 PWMx 发生器 (每个发生器有 2 个输出)
- 2 个主控时基模块
- 每个 PWM 输出具有独立的时基和占空比
- 占空比、死区、相移和频率分辨率为 1.04 ns
- 独立的故障输入和限流输入
- 冗余输出
- 真正独立输出
- 中心对齐 PWM 模式
- 输出改写控制
- 斩波模式 (也称为门控模式)
- 特殊事件触发信号
- 可从 PWMx 向模数转换器 (Analog-to-Digital Converter, ADC) 产生双触发
- PWMxL 和 PWMxH 输出引脚交换
- 独立的 PWMx 频率、占空比和相移变化
- 增强型前沿消隐 (Leading-Edge Blanking, LEB) 功能
- PWM 捕捉功能

**注：** 在中心对齐 PWM 模式下，占空比、死区、相移和频率分辨率为 8.32 ns。

图 16-1 以概念化方式显示了 PWM 模块的简化框图。

图 16-2 说明了在互补 PWM 模式下分配给每个 PWMx 输出对的模块硬件。

PWM 模块包含 8 个 PWM 发生器。模块最多具有 16 个 PWMx 输出引脚：PWM1H/PWM1L 至 PWM8H/PWM8L。对于互补输出，这 16 个 I/O 引脚组合为高 / 低端输出对。PWM1 至 PWM6 可以用于触发 ADC 转换。

### 16.2 特性说明

PWM 模块针对具有以下要求的应用而设计：

- 高 PWM 频率且高分辨率
- 能够驱动标准边沿对齐、中心对齐、互补模式和推挽模式输出
- 能够产生多相 PWM 输出

两种常见的中等功率电源转换器拓扑为推挽式和半桥式。这些设计要求由两个引脚交替切换输出 PWM 信号，这正是推挽 PWM 模式所提供的功能。

相移 PWM 描述了这样的情形：每个 PWM 发生器提供输出，但发生器输出之间的相位关系是可指定和可更改的。

多相 PWM 通常用于改善直流 / 直流转换器负载瞬态响应，以及减小输出滤波电容和电感的体积。多个直流 / 直流转换器通常并行工作，但时间上存在相移。以 250 kHz 工作时，单个 PWM 输出的周期为 4  $\mu$ s，但 4 个 PWM 通道组成的阵列 (交错工作，各工作 1  $\mu$ s) 产生的有效开关频率为 1 MHz。多相 PWM 应用通常使用固定的相位关系。

可变相位 PWM 在零电压转换 (Zero Voltage Transition, ZVT) 电源转换器中非常有用。其中，PWM 占空比总是为 50%，功率流通过改变两个 PWM 发生器之间的相对相移进行控制。

# dsPIC33EPXXGS70X/80X 系列

## 16.2.1 写保护寄存器

在 dsPIC33EPXXGS70X/80X 系列器件上，为 IOCONx 和 FCLCONx 寄存器实现了写保护功能。写保护功能可以防止对于这些寄存器的任何意外写操作。该保护功能可以通过 PWMLOCK 配置位（FDEVOPT<0>）进行控制。写保护功能的默认状态是使能（PWMLOCK = 1）。写保护功能可以通过配置 PWMLOCK = 0 来禁止。

要对这些锁定的寄存器进行写访问，用户应用程序必须向 PWMKEY 寄存器连续写入两个值（0xABCD 和 0x4321）来执行解锁操作。对 IOCONx 或 FCLCONx 寄存器的写访问必须是紧接在解锁过程之后的下一个 SFR 访问。在解锁过程和后续写访问之间不能有任何其他 SFR 访问。要写入 IOCONx 和 FCLCONx 寄存器，需要两次解锁操作。

正确的解锁序列如例 16-1 所示。

### 例 16-1: PWM 写保护寄存器解锁序列

```
; Writing to FCLCON1 register requires unlock sequence

mov #0xabcd, w10      ; Load first unlock key to w10 register
mov #0x4321, w11      ; Load second unlock key to w11 register
mov #0x0000, w0       ; Load desired value of FCLCON1 register in w0
mov w10, PWMKEY       ; Write first unlock key to PWMKEY register
mov w11, PWMKEY       ; Write second unlock key to PWMKEY register
mov w0, FCLCON1      ; Write desired value to FCLCON1 register

; Set PWM ownership and polarity using the IOCON1 register
; Writing to IOCON1 register requires unlock sequence

mov #0xabcd, w10      ; Load first unlock key to w10 register
mov #0x4321, w11      ; Load second unlock key to w11 register
mov #0xF000, w0       ; Load desired value of IOCON1 register in w0
mov w10, PWMKEY       ; Write first unlock key to PWMKEY register
mov w11, PWMKEY       ; Write second unlock key to PWMKEY register
mov w0, IOCON1       ; Write desired value to IOCON1 register
```

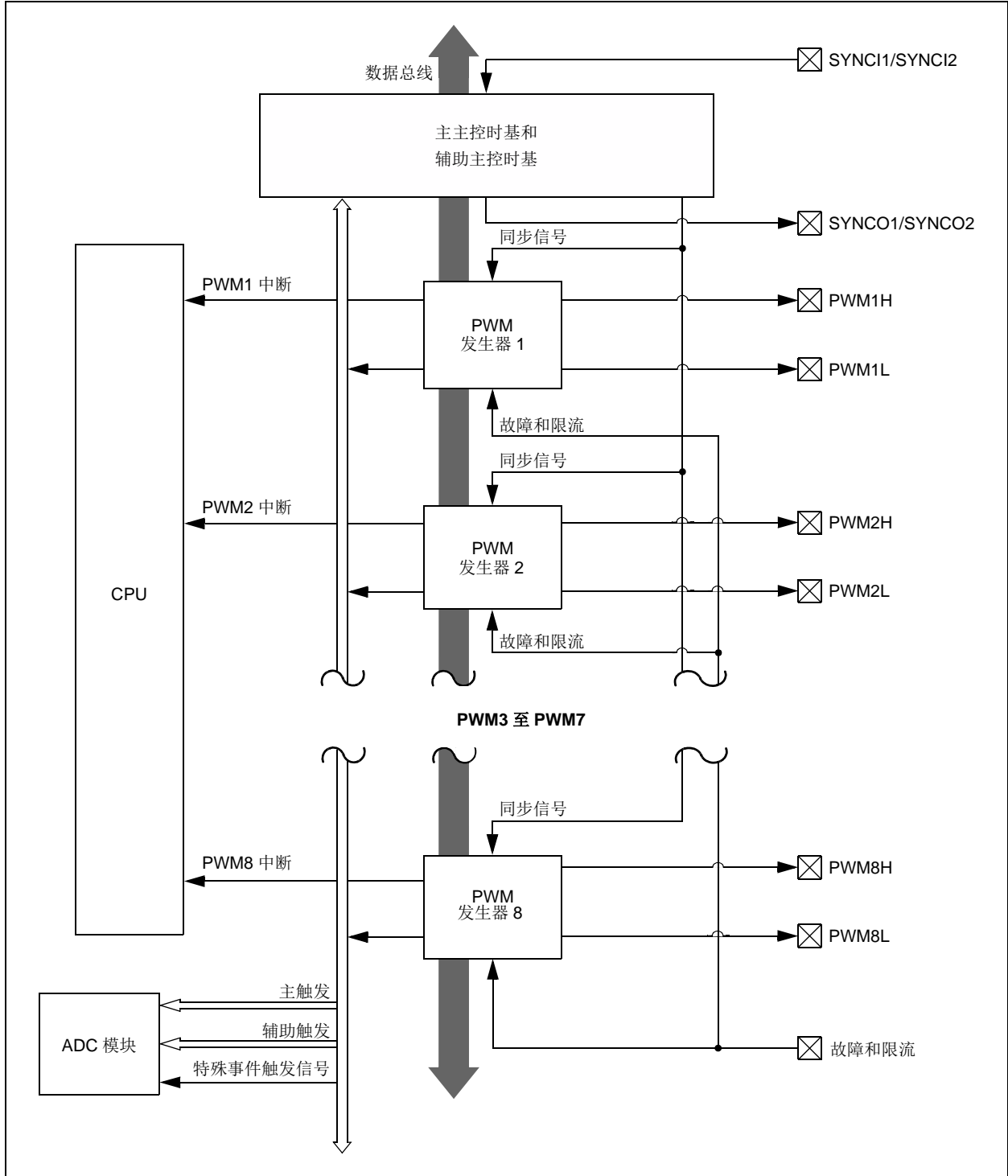
## 16.3 PWM 资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 16.3.1 主要资源

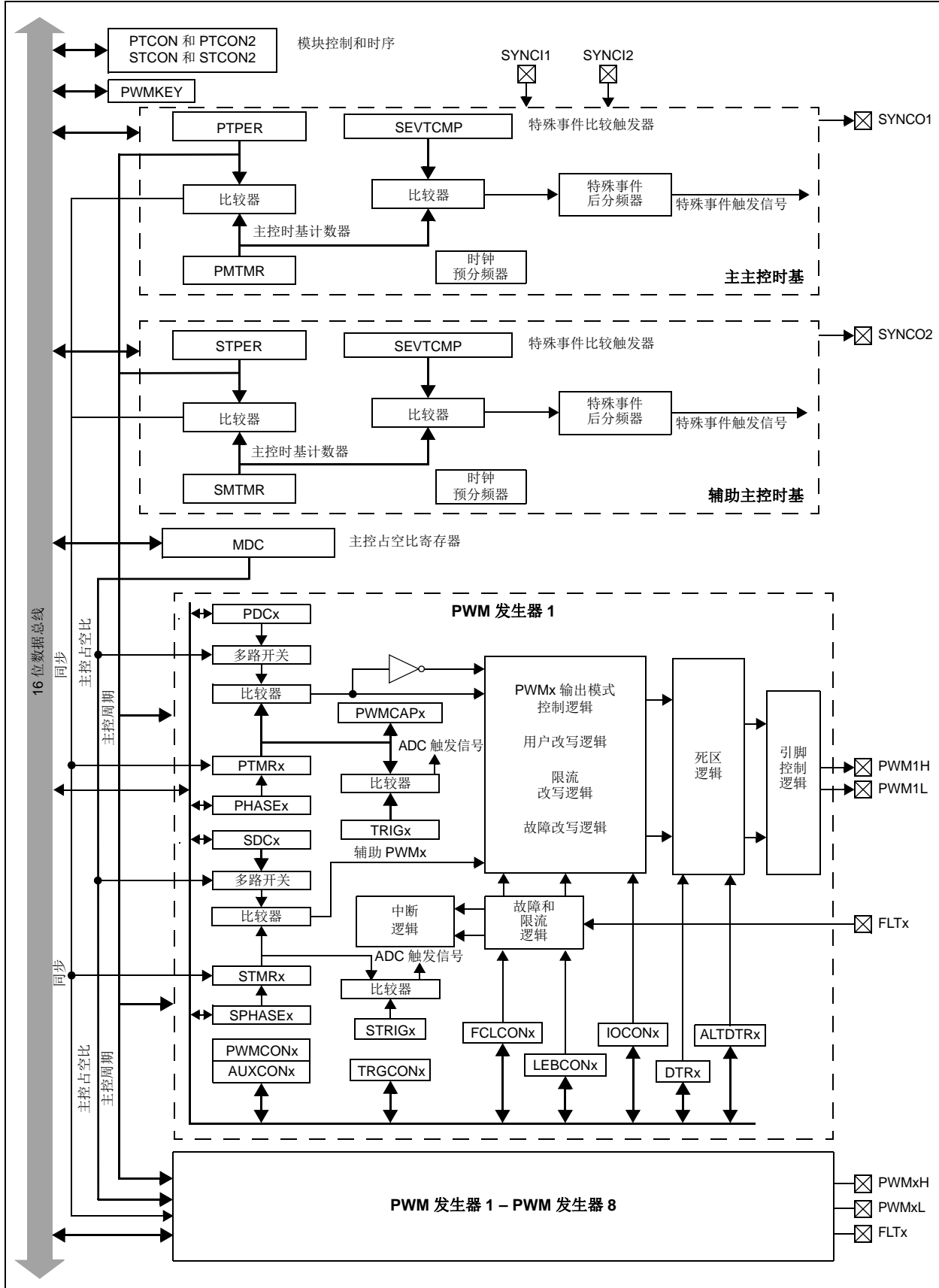
- 《dsPIC33/PIC24 系列参考手册》中的“**高速 PWM 模块**”（DS70000323）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

图 16-1: 高速 PWM 模块架构图



# dsPIC33EPXXGS70X/80X 系列

图 16-2: 高速 PWM 的简化概念框图



# dsPIC33EPXXGS70X/80X 系列

**寄存器 16-1: PTCON: PWM 时基控制寄存器**

R/W-0	U-0	R/W-0	R-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
PTEN	—	PTSIDL	SESTAT	SEIEN	EIPU <sup>(1)</sup>	SYNCPOL <sup>(1)</sup>	SYNCOEN <sup>(1)</sup>
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCEN <sup>(1)</sup>	SYNCSRC2 <sup>(1)</sup>	SYNCSRC1 <sup>(1)</sup>	SYNCSRC0 <sup>(1)</sup>	SEVTPS3 <sup>(1)</sup>	SEVTPS2 <sup>(1)</sup>	SEVTPS1 <sup>(1)</sup>	SEVTPS0 <sup>(1)</sup>
bit 7							bit 0

<b>图注:</b>	HSC = 硬件置 1/ 清零位
R = 可读位	W = 可写位
-n = POR 时的值	1 = 置 1
	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

- bit 15     **PTEN:** PWM 模块使能位  
1 = 使能 PWM 模块  
0 = 禁止 PWM 模块
- bit 14     **未实现:** 读为 0
- bit 13     **PTSIDL:** PWM 时基空闲模式停止位  
1 = PWM 时基在 CPU 空闲模式下暂停  
0 = PWM 时基在 CPU 空闲模式下运行
- bit 12     **SESTAT:** 特殊事件中断状态位  
1 = 特殊事件中断处于待处理状态  
0 = 特殊事件中断不处于待处理状态
- bit 11     **SEIEN:** 特殊事件中断允许位  
1 = 允许特殊事件中断  
0 = 禁止特殊事件中断
- bit 10     **EIPU:** 使能立即周期更新位 <sup>(1)</sup>  
1 = 立即更新有效周期寄存器  
0 = 在 PWM 周期边界处更新有效周期寄存器
- bit 9       **SYNCPOL:** 同步输入和输出极性位 <sup>(1)</sup>  
1 = SYNCIx/SYNCO1 极性反相 (低电平有效)  
0 = SYNCIx/SYNCO1 高电平有效
- bit 8       **SYNCOEN:** 主时基同步使能位 <sup>(1)</sup>  
1 = 使能 SYNCO1 输出  
0 = 禁止 SYNCO1 输出
- bit 7       **SYNCEN:** 外部时基同步使能位 <sup>(1)</sup>  
1 = 使能主时基的外部同步  
0 = 禁止主时基的外部同步
- bit 6-4     **SYNCSRC<2:0>:** 同步源选择位 <sup>(1)</sup>  
111 = 保留  
101 = 保留  
100 = 保留  
011 = PTG 触发输出 17  
010 = PTG 触发输出 16  
001 = SYNCI2  
000 = SYNCI1

**注 1:** 只有 PTEN = 0 时, 才能更改这些位。此外, 在使用 SYNCIx 功能时, 用户应用程序必须使用稍大于外部同步输入信号期望周期的值设定周期寄存器。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-1: PTCON: PWM 时基控制寄存器 (续)

bit 3-0 **SEVTPS<3:0>**: PWM 特殊事件触发信号输出后分频比选择位 <sup>(1)</sup>

1111 = 1:16, 后分频器在每发生十六个比较匹配事件时产生特殊事件触发信号

•

•

0001 = 1:2, 后分频器在每发生两个比较匹配事件时产生特殊事件触发信号

0000 = 1:1, 后分频器在每次发生比较匹配事件时产生特殊事件触发信号

**注 1:** 只有 PTEN = 0 时, 才能更改这些位。此外, 在使用 SYNCIx 功能时, 用户应用程序必须使用稍大于外部同步输入信号期望周期的值设定周期寄存器。

## 寄存器 16-2: PTCON2: PWM 时钟分频比选择寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PCLKDIV<2:0> <sup>(1)</sup>		
bit 7					bit 0		

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-3 **未实现:** 读为 0

bit 2-0 **PCLKDIV<2:0>**: PWM 输入时钟预分频比选择位 <sup>(1)</sup>

111 = 保留

110 = 64 分频, 最大 PWM 时序分辨率

101 = 32 分频, 最大 PWM 时序分辨率

100 = 16 分频, 最大 PWM 时序分辨率

011 = 8 分频, 最大 PWM 时序分辨率

010 = 4 分频, 最大 PWM 时序分辨率

001 = 2 分频, 最大 PWM 时序分辨率

000 = 1 分频, 最大 PWM 时序分辨率 (上电默认值)

**注 1:** 只有 PTEN = 0 时, 才能更改这些位。在工作期间改变时钟选择会产生不可预测的结果。



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-3: PTPER: PWM 主主控时基周期寄存器 (1,2)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PTPER<15:8>							
bit 15							
bit 8							

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PTPER<7:0>							
bit 7							
bit 0							

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-0 **PTPER<15:0>**: 主主控时基 (PMTMR) 周期值位

注 1: PWM 时基的最小值是 0x0010, 最大值是 0xFFFF8。

2: 任何小于 0x0028 的周期值都必须将最低 3 位设置为 0, 从而产生 8.32 ns 的周期分辨率 (最快附属时钟速率时)。

## 寄存器 16-4: SEVTCMP: PWM 特殊事件比较寄存器 (1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SEVTCMP<12:5>							
bit 15							
bit 8							

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
SEVTCMP<4:0>					—	—	—
bit 7							
bit 0							

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-3 **SEVTCMP<12:0>**: 特殊事件比较计数值位

bit 2-0 **未实现**: 读为 0

注 1: 一个 LSB = 1.04 ns (最快附属时钟速率时); 因此, 最小 SEVTCMP 分辨率为 8.32 ns。

# dsPIC33EPXXGS70X/80X 系列

寄存器 16-5: STCON: PWM 辅助主控时基控制寄存器

U-0	U-0	U-0	R-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	SESTAT	SEIEN	EIPU <sup>(1)</sup>	SYNCPOL	SYNCOEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SYNCEN	SYNCSRC2	SYNCSRC1	SYNCSRC0	SEVTPS3	SEVTPS2	SEVTPS1	SEVTPS0
bit 7							bit 0

<b>图注:</b>	HSC = 硬件置 1/ 清零位
R = 可读位	W = 可写位
-n = POR 时的值	1 = 置 1
	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

- bit 15-13     **未实现:** 读为 0
- bit 12       **SESTAT:** 特殊事件中断状态位  
1 = 辅助特殊事件中断处于待处理状态  
0 = 辅助特殊事件中断不处于待处理状态
- bit 11       **SEIEN:** 特殊事件中断允许位  
1 = 允许辅助特殊事件中断  
0 = 禁止辅助特殊事件中断
- bit 10       **EIPU:** 使能立即周期更新位 <sup>(1)</sup>  
1 = 立即更新有效辅助周期寄存器  
0 = 在 PWM 周期边界处更新有效辅助周期寄存器
- bit 9         **SYNCPOL:** 同步输入和输出极性位  
1 = SYNCIx/SYNCO2 极性反相 (低电平有效)  
0 = SYNCIx/SYNCO2 极性为高电平有效
- bit 8         **SYNCOEN:** 辅助主控时基同步使能位  
1 = 使能 SYNCO2 输出  
0 = 禁止 SYNCO2 输出
- bit 7         **SYNCEN:** 外部辅助主控时基同步使能位  
1 = 使能辅助时基的外部同步  
0 = 禁止辅助时基的外部同步
- bit 6-4       **SYNCSRC<2:0>:** 辅助时基同步源选择位  
111 = 保留  
101 = 保留  
100 = 保留  
011 = PTG 触发输出 17  
010 = PTG 触发输出 16  
001 = SYNCI2  
000 = SYNCI1
- bit 3-0       **SEVTPS<3:0>:** PWM 辅助特殊事件触发信号输出后分频比选择位  
1111 = 后分频比为 1:16  
0001 = 后分频比为 1:2  
•  
•  
•  
0000 = 后分频比为 1:1

注 1: 该位只适用于辅助主控时基周期。

# dsPIC33EPXXGS70X/80X 系列

**寄存器 16-6: STCON2: PWM 辅助时钟分频比选择寄存器**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PCLKDIV<2:0> <sup>(1)</sup>		
bit 7					bit 0		

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-3      **未实现:** 读为 0  
 bit 2-0      **PCLKDIV<2:0>:** PWM 输入时钟预分频比选择位 <sup>(1)</sup>  
               111 = 保留  
               110 = 64 分频, 最大 PWM 时序分辨率  
               101 = 32 分频, 最大 PWM 时序分辨率  
               100 = 16 分频, 最大 PWM 时序分辨率  
               011 = 8 分频, 最大 PWM 时序分辨率  
               010 = 4 分频, 最大 PWM 时序分辨率  
               001 = 2 分频, 最大 PWM 时序分辨率  
               000 = 1 分频, 最大 PWM 时序分辨率 (上电默认值)

**注 1:** 只有 PTEN = 0 时, 才能更改这些位。在工作期间改变时钟选择会产生不可预测的结果。

**寄存器 16-7: STPER: PWM 辅助主控时基周期寄存器 <sup>(1,2)</sup>**

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
STPER<15:8>							
bit 15							bit 8
R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
STPER<7:0>							
bit 7						bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0      **STPER<15:0>:** 辅助主控时基 (SMTMR) 周期值位

- 注 1:** PWM 时基的最小值是 0x0010, 最大值是 0xFFFF。  
**注 2:** 任何小于 0x0028 的周期值都必须将最低 3 位设置为 0, 从而产生 8.32 ns 的周期分辨率 (最快附属时钟速率时)。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-8: SSEVTCMP: PWM 辅助特殊事件比较寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEVTCMP<12:5>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
SSEVTCMP<4:0>					—	—	—
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-3        **SSEVTCMP<12:0>**: 特殊事件比较计数值位

bit 2-0        **未实现**: 读为 0

**注 1:** 一个 LSB = 1.04 ns (最快辅助时钟速率时); 因此, 最小 SSEVTCMP 分辨率为 8.32 ns。

## 寄存器 16-9: CHOP: PWM 斩波时钟发生器寄存器<sup>(1)</sup>

R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
CHPCLKEN	—	—	—	—	—	CHOPCLK6	CHOPCLK5
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHOPCLK4	CHOPCLK3	CHOPCLK2	CHOPCLK1	CHOPCLK0	—	—	—
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15        **CHPCLKEN**: 使能斩波时钟发生器位

1 = 使能斩波时钟发生器  
 0 = 禁止斩波时钟发生器

bit 14-10    **未实现**: 读为 0

bit 9-3        **CHOPCLK<6:0>**: 斩波时钟分频比位

值以 8.32 ns 为单位进行递增。斩波时钟信号的频率由以下表达式给出:  
 斩波频率 =  $1 / (16.64 * (\text{CHOP}<7:3> + 1) * \text{主主控 PWM 输入时钟周期})$

bit 2-0        **未实现**: 读为 0

**注 1:** 斩波时钟发生器使用 PTCON2 寄存器 ([寄存器 16-2](#)) 中的主 PWM 时钟预分频比 (PCLKDIV<2:0>) 工作。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-10: MDC: PWM 主控占空比寄存器 <sup>(1,2)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MDC<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
MDC<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-0      **MDC<15:0>**: PWM 主控占空比值位

**注 1:** 在 PWMx 输出上可以产生的最小脉冲宽度对应于值 0x0008, 而产生的最大脉冲宽度对应于 (周期 - 0x0008) 的值。

**2:** 当占空比接近 PWM 周期的 0% 或 100% (0 ns - 40 ns, 取决于工作模式) 时, PWM 占空比分辨率将从 1 LSB 提高到 3 LSB。

## 寄存器 16-11: PWMKEY: PWM 保护锁定 / 解锁密钥寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWMKEY<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWMKEY<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-0      **PWMKEY<15:0>**: PWM 保护锁定 / 解锁密钥值位

# dsPIC33EPXXGS70X/80X 系列

寄存器 16-12: PWMCONx: PWMx 控制寄存器 (x = 1 至 8)

R-0, HSC	R-0, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTSTAT <sup>(1)</sup>	CLSTAT <sup>(1)</sup>	TRGSTAT	FLTIEIEN	CLIEIEN	TRGIEIEN	ITB <sup>(3)</sup>	MDCS <sup>(3)</sup>
bit 15						bit 8	
R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
DTC1	DTC0	—	—	MTBS	CAM <sup>(2,3,4)</sup>	XPRES <sup>(5)</sup>	IUE
bit 7						bit 0	

**图注:** HSC = 硬件置 1/ 清零位  
R = 可读位 W = 可写位 U = 未实现位, 读为 0  
-n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

- bit 15 **FLTSTAT:** 故障中断状态位 <sup>(1)</sup>  
1 = 故障中断处于待处理状态  
0 = 没有故障中断处于待处理状态  
该位通过设置 FLTIEIEN = 0 进行清零。
- bit 14 **CLSTAT:** 限流中断状态位 <sup>(1)</sup>  
1 = 限流中断处于待处理状态  
0 = 没有限流中断处于待处理状态  
该位通过设置 CLIEIEN = 0 进行清零。
- bit 13 **TRGSTAT:** 触发中断状态位  
1 = 触发中断处于待处理状态  
0 = 没有触发中断处于待处理状态  
该位通过设置 TRGIEIEN = 0 进行清零。
- bit 12 **FLTIEIEN:** 故障中断允许位  
1 = 允许故障中断  
0 = 禁止故障中断, 并且 FLTSTAT 位清零
- bit 11 **CLIEIEN:** 限流中断允许位  
1 = 允许限流中断  
0 = 禁止限流中断, 并且 CLSTAT 位清零
- bit 10 **TRGIEIEN:** 触发中断允许位  
1 = 触发事件产生中断请求  
0 = 禁止触发事件中断, 并且 TRGSTAT 位清零
- bit 9 **ITB:** 独立时基模式位 <sup>(3)</sup>  
1 = PHASEx/SPHASEx 寄存器为该 PWMx 发生器提供时基周期  
0 = PTPER 寄存器为该 PWMx 发生器提供时序
- bit 8 **MDCS:** 主控占空比寄存器选择位 <sup>(3)</sup>  
1 = MDC 寄存器为该 PWMx 发生器提供占空比信息  
0 = PDCx 和 SDCx 寄存器为该 PWMx 发生器提供占空比信息

- 注 1:** 必须用软件清零相应的中断状态, 以及中断控制器中对应的 IFSx 位。
- 2:** 只有使能独立时基模式 (ITB = 1) 时才能使用中心对齐模式。如果 ITB = 0, 则 CAM 位会被忽略。
- 3:** 通过设置 PTEN (PTCON<15>) = 1 使能 PWMx 后, 不应更改这些位。
- 4:** 中心对齐模式会忽略占空比、相位和死区寄存器的最低 3 位。在时钟预分频比设置为最快时钟时, 可获得的最高中心对齐模式分辨率是 8.32 ns。
- 5:** 配置 CLMOD (FCLCONx<8>) = 0 和 ITB (PWMCONx<9>) = 1, 以在外部周期复位模式下工作。

## 寄存器 16-12: PWMCONx: PWMx 控制寄存器 (x = 1 至 8) (续)

bit 7-6	<b>DTC&lt;1:0&gt;</b> : 死区控制位 11 = 保留 10 = 禁止死区功能 01 = 对于互补输出模式施加负死区 00 = 对于所有输出模式施加正死区
bit 5-4	<b>未实现</b> : 读为 0
bit 3	<b>MTBS</b> : 主控时基选择位 1 = PWMx 发生器使用辅助主控时基进行同步, 并使用它作为 PWMx 发生逻辑的时钟源 (如果辅助时基可用) 0 = PWMx 发生器使用主主控时基进行同步, 并使用它作为 PWMx 发生逻辑的时钟源
bit 2	<b>CAM</b> : 中心对齐模式使能位 (2,3,4) 1 = 使能中心对齐模式 0 = 使能边沿对齐模式
bit 1	<b>XPRES</b> : 外部 PWMx 复位控制位 (5) 1 = 如果 PWMx 发生器处于独立时基模式, 则限流源复位该 PWMx 发生器的时基 0 = 外部引脚不影响 PWMx 时基
bit 0	<b>IUE</b> : 立即更新使能位 1 = 立即更新有效的占空比、相位偏移、死区和本地时基周期寄存器 0 = 对有效的占空比、相位偏移、死区和本地时基周期寄存器的更新与本地 PWMx 时基同步

- 注 1:** 必须用软件清零相应的中断状态, 以及中断控制器中对应的 IFSx 位。
- 2:** 只有使能独立时基模式 (ITB = 1) 时才能使用中心对齐模式。如果 ITB = 0, 则 CAM 位会被忽略。
- 3:** 通过设置 PTEN (PTCON<15>) = 1 使能 PWMx 后, 不应更改这些位。
- 4:** 中心对齐模式会忽略占空比、相位和死区寄存器的最低 3 位。在时钟预分频比设置为最快时钟时, 可获得的最中心对齐模式分辨率为 8.32 ns。
- 5:** 配置 CLMOD (FCLCONx<8>) = 0 和 ITB (PWMCONx<9>) = 1, 以在外部周期复位模式下工作。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-13: PDCx: PWMx 发生器占空比寄存器 (x = 1 至 8) (1,2,3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDCx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDCx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0        **PDCx<15:0>**: PWMx 发生器占空比值位

- 注 1:** 在独立 PWM 模式下, PDCx 寄存器仅控制 PWMxH 占空比。在互补、冗余和推挽 PWM 模式下, PDCx 寄存器控制 PWMxH 和 PWMxL 的占空比。
- 2:** 在 PWMx 输出上可以产生的最小脉冲宽度对应于值 0x0008, 而产生的最大脉冲宽度对应于 (周期 - 0x0008) 的值。
- 3:** 当占空比接近 PWM 周期的 0% 或 100% (0 ns - 40 ns, 取决于工作模式) 时, PWMx 占空比分辨率将从 1 LSB 提高到 3 LSB。

## 寄存器 16-14: SDCx: PWMx 辅助占空比寄存器 (x = 1 至 8) (1,2,3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDCx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SDCx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0        **SDCx<15:0>**: PWMxL 输出引脚的 PWMx 辅助占空比值位

- 注 1:** SDCx 寄存器仅在独立 PWM 模式下使用。在独立 PWM 模式下使用时, SDCx 寄存器控制 PWMxL 占空比。
- 2:** 在 PWMx 输出上可以产生的最小脉冲宽度对应于值 0x0008, 而产生的最大脉冲宽度对应于 (周期 - 0x0008) 的值。
- 3:** 当占空比接近 PWM 周期的 0% 或 100% (0 ns - 40 ns, 取决于工作模式) 时, PWMx 占空比分辨率将从 1 LSB 提高到 3 LSB。



# dsPIC33EPXXGS70X/80X 系列

寄存器 16-15: PHASEx: PWMx 主相移寄存器 (x = 1 至 8) (1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PHASEx<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PHASEx<7:0>							
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-0 PHASEx<15:0>: PWMx 发生器的相移值或独立时基周期寄存器

**注 1:** 如果 PWMCONx<9> = 0, 则依工作模式适用以下情况:

- 互补、冗余和推挽输出模式 (IOCONx<11:10> = 00、01 或 10), PHASEx<15:0> = PWMxH 和 PWMxL 输出的相移值
- 真正独立输出模式 (IOCONx<11:10> = 11), PHASEx<15:0> = 仅 PWMxH 的相移值
- 当 PHASEx/SPHASEx 寄存器提供相对于主控时基的相移时, 值的有效范围为 0x0000 至周期值

**2:** 如果 PWMCONx<9> = 1, 则依工作模式适用以下情况:

- 互补、冗余和推挽输出模式 (IOCONx<11:10> = 00、01 或 10), PHASEx<15:0> = PWMxH 和 PWMxL 的独立时基周期值
- 真正独立输出模式 (IOCONx<11:10> = 11), PHASEx<15:0> = 仅 PWMxH 的独立时基周期值
- 当 PHASEx/SPHASEx 寄存器提供本地周期时, 值的有效范围为 0x0000 至 0xFFFF

# dsPIC33EPXXGS70X/80X 系列

寄存器 16-16: **SPHASE<sub>x</sub>**: PWM<sub>x</sub> 辅助相移寄存器 (x = 1 至 8) (1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPHASE <sub>x</sub> <15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPHASE <sub>x</sub> <7:0>							
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0 **SPHASE<sub>x</sub><15:0>**: PWM<sub>xL</sub> 输出引脚辅助相位偏移位 (仅在独立 PWM 模式下使用)

注 1: 如果 PWMCON<sub>x</sub><9> = 0, 则依工作模式适用以下情况:

- 互补、冗余和推挽输出模式 (IOCON<sub>x</sub><11:10> = 00、01 或 10), SPHASE<sub>x</sub><15:0> = 不使用
- 真正独立输出模式 (IOCON<sub>x</sub><11:10> = 11), SPHASE<sub>x</sub><15:0> = 仅 PWM<sub>xL</sub> 的相移值

注 2: 如果 PWMCON<sub>x</sub><9> = 1, 则依工作模式适用以下情况:

- 互补、冗余和推挽输出模式 (IOCON<sub>x</sub><11:10> = 00、01 或 10), SPHASE<sub>x</sub><15:0> = 不使用
- 真正独立输出模式 (IOCON<sub>x</sub><11:10> = 11), SPHASE<sub>x</sub><15:0> = 仅 PWM<sub>xL</sub> 的独立时基周期值
- 当 PHASE<sub>x</sub>/SPHASE<sub>x</sub> 寄存器提供本地周期时, 值的有效范围为 0x0010-0xFFFF8

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-17: DTRx: PWMx 死区寄存器 (x = 1 至 8)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DTRx<13:8>					
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DTRx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-14    未实现: 读为 0

bit 13-0    **DTRx<13:0>**: PWMx 死区单元的无符号 14 位死区值位

## 寄存器 16-18: ALTDTRx: PWMx 备用死区寄存器 (x = 1 至 8)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ALTDTRx<13:8>					
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ALTDTRx<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-14    未实现: 读为 0

bit 13-0    **ALTDTRx<13:0>**: PWMx 死区单元的无符号 14 位备用死区值位

# dsPIC33EPXXGS70X/80X 系列

寄存器 16-19: TRGCONx: PWMx 触发控制寄存器 (x = 1 至 8)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
TRGDIV3	TRGDIV2	TRGDIV1	TRGDIV0	—	—	—	—
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DTM <sup>(1)</sup>	—	TRGSTRT5	TRGSTRT4	TRGSTRT3	TRGSTRT2	TRGSTRT1	TRGSTRT0
bit 7							bit 0

图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-12      **TRGDIV<3:0>**: 触发器 # 输出分频比位

1111 = 每发生 16 个触发事件时触发输出  
 1110 = 每发生 15 个触发事件时触发输出  
 1101 = 每发生 14 个触发事件时触发输出  
 1100 = 每发生 13 个触发事件时触发输出  
 1011 = 每发生 12 个触发事件时触发输出  
 1010 = 每发生 11 个触发事件时触发输出  
 1001 = 每发生 10 个触发事件时触发输出  
 1000 = 每发生 9 个触发事件时触发输出  
 0111 = 每发生 8 个触发事件时触发输出  
 0110 = 每发生 7 个触发事件时触发输出  
 0101 = 每发生 6 个触发事件时触发输出  
 0100 = 每发生 5 个触发事件时触发输出  
 0011 = 每发生 4 个触发事件时触发输出  
 0010 = 每发生 3 个触发事件时触发输出  
 0001 = 每发生 2 个触发事件时触发输出  
 0000 = 每次发生触发事件时触发输出

bit 11-8      **未实现**: 读为 0

bit 7          **DTM**: 双触发模式位 <sup>(1)</sup>

1 = 辅助触发事件与主触发事件组合产生 PWM 触发  
 0 = 辅助触发事件不与主触发事件组合产生 PWM 触发; 产生两个独立的 PWM 触发

bit 6          **未实现**: 读为 0

bit 5-0       **TRGSTRT<5:0>**: 触发后分频比开始使能选择位

111111 = 使能模块之后, 在产生第一个触发事件之前先等待 63 个 PWM 周期  
 •  
 •  
 •  
 000010 = 使能模块之后, 在产生第一个触发事件之前先等待 2 个 PWM 周期  
 000001 = 使能模块之后, 在产生第一个触发事件之前先等待 1 个 PWM 周期  
 000000 = 使能模块之后, 在产生第一个触发事件之前先等待 0 个 PWM 周期

注 1: 辅助 PWMx 发生器无法产生 PWM 触发中断。

# dsPIC33EPXXGS70X/80X 系列

寄存器 16-20: IOCONx: PWMx I/O 控制寄存器 (x = 1 至 8)

R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PENH	PENL	POLH	POLL	PMOD1 <sup>(1)</sup>	PMOD0 <sup>(1)</sup>	OVRENH	OVRENL
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OVRDAT1	OVRDAT0	FLTDAT1 <sup>(2)</sup>	FLTDAT0 <sup>(2)</sup>	CLDAT1 <sup>(2)</sup>	CLDAT0 <sup>(2)</sup>	SWAP	OSYNC
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15     **PENH:** PWMxH 输出引脚所有权位  
1 = PWM 模块控制 PWMxH 引脚  
0 = GPIO 模块控制 PWMxH 引脚
- bit 14     **PENL:** PWMxL 输出引脚所有权位  
1 = PWM 模块控制 PWMxL 引脚  
0 = GPIO 模块控制 PWMxL 引脚
- bit 13     **POLH:** PWMxH 输出引脚极性位  
1 = PWMxH 引脚为低电平有效  
0 = PWMxH 引脚为高电平有效
- bit 12     **POLL:** PWMxL 输出引脚极性位  
1 = PWMxL 引脚为低电平有效  
0 = PWMxL 引脚为高电平有效
- bit 11-10   **PMOD<1:0>:** PWMx I/O 引脚模式位 <sup>(1)</sup>  
11 = PWMx I/O 引脚对处于真正独立输出模式  
10 = PWMx I/O 引脚对处于推挽输出模式  
01 = PWMx I/O 引脚对处于冗余输出模式  
00 = PWMx I/O 引脚对处于互补输出模式
- bit 9       **OVRENH:** PWMxH 引脚改写使能位  
1 = OVRDAT1 为 PWMxH 引脚上的输出提供数据  
0 = PWMx 发生器为 PWMxH 引脚提供数据
- bit 8       **OVRENL:** PWMxL 引脚改写使能位  
1 = OVRDAT0 为 PWMxL 引脚上的输出提供数据  
0 = PWMx 发生器为 PWMxL 引脚提供数据
- bit 7-6     **OVRDAT<1:0>:** 使能改写时 PWMxH 和 PWMxL 引脚数据位  
如果 OVRENH = 1, 则 OVRDAT1 为 PWMxH 引脚提供数据  
如果 OVRENL = 1, 则 OVRDAT0 为 PWMxL 引脚提供数据
- bit 5-4     **FLTDAT<1:0>:** 使能 FLTMOD<1:0> 时 PWMxH 和 PWMxL 引脚状态位 <sup>(2)</sup>  
IFLTMOD (FCLCONx<15>) = 0: 正常故障模式:  
如果故障有效, 则 FLTDAT1 提供 PWMxH 引脚的状态。  
如果故障有效, 则 FLTDAT0 提供 PWMxL 引脚的状态。  
IFLTMOD (FCLCONx<15>) = 1: 独立故障模式:  
如果限流有效, 则 FLTDAT1 提供 PWMxH 引脚的状态。  
如果故障有效, 则 FLTDAT0 提供 PWMxL 引脚的状态。

注 1:  使能 PWM 模块 (PTEN = 1) 后, 不应更改这些位。  
注 2:  状态代表 PWMx 的有效 / 无效状态, 具体取决于 POLH 和 POLL 位的设置。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-20: IOCONx: PWMx I/O 控制寄存器 (x = 1 至 8) (续)

- bit 3-2 **CLDAT<1:0>**: 使能 CLMOD 时 PWMxH 和 PWMxL 引脚状态位 <sup>(2)</sup>  
**IFLTMOD (FCLCONx<15>) = 0**: 正常故障模式:  
 如果限流有效, 则 CLDAT1 提供 PWMxH 引脚的状态。  
 如果限流有效, 则 CLDAT0 提供 PWMxL 引脚的状态。  
**IFLTMOD (FCLCONx<15>) = 1**: 独立故障模式:  
 CLDAT<1:0> 位被忽略。
- bit 1 **SWAP**: 交换 PWMxH 和 PWMxL 引脚位  
 1 = PWMxH 输出信号连接到 PWMxL 引脚; PWMxL 输出信号连接到 PWMxH 引脚  
 0 = PWMxH 和 PWMxL 引脚映射到它们各自对应的引脚
- bit 0 **OSYNC**: 输出改写同步位  
 1 = 通过 OVRDAT<1:0> 位进行的输出改写与 PWMx 时基同步  
 0 = 通过 OVRDAT<1:0> 位进行的输出改写在下一个 CPU 时钟边界发生

- 注 1**: 使能 PWM 模块 (PTEN = 1) 后, 不应更改这些位。  
**注 2**: 状态代表 PWMx 的有效 / 无效状态, 具体取决于 POLH 和 POLL 位的设置。

## 寄存器 16-21: TRIGx: PWMx 主触发比较值寄存器 (x = 1 至 8)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TRGCMP<12:5>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
TRGCMP<4:0>					—	—	—
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                            0 = 清零                            x = 未知

- bit 15-3 **TRGCMP<12:0>**: 触发比较值位  
 当主 PWMx 采用本地时基时, 该寄存器包含可触发 ADC 模块的比较值。
- bit 2-0 **未实现**: 读为 0

# dsPIC33EPXXGS70X/80X 系列

寄存器 16-22: **FCLCONx: PWMx 故障限流控制寄存器 (x = 1 至 8)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IFLTMOD	CLSRC4	CLSRC3	CLSRC2	CLSRC1	CLSRC0	CLPOL <sup>(1)</sup>	CLMOD
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FLTSRC4	FLTSRC3	FLTSRC2	FLTSRC1	FLTSRC0	FLTPOL <sup>(1)</sup>	FLTMOD1	FLTMOD0
bit 7						bit 0	

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15 **IFLTMOD:** 独立故障模式使能位  
 1 = 独立故障模式: 限流输入将 FLTDAT1 映射到 PWMxH 输出, 故障输入将 FLTDAT0 映射到 PWMxL 输出; CLDAT<1:0> 位不用于改写功能  
 0 = 正常故障模式: 限流模式将 CLDAT<1:0> 位映射到 PWMxH 和 PWMxL 输出; PWM 故障模式将 FLTDAT<1:0> 映射到 PWMxH 和 PWMxL 输出

bit 14-10 **CLSRC<4:0>:** PWMx 发生器的限流控制信号源选择位

- 11111 = 保留
- 10001 = 保留
- 10000 = 模拟比较器 4
- 01111 = 模拟比较器 3
- 01110 = 模拟比较器 2
- 01101 = 模拟比较器 1
- 01100 = 故障 12
- 01011 = 故障 11
- 01010 = 故障 10
- 01001 = 故障 9
- 01000 = 故障 8
- 00111 = 故障 7
- 00110 = 故障 6
- 00101 = 故障 5
- 00100 = 故障 4
- 00011 = 故障 3
- 00010 = 故障 2
- 00001 = 故障 1
- 00000 = 保留

bit 9 **CLPOL:** PWMx 发生器的限流极性位<sup>(1)</sup>

- 1 = 选定的限流源为低电平有效
- 0 = 选定的限流源为高电平有效

bit 8 **CLMOD:** PWMx 发生器的限流模式使能位

- 1 = 使能限流模式
- 0 = 禁止限流模式

**注 1:** 只有 PTEN (PTCON<15>) = 0 时, 才能更改这些位。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-22: FCLCONx: PWMx 故障限流控制寄存器 (x = 1 至 8) (续)

- bit 7-3 **FLTSRC<4:0>**: PWMx 发生器的故障控制信号源选择位
- 11111 = 保留
  - 10001 = 保留
  - 10000 = 模拟比较器 4
  - 01111 = 模拟比较器 3
  - 01110 = 模拟比较器 2
  - 01101 = 模拟比较器 1
  - 01100 = 故障 12
  - 01011 = 故障 11
  - 01010 = 故障 10
  - 01001 = 故障 9
  - 01000 = 故障 8
  - 00111 = 故障 7
  - 00110 = 故障 6
  - 00101 = 故障 5
  - 00100 = 故障 4
  - 00011 = 故障 3
  - 00010 = 故障 2
  - 00001 = 故障 1
  - 00000 = 保留
- bit 2 **FLTPOL**: PWMx 发生器的故障极性位 <sup>(1)</sup>
- 1 = 选定的故障源为低电平有效
  - 0 = 选定的故障源为高电平有效
- bit 1-0 **FLTMOD<1:0>**: PWMx 发生器的故障模式位
- 11 = 禁止故障输入
  - 10 = 保留
  - 01 = 选定的故障源将 PWMxH 和 PWMxL 引脚强制为 FLTDATx 值 (周期)
  - 00 = 选定的故障源将 PWMxH 和 PWMxL 引脚强制为 FLTDATx 值 (锁定状态)

注 1: 只有 PTEN (PTCON<15>) = 0 时, 才能更改这些位。

## 寄存器 16-23: STRIGx: PWMx 辅助触发比较值寄存器 (x = 1 至 8) <sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STRGCMP<12:5>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
STRGCMP<4:0>					—	—	—
bit 7							bit 0

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15-3 **STRGCMP<12:0>**: 辅助触发比较值位  
当辅助 PWMx 采用本地时基时, 该寄存器包含可触发 ADC 模块的比较值。
- bit 2-0 **未实现**: 读为 0

注 1: STRIGx 无法产生 PWM 触发中断。



# dsPIC33EPXXGS70X/80X 系列

寄存器 16-24: **LEBCONx: PWMx 前沿消隐 (LEB) 控制寄存器 (x = 1 至 8)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
PHR	PHF	PLR	PLF	FLTLEBEN	CLLEBEN	—	—
bit 15						bit 8	
U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	BCH <sup>(1)</sup>	BCL <sup>(1)</sup>	BPHH	BPHL	BPLH	BPLL
bit 7						bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15      **PHR:** PWMxH 上升沿触发使能位  
 1 = PWMxH 的上升沿将触发前沿消隐计数器  
 0 = 前沿消隐忽略 PWMxH 的上升沿
- bit 14      **PHF:** PWMxH 下降沿触发使能位  
 1 = PWMxH 的下降沿将触发前沿消隐计数器  
 0 = 前沿消隐忽略 PWMxH 的下降沿
- bit 13      **PLR:** PWMxL 上升沿触发使能位  
 1 = PWMxL 的上升沿将触发前沿消隐计数器  
 0 = 前沿消隐忽略 PWMxL 的上升沿
- bit 12      **PLF:** PWMxL 下降沿触发使能位  
 1 = PWMxL 的下降沿将触发前沿消隐计数器  
 0 = 前沿消隐忽略 PWMxL 的下降沿
- bit 11      **FLTLEBEN:** 故障输入前沿消隐使能位  
 1 = 对选定故障输入应用前沿消隐  
 0 = 不对选定故障输入应用前沿消隐
- bit 10      **CLLEBEN:** 限流前沿消隐使能位  
 1 = 对选定限流输入应用前沿消隐  
 0 = 不对选定限流输入应用前沿消隐
- bit 9-6      **未实现:** 读为 0
- bit 5      **BCH:** 选定消隐信号高电平消隐使能位 <sup>(1)</sup>  
 1 = 当选定消隐信号为高电平时 (对限流和 / 或故障输入信号) 进行状态消隐  
 0 = 当选定消隐信号为高电平时不进行消隐
- bit 4      **BCL:** 选定消隐信号低电平消隐使能位 <sup>(1)</sup>  
 1 = 当选定消隐信号为低电平时 (对限流和 / 或故障输入信号) 进行状态消隐  
 0 = 当选定消隐信号为低电平时不进行消隐
- bit 3      **BPHH:** PWMxH 高电平消隐使能位  
 1 = 当 PWMxH 输出为高电平时 (对限流和 / 或故障输入信号) 进行状态消隐  
 0 = 当 PWMxH 输出为高电平时不进行消隐
- bit 2      **BPHL:** PWMxH 低电平消隐使能位  
 1 = 当 PWMxH 输出为低电平时 (对限流和 / 或故障输入信号) 进行状态消隐  
 0 = 当 PWMxH 输出为低电平时不进行消隐

**注 1:** 消隐信号通过 AUXCONx 寄存器中的 BLANKSEL<3:0> 位进行选择。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 16-24: LEBCONx: PWMx 前沿消隐 (LEB) 控制寄存器 (x = 1 至 8) (续)

- bit 1      **BPLH:** PWMxL 高电平消隐使能位  
 1 = 当 PWMxL 输出为高电平时 (对限流和 / 或故障输入信号) 进行状态消隐  
 0 = 当 PWMxL 输出为高电平时不进行消隐
- bit 0      **BPLL:** PWMxL 低电平消隐使能位  
 1 = 当 PWMxL 输出为低电平时 (对限流和 / 或故障输入信号) 进行状态消隐  
 0 = 当 PWMxL 输出为低电平时不进行消隐

注 1: 消隐信号通过 AUXCONx 寄存器中的 BLANKSEL<3:0> 位进行选择。

## 寄存器 16-25: LEBDLYx: PWMx 前沿消隐延时寄存器 (x = 1 至 8)

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	LEB<8:5>			
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
LEB<4:0>					—	—	—
bit 7							bit 0

### 图注:

R = 可读位      W = 可写位      U = 未实现位, 读为 0  
 -n = POR 时的值      1 = 置 1      0 = 清零      x = 未知

- bit 15-12      **未实现:** 读为 0
- bit 11-3      **LEB<8:0>:** 限流输入和故障输入前沿消隐延时位  
 值以 8.32 ns 为单位进行递增。
- bit 2-0      **未实现:** 读为 0

# dsPIC33EPXXGS70X/80X 系列

寄存器 16-26: **AUXCONx: PWMx 附属控制寄存器 (x = 1 至 8)**

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
HRPDIS	HRDDIS	—	—	BLANKSEL3	BLANKSEL2	BLANKSEL1	BLANKSEL0
bit 15				bit 8			

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CHOPSEL3	CHOPSEL2	CHOPSEL1	CHOPSEL0	CHOPHEN	CHOPLN
bit 7				bit 0			

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15        **HRPDIS:** 高分辨率 PWMx 周期禁止位  
 1 = 禁止高分辨率 PWMx 周期以降低功耗  
 0 = 使能高分辨率 PWMx 周期
- bit 14        **HRDDIS:** 高分辨率 PWMx 占空比禁止位  
 1 = 禁止高分辨率 PWMx 占空比以降低功耗  
 0 = 使能高分辨率 PWMx 占空比
- bit 13-12    **未实现:** 读为 0
- bit 11-8     **BLANKSEL<3:0>:** PWMx 状态消隐源选择位  
 选定的状态消隐信号将阻止限流和 / 或故障输入信号  
 (如果已通过 LEBCONx 寄存器中的 BCH 和 BCL 位使能)。  
 1001 = 保留  
 1000 = 选择 PWM8H 作为状态消隐源  
 0111 = 选择 PWM7H 作为状态消隐源  
 0110 = 选择 PWM6H 作为状态消隐源  
 0101 = 选择 PWM5H 作为状态消隐源  
 0100 = 选择 PWM4H 作为状态消隐源  
 0011 = 选择 PWM3H 作为状态消隐源  
 0010 = 选择 PWM2H 作为状态消隐源  
 0001 = 选择 PWM1H 作为状态消隐源  
 0000 = 不进行状态消隐
- bit 7-6      **未实现:** 读为 0
- bit 5-2      **CHOPSEL<3:0>:** PWMx 斩波时钟源选择位  
 选定信号将使能和禁止 (斩波) 选定的 PWMx 输出。  
 1001 = 保留  
 1000 = 选择 PWM8H 作为斩波时钟源  
 0111 = 选择 PWM7H 作为斩波时钟源  
 0110 = 选择 PWM6H 作为斩波时钟源  
 0101 = 选择 PWM5H 作为斩波时钟源  
 0100 = 选择 PWM4H 作为斩波时钟源  
 0011 = 选择 PWM3H 作为斩波时钟源  
 0010 = 选择 PWM2H 作为斩波时钟源  
 0001 = 选择 PWM1H 作为斩波时钟源  
 0000 = 选择斩波时钟发生器作为斩波时钟源
- bit 1        **CHOPHEN:** PWMxH 输出斩波使能位  
 1 = 使能 PWMxH 斩波功能  
 0 = 禁止 PWMxH 斩波功能
- bit 0        **CHOPLN:** PWMxL 输出斩波使能位  
 1 = 使能 PWMxL 斩波功能  
 0 = 禁止 PWMxL 斩波功能

# dsPIC33EPXXXGS70X/80X 系列

寄存器 16-27: PWMCAPx: PWMx 主时基捕捉寄存器 (x = 1 至 8)

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
PWMCAP<12:5> <sup>(1,2,3,4)</sup>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	U-0	U-0	U-0
PWMCAP<4:0> <sup>(1,2,3,4)</sup>					—	—	—
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-3 **PWMCAP<12:0>**: PWMx 主时基捕捉值位<sup>(1,2,3,4)</sup>

该寄存器中的值代表在限流输入上检测到前沿时捕捉的 PWMx 时基值。

bit 2-0 **未实现**: 读为 0

**注 1:** 捕捉功能仅对于主输出 (PWMxH) 可用。

**2:** 只有限流输入信号的 LEB 处理完成之后, 该功能才有效。

**3:** 最小捕捉分辨率为 8.32 ns。

**4:** 当 XPRES 位 (PWMCONx<1>) 设置为 0 时, 可以使用该功能。

## 17.0 外设触发信号发生器（PTG）模块

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“外设触发信号发生器（PTG）”（DS70669），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

### 17.1 模块简介

外设触发信号发生器（Peripheral Trigger Generator, PTG）可用于调度复杂的高速外设操作，这是使用软件难以实现的。PTG 模块使用称为“步阶”的 8 位命令，它们由用户写入 PTG 队列寄存器

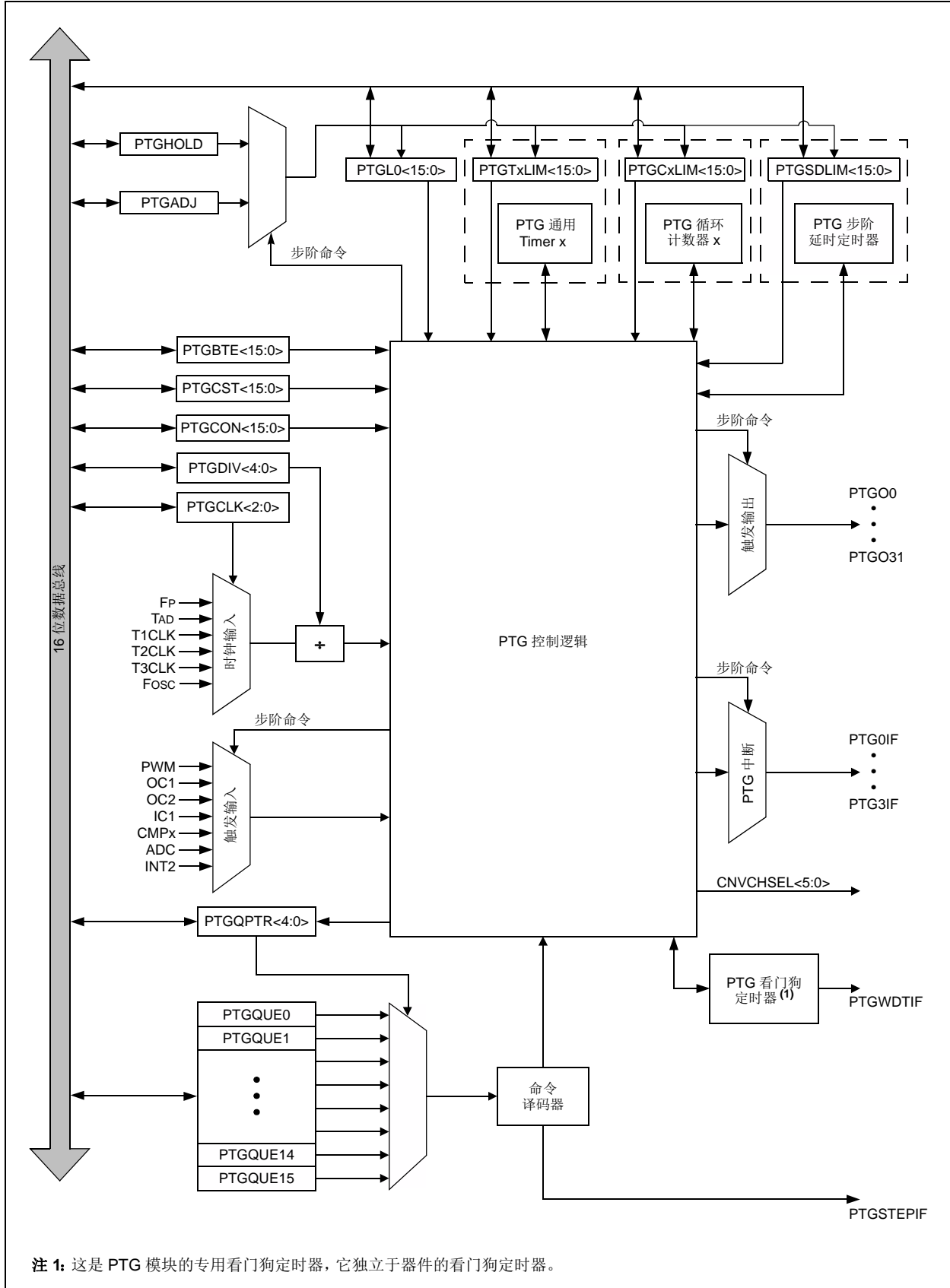
（PTGQUE0-PTQUE15），执行诸如等待输入信号、生成输出触发信号和等待定时器之类的操作。

PTG 模块具有以下主要特性：

- 多个时钟源
- 两个 16 位通用定时器
- 两个 16 位通用限制计数器
- 可配置为上升沿或下降沿触发
- 可产生处理器中断，包括：
  - 4 个可配置处理器中断
  - 单步模式下发生步阶事件时产生中断
  - PTG 看门狗定时器超时时产生中断
- 能够接收来自以下外设的触发信号：
  - ADC
  - PWM
  - 输出比较
  - 输入捕捉
  - 比较器
  - INT2
- 能够触发或同步以下外设：
  - 看门狗定时器
  - 输出比较
  - 输入捕捉
  - ADC
  - PWM
  - 比较器

# dsPIC33EPXXGS70X/80X 系列

图 17-1: PTG 框图



# dsPIC33EPXXGS70X/80X 系列

## 17.2 PTG 控制寄存器

寄存器 17-1: PTGCST: PTG 控制 / 状态寄存器

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTGEN	—	PTGSIDL	PTGTOGL	—	PTGSWT <sup>(2)</sup>	PTGSSEN	PTGIVIS
bit 15							bit 8

R/W-0	HS-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
PTGSTRT	PTGWDTO	—	—	—	—	PTGITM1 <sup>(1)</sup>	PTGITM0 <sup>(1)</sup>
bit 7							bit 0

<b>图注:</b>	HS = 硬件置 1 位
R = 可读位	W = 可写位
-n = POR 时的值	1 = 置 1
	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

- bit 15      **PTGEN:** PTG 模块使能位  
1 = 使能 PTG 模块  
0 = 禁止 PTG 模块
- bit 14      **未实现:** 读为 0
- bit 13      **PTGSIDL:** PTG 空闲模式停止位  
1 = 当器件进入空闲模式时, 模块停止工作  
0 = 在空闲模式下模块继续工作
- bit 12      **PTGTOGL:** PTG TRIG 输出翻转模式位  
1 = 每次执行 PTGTRIG 命令后翻转 PTGOx 的状态  
0 = 每次执行 PTGTRIG 命令后产生一个由 PTGPWDx 位的值决定的 PTGOx 脉冲
- bit 11      **未实现:** 读为 0
- bit 10      **PTGSWT:** PTG 软件触发位<sup>(2)</sup>  
1 = 触发 PTG 模块  
0 = 无操作 (清零该位将不起任何作用)
- bit 9        **PTGSSEN:** PTG 使能单步位  
1 = 使能单步模式  
0 = 禁止单步模式
- bit 8        **PTGIVIS:** PTG 计数器 / 定时器可视性控制位  
1 = 读取 PTGSDLIM、PTGCxLIM 或 PTGTxLIM 寄存器时返回它们相应的计数器 / 定时器寄存器 (PTGSD、PTGCx 和 PTGTx) 的当前值  
0 = 读取 PTGSDLIM、PTGCxLIM 或 PTGTxLIM 寄存器时返回先前写入这些 PTG 限制寄存器的值
- bit 7        **PTGSTRT:** 启动 PTG 序列器位  
1 = 开始连续执行命令 (连续模式)  
0 = 停止执行命令
- bit 6        **PTGWDTO:** PTG 看门狗定时器超时状态位  
1 = PTG 看门狗定时器已超时  
0 = PTG 看门狗定时器未超时
- bit 5-2      **未实现:** 读为 0

**注 1:** 这些位仅适用于 PTGWHI 和 PTGWLO 命令。  
**2:** 该位仅用于 PTGCTRL 步阶命令软件触发选项。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 17-1: PTGCST: PTG 控制 / 状态寄存器 (续)

bit 1-0      **PTGITM<1:0>**: PTG 输入触发命令工作模式位 <sup>(1)</sup>

- 11 = 在退出命令时不执行带步阶延时的单电平检测 (无论 PTGCTRL 命令如何)
- 10 = 在退出命令时执行带步阶延时的单电平检测
- 01 = 在退出命令时不执行带步阶延时的连续边沿检测 (无论 PTGCTRL 命令如何)
- 00 = 在退出命令时执行带步阶延时的连续边沿检测

- 注 1:** 这些位仅适用于 PTGWHI 和 PTGWLO 命令。  
**2:** 该位仅用于 PTGCTRL 步阶命令软件触发选项。



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 17-2: PTGCON: PTG 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGCLK2	PTGCLK1	PTGCLK0	PTGDIV4	PTGDIV3	PTGDIV2	PTGDIV1	PTGDIV0
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTGPWD3	PTGPWD2	PTGPWD1	PTGPWD0	—	PTGWDT2	PTGWDT1	PTGWDT0
bit 7						bit 0	

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-13 **PTGCLK<2:0>**: PTG 模块时钟源选择位

- 111 = CLC2
- 110 = CLC1
- 101 = PTG 模块时钟源将为 T3CLK
- 100 = PTG 模块时钟源将为 T2CLK
- 011 = PTG 模块时钟源将为 T1CLK
- 010 = PTG 模块时钟源将为 TAD
- 001 = PTG 模块时钟源将为 FOSC
- 000 = PTG 模块时钟源将为 FP

bit 12-8 **PTGDIV<4:0>**: PTG 模块时钟预分频比位

- 11111 = 32 分频
- 11110 = 31 分频
- 
- 
- 
- 00001 = 2 分频
- 00000 = 1 分频

bit 7-4 **PTGPWD<3:0>**: PTG 触发输出脉冲宽度位

- 1111 = 所有触发输出的宽度为 16 个 PTG 时钟周期
- 1110 = 所有触发输出的宽度为 15 个 PTG 时钟周期
- 
- 
- 
- 0001 = 所有触发输出的宽度为 2 个 PTG 时钟周期
- 0000 = 所有触发输出的宽度为 1 个 PTG 时钟周期

bit 3 **未实现**: 读为 0

bit 2-0 **PTGWDT<2:0>**: PTG 看门狗定时器超时计数值选择位

- 111 = 看门狗定时器在 512 个 PTG 时钟之后超时
- 110 = 看门狗定时器在 256 个 PTG 时钟之后超时
- 101 = 看门狗定时器在 128 个 PTG 时钟之后超时
- 100 = 看门狗定时器在 64 个 PTG 时钟之后超时
- 011 = 看门狗定时器在 32 个 PTG 时钟之后超时
- 010 = 看门狗定时器在 16 个 PTG 时钟之后超时
- 001 = 看门狗定时器在 8 个 PTG 时钟之后超时
- 000 = 禁止看门狗定时器

# dsPIC33EPXXGS70X/80X 系列

寄存器 17-3: PTGBTE: PTG 广播触发使能寄存器<sup>(1,2)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCTS4	ADCTS3	ADCTS2	ADCTS1	IC4TSS	IC3TSS	IC2TSS	IC1TSS
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
OC4CS	OC3CS	OC2CS	OC1CS	OC4TSS	OC3TSS	OC2TSS	OC1TSS
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15        **ADCTS4:** ADCx 的采样触发 PTGO15 位  
               1 = 在执行广播命令时产生触发信号  
               0 = 在执行广播命令时不产生触发信号
- bit 14        **ADCTS3:** ADCx 的采样触发 PTGO14 位  
               1 = 在执行广播命令时产生触发信号  
               0 = 在执行广播命令时不产生触发信号
- bit 13        **ADCTS2:** ADCx 的采样触发 PTGO13 位  
               1 = 在执行广播命令时产生触发信号  
               0 = 在执行广播命令时不产生触发信号
- bit 12        **ADCTS1:** ADCx 的采样触发 PTGO12 位  
               1 = 在执行广播命令时产生触发信号  
               0 = 在执行广播命令时不产生触发信号
- bit 11        **IC4TSS:** IC4 的触发 / 同步源位  
               1 = 在执行广播命令时产生触发 / 同步信号  
               0 = 在执行广播命令时不产生触发 / 同步信号
- bit 10        **IC3TSS:** IC3 的触发 / 同步源位  
               1 = 在执行广播命令时产生触发 / 同步信号  
               0 = 在执行广播命令时不产生触发 / 同步信号
- bit 9         **IC2TSS:** IC2 的触发 / 同步源位  
               1 = 在执行广播命令时产生触发 / 同步信号  
               0 = 在执行广播命令时不产生触发 / 同步信号
- bit 8         **IC1TSS:** IC1 的触发 / 同步源位  
               1 = 在执行广播命令时产生触发 / 同步信号  
               0 = 在执行广播命令时不产生触发 / 同步信号
- bit 7         **OC4CS:** OC4 的时钟源位  
               1 = 在执行广播命令时产生时钟脉冲  
               0 = 在执行广播命令时不产生时钟脉冲
- bit 6         **OC3CS:** OC3 的时钟源位  
               1 = 在执行广播命令时产生时钟脉冲  
               0 = 在执行广播命令时不产生时钟脉冲
- bit 5         **OC2CS:** OC2 的时钟源位  
               1 = 在执行广播命令时产生时钟脉冲  
               0 = 在执行广播命令时不产生时钟脉冲

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。  
**注 2:** 该寄存器仅用于 PTGCTRL OPTION = 1111 步阶命令。

## 寄存器 17-3: PTGBTE: PTG 广播触发使能寄存器<sup>(1,2)</sup> (续)

bit 4	<b>OC1CS:</b> OC1 的时钟源位 1 = 在执行广播命令时产生时钟脉冲 0 = 在执行广播命令时不产生时钟脉冲
bit 3	<b>OC4TSS:</b> OC4 的触发 / 同步源位 1 = 在执行广播命令时产生触发 / 同步信号 0 = 在执行广播命令时不产生触发 / 同步信号
bit 2	<b>OC3TSS:</b> OC3 的触发 / 同步源位 1 = 在执行广播命令时产生触发 / 同步信号 0 = 在执行广播命令时不产生触发 / 同步信号
bit 1	<b>OC2TSS:</b> OC2 的触发 / 同步源位 1 = 在执行广播命令时产生触发 / 同步信号 0 = 在执行广播命令时不产生触发 / 同步信号
bit 0	<b>OC1TSS:</b> OC1 的触发 / 同步源位 1 = 在执行广播命令时产生触发 / 同步信号 0 = 在执行广播命令时不产生触发 / 同步信号

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

**2:** 该寄存器仅用于 PTGCTRL OPTION = 1111 步阶命令。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 17-4: PTGT0LIM: PTG Timer0 限制寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT0LIM<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT0LIM<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-0        **PTGT0LIM<15:0>**: PTG Timer0 限制寄存器位  
通用 Timer0 限制寄存器 (仅对于 PTGT0 步阶命令有效)。

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

## 寄存器 17-5: PTGT1LIM: PTG Timer1 限制寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT1LIM<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGT1LIM<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-0        **PTGT1LIM<15:0>**: PTG Timer1 限制寄存器位  
通用 Timer1 限制寄存器 (仅对于 PTGT1 步阶命令有效)。

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 17-6: PTGSDLIM: PTG 步阶延时限制寄存器 (1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGSDLIM<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGSDLIM<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0            **PTGSDLIM<15:0>**: PTG 步阶延时限制寄存器位  
 保存一个 PTG 步阶延时值, 该值代表在步阶命令启动和步阶命令完成之间的额外 PTG 时钟数量。

- 注 1:** 对于写入 PTGSDLIM 寄存器的任何值, 都会加上一个 PTG 时钟的基本步阶延时 (步阶延时 = (PTGSDLIM) + 1)。  
**2:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

## 寄存器 17-7: PTGC0LIM: PTG 计数器 0 限制寄存器 (1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC0LIM<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC0LIM<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0            **PTGC0LIM<15:0>**: PTG 计数器 0 限制寄存器位  
 可用于指定 PTGJMPC0 步阶命令的循环计数, 或作为通用计数器 0 的限制寄存器。

- 注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 17-8: PTGC1LIM: PTG 计数器 1 限制寄存器 (1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC1LIM<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGC1LIM<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-0      **PTGC1LIM<15:0>**: PTG 计数器 1 限制寄存器位  
可用于指定 PTGJMPC1 步阶命令的循环计数, 或作为通用计数器 1 的限制寄存器。

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

## 寄存器 17-9: PTGHOLD: PTG 保持寄存器 (1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGHOLD<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGHOLD<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-0      **PTGHOLD<15:0>**: PTG 通用保持寄存器位  
保存用户提供的、要使用 PTGCOPY 命令复制到 PTGTxLIM、PTGCxLIM、PTGSDLIM 或 PTGL0 寄存器的数据。

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 17-10: PTGADJ: PTG 调节寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGADJ<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGADJ<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0            **PTGADJ<15:0>**: PTG 调节寄存器位  
 该寄存器保存用户提供的、要使用 PTGADD 命令加到 PTGTxLIM、PTGCxLIM、PTGSDLIM 或 PTGL0 寄存器的数据。

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

## 寄存器 17-11: PTGL0: PTG 立即数 0 寄存器<sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGL0<15:8>							
bit 15				bit 8			

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGL0<7:0>							
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0            **PTGL0<15:0>**: PTG 立即数 0 寄存器位  
 该寄存器保存要使用 PTGCTRL 步阶命令写入 CNVCHSEL<5:0> 位 (ADCON3L<5:0>) 的 6 位值。

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 17-12: PTGQPTR: PTG 步阶队列指针寄存器 <sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
—	—	—	PTGQPTR<4:0>					—
bit 7							bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-5            **未实现:** 读为 0  
 bit 4-0            **PTGQPTR<4:0>:** PTG 步阶队列指针寄存器位  
 该寄存器指向步阶队列中当前处于活动状态的步阶命令。

**注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。

## 寄存器 17-13: PTGQUE<sub>x</sub>: PTG 步阶队列寄存器 x (x = 0-15) <sup>(1,3)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STEP(2x + 1)<7:0> <sup>(2)</sup>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STEP(2x)<7:0> <sup>(2)</sup>							
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            **STEP(2x + 1)<7:0>:** PTG 步阶队列指针寄存器位 <sup>(2)</sup>  
 用于存储 STEP(2x + 1) 命令字节的队列位置。  
 bit 7-0            **STEP(2x)<7:0>:** PTG 步阶队列指针寄存器位 <sup>(2)</sup>  
 用于存储 STEP(2x) 命令字节的队列位置。

- 注 1:** 在 PTG 模块执行步阶命令 (PTGEN = 1 且 PTGSTRT = 1) 时, 该寄存器是只读的。  
**注 2:** 关于步阶命令编码, 请参见表 17-1。  
**注 3:** 在发生任何类型的复位时, 步阶寄存器都会保持它们的值。



# dsPIC33EPXXGS70X/80X 系列

## 17.3 步阶命令和格式

表 17-1: PTG 步阶命令格式

步阶命令字节:			
STEPx<7:0>			
CMD<3:0>		OPTION<3:0>	
bit 7	bit 4	bit 3	bit 0

bit 7-4	CMD<3:0>	步阶命令	命令说明
	0000	PTGCTRL	根据 OPTION<3:0> 的描述执行控制命令
	0001	PTGADD	根据 OPTION<3:0> 的描述, 将 PTGADJ 寄存器的内容加到目标寄存器
		PTGCOPY	根据 OPTION<3:0> 的描述, 将 PTGHOLD 寄存器的内容复制到目标寄存器
	001x	PTGSTRB	将 CMD0:OPTION<3:0> 中包含的值复制到 CNVCHSEL<5:0> 位 (ADCON3L<5:0>)
	0100	PTGWHI	根据 OPTION<3:0> 的描述, 等待来自选定 PTG 触发输入从低电平变为高电平的边沿输入
	0101	PTGWLO	根据 OPTION<3:0> 的描述, 等待来自选定 PTG 触发输入从高电平变为低电平的边沿输入
	0110	保留	保留
	0111	PTGIRQ	根据 OPTION<3:0> 的描述, 产生独立的中断请求
	100x	PTGTRIG	根据 <<CMD0>;OPTION<3:0>> 的描述, 产生独立的触发输出
	101x	PTGJMP	将 <<CMD0>;OPTION<3:0>> 中指示的值复制到 PTG 队列指针 (PTGQPTR) 处, 然后跳转到该步阶队列
	110x	PTGJMPC0	PTGC0 = PTGC0LIM: 递增 PTG 队列指针 (PTGQPTR) PTGC0 ≠ PTGC0LIM: 递增 PTG 计数器 0 (PTGC0), 并将 <<CMD0>;OPTION<3:0>> 中指示的值复制到 PTG 队列指针 (PTGQPTR) 处, 然后跳转到该步阶队列
	111x	PTGJMPC1	PTGC1 = PTGC1LIM: 递增 PTG 队列指针 (PTGQPTR) PTGC1 ≠ PTGC1LIM: 递增 PTG 计数器 1 (PTGC1), 并将 <<CMD0>;OPTION<3:0>> 中指示的值复制到 PTG 队列指针 (PTGQPTR) 处, 然后跳转到该步阶队列

- 注 1: 所有保留的命令或选项将执行, 但它们不起任何作用 (即, 作为一条 NOP 指令执行)。  
 注 2: 关于触发输出的说明, 请参见表 17-2。

# dsPIC33EPXXGS70X/80X 系列

表 17-1: PTG 步阶命令格式 (续)

bit 3-0	步阶命令	OPTION<3:0>	选项说明
	PTGCTRL <sup>(1)</sup>	0000	保留
		0001	保留
		0010	禁止 PTG 步阶延时定时器 (PTGSD)
		0011	保留
		0100	保留
		0101	保留
		0110	使能 PTG 步阶延时定时器 (PTGSD)
		0111	保留
		1000	启动并等待 PTG Timer0 与 PTG Timer0 限制寄存器发生匹配
		1001	启动并等待 PTG Timer1 与 PTG Timer1 限制寄存器发生匹配
		1010	保留
		1011	等待软件触发位从低变为高, 然后再继续 (PTGSWT = 0 至 1)
		1100	将 PTG 计数器 0 寄存器的内容复制到 CNVCHSEL<5:0> 位 (ADCON3L<5:0>)
		1101	将 PTG 计数器 1 寄存器的内容复制到 CNVCHSEL<5:0> 位 (ADCON3L<5:0>)
		1110	将 PTG 立即数 0 寄存器的内容复制到 CNVCHSEL<5:0> 位 (ADCON3L<5:0>)
		1111	产生 PTG 广播触发使能寄存器 (PTGBTE) 中指示的触发信号
	PTGADD <sup>(1)</sup>	0000	将 PTGADJ 寄存器的内容加到 PTG 计数器 0 限制寄存器 (PTGC0LIM)
		0001	将 PTGADJ 寄存器的内容加到 PTG 计数器 1 限制寄存器 (PTGC1LIM)
		0010	将 PTGADJ 寄存器的内容加到 PTG Timer0 限制寄存器 (PTGT0LIM)
		0011	将 PTGADJ 寄存器的内容加到 PTG Timer1 限制寄存器 (PTGT1LIM)
		0100	将 PTGADJ 寄存器的内容加到 PTG 步阶延时限制寄存器 (PTGSDLIM)
		0101	将 PTGADJ 寄存器的内容加到 PTG 立即数 0 寄存器 (PTGL0)
		0110	保留
		0111	保留
	PTGCOPY <sup>(1)</sup>	1000	将 PTGHOLD 寄存器的内容复制到 PTG 计数器 0 限制寄存器 (PTGC0LIM)
		1001	将 PTGHOLD 寄存器的内容复制到 PTG 计数器 1 限制寄存器 (PTGC1LIM)
		1010	将 PTGHOLD 寄存器的内容复制到 PTG Timer0 限制寄存器 (PTGT0LIM)
		1011	将 PTGHOLD 寄存器的内容复制到 PTG Timer1 限制寄存器 (PTGT1LIM)
		1100	将 PTGHOLD 寄存器的内容复制到 PTG 步阶延时限制寄存器 (PTGSDLIM)
		1101	将 PTGHOLD 寄存器的内容复制到 PTG 立即数 0 寄存器 (PTGL0)
		1110	保留
		1111	保留

注 1: 所有保留的命令或选项将执行, 但它们不起任何作用 (即, 作为一条 NOP 指令执行)。

2: 关于触发输出的说明, 请参见表 17-2。

# dsPIC33EPXXGS70X/80X 系列

**表 17-1: PTG 步阶命令格式 (续)**

bit 3-0	步阶命令	OPTION<3:0>	选项说明
	PTGWHI <sup>(1)</sup> 或 PTGWLO <sup>(1)</sup>	0000	PWM 特殊事件触发
		0001	PWM 主控时基同步输出
		0010	PWM1 中断
		0011	PWM2 中断
		0100	PWM3 中断
		0101	PWM4 中断
		0110	PWM5 中断
		0111	OC1 触发事件
		1000	OC2 触发事件
		1001	IC1 触发事件
		1010	CMP1 触发事件
		1011	CMP2 触发事件
		1100	CMP3 触发事件
		1101	CMP4 触发事件
		1110	ADC 转换完成中断
		1111	INT2 外部中断
	PTGIRQ <sup>(1)</sup>	0000	产生 PTG 中断 0
		0001	产生 PTG 中断 1
		0010	产生 PTG 中断 2
		0011	产生 PTG 中断 3
		0100	保留
		•	•
		•	•
		•	•
	1111	保留	
	PTGTRIG <sup>(2)</sup>	00000	PTGO0
		00001	PTGO1
		•	•
		•	•
		•	•
		11110	PTGO30
	11111	PTGO31	

**注 1:** 所有保留的命令或选项将执行，但它们不起任何作用（即，作为一条 NOP 指令执行）。

**2:** 关于触发输出的说明，请参见表 17-2。

# dsPIC33EPXXGS70X/80X 系列

表 17-2: PTG 输出说明

PTG 输出编号	PTG 输出说明
PTGO0	OC1 的触发 / 同步源
PTGO1	OC2 的触发 / 同步源
PTGO2	OC3 的触发 / 同步源
PTGO3	OC4 的触发 / 同步源
PTGO4	OC1 的时钟源
PTGO5	OC2 的时钟源
PTGO6	OC3 的时钟源
PTGO7	OC4 的时钟源
PTGO8	IC1 的触发 / 同步源
PTGO9	IC2 的触发 / 同步源
PTGO10	IC3 的触发 / 同步源
PTGO11	IC4 的触发 / 同步源
PTGO12	ADC 的采样触发
PTGO13	保留
PTGO14	保留
PTGO15	保留
PTGO16	PWM3 的 PWM 时基同步源
PTGO17	PWM4 的 PWM 时基同步源
PTGO18	PWM5 的 PWM 时基同步源
PTGO19	PWM6 的 PWM 时基同步源
PTGO20	保留
PTGO21	保留
PTGO22	保留
PTGO23	保留
PTGO24	保留
PTGO25	保留
PTGO26	CLC1 输入
PTGO27	CLC2 输入
PTGO28	CLC3 输入
PTGO29	CLC4 输入
PTGO30	PTG 输出至 PPS 输入选择 RPI6
PTGO31	PTG 输出至 PPS 输入选择 RPI7

## 18.0 串行外设接口 (SPI)

**注：** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“带音频编解码器支持的串行外设接口 (SPI)” (DS70005136)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

串行外设接口 (Serial Peripheral Interface, SPI) 模块是用于同其他外设或单片机进行通信的同步串行接口。这些外设可以是串行 EEPROM、移位寄存器、显示驱动器和 A/D 转换器等。SPI 模块与 Motorola® 的 SPI 和 SIOP 接口兼容。dsPIC33EPXXGS70X/80X 系列的所有器件都包含 3 个 SPI 模块。

该模块支持在两种缓冲区模式下工作。在标准模式下，数据通过单个串行缓冲区移位。在增强型缓冲区模式下，数据通过一个 FIFO 缓冲区移位。FIFO 深度取决于所配置的模式。

该模块可以发送和接收从 2 到 32 位的可变长度数据。

**注：** 在标准或增强型缓冲区模式下，都不要对 SPIxBUF 寄存器执行读 - 修改 - 写操作 (如针对位的指令)。

工作于主或从模式时，模块还支持基本的帧 SPI 协议。共支持 4 种帧 SPI 配置。

SPI3 还支持音频模式。有 4 种不同的音频模式可供使用。

- I<sup>2</sup>S
- 左对齐
- 右对齐
- PCM/DSP

在其中每种模式下，串行时钟是自由运行的，而音频数据总是在传输。

如果在两个器件之间发生音频协议数据传输，则通常一个器件是主器件，另一个是从器件。但是，音频数据可以在两个从器件之间传输。由于音频协议需要自由运行的时钟，所以主器件可以是第三方控制器。在两种情况下，主器件都会产生两个自由运行的时钟：SCKx 和 LRC (左右通道时钟 /SSx/FSYNC)。

SPI 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
- SSx: 低电平有效从选择或帧同步 I/O 脉冲

SPI 模块可以配置为使用 2、3 或 4 个引脚工作。在 3 引脚模式下，不使用 SSx。在 2 引脚模式下，SDOx 和 SSx 均不使用。

SPI 模块可以产生 3 个反映在数据通信期间所发生事件的中断。它可以产生以下类型的中断：

1. 接收中断，通过 SPIxRXIF 来指示。在以下条件下会发生该事件：
  - 接收水印中断
  - SPIROV = 1
  - SPIRBF = 1
  - SPIRBE = 1前提是在 SPIxIMSKL/H 中使能了相应的掩码位。
2. 发送中断，通过 SPIxTXIF 来指示。在以下条件下会发生该事件：
  - 发送水印中断
  - SPITUR = 1
  - SPITBF = 1
  - SPITBE = 1前提是在 SPIxIMSKL/H 中使能了相应的掩码位。
3. 通用中断，通过 SPIxIF 来指示。在以下条件下会发生该事件：
  - FRMERR = 1
  - SPIBUSY = 1
  - SRMT = 1前提是在 SPIxIMSKL/H 中使能了相应的掩码位。

标准和增强型模式下的模块框图如图 18-1 和图 18-2 所示。

**注：** 在本章中，SPI 模块统称为 SPIx，或分别称为 SPI1、SPI2 或 SPI3。特殊功能寄存器也使用类似的符号表示。例如，SPIxCON1 和 SPIxCON2 指 3 个 SPI 模块中任何一个的控制寄存器。

# dsPIC33EPXXGS70X/80X 系列

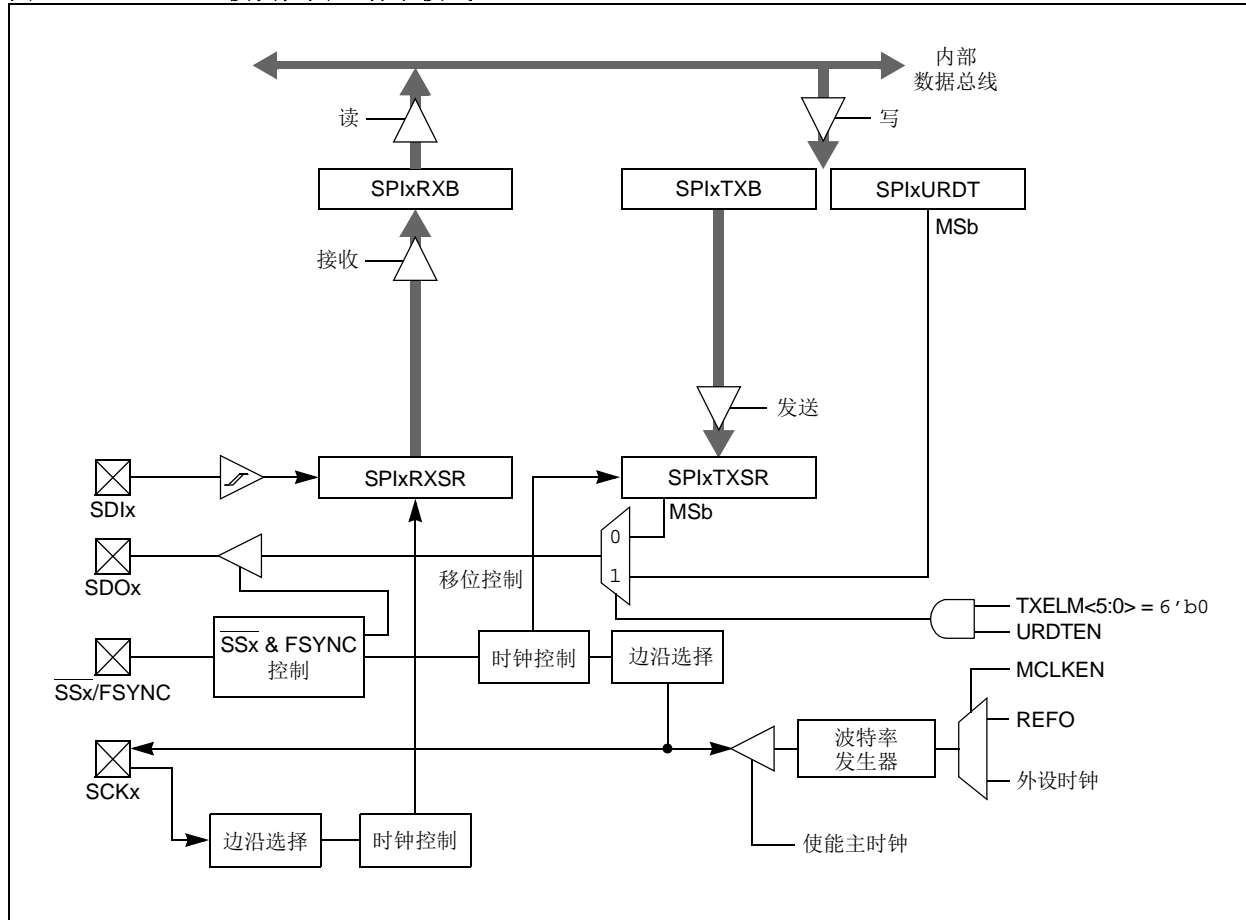
要将 SPIx 模块设置为工作于标准主模式：

1. 如果使用中断：
  - a) 将相应 IFSx 寄存器中的中断标志位清零。
  - b) 将相应 IECx 寄存器中的中断允许位置 1。
  - c) 通过写相应 IPCx 寄存器中的 SPIxIP 位来设置中断优先级。
2. 将所需的设置写入 SPIxCON1L 和 SPIxCON1H 寄存器，且 MSTEN 位 (SPIxCON1L<5>) = 1。
3. 将 SPIROV 位 (SPIxSTATL<6>) 清零。
4. 通过将 SPIEN 位 (SPIxCON1L<15>) 置 1 使能 SPIx 操作。
5. 将待发送数据写入 SPIxBUFL 和 SPIxBUFH 寄存器。发送 (和接收) 在数据写入 SPIxBUFL 和 SPIxBUFH 寄存器时立即开始。

要将 SPIx 模块设置为工作于标准从模式：

1. 将 SPIxBUF 寄存器清零。
2. 如果使用中断：
  - a) 将 SPIxBUFL 和 SPIxBUFH 寄存器清零。
  - b) 将相应 IECx 寄存器中的中断允许位置 1。
  - c) 通过写相应 IPCx 寄存器中的 SPIxIP 位来设置中断优先级。
3. 将所需的设置写入 SPIxCON1L、SPIxCON1H 和 SPIxCON2L 寄存器，且 MSTEN 位 (SPIxCON1L<5>) = 0。
4. 将 SMP 位清零。
5. 如果 CKE 位 (SPIxCON1L<8>) 置 1，则必须将 SSx 位 (SPIxCON1L<7>) 置 1 来使能 SSx 引脚。
6. 将 SPIROV 位 (SPIxSTATL<6>) 清零。
7. 通过将 SPIEN 位 (SPIxCON1L<15>) 置 1 使能 SPIx 操作。

图 18-1: SPIx 模块框图 (标准模式)



# dsPIC33EPXXGS70X/80X 系列

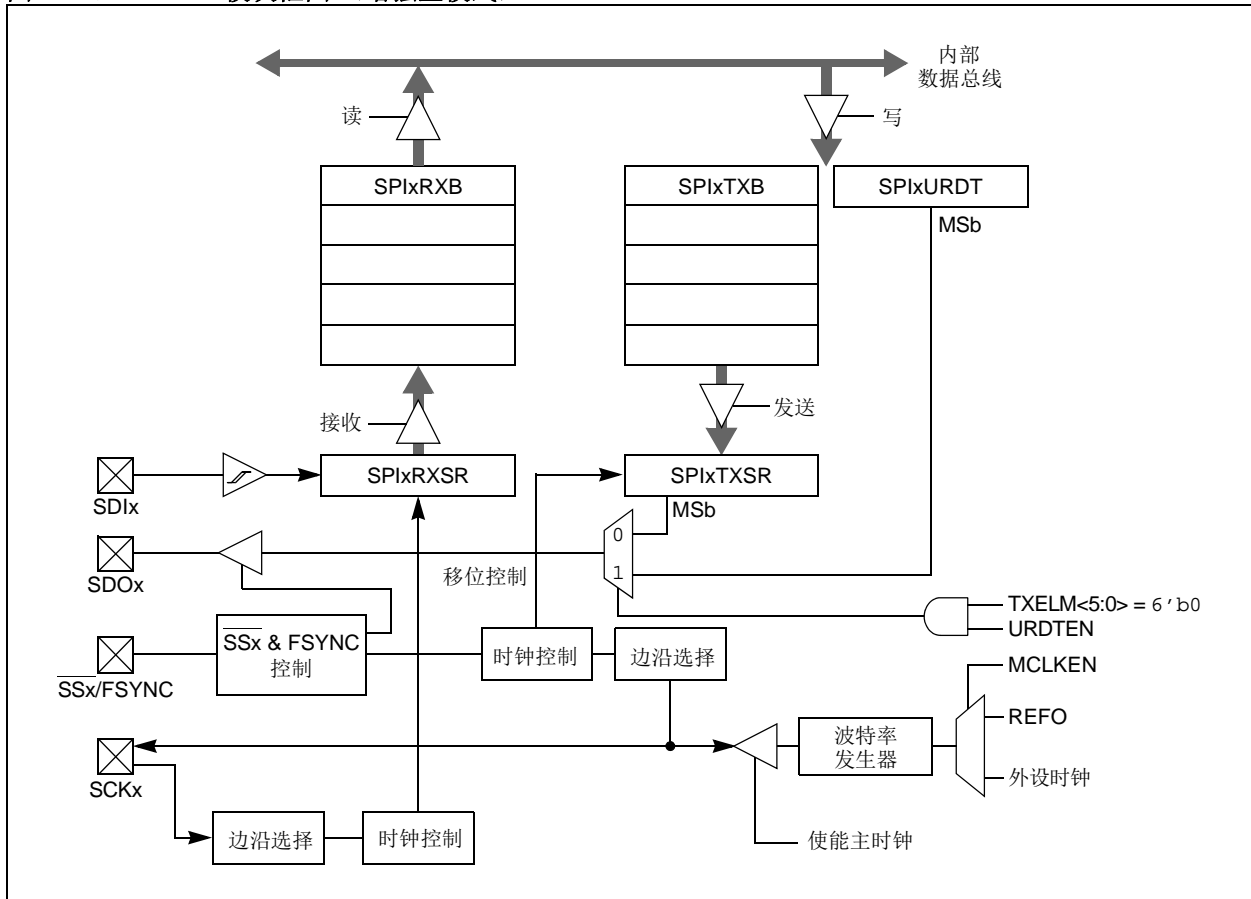
要将 SPIx 模块设置为工作于增强型缓冲区主模式：

1. 如果使用中断：
  - a) 将相应 IFSx 寄存器中的中断标志位清零。
  - b) 将相应 IECx 寄存器中的中断允许位置 1。
  - c) 写相应 IPCx 寄存器中的 SPIxIP 位。
2. 将所需的设置写入 SPIxCON1L、SPIxCON1H 和 SPIxCON2L 寄存器，且 MSTEN (SPIxCON1L<5>) = 1。
3. 将 SPIROV 位 (SPIxSTATL<6>) 清零。
4. 通过将 ENHBUF 位 (SPIxCON1L<0>) 置 1 选择增强型缓冲区模式。
5. 通过将 SPIEN 位 (SPIxCON1L<15>) 置 1 使能 SPIx 操作。
6. 将待发送数据写入 SPIxBUFL 和 SPIxBUFH 寄存器。发送 (和接收) 在数据写入 SPIxBUFL 和 SPIxBUFH 寄存器时立即开始。

要将 SPIx 模块设置为工作于增强型缓冲区从模式：

1. 将 SPIxBUFL 和 SPIxBUFH 寄存器清零。
2. 如果使用中断：
  - a) 将相应 IFSx 寄存器中的中断标志位清零。
  - b) 将相应 IECx 寄存器中的中断允许位置 1。
  - c) 通过写相应 IPCx 寄存器中的 SPIxIP 位来设置中断优先级。
3. 将所需的设置写入 SPIxCON1L、SPIxCON1H 和 SPIxCON2L 寄存器，且 MSTEN 位 (SPIxCON1L<5>) = 0。
4. 将 SMP 位清零。
5. 如果 CKE 位置 1，则必须将 SEN 位置 1 来使能 SSx 引脚。
6. 将 SPIROV 位 (SPIxSTATL<6>) 清零。
7. 通过将 ENHBUF 位 (SPIxCON1L<0>) 置 1 选择增强型缓冲区模式。
8. 通过将 SPIEN 位 (SPIxCON1L<15>) 置 1 使能 SPIx 操作。

图 18-2: SPIx 模块框图 (增强型模式)



# dsPIC33EPXXGS70X/80X 系列

要将 SPIx 模块设置为音频模式：

1. 将 SPIxBUFL 和 SPIxBUFH 寄存器清零。
2. 如果使用中断：
  - a) 将相应 IFSx 寄存器中的中断标志位清零。
  - b) 将相应 IECx 寄存器中的中断允许位置 1。
  - a) 通过写相应 IPCx 寄存器中的 SPIxIP 位来设置中断优先级。
3. 将所需的设置写入 SPIxCON1L、SPIxCON1H 和 SPIxCON2L 寄存器，且 AUDEN (SPIxCON1H<15>) = 1。
4. 将 SPIROV 位 (SPIxSTATL<6>) 清零。
5. 通过将 SPIEN 位 (SPIxCON1L<15>) 置 1 使能 SPIx 操作。
6. 将待发送数据写入 SPIxBUFL 和 SPIxBUFH 寄存器。发送 (和接收) 在数据写入 SPIxBUFL 和 SPIxBUFH 寄存器时立即开始。

## 寄存器 18-1: SPIxCON1L: SPIx 控制寄存器 1 低位字

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SPIEN	—	SPISIDL	DISSDO	MODE32 <sup>(1,4)</sup>	MODE16 <sup>(1,4)</sup>	SMP	CKE <sup>(1)</sup>
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN <sup>(2)</sup>	CKP	MSTEN	DISSDI	DISSCK	MCLKEN <sup>(3)</sup>	SPIFE	ENHBUF
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位，读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15        **SPIEN:** SPIx 使能位  
 1 = 使能模块  
 0 = 关闭并复位模块，禁止时钟，禁止中断事件产生，允许 SFR 修改
- bit 14        **未实现:** 读为 0
- bit 13        **SPISIDL:** SPIx 空闲模式停止位  
 1 = 在 CPU 空闲模式下停止工作  
 0 = 在 CPU 空闲模式下继续工作
- bit 12        **DISSDO:** 禁止 SDOx 输出端口位  
 1 = 模块不使用 SDOx 引脚；引脚由端口功能控制  
 0 = SDOx 引脚由模块控制
- bit 11-10    **MODE32 和 MODE16:** 串行字长度选择位 <sup>(1,4)</sup>

MODE32	MODE16	AUDEN	通信
1	x	0	32 位
0	1		16 位
0	0		8 位
1	1	1	24 位数据，32 位 FIFO，32 位通道 /64 位帧
1	0		32 位数据，32 位 FIFO，32 位通道 /64 位帧
0	1		16 位数据，16 位 FIFO，32 位通道 /64 位帧
0	0		16 位 FIFO，16 位通道 /32 位帧

- 注 1: 当 AUDEN (SPIxCON1H<15>) = 1 时，该模块按 CKE = 0 时的情况工作，不论其实际值如何。
- 2: 当 FRMEN = 1 时，不使用 SSEN。
- 3: 只有 SPIEN 位 = 0 时，才能写入 MCLKEN。
- 4: 该通道对于 DSP/PCM 模式没有意义，因为 LRC 会遵循 FRMSYPW。



## 寄存器 18-1: SPIxCON1L: SPIx 控制寄存器 1 低位字 (续)

bit 9	<b>SMP:</b> SPIx 数据输入采样阶段位 <u>主模式:</u> 1 = 在数据输出时间的末端采样输入数据 0 = 在数据输出时间的中间采样输入数据 <u>从模式:</u> 总是在数据输出时间的中间采样输入数据, 不论 SMP 设置如何。
bit 8	<b>CKE:</b> SPIx 时钟边沿选择位 <sup>(1)</sup> 1 = 时钟状态从有效转换到空闲时发送 0 = 时钟状态从空闲转换到有效时发送
bit 7	<b>SSEN:</b> 从选择使能位 (从模式) <sup>(2)</sup> 1 = 宏在从模式下使用 $\overline{SSx}$ 引脚; $\overline{SSx}$ 引脚用作从选择输入 0 = 宏不使用 $\overline{SSx}$ 引脚 ( $\overline{SSx}$ 引脚将由端口 I/O 控制)
bit 6	<b>CKP:</b> 时钟极性选择位 1 = 空闲状态时时钟信号为高电平; 工作状态时为低电平 0 = 空闲状态时时钟信号为低电平; 工作状态时为高电平
bit 5	<b>MSTEN:</b> 主模式使能位 1 = 主模式 0 = 从模式
bit 4	<b>DISSDI:</b> 禁止 SDIx 输入端口位 1 = 模块不使用 SDIx 引脚; 引脚由端口功能控制 0 = SDIx 引脚由模块控制
bit 3	<b>DISSCK:</b> 禁止 SCKx 输出端口位 1 = 模块不使用 SCKx 引脚; 引脚由端口功能控制 0 = SCKx 引脚由模块控制
bit 2	<b>MCLKEN:</b> 主时钟使能位 <sup>(3)</sup> 1 = 波特率发生器 (BRG) 使用 REFO 0 = BRG 使用外设时钟
bit 1	<b>SPIFE:</b> 帧同步脉冲边沿选择位 1 = 帧同步脉冲 (从空闲转换到有效的边沿) 与第一个位时钟同时出现 0 = 帧同步脉冲 (从空闲转换到有效的边沿) 在第一个位时钟之前出现
bit 0	<b>ENHBUF:</b> 增强型缓冲区使能位 1 = 使能增强型缓冲区模式 0 = 禁止增强型缓冲区模式

注 1: 当 AUDEN (SPIxCON1H<15>) = 1 时, 该模块按 CKE = 0 时的情况工作, 不论其实际值如何。

2: 当 FRMEN = 1 时, 不使用 SSEN。

3: 只有 SPIEN 位 = 0 时, 才能写入 MCLKEN。

4: 该通道对于 DSP/PCM 模式没有意义, 因为 LRC 会遵循 FRMSYPW。

# dsPIC33EPXXGS70X/80X 系列

寄存器 18-2: SPIxCON1H: SPIx 控制寄存器 1 高位字

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AUDEN <sup>(1)</sup>	SPIGNEXT	IGNROV	IGNTUR	AUDMONO <sup>(2)</sup>	URDTEN <sup>(3)</sup>	AUDMOD1 <sup>(4)</sup>	AUDMOD0 <sup>(4)</sup>
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FRMEN	FRMSYNC	FRMPOL	MSEN	FRMSYPW	FRMCNT2	FRMCNT1	FRMCNT0
bit 7						bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15        **AUDEN:** 音频编解码器支持使能位 <sup>(1)</sup>  
 1 = 使能音频协议; MSTEN 控制 SCKx 和帧 (也称为 LRC) 的方向, 并且该模块按 FRMEN = 1、FRMSYNC = MSTEN、FRMCNT<2:0> = 001 且 SMP = 0 时的情况工作, 不论它们的实际值如何  
 0 = 禁止音频协议
- bit 14        **SPIGNEXT:** SPIx 符号扩展接收 FIFO 读取数据使能位  
 1 = 对接收 FIFO 中的数据进行符号扩展  
 0 = 不对接收 FIFO 中的数据进行符号扩展
- bit 13        **IGNROV:** 忽略接收溢出位  
 1 = 接收溢出 (ROV) 不是关键错误; 在 ROV 期间, FIFO 中的数据不会被接收数据覆盖  
 0 = ROV 是关键错误, 会停止 SPI 操作
- bit 12        **IGNTUR:** 忽略发送数据不足位  
 1 = 发送数据不足 (TUR) 不是关键错误, 发送由 URDTEN 指示的数据, 直到 SPIxTXB 不为空  
 0 = TUR 是关键错误, 会停止 SPI 操作
- bit 11        **AUDMONO:** 音频数据格式发送位 <sup>(2)</sup>  
 1 = 音频数据为单声道 (即, 每个数据字通过左右声道同时发送)  
 0 = 音频数据为立体声
- bit 10        **URDTEN:** 发送数据不足数据使能位 <sup>(3)</sup>  
 1 = 在发送数据不足条件期间发送 SPIxURDT 寄存器的数据  
 0 = 在发送数据不足条件期间发送最后接收到的数据
- bit 9-8       **AUDMOD<1:0>:** 音频协议模式选择位 <sup>(4)</sup>  
 11 = PCM/DSP 模式  
 10 = 右对齐模式: 该模块按 SPIFE = 1 时的情况工作, 不论其实际值如何  
 01 = 左对齐模式: 该模块按 SPIFE = 1 时的情况工作, 不论其实际值如何  
 00 = I<sup>2</sup>S 模式: 该模块按 SPIFE = 0 时的情况工作, 不论其实际值如何
- bit 7         **FRMEN:** 帧 SPIx 支持位  
 1 = 使能帧 SPIx 支持 ( $\overline{SSx}$  引脚用作 FSYNC 输入 / 输出)  
 0 = 禁止帧 SPIx 支持

注 1: 只有 SPIEN 位 = 0 时, 才能写入 AUDEN。  
 注 2: 只有 SPIEN 位 = 0 时才能写入 AUDMONO, 并且该位仅在 AUDEN = 1 时有效。  
 注 3: 只有 IGNTUR = 1 时, URDTEN 才有效。  
 注 4: 只有 SPIEN 位 = 0 时才能写入 AUDMOD<1:0> 位, 并且仅在 AUDEN = 1 时有效。不处于 PCM/DSP 模式时, 该模块按 FRMSYPW = 1 时的情况工作, 不论其实际值如何。

## 寄存器 18-2: SPIxCON1H: SPIx 控制寄存器 1 高位字 (续)

- bit 6      **FRMSYNC:** 帧同步脉冲方向控制位  
1 = 帧同步脉冲输入 (从器件)  
0 = 帧同步脉冲输出 (主器件)
- bit 5      **FRMPOL:** 帧同步 / 从选择极性位  
1 = 帧同步脉冲 / 从选择为高电平有效  
0 = 帧同步脉冲 / 从选择为低电平有效
- bit 4      **MSEN:** 主模式从选择使能位  
1 = 使能 SPIx 从选择支持, 其极性由 FRMPOL 决定 (在主模式下在发送期间自动驱动  $\overline{SSx}$  引脚)  
0 = 禁止从选择 SPIx 支持 (SSx 引脚将由端口 I/O 控制)
- bit 3      **FRMSYPW:** 帧同步脉冲宽度位  
1 = 帧同步脉冲宽度为 1 个串行字长 (由 MODE<32,16>/WLENGTH<4:0> 定义)  
0 = 帧同步脉冲宽度为 1 个时钟 (SCK)
- bit 2-0    **FRMCNT<2:0>:** 帧同步脉冲计数器位  
控制每个同步脉冲发送的串行字数。  
111 = 保留  
110 = 保留  
101 = 每 32 个串行字产生 1 个帧同步脉冲  
100 = 每 16 个串行字产生 1 个帧同步脉冲  
011 = 每 8 个串行字产生 1 个帧同步脉冲  
010 = 每 4 个串行字产生 1 个帧同步脉冲  
001 = 每 2 个串行字产生 1 个帧同步脉冲 (由音频协议使用的值)  
000 = 每 1 个串行字产生 1 个帧同步脉冲

- 注 1:** 只有 SPIEN 位 = 0 时, 才能写入 AUDEN。
- 2:** 只有 SPIEN 位 = 0 时才能写入 AUDMONO, 并且该位仅在 AUDEN = 1 时有效。
- 3:** 只有 IGNTUR = 1 时, URDTEN 才有效。
- 4:** 只有 SPIEN 位 = 0 时才能写入 AUDMOD<1:0> 位, 并且仅在 AUDEN = 1 时有效。不处于 PCM/DSP 模式时, 该模块按 FRMSYPW = 1 时的情况工作, 不论其实际值如何。

# dsPIC33EPXXGS70X/80X 系列

寄存器 18-3: SPIxCON2L: SPIx 控制寄存器 2 低位字

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	WLENGTH<4:0> <sup>(1,2)</sup>				—
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-5      **未实现:** 读为 0

bit 4-0      **WLENGTH<4:0>:** 可变字长位 <sup>(1,2)</sup>

- 11111 = 32 位数据
- 11110 = 31 位数据
- 11101 = 30 位数据
- 11100 = 29 位数据
- 11011 = 28 位数据
- 11010 = 27 位数据
- 11001 = 26 位数据
- 11000 = 25 位数据
- 10111 = 24 位数据
- 10110 = 23 位数据
- 10101 = 22 位数据
- 10100 = 21 位数据
- 10011 = 20 位数据
- 10010 = 19 位数据
- 10001 = 18 位数据
- 10000 = 17 位数据
- 01111 = 16 位数据
- 01110 = 15 位数据
- 01101 = 14 位数据
- 01100 = 13 位数据
- 01011 = 12 位数据
- 01010 = 11 位数据
- 01001 = 10 位数据
- 01000 = 9 位数据
- 00111 = 8 位数据
- 00110 = 7 位数据
- 00101 = 6 位数据
- 00100 = 5 位数据
- 00011 = 4 位数据
- 00010 = 3 位数据
- 00001 = 2 位数据
- 00000 = 请参见 SPIxCON1L<11:10> 中的 MODE<32,16> 位

- 注 1:** 只有 AUDEN = 0 时, 这些位才有效。  
**注 2:** 通过更改这些位来改变长度并不会影响发送 / 接收 FIFO 的深度。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 18-4: SPIxSTATL: SPIx 状态寄存器低位字

U-0	U-0	U-0	R/C-0, HS	R-0, HSC	U-0	U-0	R-0, HSC
—	—	—	FRMERR	SPIBUSY	—	—	SPITUR <sup>(1)</sup>
bit 15							bit 8

R-0, HSC	R/C-0, HS	R-1, HSC	U-0	R-1, HSC	U-0	R-0, HSC	R-0, HSC
SRMT	SPIROV	SPIRBE	—	SPITBE	—	SPITBF	SPIRBF
bit 7							bit 0

<b>图注:</b>	C = 可清零位	U = 未实现位, 读为 0
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位
-n = POR 时的值	1 = 置 1	0 = 清零
		HS = 硬件置 1 位

bit 15-13 **未实现:** 读为 0

bit 12 **FRMERR:** SPIx 帧错误状态位

1 = 检测到帧错误  
0 = 未检测到帧错误

bit 11 **SPIBUSY:** SPIx 活动状态位

1 = 模块当前正忙于处理一些事务  
0 = 没有正在进行的事务 (在读取时)

bit 10-9 **未实现:** 读为 0

bit 8 **SPITUR:** SPIx 发送数据不足状态位<sup>(1)</sup>

1 = 发送缓冲区遇到发送数据不足条件  
0 = 发送缓冲区不存在发送数据不足条件

bit 7 **SRMT:** 移位寄存器空状态位

1 = 没有当前或待处理的事务 (即, SPIxTXB 和 SPIxTXSR 都不包含要发送的数据)  
0 = 当前或待处理的事务

bit 6 **SPIROV:** SPIx 接收溢出状态位

1 = 在 SPIxRXB 已满时, 完全接收到新的字节 / 半字 / 字  
0 = 未发生溢出

bit 5 **SPIRBE:** SPIx 接收缓冲区空状态位

1 = 接收缓冲区为空  
0 = 接收缓冲区不为空

**标准缓冲区模式:**

当从 SPIxBUF 读取数据来读 SPIxRXB 时, 该位由硬件自动置 1。当 SPIx 将数据从 SPIxRXSR 传输到 SPIxRXB 时, 该位由硬件自动清零。

**增强型缓冲区模式:**

指示 RXELM<5:0> = 000000。

bit 4 **未实现:** 读为 0

**注 1:** SPIEN = 0 时, SPITUR 会清零。当 IGNTUR = 1 时, SPITUR 会提供发送数据不足条件的动态状态, 但不会停止接收 / 发送操作, 不需要用软件清零。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 18-4: SPIxSTATL: SPIx 状态寄存器低位字 (续)

- bit 3      **SPITBE:** SPIx 发送缓冲区空状态位  
1 = SPIxTXB 为空  
0 = SPIxTXB 不为空  
标准缓冲区模式:  
当 SPIx 将数据从 SPIxTXB 传输到 SPIxTXSR 时, 该位由硬件自动置 1。当写 SPIxBUF 来装载 SPIxTXB 时, 该位由硬件自动清零。  
增强型缓冲区模式:  
指示 TXELM<5:0> = 000000。
- bit 2      **未实现:** 读为 0
- bit 1      **SPITBF:** SPIx 发送缓冲区满状态位  
1 = SPIxTXB 已满  
0 = SPIxTXB 未满  
标准缓冲区模式:  
当写 SPIxBUF 来装载 SPIxTXB 时, 该位由硬件自动置 1。当 SPIx 将数据从 SPIxTXB 传输到 SPIxTXSR 时, 该位由硬件自动清零。  
增强型缓冲区模式:  
指示 TXELM<5:0> = 111111。
- bit 0      **SPIRBF:** SPIx 接收缓冲区满状态位  
1 = SPIxRXB 已满  
0 = SPIxRXB 未满  
标准缓冲区模式:  
当 SPIx 将数据从 SPIxRXSR 传输到 SPIxRXB 时, 该位由硬件自动置 1。当从 SPIxBUF 读取数据来读 SPIxRXB 时, 该位由硬件自动清零。  
增强型缓冲区模式:  
指示 RXELM<5:0> = 111111。

**注 1:** SPIEN = 0 时, SPITUR 会清零。当 IGNTUR = 1 时, SPITUR 会提供发送数据不足条件的动态状态, 但不会停止接收 / 发送操作, 不需要用软件清零。

# dsPIC33EPXXGS70X/80X 系列

寄存器 18-5: SPIxSTATH: SPIx 状态寄存器高位字

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	RXELM5 <sup>(3)</sup>	RXELM4 <sup>(2)</sup>	RXELM3 <sup>(1)</sup>	RXELM2	RXELM1	RXELM0
bit 15							bit 8

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	TXELM5 <sup>(3)</sup>	TXELM4 <sup>(2)</sup>	TXELM3 <sup>(1)</sup>	TXELM2	TXELM1	TXELM0
bit 7							bit 0

<b>图注:</b>	HSC = 硬件置 1/ 清零位						
R = 可读位	W = 可写位		U = 未实现位, 读为 0				
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知				

bit 15-14    **未实现:** 读为 0

bit 13-8    **RXELM<5:0>:** 接收缓冲区元素计数位 (在增强型缓冲区模式下有效) **(1,2,3)**

bit 7-6    **未实现:** 读为 0

bit 5-0    **TXELM<5:0>:** 发送缓冲区元素计数位 (在增强型缓冲区模式下有效) **(1,2,3)**

- 注 1:** 只有 FIFODEPTH = 8 或更高值时, 才存在 RXELM3 和 TXELM3 位。  
**2:** 只有 FIFODEPTH = 16 或更高值时, 才存在 RXELM4 和 TXELM4 位。  
**3:** 只有 FIFODEPTH = 32 时, 才存在 RXELM5 和 TXELM5 位。

# dsPIC33EPXXGS70X/80X 系列

寄存器 18-6: SPIxIMSKL: SPIx 中断掩码寄存器低位字

U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0
—	—	—	FRMERREN	BUSYEN	—	—	SPITUREN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0
SRMTEN	SPIROVEN	SPIRBEN	—	SPITBEN	—	SPITBFEN	SPIRBFEN
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-13      **未实现:** 读为 0
- bit 12        **FRMERREN:** 允许通过 FRMERR 产生中断事件位  
                  1 = 帧错误产生中断事件  
                  0 = 帧错误不产生中断事件
- bit 11        **BUSYEN:** 允许通过 SPIBUSY 产生中断事件位  
                  1 = SPIBUSY 产生中断事件  
                  0 = SPIBUSY 不产生中断事件
- bit 10-9      **未实现:** 读为 0
- bit 8         **SPITUREN:** 允许通过 SPITUR 产生中断事件位  
                  1 = 发送数据不足 (TUR) 产生中断事件  
                  0 = 发送数据不足不产生中断事件
- bit 7         **SRMTEN:** 允许通过 SRMT 产生中断事件位  
                  1 = 移位寄存器为空 (SRMT) 产生中断事件  
                  0 = 移位寄存器为空不产生中断事件
- bit 6         **SPIROVEN:** 允许通过 SPIROV 产生中断事件位  
                  1 = SPIx 接收溢出 (ROV) 产生中断事件  
                  0 = SPIx 接收溢出不产生中断事件
- bit 5         **SPIRBEN:** 允许通过 SPIRBE 产生中断事件位  
                  1 = SPIx 接收缓冲区为空产生中断事件  
                  0 = SPIx 接收缓冲区为空不产生中断事件
- bit 4         **未实现:** 读为 0
- bit 3         **SPITBEN:** 允许通过 SPITBE 产生中断事件位  
                  1 = SPIx 发送缓冲区为空产生中断事件  
                  0 = SPIx 发送缓冲区为空不产生中断事件
- bit 2         **未实现:** 读为 0
- bit 1         **SPITBFEN:** 允许通过 SPITBF 产生中断事件位  
                  1 = SPIx 发送缓冲区已满产生中断事件  
                  0 = SPIx 发送缓冲区已满不产生中断事件
- bit 0         **SPIRBFEN:** 允许通过 SPIRBF 产生中断事件位  
                  1 = SPIx 接收缓冲区已满产生中断事件  
                  0 = SPIx 接收缓冲区已满不产生中断事件



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 18-7: SPIxIMSKH: SPIx 中断掩码寄存器高位字

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RXWIEN	—	RXMSK5 <sup>(1)</sup>	RXMSK4 <sup>(1,4)</sup>	RXMSK3 <sup>(1,3)</sup>	RXMSK2 <sup>(1,2)</sup>	RXMSK1 <sup>(1)</sup>	RXMSK0 <sup>(1)</sup>
bit 15							bit 8

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TXWIEN	—	TXMSK5 <sup>(1)</sup>	TXMSK4 <sup>(1,4)</sup>	TXMSK3 <sup>(1,3)</sup>	TXMSK2 <sup>(1,2)</sup>	TXMSK1 <sup>(1)</sup>	TXMSK0 <sup>(1)</sup>
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15      **RXWIEN:** 接收水印中断允许位  
 1 = RXMSK<5:0> ≤ RXELM<5:0> 时触发接收缓冲区元素水印中断  
 0 = 禁止接收缓冲区元素水印中断
- bit 14      **未实现:** 读为 0
- bit 13-8    **RXMSK<5:0>:** 接收缓冲区掩码位 **(1,2,3,4)**  
 接收掩码位; 与 RXWIEN 位配合使用。
- bit 7        **TXWIEN:** 发送水印中断允许位  
 1 = TXMSK<5:0> = TXELM<5:0> 时触发发送缓冲区元素水印中断  
 0 = 禁止发送缓冲区元素水印中断
- bit 6        **未实现:** 读为 0
- bit 5-0     **TXMSK<5:0>:** 发送缓冲区掩码位 **(1,2,3,4)**  
 发送掩码位; 与 TXWIEN 位配合使用。

- 注 1:** 高于 FIFODEPTH 的掩码值是无效的。在这种情况下, 模块不会对于任何值触发匹配。
- 2:** 只有 FIFODEPTH = 8 或更高值时, 才存在 RXMSK2 和 TXMSK2 位。
- 3:** 只有 FIFODEPTH = 16 或更高值时, 才存在 RXMSK3 和 TXMSK3 位。
- 4:** 只有 FIFODEPTH = 32 时, 才存在 RXMSK4 和 TXMSK4 位。

# dsPIC33EPXXGS70X/80X 系列

图 18-3: SPIx 主 / 从连接 (标准模式)

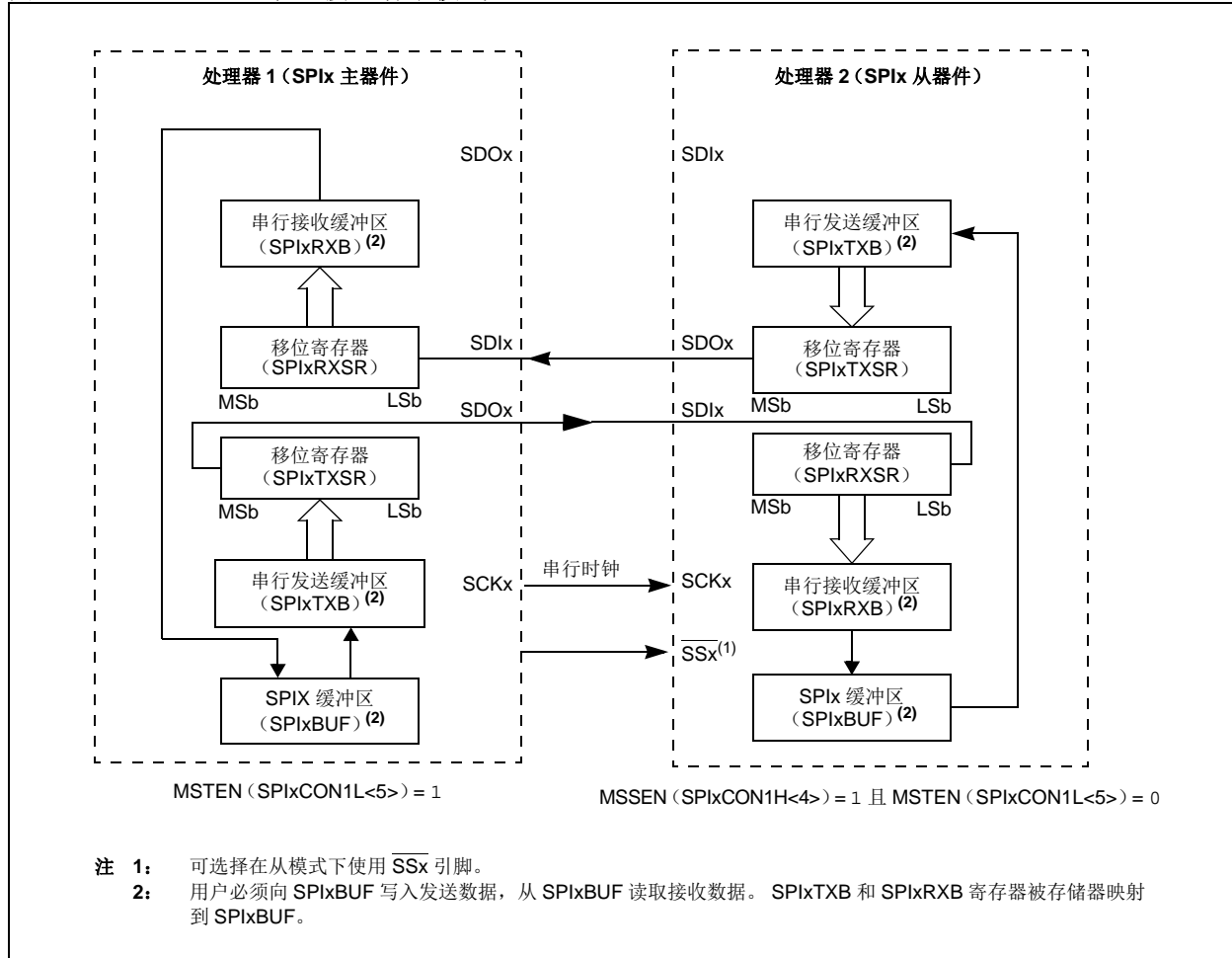


图 18-4: SPIx 主 / 从连接 (增强型缓冲区模式)

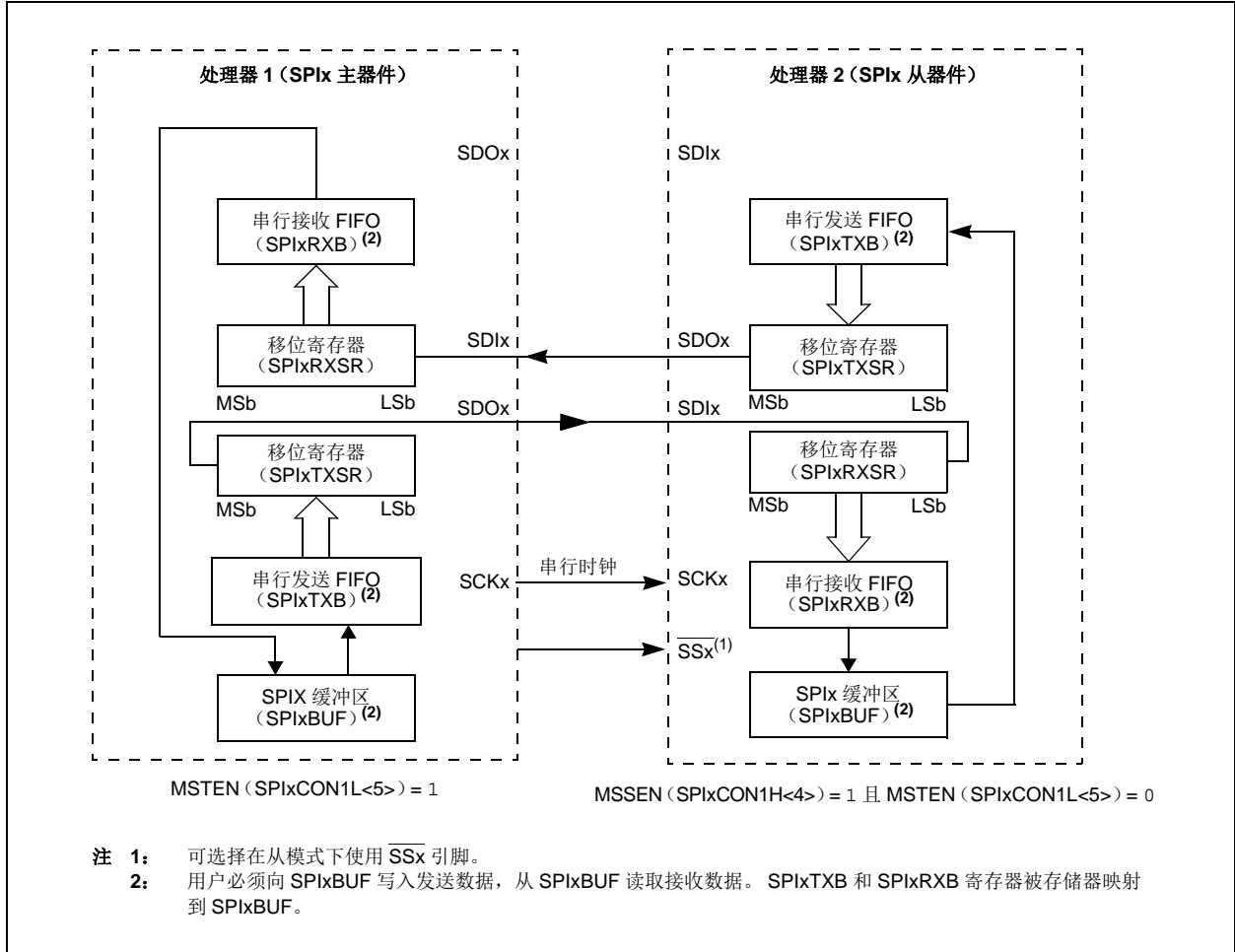
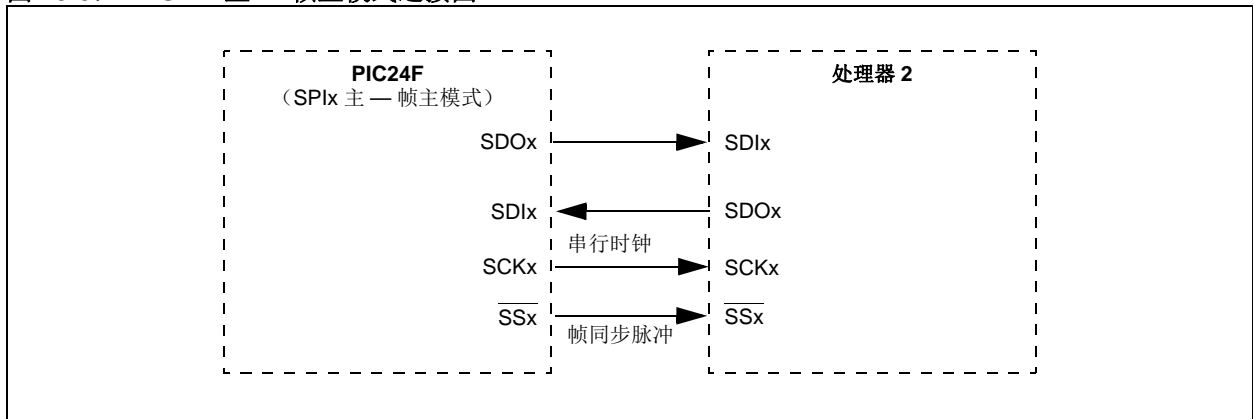


图 18-5: SPIx 主 — 帧主模式连接图



# dsPIC33EPXXGS70X/80X 系列

图 18-6: SPIx 主-帧从模式连接图

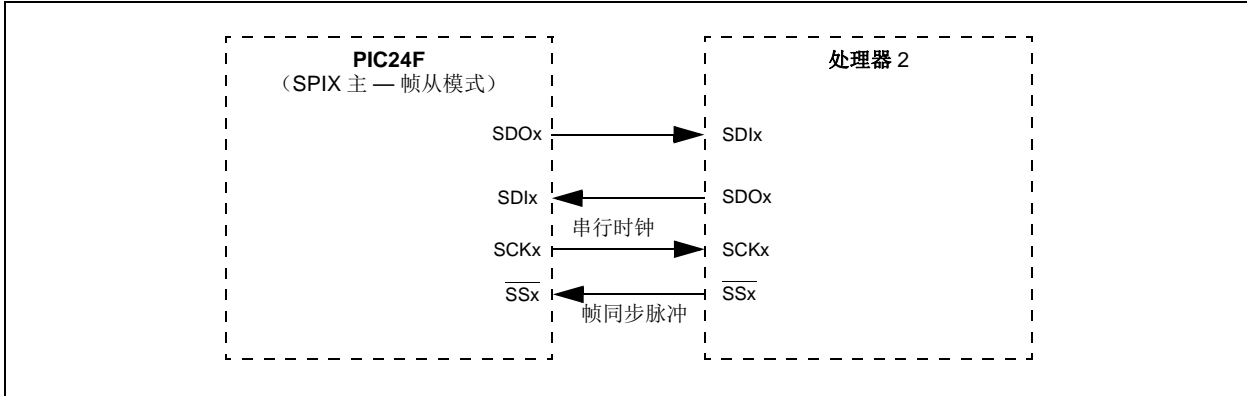


图 18-7: SPIx 从-帧主模式连接图

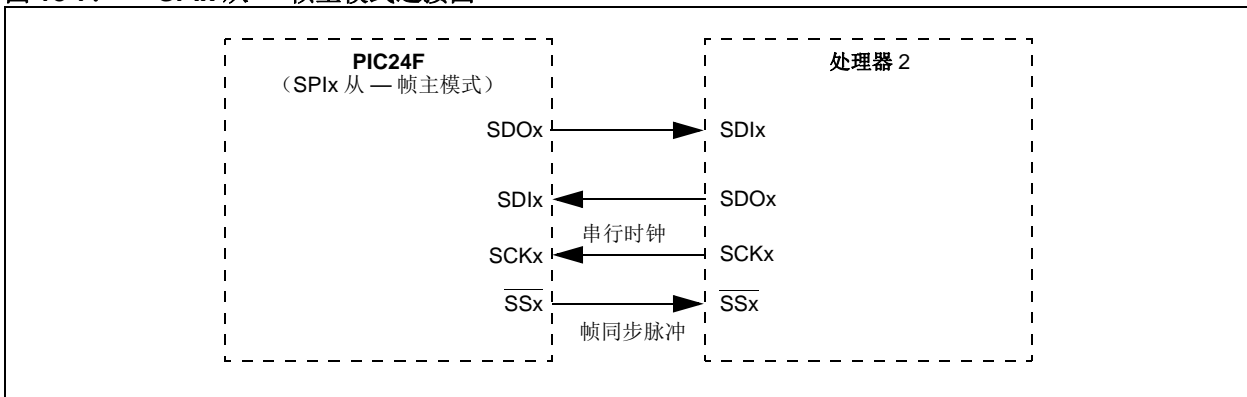
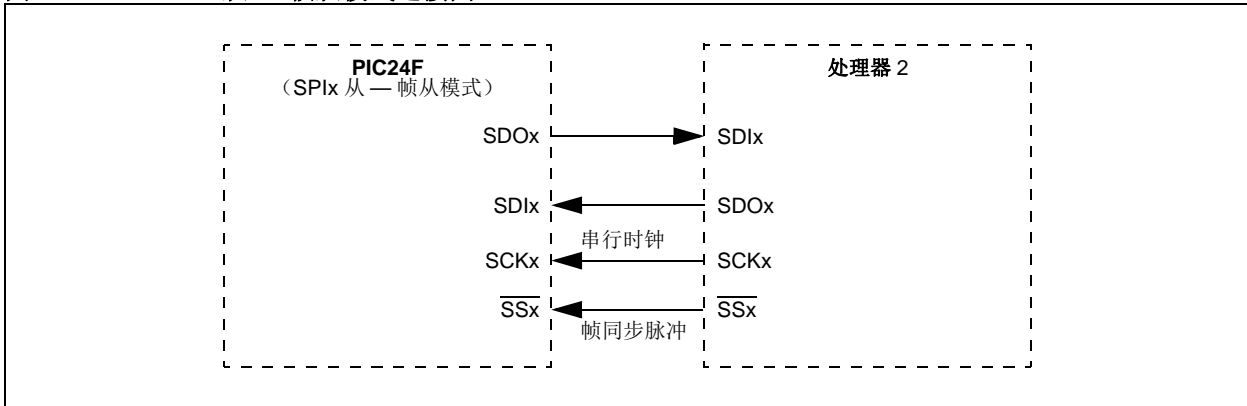


图 18-8: SPIx 从-帧从模式连接图



公式 18-1: 器件工作频率和 SPIx 时钟速度之间的关系

$$\text{波特率} = \frac{\text{FPB}}{(2 * (\text{SPIxBRG} + 1))}$$

其中:

FPB 为外设总线时钟频率。

## 19.0 I<sup>2</sup>C

- 注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“**I<sup>2</sup>C**”（DS70000195），该文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

dsPIC33EPXXGS70X/80X 系列器件包含两个 I<sup>2</sup>C 模块：I2C1 和 I2C2。

I<sup>2</sup>C 模块（16 位接口）为 I<sup>2</sup>C 串行通信标准的从模式和多主器件模式提供完全的硬件支持。

I<sup>2</sup>C 模块有一个双引脚接口：

- SCLx/ASCLx 引脚是时钟线
- SDAx/ASDAx 引脚是数据线

I<sup>2</sup>C 模块提供以下主要特性：

- I<sup>2</sup>C 接口支持主 / 从工作模式
- I<sup>2</sup>C 从模式支持 7 位和 10 位寻址
- I<sup>2</sup>C 主模式支持 7 位和 10 位寻址
- I<sup>2</sup>C 端口允许主器件和从器件之间的双向传输
- I<sup>2</sup>C 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）
- I<sup>2</sup>C 支持多主器件工作；检测总线冲突并相应地进行仲裁
- 系统管理总线（System Management Bus, SMBus）支持
- 备用 I<sup>2</sup>C 引脚映射（ASCLx/ASDAx）

## 19.1 I<sup>2</sup>C 资源

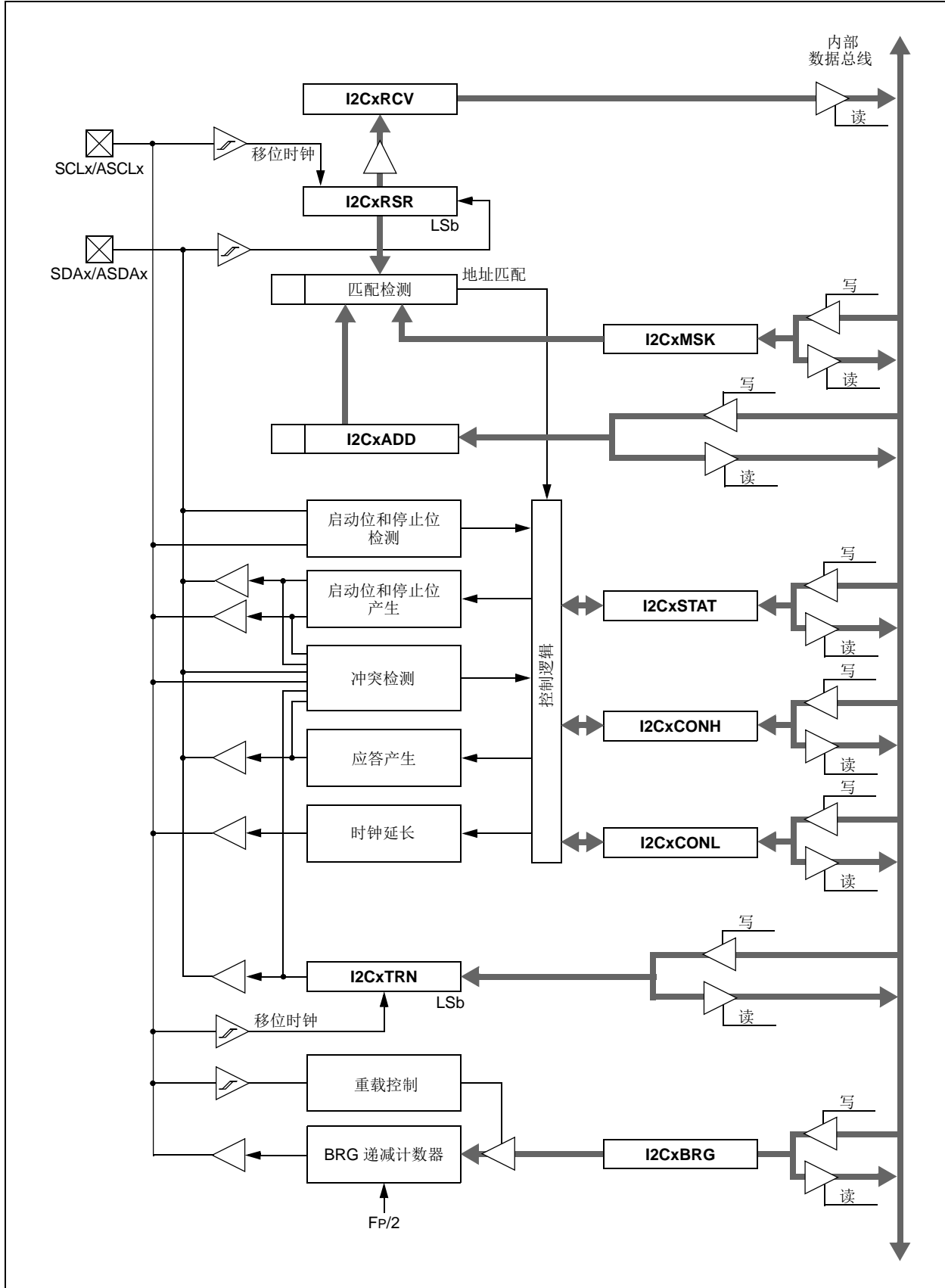
对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 19.1.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“**I<sup>2</sup>C**”（DS70000195）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

图 19-1: I2Cx 框图 (x = 1 或 2)



# dsPIC33EPXXGS70X/80X 系列

## 19.2 I<sup>2</sup>C 控制寄存器

寄存器 19-1: I2CxCONL: I2Cx 控制寄存器的低位字

R/W-0	U-0	R/W-0	R/W-1, HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	—	I2CSIDL	SCLREL	STRICT	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

<b>图注:</b>	HC = 硬件清零位
R = 可读位	W = 可写位
-n = POR 时的值	U = 未实现位, 读为 0
	0 = 清零
	1 = 置 1
	x = 未知

- bit 15     **I2CEN:** I2Cx 使能位  
 1 = 使能 I2Cx 模块, 并将 SDAx 和 SCLx 引脚配置为串口引脚  
 0 = 禁止 I2Cx 模块, 所有 I<sup>2</sup>C 引脚均由端口功能控制
- bit 14     **未实现:** 读为 0
- bit 13     **I2CSIDL:** I2Cx 空闲模式停止位  
 1 = 当器件进入空闲模式时, 模块停止工作  
 0 = 在空闲模式下模块继续工作
- bit 12     **SCLREL:** SCLx 释放控制位 (作为 I<sup>2</sup>C 从器件工作时)  
 1 = 释放 SCLx 时钟  
 0 = 保持 SCLx 时钟为低电平 (时钟延长)  
**如果 STREN = 1:**  
 该位可读可写 (即软件可以写入 0 来启动时钟延长或写入 1 来释放时钟)。在每个从器件数据字节发送开始时由硬件清零。在每个从器件地址字节接收结束时由硬件清零。在每个从器件数据字节接收结束时由硬件清零。  
**如果 STREN = 0:**  
 该位可读且可被置 1 (即软件只能写入 1 来释放时钟)。在每个从器件数据字节发送开始时由硬件清零。在每个从器件地址字节接收结束时由硬件清零。
- bit 11     **STRICT:** 严格 I2Cx 保留地址使能位  
 1 = **使能严格保留寻址:**  
 在从模式下, 器件将不应答任何保留地址。在主模式下, 允许器件产生保留地址空间内的地址。  
 0 = **应答保留寻址:**  
 在从模式下, 器件将应答任何保留地址。在主模式下, 器件不应通过保留地址寻址从器件。
- bit 10     **A10M:** 10 位从器件地址位  
 1 = I2CxADD 为 10 位从器件地址  
 0 = I2CxADD 为 7 位从器件地址
- bit 9     **DISSLW:** 禁止压摆率控制位  
 1 = 禁止压摆率控制  
 0 = 使能压摆率控制
- bit 8     **SMEN:** SMBus 输入电平位  
 1 = 使能符合 SMBus 规范的 I/O 引脚阈值  
 0 = 禁止 SMBus 输入阈值
- bit 7     **GCEN:** 广播呼叫使能位 (作为 I<sup>2</sup>C 从器件工作时)  
 1 = 允许在 I2CxRSR 中接收到广播呼叫地址时产生中断 (已使能模块接收)  
 0 = 禁止广播呼叫地址

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 19-1: I2CxCONL: I2Cx 控制寄存器的低位字 (续)

- bit 6      **STREN:** SCLx 时钟延长使能位 (作为 I<sup>2</sup>C 从器件工作时)  
与 SCLREL 位配合使用。  
1 = 使能软件控制的时钟延长  
0 = 禁止软件控制的时钟延长
- bit 5      **ACKDT:** 应答数据位 (作为 I<sup>2</sup>C 主器件工作时, 适用于主器件接收过程)  
当软件启动应答序列时将发送的值。  
1 = 在应答时发送 NACK  
0 = 在应答时发送 ACK
- bit 4      **ACKEN:** 应答序列使能位  
(作为 I<sup>2</sup>C 主器件工作时, 适用于主器件接收过程)  
1 = 在 SDAx 和 SCLx 引脚上发出应答序列, 并发送 ACKDT 数据位; 在主器件应答序列结束时由硬件清零  
0 = 应答序列不在进行中
- bit 3      **RCEN:** 接收使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 使能 I<sup>2</sup>C 接收模式; 在主器件接收完数据字节的第 8 位时由硬件清零。  
0 = 接收序列不在进行中
- bit 2      **PEN:** 停止条件使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 在 SDAx 和 SCLx 引脚上发出停止条件; 在主器件停止序列结束时由硬件清零  
0 = 停止条件不在进行中
- bit 1      **RSEN:** 重复启动条件使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 在 SDAx 和 SCLx 引脚上发出重复启动条件; 在主器件重复启动序列结束时由硬件清零  
0 = 重复启动条件不在进行中
- bit 0      **SEN:** 启动条件使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 在 SDAx 和 SCLx 引脚上发出启动条件; 在主器件启动序列结束时由硬件清零  
0 = 启动条件不在进行中



# dsPIC33EPXXGS70X/80X 系列

寄存器 19-2: I2CxCONH: I2Cx 控制寄存器的高位字

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-7     **未实现:** 读为 0
- bit 6        **PCIE:** 停止条件中断允许位 (仅在 I<sup>2</sup>C 从模式下)
  - 1 = 允许在检测到停止条件时产生中断
  - 0 = 禁止停止条件检测中断
- bit 5        **SCIE:** 启动条件中断允许位 (仅在 I<sup>2</sup>C 从模式下)
  - 1 = 允许在检测到启动或重复启动条件时产生中断
  - 0 = 禁止启动条件检测中断
- bit 4        **BOEN:** 缓冲区改写使能位 (仅在 I<sup>2</sup>C 从模式下)
  - 1 = 仅当 RBF 位 = 0 时, 在接收到地址 / 数据字节时, 更新 I2CxRCV 且产生 ACK 信号, 并忽略 I2COV 的状态
  - 0 = 仅当 I2COV 位清零时更新 I2CxRCV
- bit 3        **SDAHT:** SDAx 保持时间选择位
  - 1 = 在 SCLx 的下降沿之后在 SDAx 上至少保持 300 ns 的时间
  - 0 = 在 SCLx 的下降沿之后在 SDAx 上至少保持 100 ns 的时间
- bit 2        **SBCDE:** 从模式总线冲突检测使能位 (仅在 I<sup>2</sup>C 从模式下)
  - 1 = 允许从总线冲突中断
  - 0 = 禁止从总线冲突中断

如果在 SCLx 的上升沿, 模块输出为高电平状态时采样到 SDAx 为低电平, 则 BCL 位置 1 且总线进入空闲状态。此检测模式仅在数据和 ACK 发送序列期间有效。
- bit 1        **AHEN:** 地址保持使能位 (仅在 I<sup>2</sup>C 从模式下)
  - 1 = 在接收匹配地址字节的 SCLx 的第 8 个下降沿之后, SCLREL 位 (I2CxCONL<12>) 将清零且 SCLx 保持低电平
  - 0 = 禁止地址保持
- bit 0        **DHEN:** 数据保持使能位 (仅在 I<sup>2</sup>C 从模式下)
  - 1 = 在接收数据字节的 SCLx 的第 8 个下降沿之后, 从器件硬件清零 SCLREL (I2CxCONL<12>) 位且 SCLx 保持低电平
  - 0 = 禁止数据保持

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 19-3: I2CxSTAT: I2Cx 状态寄存器

R-0, HSC	R-0, HSC	R-0, HSC	U-0	U-0	R/C-0, HS	R-0, HSC	R-0, HSC
ACKSTAT	TRSTAT	ACKTIM	—	—	BCL	GCSTAT	ADD10
bit 15						bit 8	
R/C-0, HS	R/C-0, HS	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF
bit 7						bit 0	

<b>图注:</b>	C = 可清零位	0 = 清零	HS = 硬件置 1 位
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位	
-n = POR 时的值	1 = 置 1	U = 未实现位, 读为 0	

- bit 15     **ACKSTAT:** 应答状态位 (作为 I<sup>2</sup>C 主器件工作时, 适用于主器件发送操作)  
 1 = 接收到来自从器件的 NACK  
 0 = 接收到来自从器件的 ACK  
 在从器件应答结束时由硬件置 1 或清零。
- bit 14     **TRSTAT:** 发送状态位 (作为 I<sup>2</sup>C 主器件工作时, 适用于主器件发送操作)  
 1 = 主器件正在进行发送 (8 位 + ACK)  
 0 = 主器件不在进行发送  
 在主器件发送开始时由硬件置 1。在从器件应答结束时由硬件清零。
- bit 13     **ACKTIM:** 应答时间状态位 (仅在 I<sup>2</sup>C 从模式下)  
 1 = I<sup>2</sup>C 总线上有应答序列, 在 SCLx 的第 8 个下降沿置 1  
 0 = 无应答序列, 在 SCLx 的第 9 个上升沿清零
- bit 12-11   **未实现:** 读为 0
- bit 10     **BCL:** 主器件总线冲突检测位  
 1 = 主器件工作期间检测到了总线冲突  
 0 = 未检测到总线冲突  
 在检测到总线冲突时由硬件置 1。
- bit 9       **GCSTAT:** 广播呼叫状态位  
 1 = 接收到广播呼叫地址  
 0 = 未接收到广播呼叫地址  
 当地址与广播呼叫地址匹配时由硬件置 1。在检测到停止条件时由硬件清零。
- bit 8       **ADD10:** 10 位地址状态位  
 1 = 10 位地址匹配  
 0 = 10 位地址不匹配  
 当与匹配的 10 位地址的第 2 个字节匹配时由硬件置 1。在检测到停止条件时由硬件清零。
- bit 7       **IWCOL:** I2Cx 写冲突检测位  
 1 = 因为 I<sup>2</sup>C 模块忙, 尝试写 I2CxTRN 寄存器失败  
 0 = 未发生冲突  
 当模块忙时写 I2CxTRN 会使硬件将该位置 1 (用软件清零)。
- bit 6       **I2COV:** I2Cx 接收溢出标志位  
 1 = 当 I2CxRCV 寄存器仍然保存原先的字节时接收到了新字节  
 0 = 未发生溢出  
 尝试将数据从 I2CxRSR 传输到 I2CxRCV 时由硬件置 1 (用软件清零)。
- bit 5       **D\_A:** 数据 / 地址位 (仅在 I<sup>2</sup>C 从模式下)  
 1 = 指示上次接收的字节为数据  
 0 = 指示上次接收的字节为器件地址  
 器件地址匹配时由硬件清零。在作为从器件接收到数据字节时由硬件置 1。

## 寄存器 19-3: I2CxSTAT: I2Cx 状态寄存器 (续)

- bit 4      **P:** 停止位  
1 = 指示上次检测到停止位  
0 = 上次未检测到停止位  
当检测到启动、重复启动或停止条件时由硬件置 1 或清零。
- bit 3      **S:** 启动位  
1 = 指示上次检测到启动 (或重复启动) 位  
0 = 上次未检测到启动位  
当检测到启动、重复启动或停止条件时由硬件置 1 或清零。
- bit 2      **R\_W:** 读 / 写信息位 (仅在 I<sup>2</sup>C 从模式下)  
1 = 读 —— 表示数据自从器件输出  
0 = 写 —— 表示数据输入到从器件  
接收到 I<sup>2</sup>C 器件地址字节后由硬件置 1 或清零。
- bit 1      **RBF:** 接收缓冲区满状态位  
1 = 接收完成, I2CxRCV 已满  
0 = 接收未完成, I2CxRCV 为空  
用接收到的字节写 I2CxRCV 时由硬件置 1。用软件读 I2CxRCV 时由硬件清零。
- bit 0      **TBF:** 发送缓冲区满状态位  
1 = 发送正在进行中, I2CxTRN 已满  
0 = 发送完成, I2CxTRN 为空  
用软件写 I2CxTRN 时由硬件置 1。数据发送完成时由硬件清零。

# dsPIC33EPXXGS70X/80X 系列

寄存器 19-4: I2CxMSK: I2Cx 从模式地址掩码寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK<9:8>	
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
AMSK<7:0>							
bit 7						bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-10      **未实现:** 读为 0

bit 9-0        **AMSK<9:0>:** 地址掩码选择位

对于 10 位地址:

1 = 使能输入报文的地址中 bit Ax 的掩码; 在此位置上不需要位匹配

0 = 禁止 bit Ax 的掩码; 在此位置上需要位匹配

对于 7 位地址 (仅 I2CxMSK<6:0>):

1 = 使能输入报文的地址中 bit Ax + 1 的掩码; 在此位置上不需要位匹配

0 = 禁止 bit Ax + 1 的掩码; 在此位置上需要位匹配

## 20.0 通用异步收发器 (UART)

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息,请参见《dsPIC33/PIC24 系列参考手册》的“通用异步收发器 (UART)” (DS70000582), 该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息,请参见本数据手册中的第 4.0 节“存储器构成”。

dsPIC33EPXXGS70X/80X 系列器件包含两个 UART 模块。

通用异步收发器 (Universal Asynchronous Receiver Transmitter, UART) 模块是 dsPIC33EPXXGS70X/80X 器件系列提供的串行 I/O 模块之一。UART 是可以与外设 (如个人计算机、LIN/J2602、RS-232 和 RS-485 接口) 通信的全双工异步系统。模块还通过 UxCTS 和 UxRTS 引脚支持硬件流控制选项, 其中还包括 IrDA<sup>®</sup> 编码器和解码器。

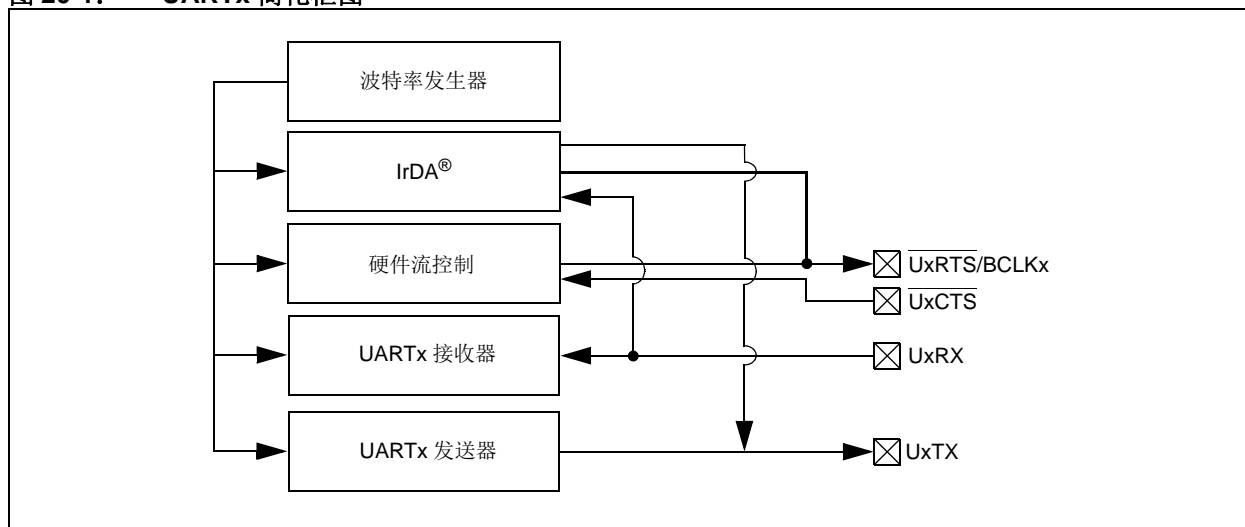
UARTx 模块的主要特性有:

- 通过 UxTX 和 UxRX 引脚进行全双工 8 位或 9 位数据传输
- 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
- 通过 UxCTS 和 UxRTS 引脚支持硬件流控制选项
- 完全集成的波特率发生器, 具有 16 位预分频器
- 当器件工作在 70 MIPS 下 16 倍模式时, 波特率范围为从 67 bps 至 4.375 Mbps
- 当器件工作在 70 MIPS 下 4 倍模式时, 波特率范围为从 267 bps 至 17.5 Mbps
- 4 级深度先进先出 (First-In First-Out, FIFO) 发送数据缓冲区
- 4 级深度 FIFO 接收数据缓冲区
- 奇偶校验错误、帧错误和缓冲区溢出错误检测
- 支持带地址检测功能的 9 位模式 (第 9 位 = 1)
- 发送中断和接收中断
- 所有 UARTx 错误条件下可分别产生中断
- 用于诊断支持的环回模式
- 支持同步字符和间隔字符
- 支持自动波特率检测
- IrDA<sup>®</sup> 编码器和解码器逻辑
- 用于 IrDA 支持的 16 倍波特率时钟输出

图 20-1 给出了 UARTx 模块的简化框图。UARTx 模块由以下至关重要的硬件组件组成:

- 波特率发生器
- 异步发送器
- 异步接收器

图 20-1: UARTx 简化框图



## 20.1 UART 有用技巧

1. 在多节点直接连接 UART 网络中，UART 接收输入会对 URXINV 位 (UxMODE<4>) 定义的相反逻辑电平作出反应；它定义空闲状态，其默认值为逻辑高电平 (即，URXINV = 0)。由于远程器件不会在同一时间初始化，所以很有可能由于 RX 线悬空，有一个器件触发启动位检测，导致在器件初始化之后接收到的第一个字节变为无效。要避免这种情况，用户应根据 URXINV 位的值，在 RX 引脚上使用上拉或下拉电阻。
  - a) 如果 URXINV = 0，则在 UxRX 引脚上使用上拉电阻。
  - b) 如果 URXINV = 1，则在 UxRX 引脚上使用下拉电阻。
2. 由于 UARTx 模块 UxRX 引脚上的活动而从休眠模式唤醒时接收的第一个字符将是无效的。在休眠模式下，外设时钟被禁止。从休眠模式唤醒之后，在振荡器系统重启并稳定之前，波特率位采样时钟相对于输入 UxRX 位时序不再同步，导致第一个字符无效；这是可以预料的。

## 20.2 UART 资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 20.2.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“通用异步收发器 (UART)” (DS70000582)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

# dsPIC33EPXXGS70X/80X 系列

## 20.3 UART 控制寄存器

寄存器 20-1: **UxMODE: UARTx 模式寄存器**

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
UARTEN <sup>(1)</sup>	—	USIDL	IREN <sup>(2)</sup>	RTSMD	—	UEN1	UEN0
bit 15							bit 8

R/W-0, HC	R/W-0	R/W-0, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSEL1	PDSEL0	STSEL
bit 7							bit 0

<b>图注:</b>	HC = 硬件清零位						
R = 可读位	W = 可写位	U = 未实现位, 读为 0					
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知				

- bit 15 **UARTEN: UARTx 使能位<sup>(1)</sup>**  
 1 = 使能 UARTx; UARTx 根据 UEN<1:0> 的定义控制所有 UARTx 引脚  
 0 = 禁止 UARTx; 由端口锁存器控制所有 UARTx 引脚; UARTx 的功耗最小
- bit 14 **未实现:** 读为 0
- bit 13 **USIDL: UARTx 空闲模式停止位**  
 1 = 当器件进入空闲模式时, 模块停止工作  
 0 = 在空闲模式下模块继续工作
- bit 12 **IREN: IrDA<sup>®</sup> 编码器和解码器使能位<sup>(2)</sup>**  
 1 = 使能 IrDA 编码器和解码器  
 0 = 禁止 IrDA 编码器和解码器
- bit 11 **RTSMD: UxRTS 引脚模式选择位**  
 1 = UxRTS 引脚处于单工模式  
 0 = UxRTS 引脚处于流控制模式
- bit 10 **未实现:** 读为 0
- bit 9-8 **UEN<1:0>: UARTx 引脚使能位**  
 11 = 使能并使用 UxTX、UxRX 和 BCLKx 引脚; UxCTS 引脚由端口锁存器控制  
 10 = 使能并使用 UxTX、UxRX、UxCTS 和 UxRTS 引脚  
 01 = 使能并使用 UxTX、UxRX 和 UxRTS 引脚; UxCTS 引脚由端口锁存器控制  
 00 = 使能并使用 UxTX 和 UxRX 引脚; UxCTS 和 UxRTS/BCLKx 引脚由端口锁存器控制
- bit 7 **WAKE: 在休眠模式下检测到启动位唤醒使能位**  
 1 = UARTx 将继续采样 UxRX 引脚; 在出现下降沿时产生中断; 在之后的上升沿由硬件清零该位  
 0 = 不使能唤醒
- bit 6 **LPBACK: UARTx 环回模式选择位**  
 1 = 使能环回模式  
 0 = 禁止环回模式

**注 1:** 有关使能 UARTx 模块进行接收或发送操作的信息, 请参见《dsPIC33/PIC24 系列参考手册》中的“通用异步收发器 (UART)” (DS70000582)。

**2:** 该功能仅在 16 倍 BRG 模式 (BRGH = 0) 下可用。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 20-1: UxMODE: UARTx 模式寄存器 (续)

- bit 5      **ABAUD:** 自动波特率使能位  
1 = 使能对下一个字符的波特率测量 —— 需要在接收其他数据前接收同步字段 (55h); 完成时由硬件清零  
0 = 禁止波特率测量或测量已完成
- bit 4      **URXINV:** UARTx 接收极性翻转位  
1 = UxRX 的空闲状态为 0  
0 = UxRX 的空闲状态为 1
- bit 3      **BRGH:** 高波特率使能位  
1 = BRG 在每个位周期内产生 4 个时钟信号 (4 倍波特率时钟, 高速模式)  
0 = BRG 在每个位周期内产生 16 个时钟信号 (16 倍波特率时钟, 标准模式)
- bit 2-1    **PDSEL<1:0>:** 奇偶校验和数据选择位  
11 = 9 位数据, 无奇偶校验  
10 = 8 位数据, 奇校验  
01 = 8 位数据, 偶校验  
00 = 8 位数据, 无奇偶校验
- bit 0      **STSEL:** 停止位选择位  
1 = 2 个停止位  
0 = 1 个停止位

- 注 1:** 有关使能 UARTx 模块进行接收或发送操作的信息, 请参见《dsPIC33/PIC24 系列参考手册》中的“通用异步收发器 (UART)” (DS70000582)。
- 2:** 该功能仅在 16 倍 BRG 模式 (BRGH = 0) 下可用。



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 20-2: UxSTA: UARTx 状态和控制寄存器

R/W-0	R/W-0	R/W-0	U-0	R/W-0, HC	R/W-0	R-0	R-1
UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN <sup>(1)</sup>	UTXBF	TRMT
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7						bit 0	

<b>图注:</b>	C = 可清零位	HC = 硬件清零位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15,13 **UTXISEL<1:0>**: UARTx 发送中断模式选择位  
 11 = 保留; 不要使用  
 10 = 当一个字符被传输到发送移位寄存器 (Transmit Shift Register, TSR) 导致发送缓冲区为空时, 产生中断  
 01 = 当最后一个字符被移出发送移位寄存器, 所有发送操作执行完毕时产生中断  
 00 = 当一个字符被传输到发送移位寄存器 (这意味着发送缓冲区中至少有一个字符) 时产生中断
- bit 14 **UTXINV**: UARTx 发送极性翻转位  
 如果 IREN = 0:  
 1 = UxTX 的空闲状态为 0  
 0 = UxTX 的空闲状态为 1  
 如果 IREN = 1:  
 1 = IrDA<sup>®</sup> 编码, UxTX 空闲状态为 1  
 0 = IrDA 编码, UxTX 空闲状态为 0
- bit 12 **未实现**: 读为 0
- bit 11 **UTXBRK**: UARTx 发送间隔位  
 1 = 在下次发送时发送同步间隔字符 —— 启动位, 后跟 12 个 0 位, 然后是停止位; 完成时由硬件清零  
 0 = 禁止或已完成同步间隔字符的发送
- bit 10 **UTXEN**: UARTx 发送使能位 <sup>(1)</sup>  
 1 = 使能发送, UARTx 控制 UxTX 引脚  
 0 = 禁止发送, 中止所有等待的发送, 缓冲区被复位; 由端口控制 UxTX 引脚
- bit 9 **UTXBF**: UARTx 发送缓冲区满状态位 (只读)  
 1 = 发送缓冲区已满  
 0 = 发送缓冲区未满; 至少还可再写入一个字符
- bit 8 **TRMT**: 发送移位寄存器空位 (只读)  
 1 = 发送移位寄存器为空, 且发送缓冲区为空 (上一次发送已完成)  
 0 = 发送移位寄存器非空, 发送在进行中或在发送缓冲区中排队
- bit 7-6 **URXISEL<1:0>**: UARTx 接收中断模式选择位  
 11 = 当 UxRSR 传输使接收缓冲区为满 (即有 4 个数据字符) 时, 中断标志位置 1  
 10 = 当 UxRSR 传输使接收缓冲区 3/4 满 (即有 3 个数据字符) 时, 中断标志位置 1  
 0x = 当接收到任一字符且将字符从 UxRSR 传输到接收缓冲区时, 中断标志位置 1; 接收缓冲区有一个或多个字符

**注 1:** 有关使能 UARTx 模块进行发送操作的信息, 请参见 《dsPIC33/PIC24 系列参考手册》中的 “通用异步收发器 (UART)” (DS70000582)。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 20-2: UxSTA: UARTx 状态和控制寄存器 (续)

- bit 5      **ADDEN:** 地址字符检测位 (接收到数据的 bit 8 = 1)  
1 = 使能地址检测模式; 如果没有选择 9 位模式, 该位将不会产生任何作用  
0 = 禁止地址检测模式
- bit 4      **RIDLE:** 接收器空闲位 (只读)  
1 = 接收器空闲  
0 = 接收器工作
- bit 3      **PERR:** 奇偶校验错误状态位 (只读)  
1 = 检测到当前字符 (接收 FIFO 顶部的字符) 的奇偶校验错误  
0 = 未检测到奇偶校验错误
- bit 2      **FERR:** 帧错误状态位 (只读)  
1 = 检测到当前字符 (接收 FIFO 顶部的字符) 的帧错误  
0 = 未检测到帧错误
- bit 1      **OERR:** 接收缓冲区溢出错误状态位 (清零 / 只读)  
1 = 接收缓冲区已溢出  
0 = 接收缓冲区未溢出: 清零原来置 1 的 OERR 位 (1 → 0 的跳变) 将使接收缓冲区和 UxRSR 复位为空状态
- bit 0      **URXDA:** UARTx 接收缓冲区中是否有数据标志位 (只读)  
1 = 接收缓冲区中有数据, 至少还有一个字符可被读取  
0 = 接收缓冲区为空

**注 1:** 有关使能 UARTx 模块进行发送操作的信息, 请参见《dsPIC33/PIC24 系列参考手册》中的“通用异步收发器 (UART)” (DS70000582)。

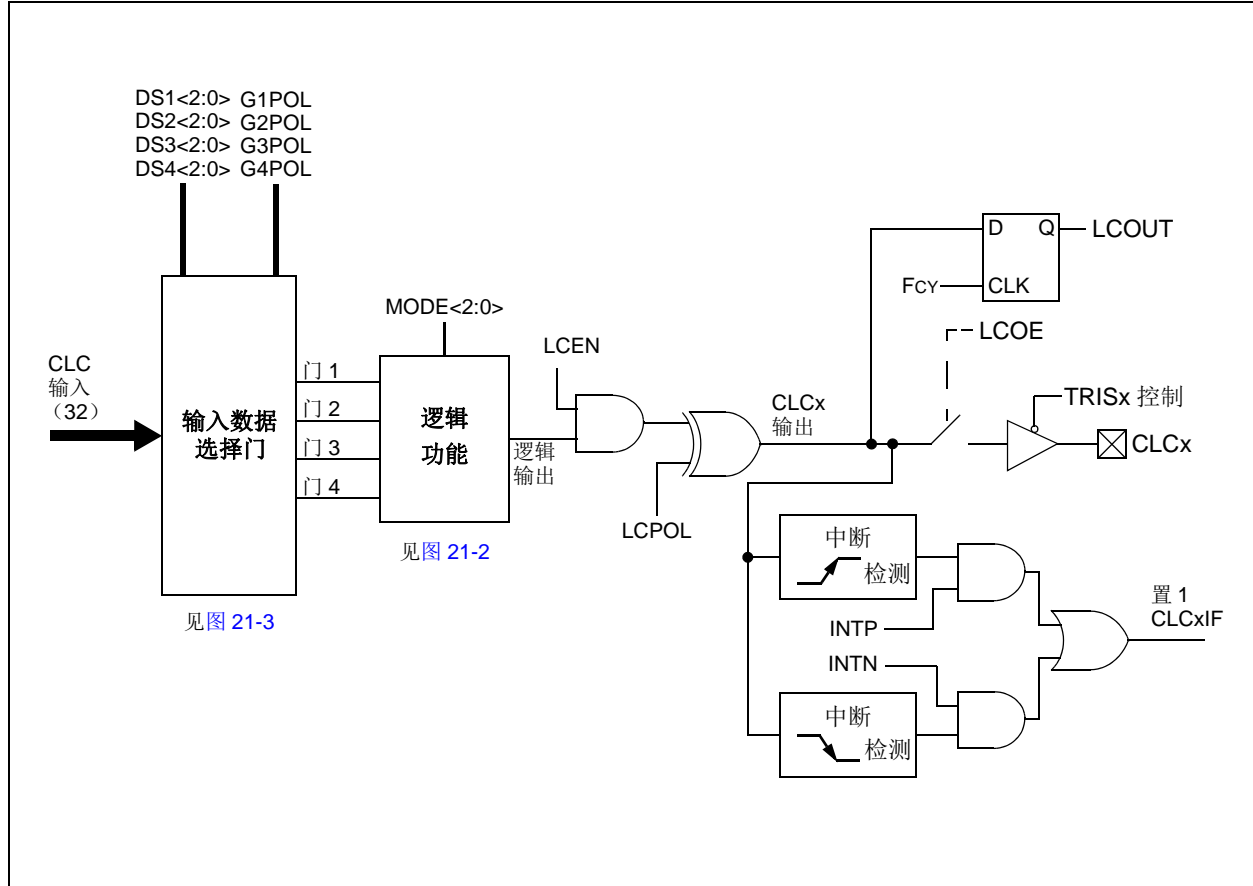
## 21.0 可配置逻辑单元 (CLC)

**注：** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“可配置逻辑单元 (CLC)” (DS70005298)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

可配置逻辑单元 (Configurable Logic Cell, CLC) 模块允许用户将一些信号的组合指定为逻辑功能的输入，并使用逻辑输出来控制其他外设或 I/O 引脚。由于 CLC 模块的操作不受软件执行限制，并支持大量的输出设计，这可以在嵌入式设计中提供更大的灵活性和可能性。

选定的逻辑功能具有 4 个输入门。这 4 个输入门使用 4 个数据源选择多路开关从最多 32 个信号的信号池中选择输入数据源。(见 P261)。图 21-1 给出了模块的概览。图 21-3 给出了数据源多路开关和逻辑输入门连接的详细信息。

图 21-1: CLCx 模块



# dsPIC33EPXXGS70X/80X 系列

图 21-2: CLCx 逻辑功能组合选项

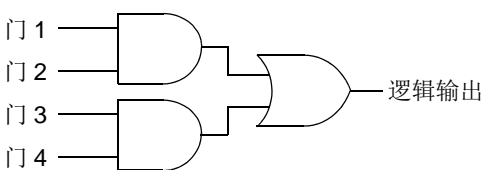
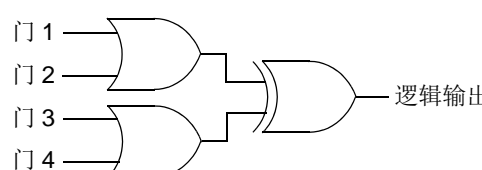
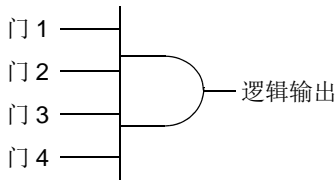
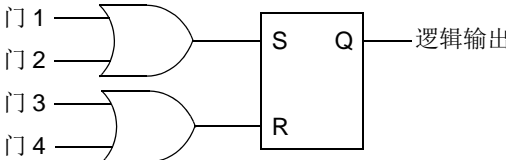
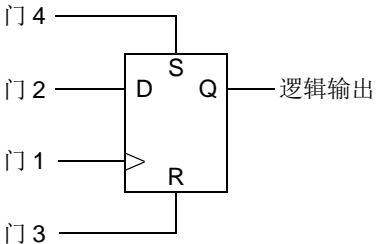
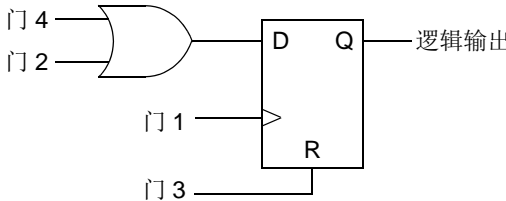
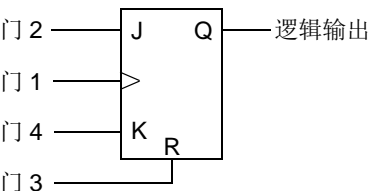
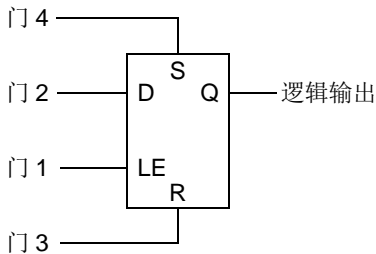
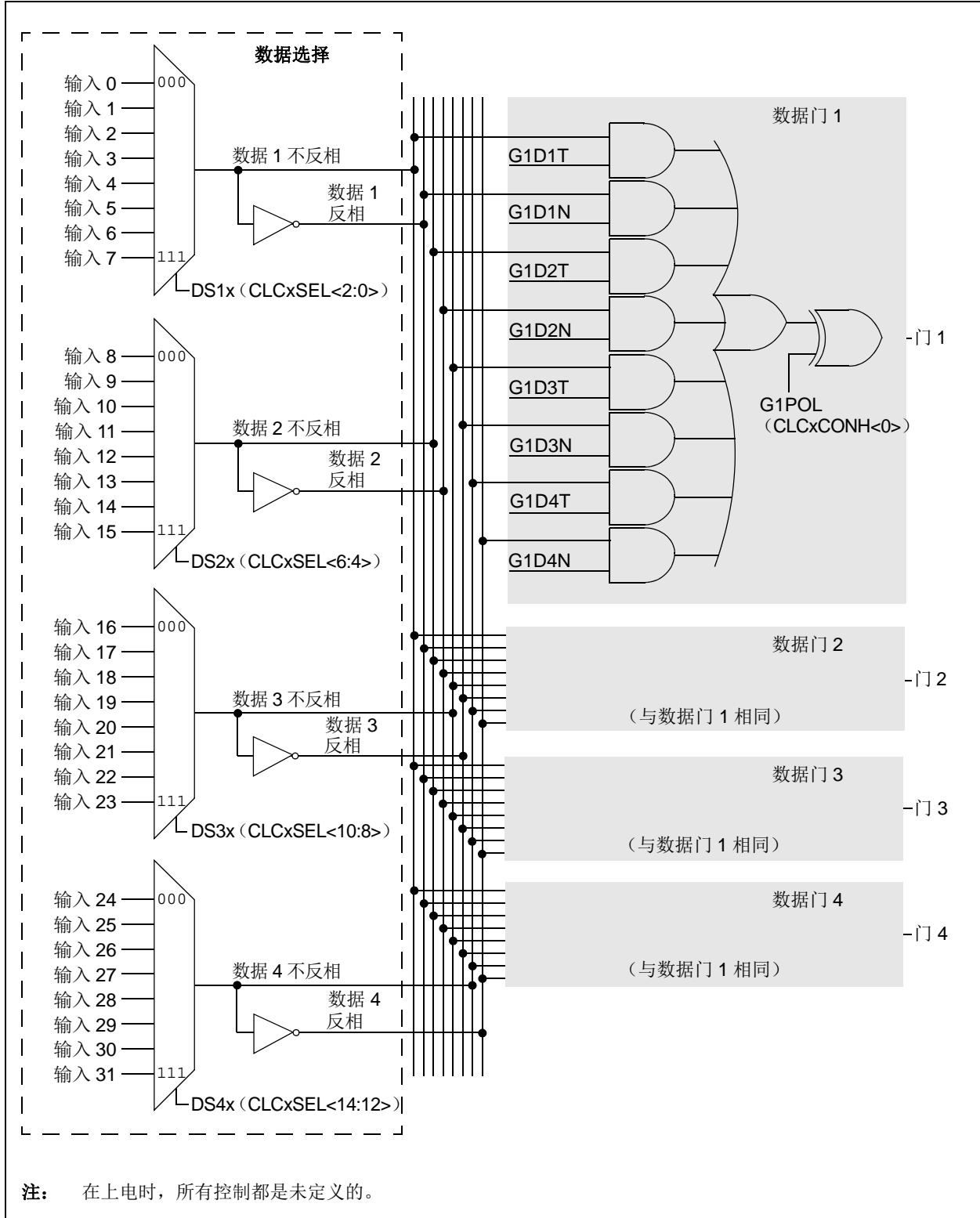
<p style="text-align: center;"><b>AND - OR</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 000</p>	<p style="text-align: center;"><b>OR - XOR</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 001</p>
<p style="text-align: center;"><b>4 输入 AND</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 010</p>	<p style="text-align: center;"><b>S-R 锁存器</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 011</p>
<p style="text-align: center;"><b>带置 1 和复位功能的 1 输入 D 触发器</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 100</p>	<p style="text-align: center;"><b>带复位功能的 2 输入 D 触发器</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 101</p>
<p style="text-align: center;"><b>带复位功能的 J-K 触发器</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 110</p>	<p style="text-align: center;"><b>带置 1 和复位功能的 1 输入透明锁存器</b></p>  <p style="text-align: center;">MODE&lt;2:0&gt; = 111</p>

图 21-3: CLCx 输入源选择图



# dsPIC33EPXXGS70X/80X 系列

## 21.1 控制寄存器

CLCx 模块由以下寄存器控制：

- CLCxCONL
- CLCxCONH
- CLCxSEL
- CLCxGLSL
- CLCxGLSH

CLCx 控制寄存器（CLCxCONL 和 CLCxCONH）用于使能模块和允许中断、控制输出使能位、选择输出极性和选择逻辑功能。此外，CLCx 控制寄存器不仅允许用户控制单元输出的逻辑极性，而且允许控制一些中间变量的逻辑极性。

CLCx 输入多路开关选择寄存器（CLCxSEL）允许用户使用 4 个数据输入选择多路开关来选择最多 4 个数据输入源。每个多路开关具有一组 8 个可用数据源。

CLCx 门逻辑输入选择寄存器（CLCxGLSL 和 CLCxGLSH）允许用户选择使用来自每个选择多路开关的哪些输出作为逻辑单元输入门的输入。每个数据源多路开关会同时输出其输出的真值和取反形式。这 8 个信号全部使能，通过逻辑单元输入门进行逻辑或运算。

**寄存器 21-1: CLCxCONL: CLCx 控制寄存器（低位字）**

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0
LCEN	—	—	—	INTP	INTN	—	—
bit 15							bit 8

R-0	R-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
LCOE	LCOUT	LCPOL	—	—	MODE2	MODE1	MODE0
bit 7							bit 0

**图注：**

R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15 **LCEN:** CLCx 使能位  
1 = 使能 CLCx，并对输入信号进行混合  
0 = 禁止 CLCx，并输出逻辑 0
- bit 14-12 **未实现:** 读为 0
- bit 11 **INTP:** CLCx 正边沿中断允许位  
1 = 当 LCOUT 上出现上升沿时，将产生中断  
0 = 不产生中断
- bit 10 **INTN:** CLCx 负边沿中断允许位  
1 = 当 LCOUT 上出现下降沿时，将产生中断  
0 = 不产生中断
- bit 9-8 **未实现:** 读为 0
- bit 7 **LCOE:** CLCx 端口使能位  
1 = 使能 CLCx 端口引脚输出  
0 = 禁止 CLCx 端口引脚输出
- bit 6 **LCOUT:** CLCx 数据输出状态位  
1 = CLCx 输出高电平  
0 = CLCx 输出低电平
- bit 5 **LCPOL:** CLCx 输出极性控制位  
1 = 模块的输出反相  
0 = 模块的输出不反相
- bit 4-3 **未实现:** 读为 0

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 21-1: CLCxCONL: CLCx 控制寄存器 (低位字) (续)

bit 2-0 **MODE<2:0>**: CLCx 模式位

- 111 = 带置 1 和复位功能的单输入透明锁存器
- 110 = 带复位功能的 JK 触发器
- 101 = 带复位功能的双输入 D 触发器
- 100 = 带置 1 和复位功能的单输入 D 触发器
- 011 = SR 锁存器
- 010 = 4 输入 AND
- 001 = 4 输入 OR-XOR
- 000 = 4 输入 AND-OR

## 寄存器 21-2: CLCxCONH: CLCx 控制寄存器 (高位字)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	G4POL	G3POL	G2POL	G1POL
bit 7							bit 0

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-4 **未实现**: 读为 0

bit 3 **G4POL**: 门 4 极性控制位

- 1 = 通道 4 逻辑输出在施加到逻辑单元时反相
- 0 = 通道 4 逻辑输出不反相

bit 2 **G3POL**: 门 3 极性控制位

- 1 = 通道 3 逻辑输出在施加到逻辑单元时反相
- 0 = 通道 3 逻辑输出不反相

bit 1 **G2POL**: 门 2 极性控制位

- 1 = 通道 2 逻辑输出在施加到逻辑单元时反相
- 0 = 通道 2 逻辑输出不反相

bit 0 **G1POL**: 门 1 极性控制位

- 1 = 通道 1 逻辑输出在施加到逻辑单元时反相
- 0 = 通道 1 逻辑输出不反相

# dsPIC33EPXXGS70X/80X 系列

寄存器 21-3: CLCxSEL: CLCx 输入多路开关选择寄存器

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	DS4<2:0>			—	DS3<2:0>		
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	DS2<2:0>			—	DS1<2:0>		
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15      **未实现:** 读为 0

bit 14-12    **DS4<2:0>:** 数据选择多路开关 4 信号选择位  
关于输入选择, 请参见表 21-1。

bit 11      **未实现:** 读为 0

bit 10-8    **DS3<2:0>:** 数据选择多路开关 3 信号选择位  
关于输入选择, 请参见表 21-1。

bit 7       **未实现:** 读为 0

bit 6-4     **DS2<2:0>:** 数据选择多路开关 2 信号选择位  
关于输入选择, 请参见表 21-1。

bit 3       **未实现:** 读为 0

bit 2-0     **DS1<2:0>:** 数据选择多路开关 1 信号选择位  
关于输入选择, 请参见表 21-1。



# dsPIC33EPXXGS70X/80X 系列

表 21-1: CLC1 多路开关输入源

	DSx<2:0>	信号源
DS1<2:0>	000	CLCINA
	001	系统时钟
	010	Timer1 匹配
	011	PWM1H
	100	PWM5L
	101	高速 PWM 时钟
	110	Timer2 匹配
	111	Timer3 匹配
DS2<2:0>	000	CLCINB
	001	CLC2 输出
	010	CMP1 输出
	011	UART1 发送输出
	100	ADC 转换结束
	101	DMA 通道 0 中断
	110	PWM1L
	111	PWM5H
DS3<2:0>	000	CLCINA
	001	CLC1 输出
	010	CMP2 输出
	011	SPI1 SDO 输出
	100	UART1 接收
	101	PWM2H
	110	PWM6L
	111	OCMP2
DS4<2:0>	000	CLCINB
	001	CLC2 输出
	010	CMP3 输出
	011	SDI1
	100	PTG
	101	ECAN1
	110	PWM2L
	111	PWM6H

# dsPIC33EPXXGS70X/80X 系列

表 21-2: CLC2 多路开关输入源

	DSx<2:0>	信号源
DS1<2:0>	000	CLCINA
	001	系统时钟
	010	Timer1 匹配
	011	PWM3H
	100	PWM7L
	101	高速 PWM 时钟
	110	Timer2 匹配
	111	Timer3 匹配
DS2<2:0>	000	CLCINB
	001	CLC1 输出
	010	CMP1 输出
	011	UART2 发送输出
	100	ADC 转换结束
	101	DMA 通道 0 中断
	110	PWM3L
	111	PWM7H
DS3<2:0>	000	CLCINA
	001	CLC2 输出
	010	CMP2 输出
	011	SPI2 SDO 输出
	100	UART2 接收
	101	PWM4H
	110	PWM8L
	111	OCMP2
DS4<2:0>	000	CLCINB
	001	CLC1 输出
	010	CMP3 输出
	011	SDI2
	100	PTG
	101	ECAN1
	110	PWM4L
	111	PWM8H

# dsPIC33EPXXGS70X/80X 系列

表 21-3: CLC3 多路开关输入源

	DSx<2:0>	信号源
DS1<2:0>	000	CLCINA
	001	系统时钟
	010	Timer1 匹配
	011	PWM5H
	100	REFO1 时钟输出
	101	高速 PWM 时钟
	110	Timer2 匹配
	111	PWM3L
DS2<2:0>	000	CLCINB
	001	CLC4 输出
	010	CMP1 输出
	011	PWM5L
	100	ADC 转换结束
	101	PWM3H
	110	ICAP1
	111	ICAP2
DS3<2:0>	000	CLCINA
	001	CLC3 输出
	010	CMP2 输出
	011	PWM6H
	100	UART1 接收
	101	DMA 通道 1 中断
	110	OCMP1
	111	PWM4L
DS4<2:0>	000	CLCINB
	001	CLC4 输出
	010	CMP3 输出
	011	PWM6L
	100	PTG
	101	PWM4H
	110	PC_PWM
	111	OCMP3

# dsPIC33EPXXGS70X/80X 系列

表 21-4: CLC4 多路开关输入源

	DSx<2:0>	信号源
DS1<2:0>	000	CLCINA
	001	PWM7H
	010	Timer1 匹配
	011	INTOSC/LPRC 时钟
	100	REFO1 时钟输出
	101	高速 PWM 时钟
	110	Timer2 匹配
	111	PWM1L
DS2<2:0>	000	CLCINB
	001	CLC3 输出
	010	CMP1 输出
	011	PWM7L
	100	ADC 转换结束
	101	PWM1H
	110	ICAP1
	111	ICAP2
DS3<2:0>	000	CLCINA
	001	CLC4 输出
	010	CMP2 输出
	011	PWM8H
	100	UART2 接收
	101	DMA 通道 1 中断
	110	OCMP1
	111	PWM2L
DS4<2:0>	000	CLCINB
	001	CLC3 输出
	010	CMP3 输出
	011	PWM8L
	100	PTG
	101	PWM2H
	110	PC_PWM
	111	OCMP3

# dsPIC33EPXXGS70X/80X 系列

寄存器 21-4: CLCxGLSL: CLCx 门逻辑输入选择低位字寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G2D4T	G2D4N	G2D3T	G2D3N	G2D2T	G2D2N	G2D1T	G2D1N
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G1D4T	G1D4N	G1D3T	G1D3N	G1D2T	G1D2N	G1D1T	G1D1N
bit 7						bit 0	

**图注:**

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **G2D4T:** 门 2 数据源 4 真值使能位

1 = 对于门 2 使能数据源 4 同相信号  
0 = 对于门 2 禁止数据源 4 同相信号

bit 14 **G2D4N:** 门 2 数据源 4 取反使能位

1 = 对于门 2 使能数据源 4 反相信号  
0 = 对于门 2 禁止数据源 4 反相信号

bit 13 **G2D3T:** 门 2 数据源 3 真值使能位

1 = 对于门 2 使能数据源 3 同相信号  
0 = 对于门 2 禁止数据源 3 同相信号

bit 12 **G2D3N:** 门 2 数据源 3 取反使能位

1 = 对于门 2 使能数据源 3 反相信号  
0 = 对于门 2 禁止数据源 3 反相信号

bit 11 **G2D2T:** 门 2 数据源 2 真值使能位

1 = 对于门 2 使能数据源 2 同相信号  
0 = 对于门 2 禁止数据源 2 同相信号

bit 10 **G2D2N:** 门 2 数据源 2 取反使能位

1 = 对于门 2 使能数据源 2 反相信号  
0 = 对于门 2 禁止数据源 2 反相信号

bit 9 **G2D1T:** 门 2 数据源 1 真值使能位

1 = 对于门 2 使能数据源 1 同相信号  
0 = 对于门 2 禁止数据源 1 同相信号

bit 8 **G2D1N:** 门 2 数据源 1 取反使能位

1 = 对于门 2 使能数据源 1 反相信号  
0 = 对于门 2 禁止数据源 1 反相信号

bit 7 **G1D4T:** 门 1 数据源 4 真值使能位

1 = 对于门 1 使能数据源 4 同相信号  
0 = 对于门 1 禁止数据源 4 同相信号

bit 6 **G1D4N:** 门 1 数据源 4 取反使能位

1 = 对于门 1 使能数据源 4 反相信号  
0 = 对于门 1 禁止数据源 4 反相信号

bit 5 **G1D3T:** 门 1 数据源 3 真值使能位

1 = 对于门 1 使能数据源 3 同相信号  
0 = 对于门 1 禁止数据源 3 同相信号

bit 4 **G1D3N:** 门 1 数据源 3 取反使能位

1 = 对于门 1 使能数据源 3 反相信号  
0 = 对于门 1 禁止数据源 3 反相信号

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 21-4: CLCxGLSL: CLCx 门逻辑输入选择低位字寄存器 (续)

bit 3	<b>G1D2T:</b> 门 1 数据源 2 真值使能位 1 = 对于门 1 使能数据源 2 同相信号 0 = 对于门 1 禁止数据源 2 同相信号
bit 2	<b>G1D2N:</b> 门 1 数据源 2 取反使能位 1 = 对于门 1 使能数据源 2 反相信号 0 = 对于门 1 禁止数据源 2 反相信号
bit 1	<b>G1D1T:</b> 门 1 数据源 1 真值使能位 1 = 对于门 1 使能数据源 1 同相信号 0 = 对于门 1 禁止数据源 1 同相信号
bit 0	<b>G1D1N:</b> 门 1 数据源 1 取反使能位 1 = 对于门 1 使能数据源 1 反相信号 0 = 对于门 1 禁止数据源 1 反相信号

# dsPIC33EPXXGS70X/80X 系列

寄存器 21-5: CLCxGLSH: CLCx 门逻辑输入选择高位字寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G4D4T	G4D4N	G4D3T	G4D3N	G4D2T	G4D2N	G4D1T	G4D1N
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
G3D4T	G3D4N	G3D3T	G3D3N	G3D2T	G3D2N	G3D1T	G3D1N
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15        **G4D4T:** 门 4 数据源 4 真值使能位  
               1 = 对于门 4 使能数据源 4 同相信号  
               0 = 对于门 4 禁止数据源 4 同相信号
- bit 14        **G4D4N:** 门 4 数据源 4 取反使能位  
               1 = 对于门 4 使能数据源 4 反相信号  
               0 = 对于门 4 禁止数据源 4 反相信号
- bit 13        **G4D3T:** 门 4 数据源 3 真值使能位  
               1 = 对于门 4 使能数据源 3 同相信号  
               0 = 对于门 4 禁止数据源 3 同相信号
- bit 12        **G4D3N:** 门 4 数据源 3 取反使能位  
               1 = 对于门 4 使能数据源 3 反相信号  
               0 = 对于门 4 禁止数据源 3 反相信号
- bit 11        **G4D2T:** 门 4 数据源 2 真值使能位  
               1 = 对于门 4 使能数据源 2 同相信号  
               0 = 对于门 4 禁止数据源 2 同相信号
- bit 10        **G4D2N:** 门 4 数据源 2 取反使能位  
               1 = 对于门 4 使能数据源 2 反相信号  
               0 = 对于门 4 禁止数据源 2 反相信号
- bit 9         **G4D1T:** 门 4 数据源 1 真值使能位  
               1 = 对于门 4 使能数据源 1 同相信号  
               0 = 对于门 4 禁止数据源 1 同相信号
- bit 8         **G4D1N:** 门 4 数据源 1 取反使能位  
               1 = 对于门 4 使能数据源 1 反相信号  
               0 = 对于门 4 禁止数据源 1 反相信号
- bit 7         **G3D4T:** 门 3 数据源 4 真值使能位  
               1 = 对于门 3 使能数据源 4 同相信号  
               0 = 对于门 3 禁止数据源 4 同相信号
- bit 6         **G3D4N:** 门 3 数据源 4 取反使能位  
               1 = 对于门 3 使能数据源 4 反相信号  
               0 = 对于门 3 禁止数据源 4 反相信号
- bit 5         **G3D3T:** 门 3 数据源 3 真值使能位  
               1 = 对于门 3 使能数据源 3 同相信号  
               0 = 对于门 3 禁止数据源 3 同相信号
- bit 4         **G3D3N:** 门 3 数据源 3 取反使能位  
               1 = 对于门 3 使能数据源 3 反相信号  
               0 = 对于门 3 禁止数据源 3 反相信号

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 21-5: CLCxGLSH: CLCx 门逻辑输入选择高位字寄存器 (续)

- bit 3      **G3D2T:** 门 3 数据源 2 真值使能位  
1 = 对于门 3 使能数据源 2 同相信号  
0 = 对于门 3 禁止数据源 2 同相信号
- bit 2      **G3D2N:** 门 3 数据源 2 取反使能位  
1 = 对于门 3 使能数据源 2 反相信号  
0 = 对于门 3 禁止数据源 2 反相信号
- bit 1      **G3D1T:** 门 3 数据源 1 真值使能位  
1 = 对于门 3 使能数据源 1 同相信号  
0 = 对于门 3 禁止数据源 1 同相信号
- bit 0      **G3D1N:** 门 3 数据源 1 取反使能位  
1 = 对于门 3 使能数据源 1 反相信号  
0 = 对于门 3 禁止数据源 1 反相信号



## 22.0 高速 12 位模数转换器 (ADC)

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息, 请参见《dsPIC33/PIC24 系列参考手册》的“12 位高速多 SAR A/D 转换器 (ADC)” (DS70005213), 该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息, 请参见本数据手册中的第 4.0 节“存储器构成”。

dsPIC33EPXXGS70X/80X 器件具有一个高速 12 位模数转换器 (ADC), 该模数转换器具有低转换延时、高分辨率和过采样能力, 可提高交流 / 直流和直流 / 直流电源转换器中的性能。

### 22.1 特性概述

高速 12 位多 SAR 模数转换器 (ADC) 包括以下特性:

- 5 个 ADC 内核: 4 个专用内核和 1 个共用内核
- 用户可为每个内核配置最高 12 位的分辨率
- 12 位分辨率时每个通道的转换速率最高为 3.25 Msps
- 低延时转换
- 最多 22 个模拟输入通道, 每个输入具有单独的 16 位转换结果寄存器
- 对于所有通道中的每个通道, 转换结果可以采用无符号或有符号数据格式
- 所有 ADC 内核上均提供单端转换和伪差分转换

- 可同时采样最多 5 个模拟输入
- 通道扫描能力
- 每个内核有多个转换触发选项, 包括:
  - PWM1 至 PWM6 (主触发和辅助触发以及限流事件触发)
  - PWM 特殊事件触发
  - Timer1/Timer2 周期匹配
  - 输出比较 1 和事件触发
  - 外部引脚触发事件 (ADTRG31)
  - 软件触发
- 两个带专用中断的集成数字比较器:
  - 多个比较选项
  - 可分配给特定的模拟输入
- 两个带专用中断的过采样滤波器:
  - 提供更高的分辨率
  - 可分配给特定的模拟输入

模块包含 5 个独立的 SAR ADC 内核。图 22-1、图 22-2 和图 22-3 给出了 12 位多 SAR ADC 的简化框图。

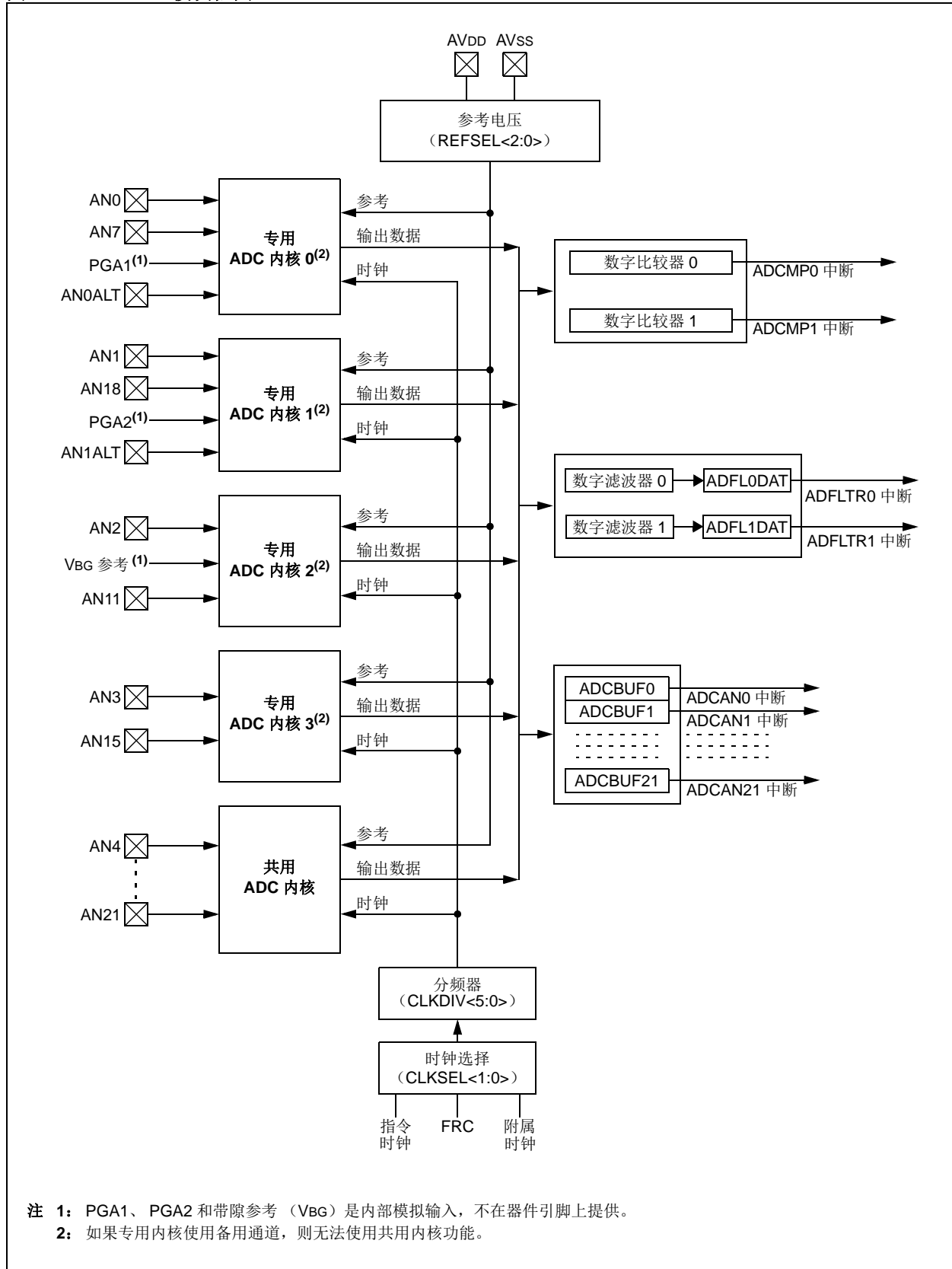
模拟输入 (通道) 通过多路开关和开关连接到每个 ADC 内核的采样保持 (Sample-and-Hold, S&H) 电路。内核使用通道信息 (输出格式、测量模式和输入编号) 来处理模拟采样。转换完成时, 结果将存储到特定模拟输入的结果缓冲区中, 并在数字滤波器和数字比较器配置为使用此特定通道中的数据时传递给二者。

ADC 模块每次最多可采样 5 个输入 (4 个输入来自专用 SAR 内核, 1 个输入来自共用 SAR 内核)。如果共用内核上有多个 ADC 输入请求转换, 模块将从编号最低的输入开始按顺序进行转换。

ADC 使每个模拟输入能够指定其自身的触发源。该功能使 ADC 可以采样和转换与依靠独立时基工作的 PWM 发生器关联的模拟输入。

# dsPIC33EPXXGS70X/80X 系列

图 22-1: ADC 模块框图



- 注 1: PGA1、PGA2 和带隙参考 (V<sub>BG</sub>) 是内部模拟输入, 不在器件引脚上提供。  
 注 2: 如果专用内核使用备用通道, 则无法使用共用内核功能。

# dsPIC33EPXXGS70X/80X 系列

图 22-2: 专用 ADC 内核 0 至 3 框图

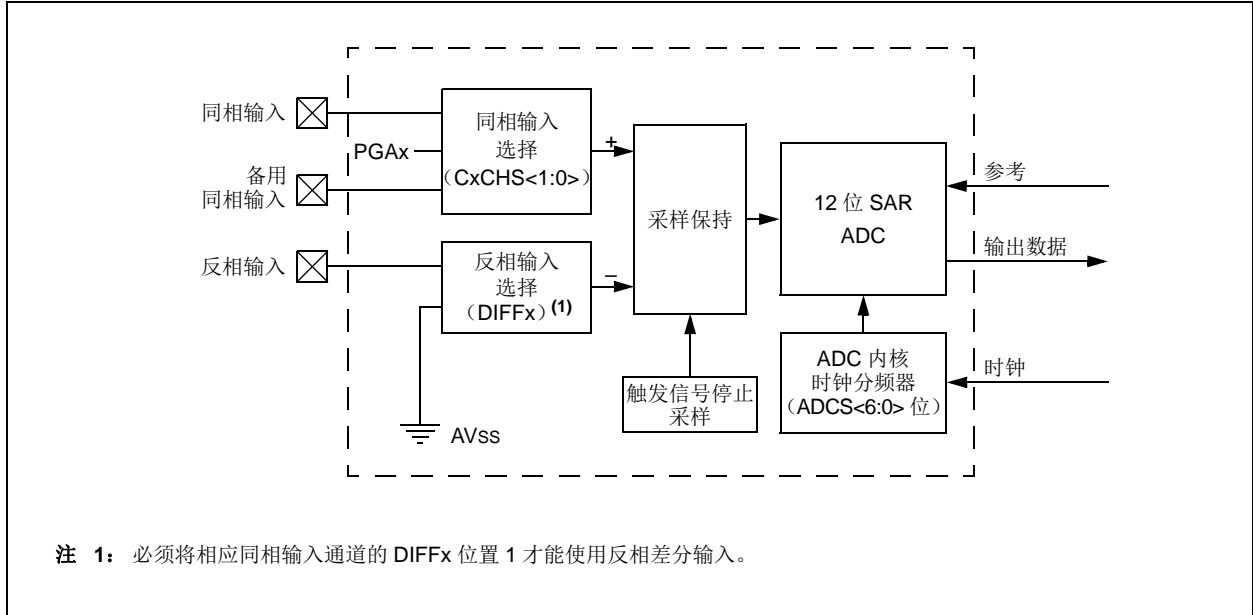
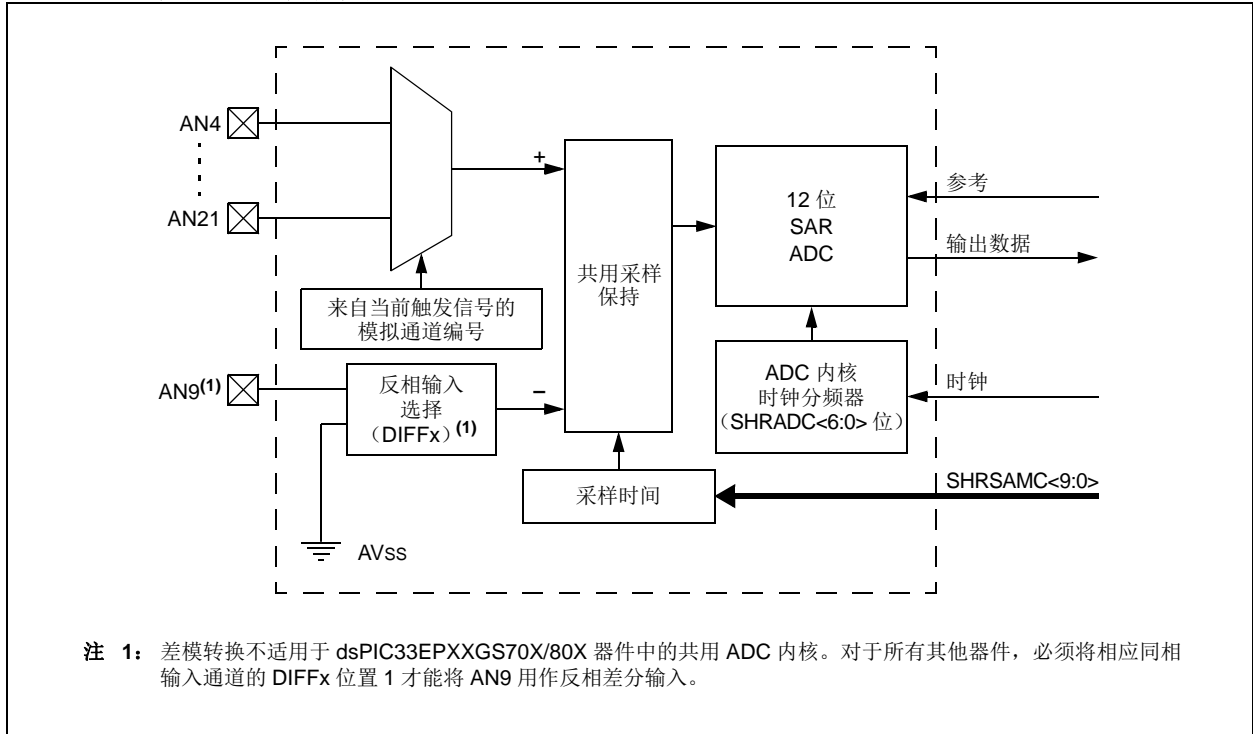


图 22-3: 共用 ADC 内核框图



# dsPIC33EPXXGS70X/80X 系列

## 22.2 模数转换器资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 22.2.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“12 位高速多 SAR A/D 转换器 (ADC)” (DS70005213)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

寄存器 22-1: ADON1L: ADC 控制寄存器 1 的低位字

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
ADON <sup>(1)</sup>	—	ADSIDL	—	—	—	—	—
bit 15							bit 8

U-0	r-0	r-0	r-0	r-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

<b>图注:</b>	r = 保留位		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15 **ADON:** ADC 使能位 <sup>(1)</sup>  
1 = 使能 ADC 模块  
0 = 禁止 ADC 模块
- bit 14 **未实现:** 读为 0
- bit 13 **ADSIDL:** ADC 空闲模式停止位  
1 = 当器件进入空闲模式时，模块停止工作  
0 = 在空闲模式下模块继续工作
- bit 12-7 **未实现:** 读为 0
- bit 6-3 **保留:** 保持为 0
- bit 2-0 **未实现:** 读为 0

**注 1:** 仅应在配置完 ADC 模块后将 ADON 位置 1。在 ADON = 1 时更改 ADC 配置位将导致不可预测的行为。

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-2: **ADCON1H: ADC 控制寄存器 1 的高位字**

r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-1	R/W-1	r-0	r-0	r-0	r-0	r-0
FORM	SHRRES1	SHRRES0	—	—	—	—	—
bit 7							bit 0

<b>图注:</b>	r = 保留位						
R = 可读位	W = 可写位	U = 未实现位, 读为 0					
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知				

- bit 15-8     **保留:** 保持为 0
- bit 7       **FORM:** 小数数据输出格式位
  - 1 = 小数
  - 0 = 整数
- bit 6-5     **SHRRES<1:0>:** 共用 ADC 内核分辨率选择位
  - 11 = 12 位分辨率
  - 10 = 10 位分辨率
  - 01 = 8 位分辨率
  - 00 = 6 位分辨率
- bit 4-0     **保留:** 保持为 0

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-3: **ADCON2L: ADC 控制寄存器 2 的低位字**

R/W-0	R/W-0	r-0	R/W-0	r-0	R/W-0	R/W-0	R/W-0
REFCIE	REFERCIE	—	EIEN	—	SHREISEL2 <sup>(1)</sup>	SHREISEL1 <sup>(1)</sup>	SHREISEL0 <sup>(1)</sup>
bit 15						bit 8	

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	SHRADCS6	SHRADCS5	SHRADCS4	SHRADCS3	SHRADCS2	SHRADCS1	SHRADCS0
bit 7						bit 0	

<b>图注:</b>	r = 保留位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15      **REFCIE:** 带隙和参考电压就绪通用中断允许位  
1 = 带隙就绪时产生通用中断  
0 = 禁止在发生带隙就绪事件时产生通用中断
- bit 14      **REFERCIE:** 带隙或参考电压错误通用中断允许位  
1 = 检测到带隙或参考电压错误时产生通用中断  
0 = 禁止在发生带隙和参考电压错误事件时产生通用中断
- bit 13      **保留:** 保持为 0
- bit 12      **EIEN:** 提前中断允许位  
1 = 为输入通道中断允许提前中断功能 (EISTATx 标志置 1 时)  
0 = 转换完成时产生各个中断 (ANxRDY 标志置 1)
- bit 11      **保留:** 保持为 0
- bit 10-8    **SHREISEL<2:0>:** 共用内核提前中断时间选择位 <sup>(1)</sup>  
111 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 8 个 TADCORE 时钟  
110 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 7 个 TADCORE 时钟  
101 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 6 个 TADCORE 时钟  
100 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 5 个 TADCORE 时钟  
011 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 4 个 TADCORE 时钟  
010 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 3 个 TADCORE 时钟  
001 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 2 个 TADCORE 时钟  
000 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 1 个 TADCORE 时钟
- bit 7      **未实现:** 读为 0
- bit 6-0    **SHRADCS<6:0>:** 共用 ADC 内核输入时钟分频比位  
这些位用于确定一个共用 TADCORE (内核时钟周期) 中的 TCORESRC (源时钟周期) 数。  
1111111 = 254 个源时钟周期  
•  
•  
•  
0000011 = 6 个源时钟周期  
0000010 = 4 个源时钟周期  
0000001 = 2 个源时钟周期  
0000000 = 2 个源时钟周期

**注 1:** 对于 6 位共用 ADC 内核分辨率 (SHRRES<1:0> = 00), SHREISEL<2:0> 设置 100 至 111 无效, 请勿使用。对于 8 位共用 ADC 内核分辨率 (SHRRES<1:0> = 01), SHREISEL<2:0> 设置 110 和 111 无效, 请勿使用。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-4: ADCON2H: ADC 控制寄存器 2 的高位字

R-0, HSC	R-0, HSC	r-0	r-0	r-0	r-0	R/W-0	R/W-0
REFRDY	REFERR	—	—	—	—	SHRSAMC9	SHRSAMC8
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SHRSAMC7	SHRSAMC6	SHRSAMC5	SHRSAMC4	SHRSAMC3	SHRSAMC2	SHRSAMC1	SHRSAMC0
bit 7						bit 0	

<b>图注:</b>	r = 保留位	U = 未实现位, 读为 0
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15      **REFRDY:** 带隙和参考电压就绪标志位

1 = 带隙就绪  
0 = 带隙未就绪

bit 14      **REFERR:** 带隙或参考电压错误标志位

1 = ADC 模块使能 (ADON = 1) 后带隙被移除  
0 = 未检测到带隙错误

bit 13-10   **保留:** 保持为 0

bit 9-0      **SHRSAMC<9:0>:** 共用 ADC 内核采样时间选择位

这些位用于指定共用 ADC 内核采样时间内的共用 ADC 内核时钟周期 (TADCORE) 数。

1111111111 = 1025 个 TADCORE

•  
•  
•

0000000001 = 3 个 TADCORE

0000000000 = 2 个 TADCORE

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-5: **ADCON3L: ADC 控制寄存器 3 的低位字**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0, HSC	R/W-0	R-0, HSC
REFSEL2	REFSEL1	REFSEL0	SUSPEND	SUSPCIE	SUSPRDY	SHRSAMP	CNVRTCH
bit 15						bit 8	
R/W-0	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SWLCTRG	SWCTRG	CNVCHSEL5	CNVCHSEL4	CNVCHSEL3	CNVCHSEL2	CNVCHSEL1	CNVCHSEL0
bit 7						bit 0	

<b>图注:</b>	U = 未实现位, 读为 0		
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13 **REFSEL<2:0>**: ADC 参考电压选择位

值	VREFH	VREFL
000	AVDD	AVSS

001-111 = **未实现**: 不使用

bit 12 **SUSPEND**: 所有 ADC 内核触发禁止位

1 = 禁止所有 ADC 内核的所有新触发事件  
0 = 可以触发所有 ADC 内核

bit 11 **SUSPCIE**: 所有 ADC 内核暂停通用中断允许位

1 = ADC 内核触发暂停 (**SUSPEND** 位 = 1) 且之前的所有转换完成 (**SUSPRDY** 位置 1) 时产生通用中断  
0 = 发生 ADC 内核暂停事件时不产生通用中断

bit 10 **SUSPRDY**: 所有 ADC 内核暂停标志位

1 = 所有 ADC 内核暂停 (**SUSPEND** 位 = 1) 且未在进行任何转换  
0 = ADC 内核正在进行之前的转换

bit 9 **SHRSAMP**: 共用 ADC 内核采样直接控制位

该位应与 **CNVRTCH** 位控制的单独通道转换触发信号配合使用。它将 **CNVCHSEL<5:0>** 位指定的模拟输入与共用 ADC 内核相连, 并允许延长采样时间。该位并非由硬件控制, 必须在转换开始前清零 (将 **CNVRTCH** 置 1)。

1 = 共用 ADC 内核采样 **CNVCHSEL<5:0>** 位指定的模拟输入  
0 = 采样由共用 ADC 内核硬件控制

bit 8 **CNVRTCH**: 软件单独通道转换触发位

1 = 为 **CNVCHSEL<5:0>** 位指定的模拟输入产生单个触发信号; 该位置 1 后, 将由硬件在下一个指令周期自动清零  
0 = 可产生下一个单独通道转换触发信号

bit 7 **SWLCTRG**: 软件电平敏感通用触发位

1 = 连续为所有通道产生触发信号, 其中软件电平敏感通用触发信号选作 **ADTRIGxL** 和 **ADTRIGxH** 寄存器中的触发源  
0 = 不产生软件电平敏感通用触发信号

bit 6 **SWCTRG**: 软件通用触发位

1 = 为所有通道产生单个触发信号, 其中通用触发信号选作 **ADTRIGxL** 和 **ADTRIGxH** 寄存器中的触发源; 该位置 1 后, 将由硬件在下一个指令周期自动清零  
0 = 准备产生下一个软件通用触发信号

bit 5-0 **CNVCHSEL<5:0>**: 软件单独通道转换触发的通道编号选择位

这些位用于定义 **CNVRTCH** 位置 1 时要转换的通道。



# dsPIC33EPXXGS70X/80X 系列

**寄存器 22-6: ADCON3H: ADC 控制寄存器 3 的高位字**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CLKSEL1	CLKSEL0	CLKDIV5	CLKDIV4	CLKDIV3	CLKDIV2	CLKDIV1	CLKDIV0
bit 15							bit 8
R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
SHREN	—	—	—	C3EN	C2EN	C1EN	C0EN
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-14 **CLKSEL<1:0>**: ADC 模块时钟源选择位

- 11 = APLL
- 10 = FRC
- 01 = FOSC (系统时钟 x 2)
- 00 = FSYS (系统时钟)

bit 13-8 **CLKDIV<5:0>**: ADC 模块时钟源分频比位

该分频比根据 CLKSEL<1:0> 位选择的 T<sub>SR</sub>C ADC 模块时钟源形成所有 ADC 内核 (共用和专用) 使用的 T<sub>CO</sub>RESRC 时钟。随后, 每个 ADC 内核通过 ADCORExH 寄存器中的 ADCS<6:0> 位或 ADCON2L 寄存器中的 SHRADCS<6:0> 位单独对 T<sub>CO</sub>RESRC 时钟进行分频, 以获得内核特定的 T<sub>AD</sub>CORE 时钟。

- 111111 = 64 个源时钟周期
- 
- 
- 
- 000011 = 4 个源时钟周期
- 000010 = 3 个源时钟周期
- 000001 = 2 个源时钟周期
- 000000 = 1 个源时钟周期

bit 7 **SHREN**: 共用 ADC 内核使能位

- 1 = 使能共用 ADC 内核
- 0 = 禁止共用 ADC 内核

bit 6-4 **未实现**: 读为 0

bit 3 **C3EN**: 专用 ADC 内核 3 使能位

- 1 = 使能专用 ADC 内核 3
- 0 = 禁止专用 ADC 内核 3

bit 2 **C2EN**: 专用 ADC 内核 2 使能位

- 1 = 使能专用 ADC 内核 2
- 0 = 禁止专用 ADC 内核 2

bit 1 **C1EN**: 专用 ADC 内核 1 使能位

- 1 = 使能专用 ADC 内核 1
- 0 = 禁止专用 ADC 内核 1

bit 0 **C0EN**: 专用 ADC 内核 0 使能位

- 1 = 使能专用 ADC 内核 0
- 0 = 禁止专用 ADC 内核 0

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-7: **ADCON4L: ADC 控制寄存器 4 的低位字**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	SAMC3EN	SAMC2EN	SAMC1EN	SAMC0EN
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-4 **未实现:** 读为 0

bit 3 **SAMC3EN:** 专用 ADC 内核 3 转换延时使能位

1 = 触发后, 转换将延时, ADC 内核将在 ADCORE3L 寄存器中的 SAMC<9:0> 位指定的时间内继续采样  
0 = 触发后, 采样立即停止, 转换将在下一个内核时钟周期开始

bit 2 **SAMC2EN:** 专用 ADC 内核 2 转换延时使能位

1 = 触发后, 转换将延时, ADC 内核将在 ADCORE2L 寄存器中的 SAMC<9:0> 位指定的时间内继续采样  
0 = 触发后, 采样立即停止, 转换将在下一个内核时钟周期开始

bit 1 **SAMC1EN:** 专用 ADC 内核 1 转换延时使能位

1 = 触发后, 转换将延时, ADC 内核将在 ADCORE1L 寄存器中的 SAMC<9:0> 位指定的时间内继续采样  
0 = 触发后, 采样立即停止, 转换将在下一个内核时钟周期开始

bit 0 **SAMC0EN:** 专用 ADC 内核 0 转换延时使能位

1 = 触发后, 转换将延时, ADC 内核将在 ADCORE0L 寄存器中的 SAMC<9:0> 位指定的时间内继续采样  
0 = 触发后, 采样立即停止, 转换将在下一个内核时钟周期开始

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-8: ADCON4H: ADC 控制寄存器 4 的高位字

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C3CHS1	C3CHS0	C2CHS1	C2CHS0	C1CHS1	C1CHS0	C0CHS1	C0CHS0
bit 7						bit 0	

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15-8      **未实现:** 读为 0
- bit 7-6      **C3CHS<1:0>:** 专用 ADC 内核 3 输入通道选择位
  - 1x = 保留
  - 01 = AN15 (DIFF3 (ADMOD0L<7>) = 1 时为差分反相输入)
  - 00 = AN3
- bit 5-4      **C2CHS<1:0>:** 专用 ADC 内核 2 输入通道选择位
  - 11 = 保留
  - 10 = VREF 带隙
  - 01 = AN11 (DIFF2 (ADMOD0L<5>) = 1 时为差分反相输入)
  - 00 = AN2
- bit 3-2      **C1CHS<1:0>:** 专用 ADC 内核 1 输入通道选择位
  - 11 = AN1ALT
  - 10 = PGA2
  - 01 = AN18 (DIFF1 (ADMOD0L<3>) = 1 时为差分反相输入)
  - 00 = AN1
- bit 1-0      **C0CHS<1:0>:** 专用 ADC 内核 0 输入通道选择位
  - 11 = AN0ALT
  - 10 = PGA1
  - 01 = AN7 (DIFF0 (ADMOD0L<1>) = 1 时为差分反相输入)
  - 00 = AN0

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-9: **ADCON5L: ADC 控制寄存器 5 的低位字**

R-0, HSC	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
SHRRDY	—	—	—	C3RDY	C2RDY	C1RDY	C0RDY
bit 15				bit 8			

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
SHRPWR	—	—	—	C3PWR	C2PWR	C1PWR	C0PWR
bit 7				bit 0			

<b>图注:</b>	U = 未实现位, 读为 0		
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15      **SHRRDY:** 共用 ADC 内核就绪标志位  
1 = ADC 内核上电并准备工作  
0 = ADC 内核未准备好进行工作
- bit 14-12    **未实现:** 读为 0
- bit 11      **C3RDY:** 专用 ADC 内核 3 就绪标志位  
1 = ADC 内核上电并准备工作  
0 = ADC 内核未准备好进行工作
- bit 10      **C2RDY:** 专用 ADC 内核 2 就绪标志位  
1 = ADC 内核上电并准备工作  
0 = ADC 内核未准备好进行工作
- bit 9        **C1RDY:** 专用 ADC 内核 1 就绪标志位  
1 = ADC 内核上电并准备工作  
0 = ADC 内核未准备好进行工作
- bit 8        **C0RDY:** 专用 ADC 内核 0 就绪标志位  
1 = ADC 内核上电并准备工作  
0 = ADC 内核未准备好进行工作
- bit 7        **SHRPWR:** 共用 ADC 内核 x 电源使能位  
1 = ADC 内核 x 上电  
0 = ADC 内核 x 断电
- bit 6-4      **未实现:** 读为 0
- bit 3        **C3PWR:** 专用 ADC 内核 3 电源使能位  
1 = ADC 内核上电  
0 = ADC 内核断电
- bit 2        **C2PWR:** 专用 ADC 内核 2 电源使能位  
1 = ADC 内核上电  
0 = ADC 内核断电
- bit 1        **C1PWR:** 专用 ADC 内核 1 电源使能位  
1 = ADC 内核上电  
0 = ADC 内核断电
- bit 0        **C0PWR:** 专用 ADC 内核 0 电源使能位  
1 = ADC 内核上电  
0 = ADC 内核断电

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-10: ADCON5H: ADC 控制寄存器 5 的高位字

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	WARMTIME3	WARMTIME2	WARMTIME1	WARMTIME0
bit 15				bit 8			

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
SHRCIE	—	—	—	C3CIE	C2CIE	C1CIE	C0CIE
bit 7				bit 0			

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-12     **未实现:** 读为 0

bit 11-8     **WARMTIME<3:0>:** ADC 专用内核 x 上电延时位  
 这些位用于通过所有 ADC 内核的内核源时钟周期 (TCORESRC) 数来确定上电延时。  
 1111 = 32768 个源时钟周期  
 1110 = 16384 个源时钟周期  
 1101 = 8192 个源时钟周期  
 1100 = 4096 个源时钟周期  
 1011 = 2048 个源时钟周期  
 1010 = 1024 个源时钟周期  
 1001 = 512 个源时钟周期  
 1000 = 256 个源时钟周期  
 0111 = 128 个源时钟周期  
 0110 = 64 个源时钟周期  
 0101 = 32 个源时钟周期  
 0100 = 16 个源时钟周期  
 00xx = 16 个源时钟周期

bit 7     **SHRCIE:** 共用 ADC 内核就绪通用中断允许位  
 1 = 当 ADC 内核上电且准备工作时产生通用中断  
 0 = 禁止在发生 ADC 内核就绪事件时产生通用中断

bit 6-4     **未实现:** 读为 0

bit 3     **C3CIE:** 专用 ADC 内核 3 就绪通用中断允许位  
 1 = 当 ADC 内核 3 上电且准备工作时产生通用中断  
 0 = 禁止在发生 ADC 内核 3 就绪事件时产生通用中断

bit 2     **C2CIE:** 专用 ADC 内核 2 就绪通用中断允许位  
 1 = 当 ADC 内核 2 上电且准备工作时产生通用中断  
 0 = 禁止在发生 ADC 内核 2 就绪事件时产生通用中断

bit 1     **C1CIE:** 专用 ADC 内核 1 就绪通用中断允许位  
 1 = 当 ADC 内核 1 上电且准备工作时产生通用中断  
 0 = 禁止在发生 ADC 内核 1 就绪事件时产生通用中断

bit 0     **C0CIE:** 专用 ADC 内核 0 就绪通用中断允许位  
 1 = 当 ADC 内核 0 上电且准备工作时产生通用中断  
 0 = 禁止在发生 ADC 内核 0 就绪事件时产生通用中断

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-11: ADCORExL: 专用 ADC 内核 x 控制寄存器的低位字 (x = 0 至 3)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SAMC<9:8>	
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SAMC<7:0>							
bit 7						bit 0	

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-10     **未实现:** 读为 0

bit 9-0       **SAMC<9:0>:** 专用 ADC 内核 x 转换延时选择位

这些位用于通过内核时钟周期 (TADCORE) 数来确定从触发事件到转换开始经过的时间。在这段时间内, ADC 内核 x 将继续采样。该功能通过 ADCON4L 寄存器中的 SAMCxEN 位使能。

1111111111 = 1025 个 TADCORE

- 
- 
- 

0000000001 = 3 个 TADCORE

0000000000 = 2 个 TADCORE

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-12: **ADCORExH**: 专用 ADC 内核 x 控制寄存器的高位字 (x = 0 至 3) <sup>(1)</sup>

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
—	—	—	EISEL2	EISEL1	EISEL0	RES1	RES0
bit 15						bit 8	

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
bit 7						bit 0	

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-13 **未实现:** 读为 0

bit 12-10 **EISEL<2:0>**: ADC 内核 x 提前中断时间选择位

- 111 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 8 个 TADCORE 时钟
- 110 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 7 个 TADCORE 时钟
- 101 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 6 个 TADCORE 时钟
- 100 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 5 个 TADCORE 时钟
- 011 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 4 个 TADCORE 时钟
- 010 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 3 个 TADCORE 时钟
- 001 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 2 个 TADCORE 时钟
- 000 = 设置提前中断, 产生中断的时间比数据就绪的时间提前 1 个 TADCORE 时钟

bit 9-8 **RES<1:0>**: ADC 内核 x 分辨率选择位

- 11 = 12 位分辨率
- 10 = 10 位分辨率
- 01 = 8 位分辨率
- 00 = 6 位分辨率

bit 7 **未实现:** 读为 0

bit 6-0 **ADCS<6:0>**: ADC 内核 x 输入时钟分频比位

这些位用于确定一个内核时钟周期 (TADCORE) 中的源时钟周期 (TCORESRC) 数。

- 11111111 = 254 个源时钟周期
- 
- 
- 
- 00000111 = 6 个源时钟周期
- 0000010 = 4 个源时钟周期
- 0000001 = 2 个源时钟周期
- 0000000 = 2 个源时钟周期

**注 1:** 对于 6 位 ADC 内核分辨率 (RES<1:0> = 00), EISEL<2:0> 位设置 100 至 111 无效, 请勿使用。对于 8 位 ADC 内核分辨率 (RES<1:0> = 01), EISEL<2:0> 位设置 110 和 111 无效, 请勿使用。

# dsPIC33EPXXGS70X/80X 系列

**寄存器 22-13: ADLVLTRGL: ADC 电平敏感触发控制寄存器的低位字**

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
LVLEN<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LVLEN<7:0>							
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0        **LVLEN<15:0>**: 相应模拟输入的电平触发使能位  
                   1 = 输入触发为电平敏感  
                   0 = 输入触发为边沿敏感

**寄存器 22-14: ADLVTRGH: ADC 电平敏感触发控制寄存器的高位字**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	LVLEN<21:16>					
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-6        **未实现**: 读为 0  
 bit 5-0        **LVLEN<21:16>**: 相应模拟输入的电平触发使能位  
                   1 = 输入触发为电平敏感  
                   0 = 输入触发为边沿敏感



# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-15: ADEIEL: ADC 提前中断允许寄存器的低位字

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EIEN<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EIEN<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0      **EIEN<15:0>**: 相应模拟输入的提前中断允许位  
 1 = 通道允许提前中断  
 0 = 通道禁止提前中断

## 寄存器 22-16: ADEIEH: ADC 提前中断允许寄存器的高位字

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	EIEN<21:16>					
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-6      **未实现**: 读为 0  
 bit 5-0      **EIEN<21:16>**: 相应模拟输入的提前中断允许位  
 1 = 通道允许提前中断  
 0 = 通道禁止提前中断

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-17: ADEISTATL: ADC 提前中断状态寄存器的低位字

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EISTAT<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EISTAT<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0        **EISTAT<15:0>**: 相应模拟输入的提前中断状态位  
 1 = 产生提前中断  
 0 = 上一次 ADCBUFx 读取后未产生提前中断

## 寄存器 22-18: ADEISTATH: ADC 提前中断状态寄存器的高位字

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	EISTAT<21:16>					
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-6        **未实现**: 读为 0  
 bit 5-0        **EISTAT<21:16>**: 相应模拟输入的提前中断状态位  
 1 = 产生提前中断  
 0 = 上一次 ADCBUFx 读取后未产生提前中断

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-19: ADMOD0L: ADC 输入模式控制寄存器 0 的低位字

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF7	SIGN7	DIFF6	SIGN6	DIFF5	SIGN5	DIFF4	SIGN4
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF3	SIGN3	DIFF2	SIGN2	DIFF1	SIGN1	DIFF0	SIGN0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-1                **DIFF<7:0>**: 相应模拟输入的差模位  
 (奇数)                1 = 通道为差分模式  
                           0 = 通道为单端模式

bit 14-0                **SIGN<7:0>**: 相应模拟输入的输出数据符号位  
 (偶数)                1 = 通道输出数据有符号  
                           0 = 通道输出数据无符号

## 寄存器 22-20: ADMOD0H: ADC 输入模式控制寄存器 0 的高位字

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF15	SIGN15	DIFF14	SIGN14	DIFF13	SIGN13	DIFF12	SIGN12
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF11	SIGN11	DIFF10	SIGN10	DIFF9	SIGN9	DIFF8	SIGN8
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-1                **DIFF<15:8>**: 相应模拟输入的差模位  
 (奇数)                1 = 通道为差分模式  
                           0 = 通道为单端模式

bit 14-0                **SIGN<15:8>**: 相应模拟输入的输出数据符号位  
 (偶数)                1 = 通道输出数据有符号  
                           0 = 通道输出数据无符号

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-21: ADMOD1L: ADC 输入模式控制寄存器 1 的低位字

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	DIFF21	SIGN21	DIFF20	SIGN20
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DIFF19	SIGN19	DIFF18	SIGN18	DIFF17	SIGN17	DIFF16	SIGN16
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-12

**未实现:** 读为 0

bit 11-1

**DIFF<21:16>:** 相应模拟输入的差模位

(奇数)

1 = 通道为差分模式

0 = 通道为单端模式

bit 10-0

**SIGN<21:16>:** 相应模拟输入的输出数据符号位

(偶数)

1 = 通道输出数据有符号

0 = 通道输出数据无符号

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-22: ADIEL: ADC 中断允许寄存器的低位字

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
IE<15:8>								
bit 15								bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
IE<7:0>								
bit 7								bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0      **IE<15:0>**: 通用中断允许位  
 1 = 相应通道允许通用中断和单独中断  
 0 = 相应通道禁止通用中断和单独中断

## 寄存器 22-23: ADIEH: ADC 中断允许寄存器的高位字

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	
—	—	—	—	—	—	—	—	
bit 15								bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
—	—	IE<21:16>						
bit 7								bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-6      **未实现**: 读为 0  
 bit 5-0      **IE<21:16>**: 通用中断允许位  
 1 = 相应通道允许通用中断和单独中断  
 0 = 相应通道禁止通用中断和单独中断

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-24: ADSTATL: ADC 数据就绪状态寄存器的低位字

R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
AN<15:8>RDY							
bit 15							bit 8

R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
AN<7:0>RDY							
bit 7							bit 0

**图注:** U = 未实现位, 读为 0  
R = 可读位 W = 可写位 HSC = 硬件置 1/ 清零位  
-n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

bit 15-0 **AN<15:0>RDY:** 相应模拟输入的通用中断允许位  
1 = 相应 ADCBUFx 寄存器中的通道转换结果已就绪  
0 = 通道转换结果未就绪

## 寄存器 22-25: ADSTATH: ADC 数据就绪状态寄存器的高位字

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	AN<21:16>RDY					
bit 7							bit 0

**图注:** U = 未实现位, 读为 0  
R = 可读位 W = 可写位 HSC = 硬件置 1/ 清零位  
-n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

bit 15-6 **未实现:** 读为 0  
bit 5-0 **AN<21:16>RDY:** 相应模拟输入的通用中断允许位  
1 = 相应 ADCBUFx 寄存器中的通道转换结果已就绪  
0 = 通道转换结果未就绪

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-26: **ADTRIGxL: ADC 通道触发 x 选择寄存器的低位字 (x = 0 至 5)**

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x+1)<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x)<4:0>				
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-13 **未实现:** 读为 0

bit 12-8 **TRGSRC(4x+1)<4:0>:** 相应模拟输入的触发源选择位

11111 = ADTRG31

11110 = PTG 触发输出 12

11101 = PWM 发生器 6 限流触发

11100 = PWM 发生器 5 限流触发

11011 = PWM 发生器 4 限流触发

11010 = PWM 发生器 3 限流触发

11001 = PWM 发生器 2 限流触发

11000 = PWM 发生器 1 限流触发

10111 = 输出比较 2 触发

10110 = 输出比较 1 触发

10101 = CLC2 输出

10100 = PWM 发生器 6 辅助触发

10011 = PWM 发生器 5 辅助触发

10010 = PWM 发生器 4 辅助触发

10001 = PWM 发生器 3 辅助触发

10000 = PWM 发生器 2 辅助触发

01111 = PWM 发生器 1 辅助触发

01110 = PWM 辅助特殊事件触发

01101 = Timer2 周期匹配

01100 = Timer1 周期匹配

01011 = CLC1 输出

01010 = PWM 发生器 6 主触发

01001 = PWM 发生器 5 主触发

01000 = PWM 发生器 4 主触发

00111 = PWM 发生器 3 主触发

00110 = PWM 发生器 2 主触发

00101 = PWM 发生器 1 主触发

00100 = PWM 特殊事件触发

00011 = 保留

00010 = 电平软件触发

00001 = 通用软件触发

00000 = 不使能触发

bit 7-5 **未实现:** 读为 0

# dsPIC33EPXXGS70X/80X 系列

---

寄存器 22-26: ADTRIGxL: ADC 通道触发 x 选择寄存器的低位字 (x = 0 至 5) (续)

bit 4-0 TRGSRC(4x)<4:0>: 相应模拟输入的触发源选择位

11111 = ADTRG31  
11110 = PTG 触发输出 30  
11101 = PWM 发生器 6 限流触发  
11100 = PWM 发生器 5 限流触发  
11011 = PWM 发生器 4 限流触发  
11010 = PWM 发生器 3 限流触发  
11001 = PWM 发生器 2 限流触发  
11000 = PWM 发生器 1 限流触发  
10111 = 输出比较 2 触发  
10110 = 输出比较 1 触发  
10101 = CLC2 输出  
10100 = PWM 发生器 6 辅助触发  
10011 = PWM 发生器 5 辅助触发  
10010 = PWM 发生器 4 辅助触发  
10001 = PWM 发生器 3 辅助触发  
10000 = PWM 发生器 2 辅助触发  
01111 = PWM 发生器 1 辅助触发  
01110 = PWM 辅助特殊事件触发  
01101 = Timer2 周期匹配  
01100 = Timer1 周期匹配  
01011 = CLC1 输出  
01010 = PWM 发生器 6 主触发  
01001 = PWM 发生器 5 主触发  
01000 = PWM 发生器 4 主触发  
00111 = PWM 发生器 3 主触发  
00110 = PWM 发生器 2 主触发  
00101 = PWM 发生器 1 主触发  
00100 = PWM 特殊事件触发  
00011 = 保留  
00010 = 电平软件触发  
00001 = 通用软件触发  
00000 = 不使能触发



# dsPIC33EPXXGS70X/80X 系列

寄存器 22-27: ADTRIGxH: ADC 通道触发 x 选择寄存器的高位字 (x = 0 至 5)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x+3)<4:0>				
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TRGSRC(4x+2)<4:0>				
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-13 未实现: 读为 0

bit 12-8 TRGSRC(4x+3)<4:0>: 相应模拟输入的触发源选择位

- 11111 = ADTRG31
- 11110 = PTG 触发输出 30
- 11101 = PWM 发生器 6 限流触发
- 11100 = PWM 发生器 5 限流触发
- 11011 = PWM 发生器 4 限流触发
- 11010 = PWM 发生器 3 限流触发
- 11001 = PWM 发生器 2 限流触发
- 11000 = PWM 发生器 1 限流触发
- 10111 = 输出比较 2 触发
- 10110 = 输出比较 1 触发
- 10101 = CLC2 输出
- 10100 = PWM 发生器 6 辅助触发
- 10011 = PWM 发生器 5 辅助触发
- 10010 = PWM 发生器 4 辅助触发
- 10001 = PWM 发生器 3 辅助触发
- 10000 = PWM 发生器 2 辅助触发
- 01111 = PWM 发生器 1 辅助触发
- 01110 = PWM 辅助特殊事件触发
- 01101 = Timer2 周期匹配
- 01100 = Timer1 周期匹配
- 01011 = CLC1 输出
- 01010 = PWM 发生器 6 主触发
- 01001 = PWM 发生器 5 主触发
- 01000 = PWM 发生器 4 主触发
- 00111 = PWM 发生器 3 主触发
- 00110 = PWM 发生器 2 主触发
- 00101 = PWM 发生器 1 主触发
- 00100 = PWM 特殊事件触发
- 00011 = 保留
- 00010 = 电平软件触发
- 00001 = 通用软件触发
- 00000 = 不使能触发

bit 7-5 未实现: 读为 0

# dsPIC33EPXXGS70X/80X 系列

---

寄存器 22-27: ADTRIGxH: ADC 通道触发 x 选择寄存器的高位字 (x = 0 至 5) (续)

bit 4-0 TRGSRC(4x+2)<4:0>: 相应模拟输入的触发源选择位

11111 = ADTRG31  
11110 = PTG 触发输出 30  
11101 = PWM 发生器 6 限流触发  
11100 = PWM 发生器 5 限流触发  
11011 = PWM 发生器 4 限流触发  
11010 = PWM 发生器 3 限流触发  
11001 = PWM 发生器 2 限流触发  
11000 = PWM 发生器 1 限流触发  
10111 = 输出比较 2 触发  
10110 = 输出比较 1 触发  
10101 = CLC2 输出  
10100 = PWM 发生器 6 辅助触发  
10011 = PWM 发生器 5 辅助触发  
10010 = PWM 发生器 4 辅助触发  
10001 = PWM 发生器 3 辅助触发  
10000 = PWM 发生器 2 辅助触发  
01111 = PWM 发生器 1 辅助触发  
01110 = PWM 辅助特殊事件触发  
01101 = Timer2 周期匹配  
01100 = Timer1 周期匹配  
01011 = CLC1 输出  
01010 = PWM 发生器 6 主触发  
01001 = PWM 发生器 5 主触发  
01000 = PWM 发生器 4 主触发  
00111 = PWM 发生器 3 主触发  
00110 = PWM 发生器 2 主触发  
00101 = PWM 发生器 1 主触发  
00100 = PWM 特殊事件触发  
00011 = 保留  
00010 = 电平软件触发  
00001 = 通用软件触发  
00000 = 不使能触发

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-28: ADCAL0L: ADC 校准寄存器 0 的低位字

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL1RDY	—	—	—	CAL1SKIP	CAL1DIFF	CAL1EN	CAL1RUN
bit 15							bit 8

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL0RDY	—	—	—	CAL0SKIP	CAL0DIFF	CAL0EN	CAL0RUN
bit 7							bit 0

<b>图注:</b>	U = 未实现位, 读为 0		
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15      **CAL1RDY:** 专用 ADC 内核 1 校准状态标志位  
1 = 专用 ADC 内核 1 已完成校准  
0 = 专用 ADC 内核 1 正在校准
- bit 14-12    **未实现:** 读为 0
- bit 11      **CAL1SKIP:** 专用 ADC 内核 1 校准旁路位  
1 = 上电后, 将不会校准专用 ADC 内核 1  
0 = 上电后, 将校准专用 ADC 内核 1
- bit 10      **CAL1DIFF:** 专用 ADC 内核 1 差模校准位  
1 = 专用 ADC 内核 1 将在差分输入模式下校准  
0 = 专用 ADC 内核 1 将在单端输入模式下校准
- bit 9        **CAL1EN:** 专用 ADC 内核 1 校准使能位  
1 = 专用 ADC 内核 1 校准位 (CALxRDY、CALxSKIP、CALxDIFF 和 CALxRUN) 可通过软件访问  
0 = 专用 ADC 内核 1 校准位被禁止
- bit 8        **CAL1RUN:** 专用 ADC 内核 1 校准启动位  
1 = 如果该位由软件置 1, 专用 ADC 内核 1 校准周期将启动; 该位由硬件自动清零  
0 = 软件可启动下一个校准周期
- bit 7        **CAL0RDY:** 专用 ADC 内核 0 校准状态标志位  
1 = 专用 ADC 内核 0 已完成校准  
0 = 专用 ADC 内核 0 正在校准
- bit 6-4      **未实现:** 读为 0
- bit 3        **CAL0SKIP:** 专用 ADC 内核 0 校准旁路位  
1 = 上电后, 将不会校准专用 ADC 内核 0  
0 = 上电后, 将校准专用 ADC 内核 0
- bit 2        **CAL0DIFF:** 专用 ADC 内核 0 差模校准位  
1 = 专用 ADC 内核 0 将在差分输入模式下校准  
0 = 专用 ADC 内核 0 将在单端输入模式下校准
- bit 1        **CAL0EN:** 专用 ADC 内核 0 校准使能位  
1 = 专用 ADC 内核 0 校准位 (CALxRDY、CALxSKIP、CALxDIFF 和 CALxRUN) 可通过软件访问  
0 = 专用 ADC 内核 0 校准位被禁止
- bit 0        **CAL0RUN:** 专用 ADC 内核 0 校准启动位  
1 = 如果该位由软件置 1, 专用 ADC 内核 0 校准周期将启动; 该位由硬件自动清零  
0 = 软件可启动下一个校准周期

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-29: ADCAL0H: ADC 校准寄存器 0 的高位字

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL3RDY	—	—	—	CAL3SKIP	CAL3DIFF	CAL3EN	CAL3RUN
bit 15						bit 8	

R-0, HSC	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CAL2RDY	—	—	—	CAL2SKIP	CAL2DIFF	CAL2EN	CAL2RUN
bit 7						bit 0	

<b>图注:</b>	U = 未实现位, 读为 0		
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15      **CAL3RDY:** 专用 ADC 内核 3 校准状态标志位  
1 = 专用 ADC 内核 3 已完成校准  
0 = 专用 ADC 内核 3 正在校准
- bit 14-12    **未实现:** 读为 0
- bit 11      **CAL3SKIP:** 专用 ADC 内核 3 校准旁路位  
1 = 上电后, 将不会校准专用 ADC 内核 3  
0 = 上电后, 将校准专用 ADC 内核 3
- bit 10      **CAL3DIFF:** 专用 ADC 内核 3 差模校准位  
1 = 专用 ADC 内核 3 将在差分输入模式下校准  
0 = 专用 ADC 内核 3 将在单端输入模式下校准
- bit 9        **CAL3EN:** 专用 ADC 内核 3 校准使能位  
1 = 专用 ADC 内核 3 校准位 (CALxRDY、CALxSKIP、CALxDIFF 和 CALxRUN) 可通过软件访问  
0 = 专用 ADC 内核 3 校准位被禁止
- bit 8        **CAL3RUN:** 专用 ADC 内核 3 校准启动位  
1 = 如果该位由软件置 1, 专用 ADC 内核 3 校准周期将启动; 该位由硬件自动清零  
0 = 软件可启动下一个校准周期
- bit 7        **CAL2RDY:** 专用 ADC 内核 2 校准状态标志位  
1 = 专用 ADC 内核 2 已完成校准  
0 = 专用 ADC 内核 2 正在校准
- bit 6-4      **未实现:** 读为 0
- bit 3        **CAL2SKIP:** 专用 ADC 内核 2 校准旁路位  
1 = 上电后, 将不会校准专用 ADC 内核 2  
0 = 上电后, 将校准专用 ADC 内核 2
- bit 2        **CAL2DIFF:** 专用 ADC 内核 2 差模校准位  
1 = 专用 ADC 内核 2 将在差分输入模式下校准  
0 = 专用 ADC 内核 2 将在单端输入模式下校准
- bit 1        **CAL2EN:** 专用 ADC 内核 2 校准使能位  
1 = 专用 ADC 内核 2 校准位 (CALxRDY、CALxSKIP、CALxDIFF 和 CALxRUN) 可通过软件访问  
0 = 专用 ADC 内核 2 校准位被禁止
- bit 0        **CAL2RUN:** 专用 ADC 内核 2 校准启动位  
1 = 如果该位由软件置 1, 专用 ADC 内核 2 校准周期将启动; 该位由硬件自动清零  
0 = 软件可启动下一个校准周期

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 22-30: ADCAL1H: ADC 校准寄存器 1 的高位字

R/W-0, HS	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CSHRRDY	—	—	—	CSHRSKIP	CSHRDIFF	CSHREN	CSHRRUN
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

<b>图注:</b>	HS = 硬件置 1 位
R = 可读位	W = 可写位
-n = POR 时的值	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

- bit 15      **CSHRRDY:** 共用 ADC 内核校准状态标志位  
1 = 共用 ADC 内核已完成校准  
0 = 共用 ADC 内核正在校准
- bit 14-12    **未实现:** 读为 0
- bit 11      **CSHRSKIP:** 共用 ADC 内核校准旁路位  
1 = 上电后, 将不会校准共用 ADC 内核  
0 = 上电后, 将校准共用 ADC 内核
- bit 10      **CSHRDIFF:** 共用 ADC 内核差模校准位  
1 = 共用 ADC 内核将在差分输入模式下校准  
0 = 共用 ADC 内核将在单端输入模式下校准
- bit 9        **CSHREN:** 共用 ADC 内核校准使能位  
1 = 共用 ADC 内核校准位 (CSHRRDY、CSHRSKIP、CSHRDIFF 和 CSHRRUN) 可通过软件访问  
0 = 共用 ADC 内核校准位被禁止
- bit 8        **CSHRRUN:** 共用 ADC 内核校准启动位  
1 = 如果该位由软件置 1, 共用 ADC 内核校准周期将启动; 该位由硬件自动清零  
0 = 软件可启动下一个校准周期
- bit 7-0     **未实现:** 读为 0

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-31: ADCMPxCON: ADC 数字比较器 x 控制寄存器 (x = 0 或 1)

U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC	R-0, HSC
—	—	—	CHNL4	CHNL3	CHNL2	CHNL1	CHNL0
bit 15							bit 8

R/W-0	R/W-0	R-0, HC, HS	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPEN	IE	STAT	BTWN	HIHI	HILO	LOHI	LOLO
bit 7							bit 0

<b>图注:</b>	HC = 硬件清零位	U = 未实现位, 读为 0
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位
-n = POR 时的值	1 = 置 1	0 = 清零
		HS = 硬件置 1 位

bit 15-13 **未实现:** 读为 0

bit 12-8 **CHNL<4:0>:** 输入通道编号位  
 如果比较器检测到某个通道发生事件, 则该通道编号将写入这些位。  
 11111 = 保留  
 •  
 •  
 10110 = 保留  
 10101 = AN21  
 10100 = AN20  
 •  
 •  
 00001 = AN1  
 00000 = AN0

bit 7 **CMPEN:** 比较器使能位  
 1 = 使能比较器  
 0 = 禁止比较器并清零 STAT 状态位

bit 6 **IE:** 比较器通用 ADC 中断允许位  
 1 = 如果比较器检测到比较事件, 将产生通用 ADC 中断  
 0 = 比较器将不产生通用 ADC 中断

bit 5 **STAT:** 比较器事件状态位  
 当从 CHNL<4:0> 位读取通道编号时, 该位由硬件清零。  
 1 = 上一次读取 CHNL<4:0> 位后已检测到比较事件  
 0 = 上一次读取 CHNL<4:0> 位后尚未检测到比较事件

bit 4 **BTWN:** 介于下限 / 上限间的比较器事件位  
 1 = 当  $ADCMPxLO \leq ADCBUFx < ADCMPxHI$  时, 产生数字比较器事件  
 0 = 当  $ADCMPxLO \leq ADCBUFx < ADCMPxHI$  时, 不产生数字比较器事件

bit 3 **HIHI:** 高于上限的比较器事件位  
 1 = 当  $ADCBUFx \geq ADCMPxHI$  时, 产生数字比较器事件  
 0 = 当  $ADCBUFx \geq ADCMPxHI$  时, 不产生数字比较器事件

bit 2 **HILO:** 低于上限的比较器事件位  
 1 = 当  $ADCBUFx < ADCMPxHI$  时, 产生数字比较器事件  
 0 = 当  $ADCBUFx < ADCMPxHI$  时, 不产生数字比较器事件

bit 1 **LOHI:** 高于下限的比较器事件位  
 1 = 当  $ADCBUFx \geq ADCMPxLO$  时, 产生数字比较器事件  
 0 = 当  $ADCBUFx \geq ADCMPxLO$  时, 不产生数字比较器事件

bit 0 **LOLO:** 低于下限的比较器事件位  
 1 = 当  $ADCBUFx < ADCMPxLO$  时, 产生数字比较器事件  
 0 = 当  $ADCBUFx < ADCMPxLO$  时, 不产生数字比较器事件

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-32: **ADCMPxENL**: ADC 数字比较器 x 通道使能寄存器的低位字 (x = 0 或 1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPEN<15:8>							
bit 15							bit 8

R/W/0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPEN<7:0>							
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-0 **CMPEN<15:0>**: 相应输入通道的比较器使能位  
 1 = 比较器使用相应通道的转换结果  
 0 = 比较器不使用相应通道的转换结果

寄存器 22-33: **ADCMPxENH**: ADC 数字比较器 x 通道使能寄存器的高位字 (x = 0 或 1)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CMPEN<21:16>					
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-6 **未实现**: 读为 0  
 bit 5-0 **CMPEN<21:16>**: 相应输入通道的比较器使能位  
 1 = 比较器使用相应通道的转换结果  
 0 = 比较器不使用相应通道的转换结果

# dsPIC33EPXXGS70X/80X 系列

寄存器 22-34: **ADFLxCON: ADC 数字滤波器 x 控制寄存器 (x = 0 或 1)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0, HSC
FLEN	MODE1	MODE0	OVRSAM2	OVRSAM1	OVRSAM0	IE	RDY
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FLCHSEL4	FLCHSEL3	FLCHSEL2	FLCHSEL1	FLCHSEL0
bit 7							bit 0

<b>图注:</b>	U = 未实现位, 读为 0		
R = 可读位	W = 可写位	HSC = 硬件置 1/ 清零位	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15      **FLEN:** 滤波器使能位  
 1 = 使能滤波器  
 0 = 禁止滤波器并清零 RDY 位
- bit 14-13    **MODE<1:0>:** 滤波器模式位  
 11 = 平均模式  
 10 = 保留  
 01 = 保留  
 00 = 过采样模式
- bit 12-10    **OVRSAM<2:0>:** 滤波器平均 / 过采样率位  
如果 MODE<1:0> = 00:  
 111 = 128x (ADFLxDAT 寄存器中的 16 位结果采用 12.4 格式)  
 110 = 32x (ADFLxDAT 寄存器中的 15 位结果采用 12.3 格式)  
 101 = 8x (ADFLxDAT 寄存器中的 14 位结果采用 12.2 格式)  
 100 = 2x (ADFLxDAT 寄存器中的 13 位结果采用 12.1 格式)  
 011 = 256x (ADFLxDAT 寄存器中的 16 位结果采用 12.4 格式)  
 010 = 64x (ADFLxDAT 寄存器中的 15 位结果采用 12.3 格式)  
 001 = 16x (ADFLxDAT 寄存器中的 14 位结果采用 12.2 格式)  
 000 = 4x (ADFLxDAT 寄存器中的 13 位结果采用 12.1 格式)  
如果 MODE<1:0> = 11 (所有情况下 ADFLxDAT 寄存器中的 12 位结果):  
 111 = 256x  
 110 = 128x  
 101 = 64x  
 100 = 32x  
 011 = 16x  
 010 = 8x  
 001 = 4x  
 000 = 2x
- bit 9        **IE:** 滤波器通用 ADC 中断允许位  
 1 = 当滤波器结果就绪时, 产生通用 ADC 中断  
 0 = 滤波器将不产生通用 ADC 中断
- bit 8        **RDY:** 过采样滤波器数据就绪标志位  
 当从 ADFLxDAT 寄存器读取结果时, 该位由硬件清零。  
 1 = ADFLxDAT 寄存器中的数据已就绪  
 0 = ADFLxDAT 寄存器已被读取, ADFLxDAT 寄存器中的新数据未就绪
- bit 7-5      **未实现:** 读为 0



# dsPIC33EPXXGS70X/80X 系列

---

寄存器 22-34: ADFLxCON: ADC 数字滤波器 x 控制寄存器 (x = 0 或 1) (续)

bit 4-0      **FLCHSEL<4:0>**: 过采样滤波器输入通道选择位

11111 = 保留

•

•

•

10110 = 保留

10101 = AN21

10100 = AN20

•

•

•

00001 = AN1

00000 = AN0

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 23.0 控制器局域网 (CAN) 模块 (仅限 dsPIC33EPXXGS80X 器件)

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息, 请参见《dsPIC33/PIC24 系列参考手册》的“增强型控制器局域网 (ECAN™)” (DS70353), 该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息, 请参见本数据手册中的第 4.0 节“存储器构成”。

### 23.1 概述

控制器局域网 (Controller Area Network, CAN) 模块是一个串行接口, 用于同其他 CAN 模块或单片机进行通信。此接口 / 协议是针对允许在噪声环境下通信而设计的。dsPIC33EPXXGS80X 器件包含两个 CAN 模块。

CAN 模块是一个通信控制器, 实现了 BOSCH CAN 规范中定义的 CAN 2.0 A/B 协议。该模块支持 CAN 1.2、CAN 2.0A、CAN 2.0B Passive 和 CAN 2.0B Active 版本的协议。该模块实现了一种完整的 CAN 系统。但是本数据手册不讨论 CAN 规范。读者可参阅 BOSCH CAN 规范了解更多详细信息。

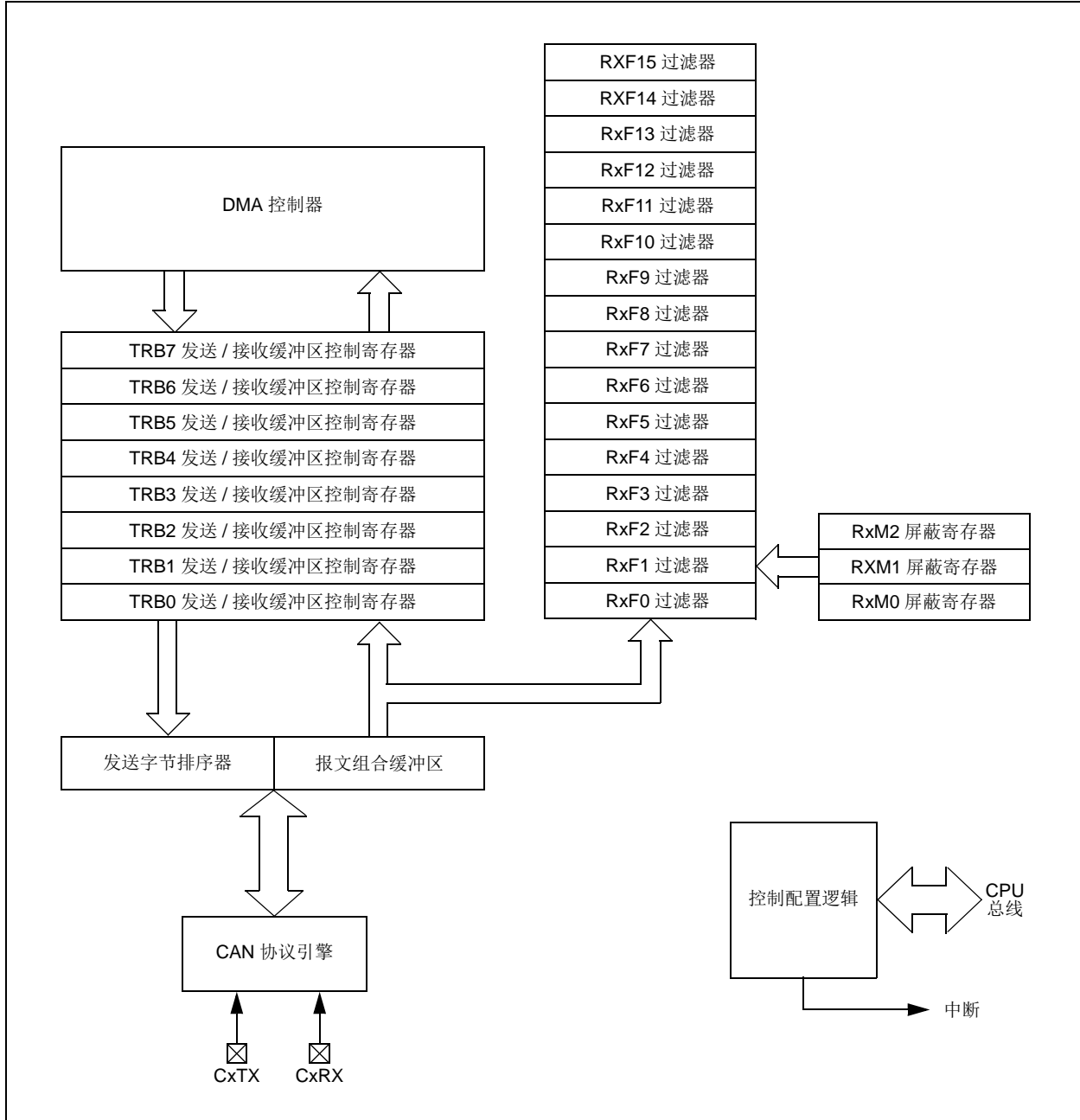
CAN 模块具有以下特性:

- 实现了 CAN 协议 CAN 1.2、CAN 2.0A 和 CAN 2.0B
- 支持标准数据帧和扩展数据帧
- 0-8 字节数据长度
- 最高 1 Mbps 的可编程比特率
- 自动响应远程发送请求
- 最多 8 个发送缓冲区, 可由应用程序指定优先级和中止功能 (每个缓冲区最多包含 8 个字节的数据)
- 最多 32 个接收缓冲区 (每个缓冲区最多包含 8 个字节的数据)
- 最多 16 个完全 (标准 / 扩展标识符) 的接收过滤器
- 3 个完全接收过滤屏蔽寄存器
- DeviceNet™ 寻址支持
- 集成了低通滤波器的可编程唤醒功能
- 支持自检操作的可编程环回模式
- 通过中断功能在出现任何 CAN 接收器和发送器错误状态时发出中断信号
- 可编程时钟源
- 与输入捕捉模块 2 (IC2) 的可编程连接, 以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 / 控制模块组成。CAN 协议引擎处理在 CAN 总线上接收和发送报文的所有功能。通过首先装载相应的数据寄存器发送报文。可通过读取相应的寄存器检测状态和错误。将对在 CAN 总线上检测到的任何报文进行错误检测, 并随后将其与过滤器进行比较以判断是否要将其接收并存储到接收寄存器之一中。

# dsPIC33EPXXGS70X/80X 系列

图 23-1: CANx 模块框图



## 23.2 工作模式

用户可以选择 CANx 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
- 监听所有报文模式
- 环回模式

可通过设置 REQOP<2:0> 位 (CxCTRL1<10:8>) 请求所需模式。通过监视 OPMODE<2:0> 位 (CxCTRL1<7:5>) 可以确认进入的模式。通常在总线上检测到至少 11 个连续隐性位表明总线空闲时才允许改变模式，在此之前不会改变模块的工作模式和 OPMODEx 位。

# dsPIC33EPXXGS70X/80X 系列

## 23.3 CAN 控制寄存器

寄存器 23-1: CxCTRL1: CANx 控制寄存器 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
—	—	CSIDL	ABAT	CANCKS	REQOP2	REQOP1	REQOP0
bit 15							bit 8
R-1	R-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
OPMODE2	OPMODE1	OPMODE0	—	CANCAP	—	—	WIN
bit 7							bit 0

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-14     **未实现:** 读为 0
- bit 13       **CSIDL:** CANx 空闲模式停止位  
1 = 当器件进入空闲模式时, 模块停止工作  
0 = 在空闲模式下模块继续工作
- bit 12       **ABAT:** 中止所有等待发送的位  
1 = 通知所有发送缓冲区中止发送  
0 = 模块将在所有发送中止时清零该位
- bit 11       **CANCKS:** CANx 模块时钟 (FCAN) 源选择位  
1 = FCAN 等于 2 \* FP  
0 = FCAN 等于 FP
- bit 10-8     **REQOP<2:0>:** 请求工作模式位  
111 = 设置为监听所有报文模式  
110 = 保留  
101 = 保留  
100 = 设置为配置模式  
011 = 设置为监听模式  
010 = 设置为环回模式  
001 = 设置为禁止模式  
000 = 设置为正常工作模式
- bit 7-5      **OPMODE<2:0>:** 工作模式位  
111 = 模块处于监听所有报文模式  
110 = 保留  
101 = 保留  
100 = 模块处于配置模式  
011 = 模块处于监听模式  
010 = 模块处于环回模式  
001 = 模块处于禁止模式  
000 = 模块处于正常工作模式
- bit 4        **未实现:** 读为 0
- bit 3        **CANCAP:** CANx 报文接收定时器捕捉事件使能位  
1 = 使能基于 CAN 报文接收的输入捕捉  
0 = 禁止 CAN 捕捉
- bit 2-1      **未实现:** 读为 0
- bit 0        **WIN:** SFR 映射窗口选择位  
1 = 使用过滤器窗口  
0 = 使用缓冲区窗口

# dsPIC33EPXXGS70X/80X 系列

寄存器 23-2: CxCTRL2: CANx 控制寄存器 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	DNCNT<4:0>				
bit 7						bit 0	

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-5      **未实现:** 读为 0  
 bit 4-0      **DNCNT<4:0>:** DeviceNet™ 过滤器位编号位  
               10010-11111 = 无效的选择  
               10001 = 最多可将数据字节 3 的 bit 6 与 EID<17> 作比较  
               •  
               •  
               •  
               00001 = 最多可将数据字节 1 的 bit 7 与 EID<0> 作比较  
               00000 = 不比较数据字节

# dsPIC33EPXXGS70X/80X 系列

寄存器 23-3: CxVEC: CANx 中断编码寄存器

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0
bit 15							bit 8

U-0	R-1	R-0	R-0	R-0	R-0	R-0	R-0
—	ICODE6	ICODE5	ICODE4	ICODE3	ICODE2	ICODE1	ICODE0
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-13     **未实现:** 读为 0

bit 12-8     **FILHIT<4:0>:** 选中过滤器的编号位

10000-11111 = 保留

01111 = 过滤器 15

•

•

•

00001 = 过滤器 1

00000 = 过滤器 0

bit 7         **未实现:** 读为 0

bit 6-0     **ICODE<6:0>:** 中断标志编码位

1000101-1111111 = 保留

1000100 = FIFO 几乎满中断

1000011 = 接收器溢出中断

1000010 = 唤醒中断

1000001 = 错误中断

1000000 = 无中断

•

•

•

0010000-0111111 = 保留

0001111 = RB15 缓冲区中断

•

•

•

0001001 = RB9 缓冲区中断

0001000 = RB8 缓冲区中断

0000111 = TRB7 缓冲区中断

0000110 = TRB6 缓冲区中断

0000101 = TRB5 缓冲区中断

0000100 = TRB4 缓冲区中断

0000011 = TRB3 缓冲区中断

0000010 = TRB2 缓冲区中断

0000001 = TRB1 缓冲区中断

0000000 = TRB0 缓冲区中断

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-4: CxFCTRL: CANx FIFO 控制寄存器

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
DMABS2	DMABS1	DMABS0	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FSA4	FSA3	FSA2	FSA1	FSA0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-13      **DMABS<2:0>**: DMA 缓冲区大小位

111 = 保留  
 110 = RAM 中有 32 个缓冲区  
 101 = RAM 中有 24 个缓冲区  
 100 = RAM 中有 16 个缓冲区  
 011 = RAM 中有 12 个缓冲区  
 010 = RAM 中有 8 个缓冲区  
 001 = RAM 中有 6 个缓冲区  
 000 = RAM 中有 4 个缓冲区

bit 12-5      未实现: 读为 0

bit 4-0      **FSA<4:0>**: FIFO 区域从哪个缓冲区开始位

11111 = 接收缓冲区 RB31  
 11110 = 接收缓冲区 RB30  
 •  
 •  
 •  
 00001 = 发送 / 接收缓冲区 TRB1  
 00000 = 发送 / 接收缓冲区 TRB0



# dsPIC33EPXXGS70X/80X 系列

寄存器 23-5: CxFIFO: CANx FIFO 状态寄存器

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FBP5	FBP4	FBP3	FBP2	FBP1	FBP0
bit 15							bit 8

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FNRB5	FNRB4	FNRB3	FNRB2	FNRB1	FNRB0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-14      **未实现:** 读为 0

bit 13-8      **FBP<5:0>:** FIFO 缓冲区指针位

- 011111 = RB31 缓冲区
- 011110 = RB30 缓冲区
- 
- 
- 
- 000001 = TRB1 缓冲区
- 000000 = TRB0 缓冲区

bit 7-6      **未实现:** 读为 0

bit 5-0      **FNRB<5:0>:** FIFO 下一个读缓冲区指针位

- 011111 = RB31 缓冲区
- 011110 = RB30 缓冲区
- 
- 
- 
- 000001 = TRB1 缓冲区
- 000000 = TRB0 缓冲区

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-6: CxINTF: CANx 中断标志寄存器

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN
bit 15							bit 8

R/C-0	R/C-0	R/C-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0
IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF
bit 7							bit 0

<b>图注:</b>	C = 可写位, 但只能写入 0 来清零该位
R = 可读位	W = 可写位
-n = POR 时的值	U = 未实现位, 读为 0
	1 = 置 1
	0 = 清零
	x = 未知

- bit 15-14 **未实现:** 读为 0
- bit 13 **TXBO:** 发送器处于错误状态总线关闭位  
1 = 发送器处于总线关闭状态  
0 = 发送器不处于总线关闭状态
- bit 12 **TXBP:** 发送器处于错误状态总线被动位  
1 = 发送器处于总线被动状态  
0 = 发送器不处于总线被动状态
- bit 11 **RXBP:** 接收器处于错误状态总线被动位  
1 = 接收器处于总线被动状态  
0 = 接收器不处于总线被动状态
- bit 10 **TXWAR:** 发送器处于错误状态警告位  
1 = 发送器处于错误警告状态  
0 = 发送器不处于错误警告状态
- bit 9 **RXWAR:** 接收器处于错误状态警告位  
1 = 接收器处于错误警告状态  
0 = 接收器不处于错误警告状态
- bit 8 **EWARN:** 发送器或接收器处于错误状态警告位  
1 = 发送器或接收器处于错误警告状态  
0 = 发送器或接收器不处于错误警告状态
- bit 7 **IVRIF:** 收到无效报文中断标志位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 6 **WAKIF:** 总线唤醒活动中断标志位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 5 **ERRIF:** 错误中断标志位 (CxINTF<13:8> 寄存器中的多个中断源)  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 4 **未实现:** 读为 0
- bit 3 **FIFOIF:** FIFO 几乎满中断标志位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 2 **RBOVIF:** 接收缓冲区溢出中断标志位  
1 = 产生了中断请求  
0 = 未产生中断请求

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-6: CxINTF: CANx 中断标志寄存器 (续)

- bit 1      **RBIF:** 接收缓冲区中断标志位  
 1 = 产生了中断请求  
 0 = 未产生中断请求
- bit 0      **TBIF:** 发送缓冲区中断标志位  
 1 = 产生了中断请求  
 0 = 未产生中断请求

## 寄存器 23-7: CxINTE: CANx 中断允许寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

- bit 15-8      **未实现:** 读为 0
- bit 7      **IVRIE:** 收到无效报文中断允许位  
 1 = 允许中断请求  
 0 = 禁止中断请求
- bit 6      **WAKIE:** 总线唤醒活动中断允许位  
 1 = 允许中断请求  
 0 = 禁止中断请求
- bit 5      **ERRIE:** 错误中断允许位  
 1 = 允许中断请求  
 0 = 禁止中断请求
- bit 4      **未实现:** 读为 0
- bit 3      **FIFOIE:** FIFO 几乎满中断允许位  
 1 = 允许中断请求  
 0 = 禁止中断请求
- bit 2      **RBOVIE:** 接收缓冲区溢出中断允许位  
 1 = 允许中断请求  
 0 = 禁止中断请求
- bit 1      **RBIE:** 接收缓冲区中断允许位  
 1 = 允许中断请求  
 0 = 禁止中断请求
- bit 0      **TBIE:** 发送缓冲区中断允许位  
 1 = 允许中断请求  
 0 = 禁止中断请求

# dsPIC33EPXXGS70X/80X 系列

**寄存器 23-8: CxEC: CANx 发送 / 接收错误计数寄存器**

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TERRCNT7	TERRCNT6	TERRCNT5	TERRCNT4	TERRCNT3	TERRCNT2	TERRCNT1	TERRCNT0
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RERRCNT7	RERRCNT6	RERRCNT5	RERRCNT4	RERRCNT3	RERRCNT2	RERRCNT1	RERRCNT0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            **TERRCNT<7:0>**: 发送错误计数位

bit 7-0             **RERRCNT<7:0>**: 接收错误计数位

**寄存器 23-9: CxCFG1: CANx 波特率配置寄存器 1**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            **未实现**: 读为 0

bit 7-6             **SJW<1:0>**: 同步跳转宽度位

- 11 = 长度为 4 x T<sub>Q</sub>
- 10 = 长度为 3 x T<sub>Q</sub>
- 01 = 长度为 2 x T<sub>Q</sub>
- 00 = 长度为 1 x T<sub>Q</sub>

bit 5-0             **BRP<5:0>**: 波特率预分频比位

- 11 1111 = T<sub>Q</sub> = 2 x 64 x 1/FCAN
- 
- 
- 
- 00 0010 = T<sub>Q</sub> = 2 x 3 x 1/FCAN
- 00 0001 = T<sub>Q</sub> = 2 x 2 x 1/FCAN
- 00 0000 = T<sub>Q</sub> = 2 x 1 x 1/FCAN

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-10: CxCFG2: CANx 波特率配置寄存器 2

U-0	R/W-x	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	WAKFIL	—	—	—	SEG2PH2	SEG2PH1	SEG2PH0
bit 15					bit 8		

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SEG2PHTS	SAM	SEG1PH2	SEG1PH1	SEG1PH0	PRSEG2	PRSEG1	PRSEG0
bit 7					bit 0		

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15      **未实现:** 读为 0

bit 14      **WAKFIL:** 选择是否使用 CAN 总线线路滤波器唤醒的位

1 = 使用 CAN 总线线路滤波器来唤醒

0 = 不使用 CAN 总线线路滤波器来唤醒

bit 13-11    **未实现:** 读为 0

bit 10-8     **SEG2PH<2:0>:** 相位缓冲段 2 位

111 = 长度为 8 x T<sub>Q</sub>

•

•

•

000 = 长度为 1 x T<sub>Q</sub>

bit 7        **SEG2PHTS:** 相位缓冲段 2 时间选择位

1 = 可自由编程

0 = SEG1PH<sub>x</sub> 位的最大值与信息处理时间 (Information Processing Time, IPT) 中的较大值

bit 6        **SAM:** CAN 总线线路采样位

1 = 在采样点对总线线路采样三次

0 = 在采样点对总线线路采样一次

bit 5-3     **SEG1PH<2:0>:** 相位缓冲段 1 位

111 = 长度为 8 x T<sub>Q</sub>

•

•

•

000 = 长度为 1 x T<sub>Q</sub>

bit 2-0     **PRSEG<2:0>:** 传播时间段位

111 = 长度为 8 x T<sub>Q</sub>

•

•

•

000 = 长度为 1 x T<sub>Q</sub>

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-11: CxFEN1: CANx 接收过滤器使能寄存器 1

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-0      **FLTEN<15:0>**: 使能过滤器 n 接收报文位  
 1 = 使能过滤器 n  
 0 = 禁止过滤器 n

## 寄存器 23-12: CxBUFPNT1: CANx 过滤器 0-3 缓冲区指针寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3BP3	F3BP2	F3BP1	F3BP0	F2BP3	F2BP2	F2BP1	F2BP0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F1BP3	F1BP2	F1BP1	F1BP0	F0BP3	F0BP2	F0BP1	F0BP0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-12      **F3BP<3:0>**: 过滤器 3 的接收缓冲区屏蔽位  
 1111 = 满足过滤条件的数据被接收到接收 FIFO 缓冲区中  
 1110 = 满足过滤条件的数据被接收到接收缓冲区 14 中  
 •  
 •  
 •  
 0001 = 满足过滤条件的数据被接收到接收缓冲区 1 中  
 0000 = 满足过滤条件的数据被接收到接收缓冲区 0 中

bit 11-8      **F2BP<3:0>**: 过滤器 2 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)

bit 7-4      **F1BP<3:0>**: 过滤器 1 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)

bit 3-0      **F0BP<3:0>**: 过滤器 0 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-13: CxBUFNT2: CANx 过滤器 4-7 缓冲区指针寄存器 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7BP3	F7BP2	F7BP1	F7BP0	F6BP3	F6BP2	F6BP1	F6BP0
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F5BP3	F5BP2	F5BP1	F5BP0	F4BP3	F4BP2	F4BP1	F4BP0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-12      **F7BP<3:0>**: 过滤器 7 的接收缓冲区屏蔽位  
 1111 = 满足过滤条件的数据被接收到接收 FIFO 缓冲区中  
 1110 = 满足过滤条件的数据被接收到接收缓冲区 14 中  
 •  
 •  
 •  
 0001 = 满足过滤条件的数据被接收到接收缓冲区 1 中  
 0000 = 满足过滤条件的数据被接收到接收缓冲区 0 中

bit 11-8      **F6BP<3:0>**: 过滤器 6 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)  
 bit 7-4      **F5BP<3:0>**: 过滤器 5 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)  
 bit 3-0      **F4BP<3:0>**: 过滤器 4 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)

## 寄存器 23-14: CxBUFNT3: CANx 过滤器 8-11 缓冲区指针寄存器 3

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F11BP3	F11BP2	F11BP1	F11BP0	F10BP3	F10BP2	F10BP1	F10BP0
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F9BP3	F9BP2	F9BP1	F9BP0	F8BP3	F8BP2	F8BP1	F8BP0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-12      **F11BP<3:0>**: 过滤器 11 的接收缓冲区屏蔽位  
 1111 = 满足过滤条件的数据被接收到接收 FIFO 缓冲区中  
 1110 = 满足过滤条件的数据被接收到接收缓冲区 14 中  
 •  
 •  
 •  
 0001 = 满足过滤条件的数据被接收到接收缓冲区 1 中  
 0000 = 满足过滤条件的数据被接收到接收缓冲区 0 中

bit 11-8      **F10BP<3:0>**: 过滤器 10 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)  
 bit 7-4      **F9BP<3:0>**: 过滤器 9 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)  
 bit 3-0      **F8BP<3:0>**: 过滤器 8 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-15: CxBUFPNT4: CANx 过滤器 12-15 缓冲区指针寄存器 4

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15BP3	F15BP2	F15BP1	F15BP0	F14BP3	F14BP2	F14BP1	F14BP0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F13BP3	F13BP2	F13BP1	F13BP0	F12BP3	F12BP2	F12BP1	F12BP0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-12     **F15BP<3:0>**: 过滤器 15 的接收缓冲区屏蔽位  
1111 = 满足过滤条件的数据被接收到接收 FIFO 缓冲区中  
1110 = 满足过滤条件的数据被接收到接收缓冲区 14 中  
•  
•  
•  
0001 = 满足过滤条件的数据被接收到接收缓冲区 1 中  
0000 = 满足过滤条件的数据被接收到接收缓冲区 0 中

bit 11-8     **F14BP<3:0>**: 过滤器 14 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)

bit 7-4     **F13BP<3:0>**: 过滤器 13 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)

bit 3-0     **F12BP<3:0>**: 过滤器 12 的接收缓冲区屏蔽位 (与 bit 15-12 的值相同)



# dsPIC33EPXXGS70X/80X 系列

**寄存器 23-16: CxRxFnSID: CANx 接收过滤器 n 标准标识符寄存器 (n = 0-15)**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15							bit 8
R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16
bit 7							bit 0

**图注:**

R = 可读位  
-n = POR 时的值

W = 可写位  
1 = 置 1

U = 未实现位, 读为 0  
0 = 清零

x = 未知

- bit 15-5      **SID<10:0>**: 标准标识符位  
1 = 报文地址位 SIDx 必须为 1 以匹配过滤器  
0 = 报文地址位 SIDx 必须为 0 以匹配过滤器
- bit 4      **未实现**: 读为 0
- bit 3      **EXIDE**: 扩展标识符使能位  
如果 MIDE = 1:  
1 = 仅匹配带有扩展标识符地址的报文  
0 = 仅匹配带有标准标识符地址的报文  
如果 MIDE = 0:  
忽略 EXIDE 位。
- bit 2      **未实现**: 读为 0
- bit 1-0      **EID<17:16>**: 扩展标识符位  
1 = 报文地址位 EIDx 必须为 1 以匹配过滤器  
0 = 报文地址位 EIDx 必须为 0 以匹配过滤器

**寄存器 23-17: CxRxFnEID: CANx 接收过滤器 n 扩展标识符寄存器 (n = 0-15)**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<15:8>							
bit 15							bit 8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<7:0>							
bit 7							bit 0

**图注:**

R = 可读位  
-n = POR 时的值

W = 可写位  
1 = 置 1

U = 未实现位, 读为 0  
0 = 清零

x = 未知

- bit 15-0      **EID<15:0>**: 扩展标识符位  
1 = 报文地址位 EIDx 必须为 1 以匹配过滤器  
0 = 报文地址位 EIDx 必须为 0 以匹配过滤器

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-18: CxFMSKSEL1: CANx 过滤器 7-0 屏蔽选择寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7MSK1	F7MSK0	F6MSK1	F6MSK0	F5MSK1	F5MSK0	F4MSK1	F4MSK0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3MSK1	F3MSK0	F2MSK1	F2MSK0	F1MSK1	F1MSK0	F0MSK1	F0MSK0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-14      **F7MSK<1:0>**: 过滤器 7 的屏蔽源位  
                     11 = 保留  
                     10 = 接收屏蔽寄存器 2 包含屏蔽值  
                     01 = 接收屏蔽寄存器 1 包含屏蔽值  
                     00 = 接收屏蔽寄存器 0 包含屏蔽值
- bit 13-12      **F6MSK<1:0>**: 过滤器 6 的屏蔽源位 (与 bit 15-14 的值相同)
- bit 11-10      **F5MSK<1:0>**: 过滤器 5 的屏蔽源位 (与 bit 15-14 的值相同)
- bit 9-8        **F4MSK<1:0>**: 过滤器 4 的屏蔽源位 (与 bit 15-14 的值相同)
- bit 7-6        **F3MSK<1:0>**: 过滤器 3 的屏蔽源位 (与 bit 15-14 的值相同)
- bit 5-4        **F2MSK<1:0>**: 过滤器 2 的屏蔽源位 (与 bit 15-14 的值相同)
- bit 3-2        **F1MSK<1:0>**: 过滤器 1 的屏蔽源位 (与 bit 15-14 的值相同)
- bit 1-0        **F0MSK<1:0>**: 过滤器 0 的屏蔽源位 (与 bit 15-14 的值相同)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-19: CxFMSKSEL2: CANx 过滤器 15-8 屏蔽选择寄存器 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15MSK1	F15MSK0	F14MSK1	F14MSK0	F13MSK1	F13MSK0	F12MSK1	F12MSK0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F11MSK1	F11MSK0	F10MSK1	F10MSK0	F9MSK1	F9MSK0	F8MSK1	F8MSK0
bit 7							bit 0

### 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-14 **F15MSK<1:0>**: 过滤器 15 的屏蔽源位

11 = 保留

10 = 接收屏蔽寄存器 2 包含屏蔽值

01 = 接收屏蔽寄存器 1 包含屏蔽值

00 = 接收屏蔽寄存器 0 包含屏蔽值

bit 13-12 **F14MSK<1:0>**: 过滤器 14 的屏蔽源位 (与 bit 15-14 的值相同)

bit 11-10 **F13MSK<1:0>**: 过滤器 13 的屏蔽源位 (与 bit 15-14 的值相同)

bit 9-8 **F12MSK<1:0>**: 过滤器 12 的屏蔽源位 (与 bit 15-14 的值相同)

bit 7-6 **F11MSK<1:0>**: 过滤器 11 的屏蔽源位 (与 bit 15-14 的值相同)

bit 5-4 **F10MSK<1:0>**: 过滤器 10 的屏蔽源位 (与 bit 15-14 的值相同)

bit 3-2 **F9MSK<1:0>**: 过滤器 9 的屏蔽源位 (与 bit 15-14 的值相同)

bit 1-0 **F8MSK<1:0>**: 过滤器 8 的屏蔽源位 (与 bit 15-14 的值相同)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-20: CxRXMnSID: CANx 接收过滤器屏蔽器 n 标准标识符寄存器 (n = 0-2)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15						bit 8	

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	MIDE	—	EID17	EID16
bit 7						bit 0	

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-5      **SID<10:0>**: 标准标识符位  
 1 = 过滤器比较操作包含 SIDx 位  
 0 = 过滤器比较操作与 SIDx 位无关
- bit 4          **未实现**: 读为 0
- bit 3          **MIDE**: 标识符接收模式位  
 1 = 只匹配与过滤器中 EXIDE 位对应的报文类型 (标准或扩展地址)  
 0 = 如果过滤器匹配, 则与标准或扩展地址报文匹配 (即, 如果 (过滤器 SIDx) = (报文 SIDx) 或如果 (过滤器 SIDx/EIDx) = (报文 SIDx/EIDx))
- bit 2          **未实现**: 读为 0
- bit 1-0        **EID<17:16>**: 扩展标识符位  
 1 = 过滤器比较操作包含 EIDx 位  
 0 = 过滤器比较操作与 EIDx 位无关

## 寄存器 23-21: CxRXMnEID: CANx 接收过滤器屏蔽器 n 扩展标识符寄存器 (n = 0-2)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<15:8>							
bit 15						bit 8	

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID<7:0>							
bit 7						bit 0	

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 15-0      **EID<15:0>**: 扩展标识符位  
 1 = 过滤器比较操作包含 EIDx 位  
 0 = 过滤器比较操作与 EIDx 位无关

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-22: CxRXFUL1: CANx 接收缓冲区满寄存器 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<15:8>							
bit 15				bit 8			

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<7:0>							
bit 7				bit 0			

### 图注:

C = 可写位, 但只能写入 0 来清零该位			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0      **RXFUL<15:0>**: 接收缓冲区 n 满位  
 1 = 缓冲区已满 (由模块置 1)  
 0 = 缓冲区为空 (由用户软件清零)

## 寄存器 23-23: CxRXFUL2: CANx 接收缓冲区满寄存器 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<31:24>							
bit 15				bit 8			

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL<23:16>							
bit 7				bit 0			

### 图注:

C = 可写位, 但只能写入 0 来清零该位			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-0      **RXFUL<31:16>**: 接收缓冲区 n 满位  
 1 = 缓冲区已满 (由模块置 1)  
 0 = 缓冲区为空 (由用户软件清零)

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 23-24: CxRXOVF1: CANx 接收缓冲区溢出寄存器 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<15:8>							
bit 15				bit 8			

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<7:0>							
bit 7				bit 0			

**图注:** C = 可写位, 但只能写入 0 来清零该位  
R = 可读位 W = 可写位 U = 未实现位, 读为 0  
-n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

bit 15-0 **RXOVF<15:0>**: 接收缓冲区 n 溢出位  
1 = 模块尝试对一个已满的缓冲区执行写操作 (由模块置 1)  
0 = 无溢出条件 (由用户软件清零)

## 寄存器 23-25: CxRXOVF2: CANx 接收缓冲区溢出寄存器 2

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<31:24>							
bit 15				bit 8			

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF<23:16>							
bit 7				bit 0			

**图注:** C = 可写位, 但只能写入 0 来清零该位  
R = 可读位 W = 可写位 U = 未实现位, 读为 0  
-n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

bit 15-0 **RXOVF<31:16>**: 接收缓冲区 n 溢出位  
1 = 模块尝试对一个已满的缓冲区执行写操作 (由模块置 1)  
0 = 无溢出条件 (由用户软件清零)

# dsPIC33EPXXGS70X/80X 系列

寄存器 23-26: CxTRmnCON: CANx 发送 / 接收缓冲区 mn 控制寄存器 (m = 0,2,4,6 ; n = 1,3,5,7)

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENn	TXABTn	TXLARBn	TXERRn	TXREQn	RTRENn	TXnPRI1	TXnPRI0
bit 15							bit 8

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENm	TXABTm <sup>(1)</sup>	TXLARBm <sup>(1)</sup>	TXERRm <sup>(1)</sup>	TXREQm	RTRENm	TXmPRI1	TXmPRI0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 请参见控制缓冲区 n 中 bit 7-0 的定义。

bit 7 **TXENm**: 发送 / 接收缓冲区 m 选择位

1 = 缓冲区 TRBm 是发送缓冲区

0 = 缓冲区 TRBm 是接收缓冲区

bit 6 **TXABTm**: 报文中止位<sup>(1)</sup>

1 = 报文中止

0 = 成功完成报文发送

bit 5 **TXLARBm**: 报文仲裁失败位<sup>(1)</sup>

1 = 报文在发送过程中仲裁失败

0 = 报文在发送过程中没有仲裁失败

bit 4 **TXERRm**: 在发送过程中检测到错误位<sup>(1)</sup>

1 = 发送报文时发生总线错误

0 = 发送报文时未发生总线错误

bit 3 **TXREQm**: 报文发送请求位

1 = 请求发送报文; 当报文发送成功时, 该位会自动清零

0 = 在该位置 1 的情况下清零该位 (= 0) 将请求中止报文

bit 2 **RTRENm**: 自动远程发送使能位

1 = 当接收到远程发送时, TXREQx 置 1

0 = 当接收到远程发送时, TXREQx 不受影响

bit 1-0 **TXmPRI<1:0>**: 报文发送优先级位

11 = 最高报文优先级

10 = 中高报文优先级

01 = 中低报文优先级

00 = 最低报文优先级

注 1: 当 TXREQmn 置 1 时清零该位。

注: 缓冲区、SIDx、EIDx、DLCx、数据字段和接收状态寄存器位于 DMA RAM 中。

# dsPIC33EPXXGS70X/80X 系列

## 23.4 CAN 报文缓冲区

CAN 报文缓冲区是 RAM 存储区的一部分。它们不是 CAN 特殊功能寄存器。用户应用程序必须直接写入为 CAN 报文缓冲区配置的 RAM 区。缓冲区的位置和大小由用户应用程序定义。

### 缓冲器 21-1: CANx 报文缓冲区字 0

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
—	—	—	SID10	SID9	SID8	SID7	SID6	
bit 15								bit 8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
SID5	SID4	SID3	SID2	SID1	SID0	SRR	IDE	
bit 7								bit 0

#### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-13      **未实现:** 读为 0  
 bit 12-2      **SID<10:0>:** 标准标识符位  
 bit 1          **SRR:** 替代远程请求位  
                 **当 IDE = 0 时:**  
                 1 = 报文将请求远程发送  
                 0 = 正常报文  
                 **当 IDE = 1 时:**  
                 **SRR** 位必须设置为 1。  
 bit 0          **IDE:** 扩展标识符位  
                 1 = 报文将发送扩展标识符  
                 0 = 报文将发送标准标识符

### 缓冲器 21-2: CANx 报文缓冲区字 1

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	
—	—	—	—	EID<17:14>				
bit 15								bit 8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	
EID<13:6>								
bit 7								bit 0

#### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-12      **未实现:** 读为 0  
 bit 11-0      **EID<17:6>:** 扩展标识符位



# dsPIC33EPXXGS70X/80X 系列

## 缓冲器 21-3: CANx 报文缓冲区字 2

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID5	EID4	EID3	EID2	EID1	EID0	RTR	RB1
bit 15							bit 8
U-x	U-x	U-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	RB0	DLC3	DLC2	DLC1	DLC0
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-10      **EID<5:0>**: 扩展标识符位

bit 9          **RTR**: 远程发送请求位

当 IDE = 1 时:

1 = 报文将请求远程发送

0 = 正常报文

当 IDE = 0 时:

RTR 位被忽略。

bit 8          **RB1**: 保留位 1

用户必须按 CAN 协议将该位设置为 0。

bit 7-5        **未实现**: 读为 0

bit 4          **RB0**: 保留位 0

用户必须按 CAN 协议将该位设置为 0。

bit 3-0        **DLC<3:0>**: 数据长度编码位

## 缓冲器 21-4: CANx 报文缓冲区字 3

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 1<15:8>							
bit 15							bit 8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 0<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8      **字节 1<15:8>**: CANx 报文字节 1 位

bit 7-0        **字节 0<7:0>**: CANx 报文字节 0 位

# dsPIC33EPXXGS70X/80X 系列

## 缓冲器 21-5: CANx 报文缓冲区字 4

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 3<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 2<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            字节 3<15:8>: CANx 报文字节 3 位

bit 7-0            字节 2<7:0>: CANx 报文字节 2 位

## 缓冲器 21-6: CANx 报文缓冲区字 5

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 5<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 4<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15-8            字节 5<15:8>: CANx 报文字节 5 位

bit 7-0            字节 4<7:0>: CANx 报文字节 4 位

# dsPIC33EPXXGS70X/80X 系列

## 缓冲器 21-7: CANx 报文缓冲区字 6

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 7<15:8>							
bit 15							bit 8

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
字节 6<7:0>							
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-8              **字节 7<15:8>**: CANx 报文字节 7 位

bit 7-0              **字节 6<7:0>**: CANx 报文字节 6 位

## 缓冲器 21-8: CANx 报文缓冲区字 7

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	FILHIT<4:0> <sup>(1)</sup>				
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
 -n = POR 时的值              1 = 置 1                      0 = 清零                      x = 未知

bit 15-13              **未实现**: 读为 0

bit 12-8              **FILHIT<4:0>**: 选中过滤器的编码位 <sup>(1)</sup>  
 对导致写入该缓冲区的过滤器的编号进行编码。

bit 7-0              **未实现**: 读为 0

**注 1:**      模块只能针对接收缓冲区执行写操作, 不用于发送缓冲区。

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 24.0 高速模拟比较器

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“**高速模拟比较器模块**” (DS70005128)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节 “存储器构成”**。

高速模拟比较器模块用于监视对于 CPU 和 ADC 可能速度太快而无法捕捉的瞬态电流和 / 或电压。

## 24.1 特性概述

开关电源 (Switch Mode Power Supply, SMPS) 比较器模块提供以下主要特性:

- 4 个轨到轨模拟比较器
- 每个模拟比较器均具有专用的 12 位 DAC
- 每个比较器最多具有 6 个可选输入源:
  - 4 个外部输入
  - 2 个来自 PGAx 模块的内部输入
- 可编程比较器滞后
- 可编程输出极性
- 最多 2 个 DAC 输出连接至器件引脚
- 用于 DAC 的多个参考电压:
  - 外部参考 (EXTREF1 或 EXTREF2)
  - AVDD
- 可产生中断
- PWM 功能支持:
  - PWM 占空比控制
  - PWM 周期控制
  - PWM 故障检测

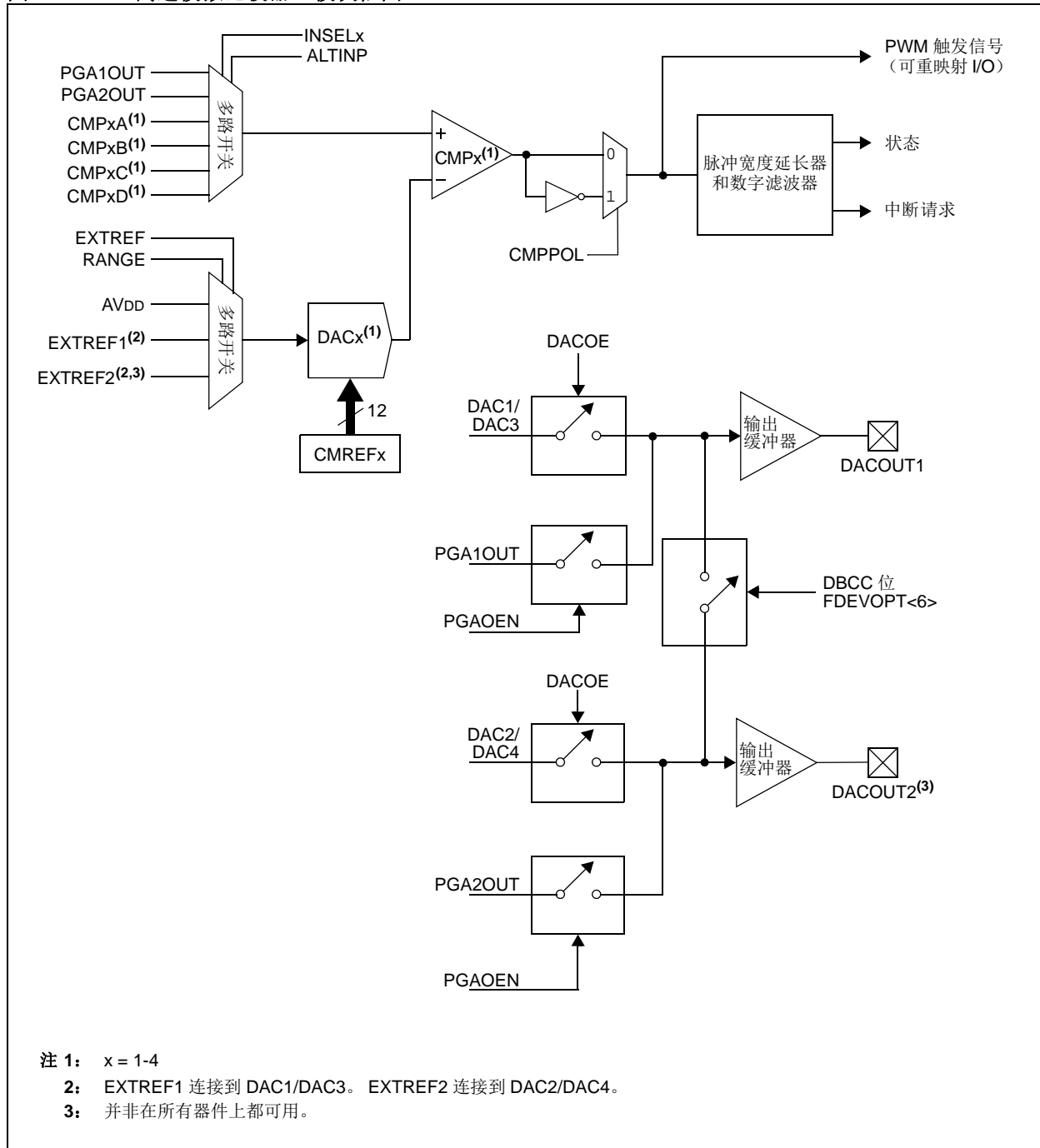
# dsPIC33EPXXGS70X/80X 系列

## 24.2 模块说明

图24-1给出了高速模拟比较器模块中的一个模拟比较器的功能框图。模拟比较器提供高速操作，典型延时为15 ns。比较器的反相输入端总是连接到DACx电路。比较器的同相输入端连接到一个用于选择所需输入源引脚的模拟多路开关。

模拟比较器的输入引脚通常与模数转换器（ADC）模块使用的引脚共用。比较器和ADC可同时使用相同的引脚。此功能允许用户用ADC测量输入电压，同时用比较器检测瞬态电压。

图 24-1: 高速模拟比较器 x 模块框图



## 24.3 模块应用

通过该模块，SMPS dsPIC<sup>®</sup> DSC 器件提供监视电源转换应用中的电压和电流的方法。它可以检测瞬态条件并触发 dsPIC DSC 处理器和 / 或外设进行相应处理，无需处理器和 ADC 持续监视电压或电流，让 dsPIC DSC 可以执行其他任务。

比较器模块具有一个高速比较器和一个关联的 12 位 DAC，DAC 为比较器的反相输入端提供可编程的参考电压。比较器输出的极性可由用户编程。模块的输出可用于以下模式：

- 产生中断
- 触发 ADC 采样和转换过程
- 截断 PWM 信号（限流）
- 截断 PWM 周期（电流最小）
- 禁止 PWM 输出（故障锁定）

比较器模块的输出可同时用于多种模式，例如：当检测到的电压超出预期值时，比较器输出可用于 1) 产生中断，2) 使 ADC 对其进行采样和转换，以及 3) 截断 PWM 输出。

当模拟输入电压超出设定的阈值电压时，比较器模块还可用于将系统从休眠或空闲模式唤醒。

## 24.4 数模比较器（DAC）

每个模拟比较器均具有一个专用的 12 位 DAC，用于通过 CMPxDAC 寄存器来编程比较器阈值电压。DAC 参考电压源通过 CMPxCON 寄存器中的 EXTREF 和 RANGE 位来选择。

EXTREF 位用于将外部参考电压 EXTREFx 或内部源选作参考电压源。用户可以利用 EXTREFx 输入连接到更适合其应用的参考电压。选择内部参考电压时，RANGE 位会将 AVDD 使能为 DAC 的参考电压源。

**注：** EXTREF2 并非在所有器件上都可用。

每个 DACx 在 CMPxCON 寄存器中都有一个输出使能位 DACOE，用于使能 DACx 参考电压以连接至外部输出引脚（DACOUTx）。有关将 DACx 输出电压连接至 DACOUTx 引脚的信息，请参见图 24-1。

- 注 1：** 需要在软件中确保不将多个 DACOE 位置 1。如果有多个比较器使能 DACx 输出，则 DACOUTx 引脚上的输出将是不确定的。
- 2：** DACOUT2 并非在所有器件上都可用。

## 24.5 脉冲宽度延长器和数字逻辑

模拟比较器可以响应非常快速的瞬态信号。为比较器输出指定所需极性之后，信号会被传递给脉冲宽度延长电路。脉冲宽度延长电路具有异步设置功能和延时电路，可确保最小脉冲宽度为 3 个系统时钟周期，从而使连接的电路可以正确响应窄脉冲事件。

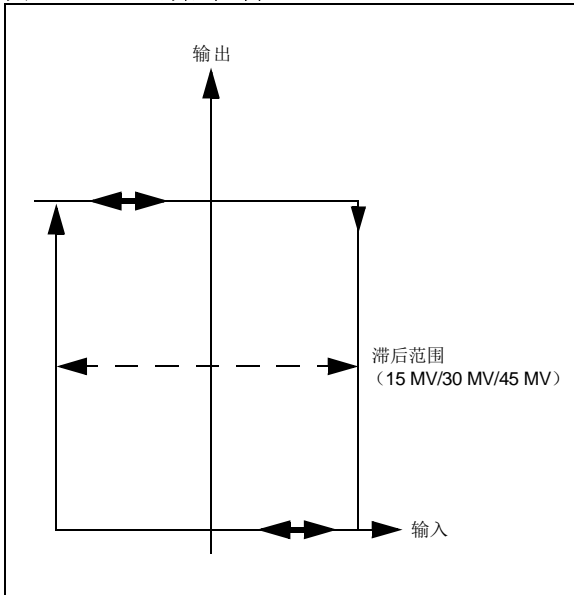
脉冲宽度延长电路之后是一个数字滤波器。数字滤波器通过 CMPxCON 寄存器中的 FLTREN 位使能。数字滤波器使用通过 CMPxCON 寄存器中的 FCLKSEL 位指定的时钟工作。对于选定的时钟，比较器信号必须在高电平或低电平状态保持稳定至少 3 个时钟周期，这样它才能通过数字滤波器。

## 24.6 滞后

模块的另一个功能是滞后控制。通过 CMPxCON 寄存器的 HYSSEL<1:0> 位，可以使能或禁止滞后，以及控制其幅值。有 3 个不同的值可供使用：15 mV、30 mV 和 45 mV。此外，还可以选择对哪个边沿（上升或下降）应用滞后。

滞后控制可以防止由于输入上的小扰动（噪声），导致比较器输出不断改变状态（见图 24-2）。

图 24-2： 滞后控制



## 24.7 模拟比较器资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 24.7.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“高速模拟比较器模块”（DS70005128）
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具



# dsPIC33EPXXGS70X/80X 系列

寄存器 24-1: **CMPxCON**: 比较器 x 控制寄存器

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMPON	—	CMPSIDL	HYSSEL1	HYSSEL0	FLTREN	FCLKSEL	DACOE
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	HC-0, HS	R/W-0	R/W-0	R/W-0
INSEL1	INSEL0	EXTREF	HYSPOL	CMPSTAT	ALTINP	CMPPOL	RANGE
bit 7							bit 0

<b>图注:</b>	HC = 硬件清零位	HS = 硬件置 1 位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15      **CMPON**: 比较器工作模式位  
 1 = 使能比较器模块  
 0 = 禁止比较器模块 (降低功耗)
- bit 14      **未实现**: 读为 0
- bit 13      **CMPSIDL**: 比较器空闲模式停止位  
 1 = 当器件进入空闲模式时, 模块停止工作  
 0 = 在空闲模式下模块继续工作  
 如果器件具有多个比较器, 将任一 **CMPSIDL** 位设置为 1, 将在空闲模式下禁止所有比较器。
- bit 12-11   **HYSSEL<1:0>**: 比较器滞后选择位  
 11 = 45 mV 滞后  
 10 = 30 mV 滞后  
 01 = 15 mV 滞后  
 00 = 不选择任何滞后
- bit 10      **FLTREN**: 数字滤波器使能位  
 1 = 使能数字滤波器  
 0 = 禁止数字滤波器
- bit 9        **FCLKSEL**: 数字滤波器和脉冲宽度延长器时钟选择位  
 1 = 数字滤波器和脉冲宽度延长器使用 PWM 时钟工作  
 0 = 数字滤波器和脉冲宽度延长器使用系统时钟工作
- bit 8        **DACOE**: DACx 输出使能位  
 1 = DACx 模拟电压连接到 DACOUTx 引脚 <sup>(1)</sup>  
 0 = DACx 模拟电压不连接到 DACOUTx 引脚
- bit 7-6     **INSEL<1:0>**: 比较器输入源选择位  
如果 ALTINP = 0, 则从比较器输入中选择:  
 11 = 选择 CMPxD 输入引脚  
 10 = 选择 CMPxC 输入引脚  
 01 = 选择 CMPxB 输入引脚  
 00 = 选择 CMPxA 输入引脚  
如果 ALTINP = 1, 则从备用输入中选择:  
 11 = 保留  
 10 = 保留  
 01 = 选择 PGA2 输出  
 00 = 选择 PGA1 输出

**注 1:** 在任意给定时刻, DACOUTx 只能与单个比较器关联。软件必须确保没有多个比较器通过置 1 相应的 DACOE 位使能 DACx 输出。

# dsPIC33EPXXGS70X/80X 系列

---

## 寄存器 24-1: CMPxCON: 比较器 x 控制寄存器 (续)

- bit 5      **EXTREF:** 外部参考电压源使能位  
1 = 外部参考电压源为 DACx 提供参考电压 (最大 DAC 电压由外部电压源决定)  
0 = AVDD 为 DACx 提供参考电压 (最大 DAC 电压为 AVDD)
- bit 4      **HYSPOL:** 比较器滞后极性选择位  
1 = 在比较器输出的下降沿应用滞后  
0 = 在比较器输出的上升沿应用滞后
- bit 3      **CMPSTAT:** 比较器电流状态位  
反映比较器 x 的电流输出状态, 包括 CMPPOL 位的设置。
- bit 2      **ALTINP:** 备用输入选择位  
1 = INSEL<1:0> 位用于选择备用输入  
0 = INSEL<1:0> 位用于选择比较器输入
- bit 1      **CMPPOL:** 比较器输出极性控制位  
1 = 输出反相  
0 = 输出不反相
- bit 0      **RANGE:** DACx 输出电压范围选择位  
1 = AVDD 为最大 DACx 输出电压  
0 = 未实现, 不使用

**注 1:** 在任意给定时刻, DACOUTx 只能与单个比较器关联。软件必须确保没有多个比较器通过置 1 相应的 DACOE 位使能 DACx 输出。

# dsPIC33EPXXGS70X/80X 系列

寄存器 24-2: **CMPxDAC: 比较器 x DAC 控制寄存器**

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	CMREF<11:8>			
bit 15				bit 8			
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMREF<7:0>							
bit 7				bit 0			

**图注:**

R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-12     **未实现:** 读为 0

bit 11-0     **CMREF<11:0>:** 比较器参考电压选择位

111111111111

- 
- 
- 
- 
- 
- 
- 

$$= ([\text{CMREF}<11:0>] * (\text{AVDD}) / 4096) \text{ V } (\text{EXTREF} = 0)$$

$$\text{或 } ([\text{CMREF}<11:0>] * (\text{EXTREF}) / 4096) \text{ V } (\text{EXTREF} = 1)$$

000000000000

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 25.0 可编程增益放大器 (PGA)

**注 1:** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息, 请参见《dsPIC33/PIC24 系列参考手册》的“可编程增益放大器 (PGA)” (DS70005146), 该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

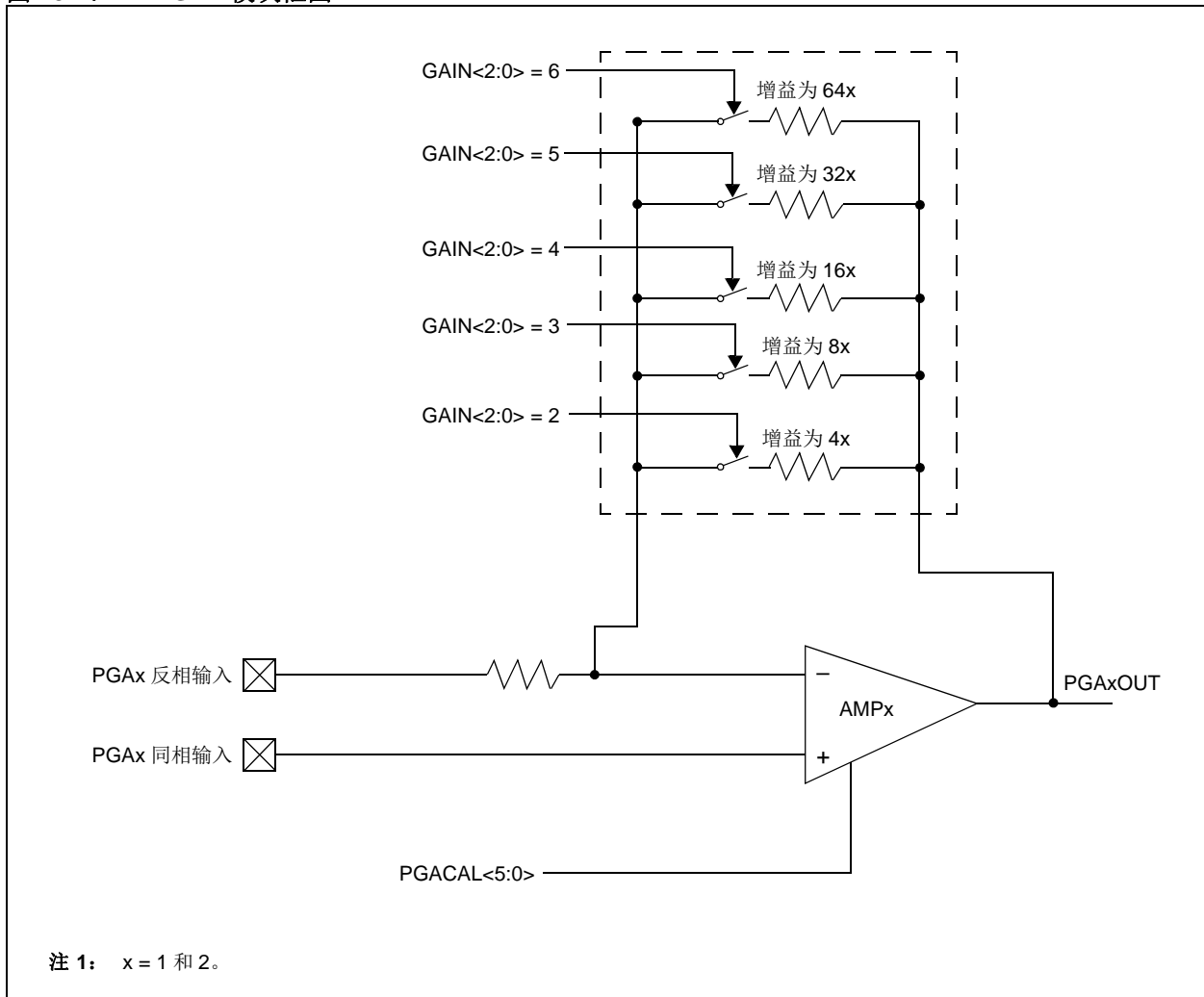
**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息, 请参见本数据手册中的第 4.0 节“存储器构成”。

dsPIC33EPXXGS70X/80X 系列器件具有两个可编程增益放大器 (PGA1 和 PGA2)。PGA 是基于运放且具有用户可编程增益的同相放大器。PGA 的输出可连接到模数转换器的多个专用采样保持输入和 / 或高速模拟比较器模块。PGA 有五个可选增益, 可用作以地为参考的放大器 (单端), 也可与独立地参考点配合使用。

PGA 模块的主要特性包括:

- 单端或独立地参考
- 可选增益: 4x、8x、16x、32x 和 64x
- 高增益带宽
- 轨到轨输出电压
- 宽输入电压范围

图 25-1: PGAx 模块框图



# dsPIC33EPXXGS70X/80X 系列

## 25.1 模块说明

可编程增益放大器用于将低电压（即负载电阻 / 电流检测电阻两端的电压）放大，从而提高测量信号的信噪比。PGA<sub>x</sub> 输出电压可通过 ADC 模块上的四个专用采样保持电路中的任意一个进行读取。输出电压也可馈入比较器模块以进行过流 / 过压保护。图 25-2 给出了 PGA<sub>x</sub> 模块的功能框图。有关互连的更多详细信息，请参见第 22.0 节“高速 12 位模数转换器 (ADC)”和第 24.0 节“高速模拟比较器”。

PGA<sub>x</sub> 模块的增益可通过 PGA<sub>x</sub>CON 寄存器中的 GAIN<2:0> 位来选择。共有五个可选增益，其范围为 4x 至 64x。PGA<sub>x</sub>CON 寄存器中的 SELPI<2:0> 和 SELNI<2:0> 位用于为 PGA<sub>x</sub> 模块从四对同相 / 反相输入

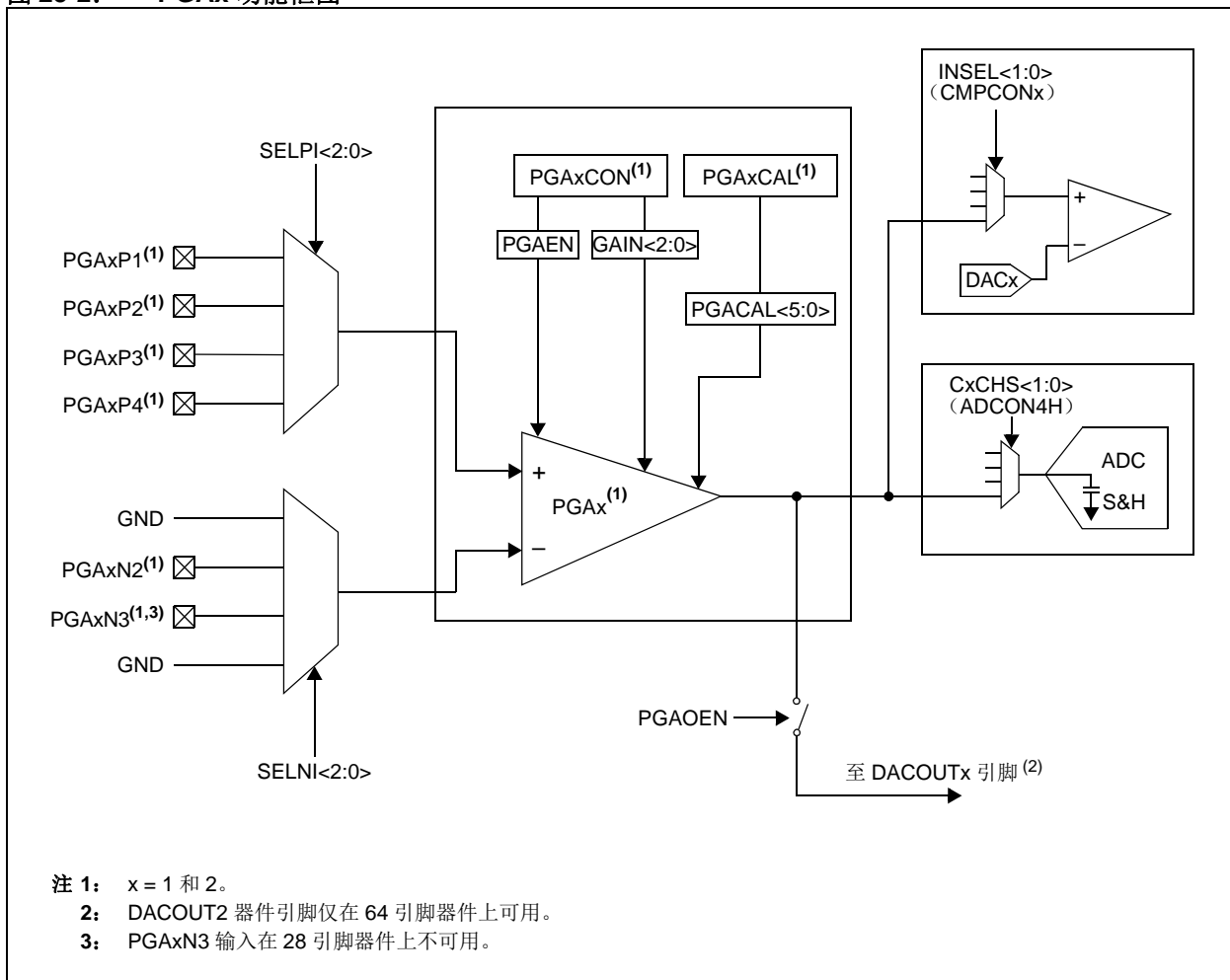
中选择一对。对于单端应用，SELNI<2:0> 位将选择地作为反相输入源。要提供独立地参考，可将 PGA<sub>x</sub>N2 和 PGA<sub>x</sub>N3 引脚作为 PGA<sub>x</sub> 模块的反相输入源。

**注 1:** 并非所有 PGA 同相 / 反相输入在所有器件上均可用。关于可用的输入源引脚，请参见具体器件的引脚分配。

可通过将 PGA<sub>x</sub>CON 寄存器中的 PGOEN 位置 1，以将 PGA<sub>x</sub> 模块的输出电压连接至 DACOUT<sub>x</sub> 引脚。当使能 PGOEN 位时，PGA1 的输出电压连接至 DACOUT1，PGA2 的输出电压连接至 DACOUT2。对于只有一个 DACOUT<sub>x</sub> 引脚的器件，可通过配置 FDEVOPT 寄存器中的 DBCC 配置位 (FDEVOPT<6>) 将 PGA2 的输出电压连接至 DACOUT1。

如果 DAC<sub>x</sub> 输出电压和 PGA<sub>x</sub> 输出电压均连接至 DACOUT<sub>x</sub> 引脚，则产生的输出电压将为这两个信号的组合。PGA<sub>x</sub> 模块和 DAC<sub>x</sub> 模块之间未分配优先级。

图 25-2: PGA<sub>x</sub> 功能框图



# dsPIC33EPXXGS70X/80X 系列

## 25.2 PGA 资源

对于本数据手册中列出的器件，Microchip 网站的主要产品页面中提供了许多有用的资源。该产品页面包含最新更新和附加信息。

### 25.2.1 主要资源

- 《dsPIC33/PIC24 系列参考手册》中的“可编程增益放大器 (PGA)” (DS70005146)
- 代码示例
- 应用笔记
- 软件库
- 网上研讨会
- 所有相关的《dsPIC33/PIC24 系列参考手册》章节
- 开发工具

### 寄存器 25-1: PGAxCON: PGAx 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PGAEN	PGAOEN	SELPI2	SELPI1	SELPI0	SELNI2	SELNI1	SELNI0
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	GAIN2	GAIN1	GAIN0
bit 7				bit 0			

#### 图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15      **PGAEN:** PGAx 使能位  
 1 = 使能 PGAx 模块  
 0 = 禁止 PGAx 模块 (降低功耗)
- bit 14      **PGAOEN:** PGAx 输出使能位  
 1 = PGAx 输出连接到 DACOUTx 引脚  
 0 = PGAx 输出不连接到 DACOUTx 引脚
- bit 13-11   **SELPI<2:0>:** PGAx 同相输入选择位  
 111 = 保留  
 110 = 保留  
 101 = 保留  
 100 = 保留  
 011 = PGAxP4  
 010 = PGAxP3  
 001 = PGAxP2  
 000 = PGAxP1
- bit 10-8    **SELNI<2:0>:** PGAx 反相输入选择位  
 111 = 保留  
 110 = 保留  
 101 = 保留  
 100 = 保留  
 011 = 地 (单端模式)  
 010 = PGAxN3  
 001 = PGAxN2  
 000 = 地 (单端模式)
- bit 7-3     **未实现:** 读为 0

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 25-1: PGAxCON: PGAx 控制寄存器 (续)

bit 2-0 **GAIN<2:0>**: PGAx 增益选择位

111 = 保留  
110 = 增益为 64x  
101 = 增益为 32x  
100 = 增益为 16x  
011 = 增益为 8x  
010 = 增益为 4x  
001 = 保留  
000 = 保留

## 寄存器 25-2: PGAxCAL: PGAx 校准寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PGACAL<5:0>					
bit 7							bit 0

### 图注:

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                          0 = 清零                          x = 未知

bit 15-6 **未实现**: 读为 0

bit 5-0 **PGACAL<5:0>**: PGAx 失调校准位

模块使能前, PGA1 和 PGA2 的校准值必须从闪存地址 0x800E48 和 0x800E4C 分别复制到这些位中。更多信息, 请参见第 27.0 节 “特殊功能” 中的校准数据地址表 (表 27-3)。





# dsPIC33EPXXGS70X/80X 系列

## 26.3 电流源控制寄存器

寄存器 26-1: ISRCCON: 恒流源控制寄存器

R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
ISRCEN	—	—	—	—	OUTSEL2	OUTSEL1	OUTSEL0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ISRCCAL5	ISRCCAL4	ISRCCAL3	ISRCCAL2	ISRCCAL1	ISRCCAL0
bit 7							bit 0

**图注:**

R = 可读位                      W = 可写位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 15            **ISRCEN:** 恒流源使能位

1 = 使能电流源  
0 = 禁止电流源

bit 14-11        **未实现:** 读为 0

bit 10-8         **OUTSEL<2:0>:** 输出恒流选择位

111 = 保留  
110 = 保留  
101 = 保留  
100 = 输入引脚 ISRC4 (AN4)  
011 = 输入引脚 ISRC3 (AN5)  
010 = 输入引脚 ISRC2 (AN6)  
001 = 输入引脚 ISRC1 (AN12)  
000 = 不选择任何输出

bit 7-6           **未实现:** 读为 0

bit 5-0           **ISRCCAL<5:0>:** 恒流源校准位

模块使能前, 校准值必须从闪存地址 0x800E78 复制到这些位中。更多信息, 请参见第 27.0 节 “特殊功能” 中的校准数据地址表 (表 27-3)。

## 27.0 特殊功能

**注：** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的“**器件配置**”（DS70000618）、“**看门狗定时器和节能模式**”（DS70615）和“**CodeGuard™ 中等安全性**”（DS70005182），文档可从 Microchip 网站（[www.microchip.com](http://www.microchip.com)）下载。

dsPIC33EPXXGS70X/80X 系列器件具有几项特殊功能，旨在最大限度地提高应用的灵活性和可靠性，并通过减少外部元件将成本降至最低。提供的特殊功能包括：

- 灵活的配置
- 看门狗定时器（WDT）
- 代码保护和 CodeGuard™ 安全性
- JTAG 边界扫描接口
- 在线串行编程（ICSP）
- 在线仿真
- 欠压复位（Brown-out Reset, BOR）

## 27.1 配置位

在 dsPIC33EPXXGS70X/80X 系列器件中，配置字以易失性存储方式实现。这就意味着在器件每次上电时都必须对配置数据进行编程。配置数据存储在片上程序存储空间的末尾，称为闪存配置字。表 27-1 列出了它们的具体位置，表 27-2 给出了其详细说明。器件复位期间，配置数据会自动从闪存配置字装入到相应的配置影子寄存器中。

对于在双分区闪存模式下工作的器件，BSEQx 位（FBTSEQ<11:0>）决定哪个分区在启动时成为活动分区，该分区的配置字将装入配置影子寄存器。

**注：** 所有类型的器件复位都会重新装入配置数据。

在为这些器件创建应用程序时，用户应总是在其代码中告知编译器特别为配置数据分配闪存配置字地址，以确保当编译代码时程序代码不会存储到该地址。从配置空间执行程序代码会导致器件复位。

**注：** 对程序存储器的最后一页执行页擦除操作会清零闪存配置字。

表 27-1: 配置寄存器映射<sup>(3)</sup>

名称	地址	器件存储器大小 (KB)	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	
FSEC	00AF80	64	—	AIVTDIS	—	—	—	CSS<2:0>			CWRP	GSS<1:0>		GWRP	—	BSEN	BSS<1:0>		BWRP	
	015780	128																		
FBSLIM	00AF90	64	—	—	—	—	BSLIM<12:0>													
	015790	128																		
FSIGN	00AF90	64	—	保留 <sup>(2)</sup>	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
	015794	128																		
FOSCSSEL	00AF98	64	—	—	—	—	—	—	—	—	—	IESO	—	—	—	—	FNOSC<2:0>			
	015798	128																		
FOSC	00AF9C	64	—	—	—	—	—	—	—	—	PLLKEN	FCKSM<1:0>		IOL1WAY	—	—	OSCIOFNC	POSCMD<1:0>		
	01579C	128																		
FWDT	00AFA0	64	—	—	—	—	—	—	—	—	WDTWIN<1:0>		WINDIS	WDTEN<1:0>		WDTPRE	WDTPOST<3:0>			
	0157A0	128																		
FPOR	00AFA4	64	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	保留 <sup>(1)</sup>	
	0157A4	128																		
FICD	00AFA8	64	—	BTSWP	—	—	—	—	—	—	—	保留 <sup>(1)</sup>	—	JTAGEN	—	—	—	ICS<1:0>		
	0157A8	128																		
FDEVOPT	00AFAC	64	—	—	—	—	—	—	—	—	—	—	DBCC	—	ALTI2C2	ALTI2C1	保留 <sup>(1)</sup>	—	PWMLOCK	
	0157AC	128																		
FALTREG	00AFB0	64	—	—	CTXT4<2:0>			—	CTXT3<3:0>			—	CTXT2 <2:0>		—	CTXT1 <2:0>				
	0157B0	128																		
FBTSEQ	00AFFC	64	IBSEQ<11:0>						BSEQ<11:0>											
	0157FC	128																		
FBOOT <sup>(4)</sup>	801000	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BTMODE<1:0>	

- 注 1: 这些位保留, 必须编程为 1。  
 注 2: 该位保留, 必须编程为 0。  
 注 3: 在双分区闪存模式下工作时, 每个分区都将具有专用的配置寄存器。器件复位时, 活动分区的配置值将在启动时读取, 但在软交换期间, 新的活动分区的配置设置将被忽略。  
 注 4: FBOOT 位于配置存储空间中。

# dsPIC33EPXXGS70X/80X 系列

表 27-2: 配置位说明

位域	说明
BSS<1:0>	引导段代码保护级别位 11 = 引导段不被代码保护 (BWRP 写保护除外) 10 = 标准安全性 0x = 高安全性
BSEN	引导段控制位 1 = 未使能引导段 0 = 引导段大小由 BSLIM<12:0> 位决定
BWRP	引导段写保护位 1 = 引导段可以写入 0 = 引导段被写保护
BSLIM<12:0>	引导段闪存页地址限制位 包含最后一个有效的引导段页。要编程的值为页地址的补码，这样编程额外的 0 只会增加引导段的大小 (即, 0x1FFD = 2 页或 1024 个指令字)。
GSS<1:0>	通用段代码保护级别位 11 = 用户程序存储区不被代码保护 10 = 标准安全性 0x = 高安全性
GWRP	通用段写保护位 1 = 用户程序存储区不被写保护 0 = 用户程序存储区被写保护
CWRP	配置段写保护位 1 = 配置数据不被写保护 0 = 配置数据被写保护
CSS<2:0>	配置段代码保护级别位 111 = 配置数据不被代码保护 110 = 标准安全性 10x = 增强安全性 0xx = 高安全性
BTSWP	BOOTSWP 指令使能 / 禁止位 1 = 禁止 BOOTSWP 指令 0 = 使能 BOOTSWP 指令
BSEQ<11:0>	引导序列号位 (仅限双分区模式) 相对值, 用于定义器件复位后的活动分区; 包含较小引导号的分区将成为活动分区。
IBSEQ<11:0>	引导序列号反码位 (仅限双分区模式) BSEQ<11:0> 的反码; 必须由用户计算, 写入后才能进行器件编程。如果 BSEQx 和 IBSEQx 并非互为反码, 则引导序列号被视为无效。
AIVTDIS <sup>(1)</sup>	备用中断向量表位 1 = 禁止备用中断向量表 0 = 使能备用中断向量表 (INTCON2<8> = 1 时)
IESO	双速振荡器启动使能位 1 = 使用 FRC 启动器件, 然后自动切换到就绪的用户选择的振荡器源 0 = 使用用户选择的振荡器源启动器件
PWMLOCK	PWM 锁定使能位 1 = 只有在密钥序列之后, 才能对某些 PWM 寄存器进行写操作 0 = 无需密钥序列即可对 PWM 寄存器进行写操作

注 1: 必须存在引导段才能使用备用中断向量表。

# dsPIC33EPXXGS70X/80X 系列

表 27-2: 配置位说明 (续)

位域	说明
FNOSC<2:0>	振荡器选择位 111 = N 分频快速 RC 振荡器 (FRCDIVN) 110 = 16 分频快速 RC 振荡器 101 = 低功耗 RC 振荡器 (LPRC) 100 = 保留; 不要使用 011 = 带 PLL 模块的主振荡器 (XT+PLL、HS+PLL 和 EC+PLL) 010 = 主振荡器 (XT、HS 和 EC) 001 = 带 PLL 模块的 N 分频快速 RC 振荡器 (FRCPLL) 000 = 快速 RC 振荡器 (FRC)
FCKSM<1:0>	时钟切换模式位 1x = 禁止时钟切换, 禁止故障保护时钟监视器 01 = 使能时钟切换, 禁止故障保护时钟监视器 00 = 使能时钟切换, 使能故障保护时钟监视器
IOL1WAY	外设引脚选择配置位 1 = 只允许一次重新配置 0 = 允许多次重新配置
OSCIOFNC	OSC2 引脚功能位 (XT 和 HS 模式除外) 1 = OSC2 为时钟输出 0 = OSC2 为通用数字 I/O 引脚
POSCMD<1:0>	主振荡器模式选择位 11 = 禁止主振荡器 10 = HS 晶振模式 01 = XT 晶振模式 00 = EC (外部时钟) 模式
WDTEN<1:0>	看门狗定时器使能位 11 = 总是使能看门狗定时器 (不能禁止 LPRC 振荡器; 清零 RCON 寄存器中的 SWDTEN 位将不起作用) 10 = 通过用户软件使能 / 禁止看门狗定时器 (可通过清零 RCON 寄存器中的 SWDTEN 位来禁止 LPRC) 01 = 看门狗定时器仅在器件正常工作时使能, 在休眠模式下禁止; 该模式下禁止软件控制 00 = 禁止看门狗定时器和 SWDTEN 位
WINDIS	看门狗定时器窗口使能位 1 = 看门狗定时器处于非窗口模式 0 = 看门狗定时器处于窗口模式
PLLKEN	PLL 锁定使能位 1 = 使能 PLL 锁定 0 = 禁止 PLL 锁定
WDTPRE	看门狗定时器预分频比位 1 = 1:128 0 = 1:32
WDTPOST<3:0>	看门狗定时器后分频比位 1111 = 1:32,768 1110 = 1:16,384 • • • 0001 = 1:2 0000 = 1:1

注 1: 必须存在引导段才能使用备用中断向量表。

# dsPIC33EPXXGS70X/80X 系列

表 27-2: 配置位说明 (续)

位域	说明
WDTWIN<1:0>	看门狗定时器窗口选择位 11 = WDT 窗口为 WDT 周期的 25% 10 = WDT 窗口为 WDT 周期的 37.5% 01 = WDT 窗口为 WDT 周期的 50% 00 = WDT 窗口为 WDT 周期的 75%
ALTI2C1	备用 I2C1 引脚位 1 = I2C1 映射到 SDA1/SCL1 引脚 0 = I2C1 映射到 ASDA1/ASCL1 引脚
ALTI2C2	备用 I2C2 引脚位 1 = I2C2 映射到 SDA2/SCL2 引脚 0 = I2C2 映射到 ASDA2/ASCL2 引脚
JTAGEN	JTAG 使能位 1 = 使能 JTAG 0 = 禁止 JTAG
ICS<1:0>	ICD 通信通道选择位 11 = 通过 PGEC1 和 PGED1 进行通信 10 = 通过 PGEC2 和 PGED2 进行通信 01 = 通过 PGEC3 和 PGED3 进行通信 00 = 保留; 不要使用
DBCC	DACx 输出交叉连接选择位 1 = DAC 输出间无交叉连接 0 = DACOUT1 和 DACOUT2 互连
CTXT1<2:0>	备用工作寄存器组 1 中断优先级 (IPL) 选择位 111 = 保留 110 = 分配 IPL 级别 7 101 = 分配 IPL 级别 6 100 = 分配 IPL 级别 5 011 = 分配 IPL 级别 4 010 = 分配 IPL 级别 3 001 = 分配 IPL 级别 2 000 = 分配 IPL 级别 1
CTXT2<2:0>	备用工作寄存器组 2 中断优先级 (IPL) 选择位 111 = 保留 110 = 分配 IPL 级别 7 101 = 分配 IPL 级别 6 100 = 分配 IPL 级别 5 011 = 分配 IPL 级别 4 010 = 分配 IPL 级别 3 001 = 分配 IPL 级别 2 000 = 分配 IPL 级别 1
CTXT3<2:0>	备用工作寄存器组 3 中断优先级 (IPL) 选择位 111 = 保留 110 = 分配 IPL 级别 7 101 = 分配 IPL 级别 6 100 = 分配 IPL 级别 5 011 = 分配 IPL 级别 4 010 = 分配 IPL 级别 3 001 = 分配 IPL 级别 2 000 = 分配 IPL 级别 1

注 1: 必须存在引导段才能使用备用中断向量表。

# dsPIC33EPXXGS70X/80X 系列

表 27-2: 配置位说明 (续)

位域	说明
CTXT4<2:0>	备用工作寄存器组 4 中断优先级 (IPL) 选择位 111 = 保留 110 = 分配 IPL 级别 7 101 = 分配 IPL 级别 6 100 = 分配 IPL 级别 5 011 = 分配 IPL 级别 4 010 = 分配 IPL 级别 3 001 = 分配 IPL 级别 2 000 = 分配 IPL 级别 1
BTMODE<1:0>	引导模式配置位 11 = 单分区模式 10 = 双分区模式 01 = 受保护双分区模式 00 = 特权双分区模式

注 1: 必须存在引导段才能使用备用中断向量表。



# dsPIC33EPXXXGS70X/80X 系列

## 27.2 器件校准和标识

dsPIC33EPXXXGS70X/80X 系列器件上的 PGAx 和电流源模块需要使用校准数据寄存器来提高模块在宽工作范围内的性能。这些校准寄存器为只读寄存器，存储在配置存储空间中。使能模块之前，必须读取校准数据（通过 TBLPAG 和表读指令）并将其装入相应的 SFR 寄存器中。表 27-3 列出了器件校准地址。

dsPIC33EPXXXGS70X/80X 器件在配置存储空间末尾处附近有两个标识寄存器，分别存储器件 ID（DEVID）和器件版本（DEVREV）。这两个寄存器用于确定器件的掩膜、型号和生产信息。这两个寄存器为只读寄存器，如寄存器 27-1 和寄存器 27-2 所示。

表 27-3: 器件校准地址<sup>(1)</sup>

校准名称	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
PGA1CAL	800E48	—	—	—	—	—	—	—	—	—	—	—	PGA1 校准数据					
PGA2CAL	800E4C	—	—	—	—	—	—	—	—	—	—	—	PGA2 校准数据					
ISRCCAL	800E78	—	—	—	—	—	—	—	—	—	—	—	电流源校准数据					

注 1: 使能模块之前，必须将校准数据复制到相应的 SFR 寄存器中。

# dsPIC33EPXXGS70X/80X 系列

## 寄存器 27-1: DEVID: 器件 ID 寄存器

R	R	R	R	R	R	R	R	R
DEVID<23:16>								
bit 23								bit 16

R	R	R	R	R	R	R	R	R
DEVID<15:8>								
bit 15								bit 8

R	R	R	R	R	R	R	R	R
DEVID<7:0>								
bit 7								bit 0

图注: R = 只读位 U = 未实现位

bit 23-0 DEVID<23:0>: 器件标识符位

## 寄存器 27-2: DEVREV: 器件版本寄存器

R	R	R	R	R	R	R	R	R
DEVREV<23:16>								
bit 23								bit 16

R	R	R	R	R	R	R	R	R
DEVREV<15:8>								
bit 15								bit 8

R	R	R	R	R	R	R	R	R
DEVREV<7:0>								
bit 7								bit 0

图注: R = 只读位 U = 未实现位

bit 23-0 DEVREV<23:0>: 器件版本位

## 27.3 用户 OTP 存储器

dsPIC33EPXXGS70X/80X 系列器件包含 64 字的可一次性编程 (One-Time-Programmable, OTP) 用户存储器, 地址范围为 0x800F80 到 0x800FFC。用户 OTP 字可用于存储校验和、代码版本、产品信息 (如序列号、系统制造日期和制造批号) 和其他特定于应用的信息。这些字只能在编程时写入一次, 在运行时不能写入但可以读取。

## 27.4 片上稳压器

所有 dsPIC33EPXXGS70X/80X 系列器件使用标称值为 1.8V 的电压为其内核数字逻辑供电。对于需要工作在更高的典型电压值 (如 3.3V) 的设计, 这可能会产生冲突。为了简化系统设计, dsPIC33EPXXGS70X/80X 系列中的所有器件均包含一个片上稳压器, 可使器件内核逻辑通过 VDD 工作。

稳压器通过其他 VDD 引脚为内核供电。必须将一个低 ESR (小于 1Ω) 电容 (例如钽电容或陶瓷电容) 连接到 VCAP 引脚 (图 27-1)。这有助于维持稳压器的稳定性。滤波电容的推荐值在第 30.0 节 “电气特性” 的表 30-5 中提供。

**注:** 低 ESR 电容要尽可能靠近 VCAP 引脚放置, 这一点很重要。

## 27.5 欠压复位 (BOR)

欠压复位 (BOR) 模块基于内部参考电压电路, 该电路监视经过稳压的电源电压 VCAP。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毛刺 (例如, 由于电力传输线路不良造成的交流周期波形部分丢失, 或者由于接入大感性负载时电流消耗过大造成电压骤降) 产生。

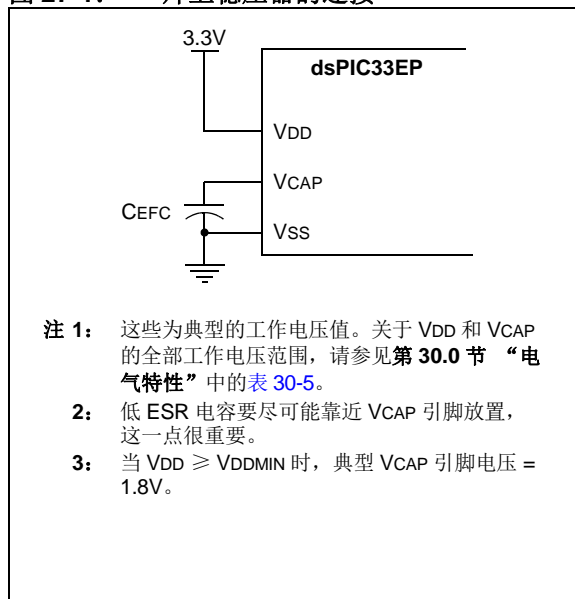
BOR 将产生复位器件的复位脉冲。BOR 会根据器件配置位 (FNOSC<2:0> 和 POSCMD<1:0>) 的值选择时钟源。

如果选择了振荡器模式, BOR 将激活振荡器起振定时器 (Oscillator Start-up Timer, 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 (OSCCON<5>) 为 1。

同时, 将在内部复位释放前应用上电延时定时器 (PWRT) 延时 (TPWRT)。如果 TPWRT = 0 且使用了晶振, 那么会应用 TFSCM 的标称延时, 这种情况下总的延时为 TFSCM。关于具体 TFSCM 值, 请参见第 30.0 节 “电气特性” 的表 30-23 中的参数 SY35。

BOR 状态位 (RCON<1>) 置 1, 指示发生了 BOR。BOR 电路在休眠或空闲模式下继续工作, 当 VDD 下降到 BOR 阈值电压以下时将复位器件。

图 27-1: 片上稳压器的连接 (1,2,3)



# dsPIC33EPXXGS70X/80X 系列

## 27.6 看门狗定时器 (WDT)

对于 dsPIC33EPXXGS70X/80X 系列器件，WDT 由 LPRC 振荡器驱动。当使能 WDT 时，时钟源也将使能。

### 27.6.1 预分频器 / 后分频器

由 LPRC 提供的 WDT 时钟源的频率标称值为 32 kHz。该信号输入给可配置为 5 位 (32 分频) 或 7 位 (128 分频) 工作的预分频器。预分频比由 WDTPRE 配置位设置。使用 32 kHz 输入，预分频器将产生 WDT 超时周期 (TWDT)，如表 30-23 中的参数 SY12 所示。

分频比可变的后分频器对 WDT 预分频器的输出进行分频，以获得范围较宽的超时周期。后分频比由 WDTPOST<3:0> 配置位 (FWDT<3:0>) 控制，这些配置位允许选择 16 种设置，从 1:1 至 1:32,768。使用预分频器和后分频器，可以使超时周期的范围扩展到 1 ms 至 131 秒。

WDT、预分频器和后分频器在以下条件下复位：

- 任何器件复位时
- 在完成时钟切换时，无论时钟切换是由软件（即，在更改 NOSCx 位后将 OSWEN 位置 1）引起还是由硬件（即，故障保护时钟监视器）引起
- 执行 PWRSAV 指令时（即，进入休眠或空闲模式）
- 当器件退出休眠或空闲模式恢复正常工作时
- 当在正常执行过程中执行 CLRWDI 指令时

**注：** 执行 CLRWDI 和 PWRSAV 指令会将预分频器和后分频器的计数值清零。

### 27.6.2 休眠和空闲模式

如果使能了 WDT，它将在休眠或空闲模式下继续运行。当发生 WDT 超时，将唤醒器件并从执行 PWRSAV 指令处继续执行代码。在器件唤醒后，需要用软件将相应的 SLEEP 或 IDLE 位 (RCON<3:2>) 清零。

### 27.6.3 使能 WDT

WDT 的使能或禁止由 FWDT 配置寄存器中的 WDTEN<1:0> 配置位控制。当 WDTEN<1:0> 配置位已编程为 0b11 时，WDT 始终使能。

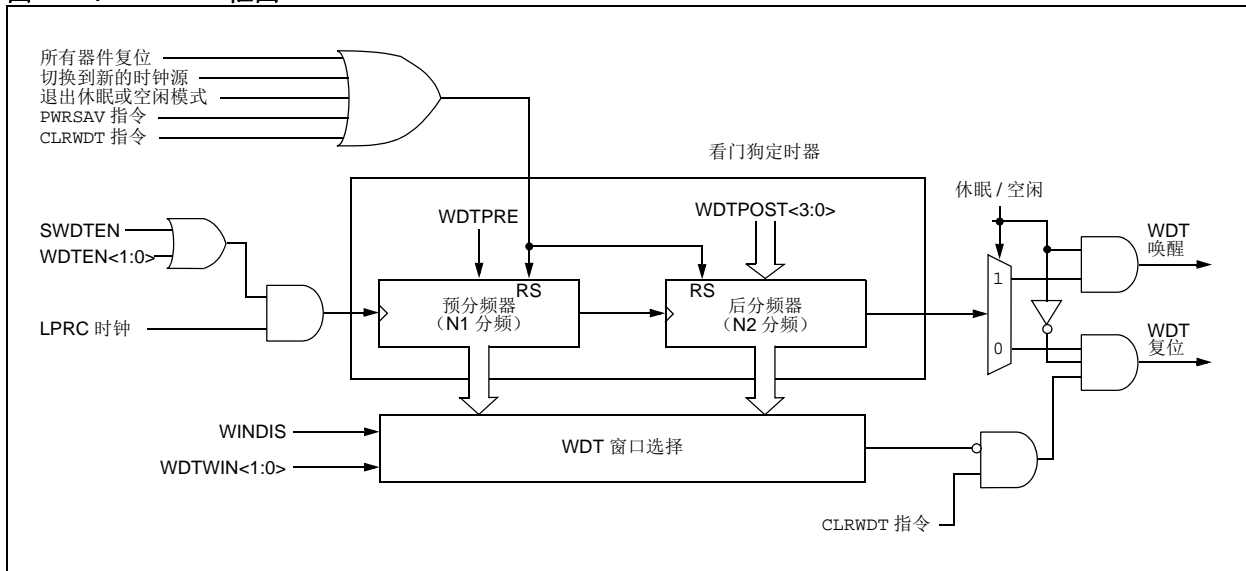
当 WDTEN<1:0> 配置位已编程为 0b10 时，可以选择用软件控制 WDT。通过将 SWDTEN 控制位 (RCON<5>) 置 1 来用软件使能 WDT。任何器件复位都会导致 SWDTEN 控制位清零。软件 WDT 选项允许用户应用程序在关键代码段使能 WDT 并在非关键代码段禁止 WDT，从而最大限度地降低功耗。

WDT 超时标志位 WDT0 (RCON<4>) 不会在 WDT 超时后自动清零。要检测后续的 WDT 事件，必须用软件将该标志位清零。

### 27.6.4 WDT 窗口

看门狗定时器有一种可选的窗口模式，可通过编程 WDT 配置寄存器中的 WINDIS 位 (FWDT<7>) 来使能该模式。在窗口模式 (WINDIS = 0) 下，WDT 应根据可编程看门狗定时器窗口选择位 (WDTWIN<1:0>) 的设置进行清零。

图 27-2: WDT 框图



## 27.7 JTAG 接口

dsPIC33EPXXXGS70X/80X 系列器件实现了一个 JTAG 接口，该接口支持边界扫描器件测试。关于该接口的详细信息将在文档以后的版本中提供。

**注：** 关于 JTAG 接口使用、配置和操作方面的更多信息，请参见《dsPIC33/PIC24 系列参考手册》中的“编程和诊断”（DS70608）。

## 27.8 在线串行编程（ICSP）

可以在最终的应用电路中对 dsPIC33EPXXXGS70X/80X 系列器件进行串行编程。只需要 5 根线即可实现这一操作，其中时钟线、数据线各一根，其余 3 根分别是电源线、地线和编程序列线。串行编程允许用户在生产电路板时使用未编程器件，而仅在产品交付之前才对器件进行编程，从而可以使用最新版本的固件或者定制固件进行编程。关于在线串行编程（ICSP）的详细信息，请参见《带易失性配置位的 dsPIC33E/PIC24E 器件的闪存编程规范》（DS70663A\_CN）。

可使用 3 对编程时钟 / 数据引脚中的任意一对：

- PGEC1 和 PGED1
- PGEC2 和 PGED2
- PGEC3 和 PGED3

## 27.9 在线调试器

当选择 MPLAB® ICD 3 或者 REAL ICE 仿真器作为调试器时，将使能在线调试功能。该功能允许与 MPLAB IDE 配合使用来进行简单的调试。通过 PGECx（仿真 / 调试时钟）和 PGEDx（仿真 / 调试数据）引脚功能控制调试功能。

可使用 3 对调试时钟 / 数据引脚中的任意一对：

- PGEC1 和 PGED1
- PGEC2 和 PGED2
- PGEC3 和 PGED3

要使用器件的在线调试功能，就必须在设计中对 MCLR、VDD、VSS 和 PGECx/PGEDx 引脚对进行 ICSP 连接。此外，当使能该功能时，某些资源就不能用于一般用途了。这些资源包括数据 RAM 的前 80 字节和两个 I/O 引脚（PGECx 和 PGEDx）。

## 27.10 代码保护和 CodeGuard 安全性

dsPIC33EPXXXGS70X/80X 器件提供了多个安全级别，用于保护各方的知识产权。闪存程序存储器保护可分为以下三个段：引导段（Boot Segment, BS）、通用段（General Segment, GS）和配置段（Configuration Segment, CS）。引导段的安全权限最高，可被认为在访问其他段时的限制是有限的。通用段的安全权限最低，用于最终用户的系统代码。配置段仅包含器件的用户配置数据，位于程序存储空间的末尾。

代码保护功能由配置寄存器 FSEC 和 FBSLIM 控制。FSEC 寄存器用于控制各个段的代码保护级别以及是否被写保护。BS 和 GS 的大小将取决于 BSLIM<12:0> 位设置以及是否使能了备用中断向量表（AIVT）。BSLIM<12:0> 位用于定义 BS 的页数，其中每页包含 512 个指令字。最小的 BS 大小为一页，其中包含中断向量表（IVT）和代码保护的 256 个指令字。

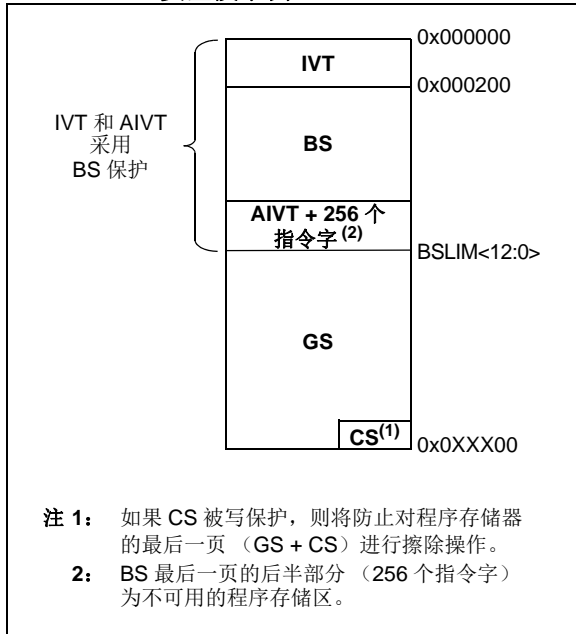
如果使能了 AIVT，则 BS 的最后一页将包含 AIVT，而不包含任何 BS 代码。使能 AIVT 时，最小的 BS 大小为两页（1024 个指令字），其中一页用于 IVT 和 BS 代码，另一页用于 AIVT。BS 的写保护不包含 AIVT。BS 的最后一页始终可通过 BS 代码编程或擦除。通用段将从下一页开始，占用程序闪存中除闪存配置字以外的剩余部分。只有在未使能 BS 时，IVT 才将采用 GS 安全性。当 BS 和 GS 中任一安全段已使能写保护时，将防止对 IVT 进行编程或页擦除操作。

**注：** 关于 CodeGuard 安全性的使用、配置和操作的更多信息，请参见《dsPIC33/PIC24 系列参考手册》中的“CodeGuard™ 中级安全性”（DS70005182）。

# dsPIC33EPXXXGS70X/80X 系列

图 27-3 给出了器件的各个安全段。图中显示了全部三个段，但并非都是必需的。如果只需要基本代码保护，则可以根据需要单独使能 GS 或者组合使能 GS 与 CS。

图 27-3: dsPIC33EPXXXGS70X/80X 器件的安全段示例



dsPIC33EPXXXGS70X/80X 系列器件可在双分区模式下工作，每个分区都需要一定的安全性。在双分区模式下工作时，活动分区和非活动分区均包含复位向量、中断向量表（IVT 和 AIVT（如果已使能））和闪存配置字的唯一副本。两个分区均包含之前所述的三个安全段。代码不能从非活动分区执行，可被活动分区编程和读取，但要受到已定义代码保护的限制。图 27-4 和图 27-5 显示了器件在双分区模式下工作时的各个安全段。

器件也可在受保护双分区模式或特权双分区模式下工作。在受保护双分区模式下，分区 1 具有永久的擦除 / 写保护。该实现可用作“出厂默认”模式，该模式下在分区 1 中存储了故障保护备份映像。例如，可将故障保护自举程序与故障保护备份代码映像一同置于分区 1 中，可在分区 2 的闪存更新失败时使用或重新写入分区 2。

特权双分区模式执行与受保护双分区模式相同的功能，但为了避免引导段与通用段中的代码使用发生冲突，会受到额外限制。

图 27-4: dsPIC33EP64GS70X/80X 器件的安全段示例（双分区模式）

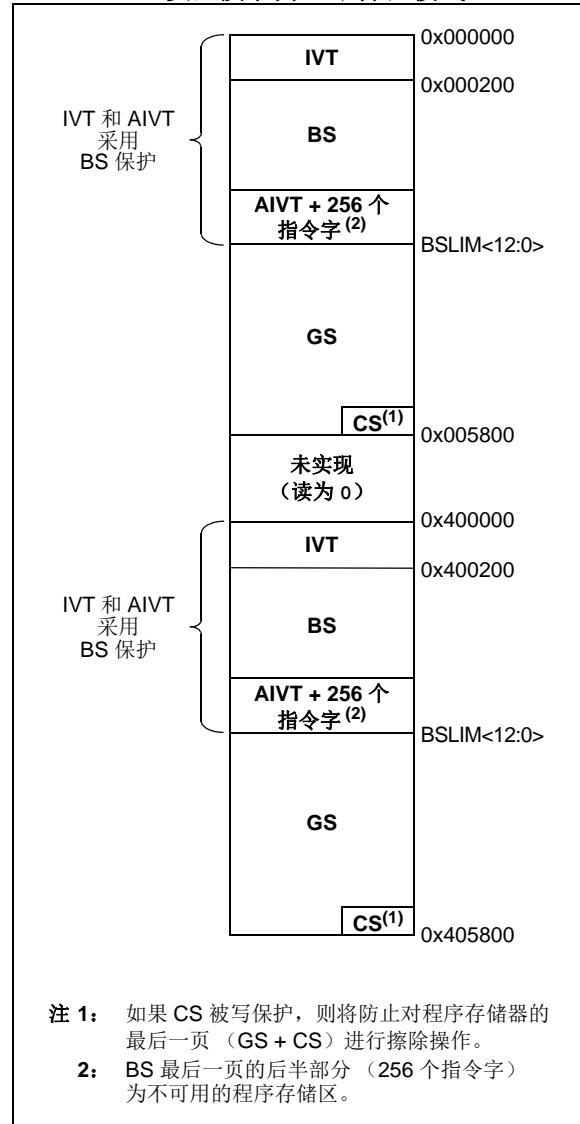
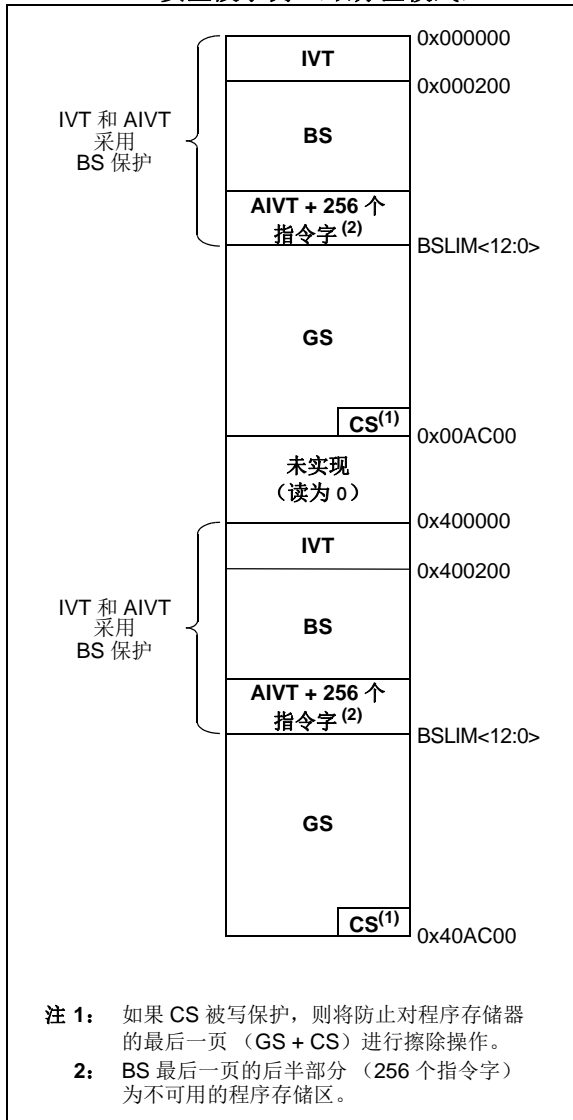


图 27-5: dsPIC33EP128GS70X/80X 器件的安全段示例（双分区模式）



# dsPIC33EPXXXGS70X/80X 系列

---

注:



## 28.0 指令集汇总

**注：** 本数据手册总结了 dsPIC33EPXXGS70X/80X 系列器件的特性。但是不应把本数据手册当作无所不包的参考资料来使用。如需了解本数据手册的补充信息，请参见《dsPIC33/PIC24 系列参考手册》的相关章节，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

dsPIC33EP 系列器件的指令集与 dsPIC30F 和 dsPIC33F 系列器件的指令集几乎完全相同。

大部分指令的长度为一个程序存储字 (24 位)。只有三条指令需要两个程序存储单元。

每一条单字指令长 24 位，分为一个指定指令类型的 8 位操作码和进一步指定指令操作的一个或多个操作数。

指令集是高度正交的，分为 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 28-1 列出了在说明指令时使用的通用符号。

表 28-2 是 dsPIC33E 指令集的汇总，列出了所有指令以及每条指令影响的状态标志。

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
- 保存结果的目标寄存器，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) 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 (由一个立即数值指定，或者由寄存器 Wb 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 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，涉及算术或逻辑运算的立即数指令，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 地址空间预取目标寄存器
- 累加器回写目标寄存器

与乘法无关的其他 DSP 指令使用的操作数可能包括：

- 要使用的累加器 (必需)
- 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：

- 程序存储器地址
- 表读和表写指令的模式

# dsPIC33EPXXGS70X/80X 系列

大多数指令都是单字指令。设计了一些双字指令，用 48 位来提供所需信息。在第二个字中，高 8 位全为 0。如果指令自身将第二个字当作一条指令来执行的话，它将作为一条 NOP 指令来执行。

双字指令执行需要两个指令周期。

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、指令执行结果改变了程序计数器或者执行了 PSV 或表读操作。在这些情况下，指令执行需要多个指

令周期，在额外的指令周期中执行 NOP 指令。某些可能涉及到跳过后续指令的指令，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。

**注：** 关于指令集的更多详细信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157F\_CN)。

**表 28-1: 操作码说明中使用的符号**

字段	说明
#text	表示由 text 定义的立即数
(text)	表示 text 的内容
[text]	表示由 text 寻址的存储单元
{ }	可选字段或操作
a ∈ {b, c, d}	a 从值 b、c 和 d 的集合中选择
<n:m>	寄存器位域
.b	字节模式选择
.d	双字模式选择
.S	影子寄存器选择
.w	字模式选择 (默认)
Acc	两个累加器 {A, B} 之一
AWB	累加器回写目标地址寄存器 ∈ {W13, [W13]+ = 2}
bit4	4 位选择字段 (用于字寻址指令) ∈ {0..15}
C, DC, N, OV, Z	MCU 状态位: 进位、半进位、负、溢出和全零标志位
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 ∈ {0x0000...0x1FFF}
lit1	1 位无符号立即数 ∈ {0,1}
lit4	4 位无符号立即数 ∈ {0...15}
lit5	5 位无符号立即数 ∈ {0...31}
lit8	8 位无符号立即数 ∈ {0...255}
lit10	10 位无符号立即数 ∈ {0...255} (对于字节模式) 或 {0:1023} (对于字模式)
lit14	14 位无符号立即数 ∈ {0...16384}
lit16	16 位无符号立即数 ∈ {0...65535}
lit23	23 位无符号立即数 ∈ {0...8388608}; LSb 必须为 0
无	字段无需内容, 可为空
OA, OB, SA, SB	DSP 状态位: ACCA 溢出、ACCB 溢出、ACCA 饱和和 ACCB 饱和
PC	程序计数器
Slit10	10 位有符号立即数 ∈ {-512...511}
Slit16	16 位有符号立即数 ∈ {-32768...32767}
Slit6	6 位有符号立即数 ∈ {-16...16}
Wb	基本 W 寄存器 ∈ {W0...W15}
Wd	目标 W 寄存器 ∈ {Wd, [Wd], [Wd++], [Wd--], [++Wd], [--Wd]}
Wdo	目标 W 寄存器 ∈ {Wnd, [Wnd], [Wnd++], [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb]}
Wm, Wn	被除数和除数工作寄存器对 (直接寻址)

# dsPIC33EPXXGS70X/80X 系列

表 28-1: 操作码说明中使用的符号 (续)

字段	说明
Wm*Wm	用于平方指令的被乘数和乘数工作寄存器对 $\in \{W4 * W4, W5 * W5, W6 * W6, W7 * W7\}$
Wm*Wn	用于 DSP 指令的被乘数和乘数工作寄存器对 $\in \{W4 * W5, W4 * W6, W4 * W7, W5 * W6, W5 * W7, W6 * W7\}$
Wn	16 个工作寄存器之一 $\in \{W0...W15\}$
Wnd	16 个目标工作寄存器之一 $\in \{W0...W15\}$
Wns	16 个源工作寄存器之一 $\in \{W0...W15\}$
WREG	W0 (文件寄存器指令中使用的工作寄存器)
Ws	源 W 寄存器 $\in \{Ws, [Ws], [Ws++] , [Ws--], [++Ws], [--Ws]\}$
Wso	源 W 寄存器 $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb]\}$
Wx	用于 DSP 指令的 X 数据空间预取地址寄存器 $\in \{[W8] + = 6, [W8] + = 4, [W8] + = 2, [W8], [W8] - = 6, [W8] - = 4, [W8] - = 2, [W9] + = 6, [W9] + = 4, [W9] + = 2, [W9], [W9] - = 6, [W9] - = 4, [W9] - = 2, [W9 + W12], \text{无}\}$
Wxd	用于 DSP 指令的 X 数据空间预取目标寄存器 $\in \{W4...W7\}$
Wy	用于 DSP 指令的 Y 数据空间预取地址寄存器 $\in \{[W10] + = 6, [W10] + = 4, [W10] + = 2, [W10], [W10] - = 6, [W10] - = 4, [W10] - = 2, [W11] + = 6, [W11] + = 4, [W11] + = 2, [W11], [W11] - = 6, [W11] - = 4, [W11] - = 2, [W11 + W12], \text{无}\}$
Wyd	用于 DSP 指令的 Y 数据空间预取目标寄存器 $\in \{W4...W7\}$

# dsPIC33EPXXGS70X/80X 系列

表 28-2: 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 <sup>(1)</sup>	影响的状态标志
1	ADD	ADD Acc	累加器相加	1	1	OA,OB,SA,SB
		ADD f	$f = f + WREG$	1	1	C,DC,N,OV,Z
		ADD f, WREG	$WREG = f + WREG$	1	1	C,DC,N,OV,Z
		ADD #lit10, Wn	$Wd = lit10 + Wd$	1	1	C,DC,N,OV,Z
		ADD Wb, Ws, Wd	$Wd = Wb + Ws$	1	1	C,DC,N,OV,Z
		ADD Wb, #lit5, Wd	$Wd = Wb + lit5$	1	1	C,DC,N,OV,Z
		ADD Wso, #Slit4, Acc	将 16 位有符号立即数加到累加器	1	1	OA,OB,SA,SB
2	ADDC	ADDC f	$f = f + WREG + (C)$	1	1	C,DC,N,OV,Z
		ADDC f, WREG	$WREG = f + WREG + (C)$	1	1	C,DC,N,OV,Z
		ADDC #lit10, Wn	$Wd = lit10 + Wd + (C)$	1	1	C,DC,N,OV,Z
		ADDC Wb, Ws, Wd	$Wd = Wb + Ws + (C)$	1	1	C,DC,N,OV,Z
		ADDC Wb, #lit5, Wd	$Wd = Wb + lit5 + (C)$	1	1	C,DC,N,OV,Z
3	AND	AND f	$f = f .AND. WREG$	1	1	N,Z
		AND f, WREG	$WREG = f .AND. WREG$	1	1	N,Z
		AND #lit10, Wn	$Wd = lit10 .AND. Wd$	1	1	N,Z
		AND Wb, Ws, Wd	$Wd = Wb .AND. Ws$	1	1	N,Z
		AND Wb, #lit5, Wd	$Wd = Wb .AND. lit5$	1	1	N,Z
4	ASR	ASR f	$f =$ 算术右移 $f$	1	1	C,N,OV,Z
		ASR f, WREG	$WREG =$ 算术右移 $f$	1	1	C,N,OV,Z
		ASR Ws, Wd	$Wd =$ 算术右移 $Ws$	1	1	C,N,OV,Z
		ASR Wb, Wns, Wnd	$Wnd =$ 将 $Wb$ 算术右移 $Wns$ 位	1	1	N,Z
		ASR Wb, #lit5, Wnd	$Wnd =$ 将 $Wb$ 算术右移 $lit5$ 位	1	1	N,Z
5	BCLR	BCLR f, #bit4	将 $f$ 中的指定位清零	1	1	无
		BCLR Ws, #bit4	将 $Ws$ 中的指定位清零	1	1	无
6	BOOTSWP	BOOTSWP	交换活动和非活动闪存程序空间	1	2	无
7	BRA	BRA C, Expr	如果有进位则跳转	1	1 (4)	无
		BRA GE, Expr	如果大于或等于则跳转	1	1 (4)	无
		BRA GEU, Expr	如果无符号大于或等于则跳转	1	1 (4)	无
		BRA GT, Expr	如果大于则跳转	1	1 (4)	无
		BRA GTU, Expr	如果无符号大于则跳转	1	1 (4)	无
		BRA LE, Expr	如果小于或等于则跳转	1	1 (4)	无
		BRA LEU, Expr	如果无符号小于或等于则跳转	1	1 (4)	无
		BRA LT, Expr	如果小于则跳转	1	1 (4)	无
		BRA LTU, Expr	如果无符号小于则跳转	1	1 (4)	无
		BRA N, Expr	如果为负则跳转	1	1 (4)	无
		BRA NC, Expr	如果没有进位则跳转	1	1 (4)	无
		BRA NN, Expr	如果不为负则跳转	1	1 (4)	无
		BRA NOV, Expr	如果未溢出则跳转	1	1 (4)	无
		BRA NZ, Expr	如果不为零则跳转	1	1 (4)	无
		BRA OA, Expr	如果累加器 A 溢出则跳转	1	1 (4)	无
		BRA OB, Expr	如果累加器 B 溢出则跳转	1	1 (4)	无
		BRA OV, Expr	如果溢出则跳转	1	1 (4)	无
		BRA SA, Expr	如果累加器 A 饱和则跳转	1	1 (4)	无
		BRA SB, Expr	如果累加器 B 饱和则跳转	1	1 (4)	无
		BRA Expr	无条件跳转	1	4	无
BRA Z, Expr	如果为零则跳转	1	1 (4)	无		
BRA Wn	计算跳转	1	4	无		
8	BSET	BSET f, #bit4	将 $f$ 中的指定位置 1	1	1	无
		BSET Ws, #bit4	将 $Ws$ 中的指定位置 1	1	1	无

注 1: 对非 CPU SFR 执行读操作和读 - 修改 - 写操作 (例如, 位操作和逻辑运算) 时需要一个额外的指令周期。

# dsPIC33EPXXGS70X/80X 系列

表 28-2: 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 <sup>(1)</sup>	影响的状态标志
9	BSW	BSW.C Ws,Wb	将 C 位内容写入 Ws<Wb>	1	1	无
		BSW.Z Ws,Wb	将 Z 位内容取反写入 Ws<Wb>	1	1	无
10	BTG	BTG f,#bit4	将 f 中的指定位取反	1	1	无
		BTG Ws,#bit4	将 Ws 中的指定位取反	1	1	无
11	BTSC	BTSC f,#bit4	对 f 中的指定位进行测试, 如果为零则跳过	1	1 (2 或 3)	无
		BTSC Ws,#bit4	对 Ws 中的指定位进行测试, 如果为零则跳过	1	1 (2 或 3)	无
12	BTSS	BTSS f,#bit4	对 f 中的指定位进行测试, 如果为 1 则跳过	1	1 (2 或 3)	无
		BTSS Ws,#bit4	对 Ws 中的指定位进行测试, 如果为 1 则跳过	1	1 (2 或 3)	无
13	BTST	BTST f,#bit4	对 f 中的指定位进行测试	1	1	Z
		BTST.C Ws,#bit4	对 Ws 中的指定位进行测试, 并将其值存储到 C	1	1	C
		BTST.Z Ws,#bit4	对 Ws 中的指定位进行测试, 并将其反码存储到 Z	1	1	Z
		BTST.C Ws,Wb	对 Ws<Wb> 位进行测试, 并将其值存储到 C	1	1	C
		BTST.Z Ws,Wb	对 Ws<Wb> 位进行测试, 并将其反码存储到 Z	1	1	Z
14	BTSTS	BTSTS f,#bit4	对 f 中的指定位进行测试, 并将 f 中的该位置 1	1	1	Z
		BTSTS.C Ws,#bit4	对 Ws 中的指定位进行测试, 并将其值存储到 C, 然后将 Ws 中的该位置 1	1	1	C
		BTSTS.Z Ws,#bit4	对 Ws 中的指定位进行测试, 并将其反码存储到 Z, 然后将 Ws 中的该位置 1	1	1	Z
15	CALL	CALL lit23	调用子程序	2	4	SFA
		CALL Wn	间接调用子程序	1	4	SFA
		CALL.L Wn	间接调用子程序 (长地址)	1	4	SFA
16	CLR	CLR f	f = 0x0000	1	1	无
		CLR WREG	WREG = 0x0000	1	1	无
		CLR Ws	Ws = 0x0000	1	1	无
		CLR Acc,Wx,Wxd,Wy,Wyd,AWB	将累加器清零	1	1	OA,OB,SA,SB
17	CLRWDT	CLRWDT	将看门狗定时器清零	1	1	WDTO,Sleep
18	COM	COM f	f = $\bar{f}$	1	1	N,Z
		COM f,WREG	WREG = $\bar{f}$	1	1	N,Z
		COM Ws,Wd	Wd = $\bar{W}s$	1	1	N,Z
19	CP	CP f	比较 f 和 WREG	1	1	C,DC,N,OV,Z
		CP Wb,#lit8	比较 Wb 和 lit8	1	1	C,DC,N,OV,Z
		CP Wb,Ws	比较 Wb 和 Ws (Wb - Ws)	1	1	C,DC,N,OV,Z
20	CP0	CP0 f	比较 f 和 0x0000	1	1	C,DC,N,OV,Z
		CP0 Ws	比较 Ws 和 0x0000	1	1	C,DC,N,OV,Z
21	CPB	CPB f	带借位比较 f 和 WREG	1	1	C,DC,N,OV,Z
		CPB Wb,#lit8	带借位比较 Wb 和 lit8	1	1	C,DC,N,OV,Z
		CPB Wb,Ws	带借位比较 Wb 和 Ws (Wb - Ws - C)	1	1	C,DC,N,OV,Z
22	CPSEQ	CPSEQ Wb,Wn	比较 Wb 和 Wn, 如果相等则跳过	1	1 (2 或 3)	无
	CPBEQ	CPBEQ Wb,Wn,Expr	比较 Wb 和 Wn, 如果相等则跳转	1	1 (5)	无
23	CPSGT	CPSGT Wb,Wn	比较 Wb 和 Wn, 如果大于则跳过	1	1 (2 或 3)	无
	CPBGT	CPBGT Wb,Wn,Expr	比较 Wb 和 Wn, 如果大于则跳转	1	1 (5)	无
24	CPSLT	CPSLT Wb,Wn	比较 Wb 和 Wn, 如果小于则跳过	1	1 (2 或 3)	无
	CPBLT	CPBLT Wb,Wn,Expr	比较 Wb 和 Wn, 如果小于则跳转	1	1 (5)	无
25	CPSNE	CPSNE Wb,Wn	比较 Wb 和 Wn, 如果不相等则跳过	1	1 (2 或 3)	无
	CPBNE	CPBNE Wb,Wn,Expr	比较 Wb 和 Wn, 如果不相等则跳转	1	1 (5)	无

注 1: 对非 CPU SFR 执行读操作和读 - 修改 - 写操作 (例如, 位操作和逻辑运算) 时需要一个额外的指令周期。

# dsPIC33EPXXGS70X/80X 系列

表 28-2: 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 <sup>(1)</sup>	影响的状态标志
26	CTXTSWP	CTXTSWP #lit3	将 CPU 寄存器现场切换至 lit3 定义的现场	1	2	无
		CTXTSWP Wn	将 CPU 寄存器现场切换至 Wn 定义的现场	1	2	无
27	DAW	DAW Wn	Wn = 十进制调整 Wn	1	1	C
28	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
29	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
30	DISI	DISI #lit14	在 k 个指令周期内禁止中断	1	1	无
31	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
32	DIVF	DIVF Wm, Wn	有符号 16/16 位小数除法	1	18	N,Z,C,OV
33	DO	DO #lit15, Expr	执行 DO 循环代码到 PC + Expr, 执行次数为 lit15 + 1 次	2	2	无
		DO Wn, Expr	执行 DO 循环代码到 PC + Expr, 执行次数为 (Wn) + 1 次	2	2	无
34	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
35	EDAC	EDAC Wm*Wm, Acc, Wx, Wy, Wxd	欧几里德距离	1	1	OA,OB,OAB,SA,SB,SAB
36	EXCH	EXCH Wns, Wnd	交换 Wns 和 Wnd 的内容	1	1	无
37	FBCL	FBCL Ws, Wnd	从左边 (MSb) 开始查找第一个位变化	1	1	C
38	FF1L	FF1L Ws, Wnd	从左边 (MSb) 开始查找第一个 1	1	1	C
39	FF1R	FF1R Ws, Wnd	从右边 (LSb) 开始查找第一个 1	1	1	C
40	GOTO	GOTO Expr	跳转到地址	2	4	无
		GOTO Wn	间接跳转到地址	1	4	无
		GOTO.L Wn	间接跳转到地址 (长地址)	1	4	无
41	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
42	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
43	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
44	LAC	LAC Wso, #S1it4, Acc	装载累加器	1	1	OA,OB,OAB,SA,SB,SAB
45	LNK	LNK #lit14	分配堆栈帧	1	1	SFA
46	LSR	LSR f	$f =$ 逻辑右移 f	1	1	C,N,OV,Z
		LSR f, WREG	$WREG =$ 逻辑右移 f	1	1	C,N,OV,Z
		LSR Ws, Wd	$Wd =$ 逻辑右移 Ws	1	1	C,N,OV,Z
		LSR Wb, Wns, Wnd	Wnd = 将 Wb 逻辑右移 Wns 位	1	1	N,Z
		LSR Wb, #lit5, Wnd	Wnd = 将 Wb 逻辑右移 lit5 位	1	1	N,Z
47	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

注 1: 对非 CPU SFR 执行读操作和读 - 修改 - 写操作 (例如, 位操作和逻辑运算) 时需要一个额外的指令周期。

# dsPIC33EPXXGS70X/80X 系列

**表 28-2: 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 <sup>(1)</sup>	影响的状态标志
48	MOV	MOV f, Wn	将 f 中的内容送入 Wn	1	1	无
		MOV f	将 f 中的内容送入 f	1	1	无
		MOV f, WREG	将 f 中的内容送入 WREG	1	1	无
		MOV #lit16, Wn	将 16 位立即数送入 Wn	1	1	无
		MOV.b #lit8, Wn	将 8 位立即数送入 Wn	1	1	无
		MOV Wn, f	将 Wn 中的内容送入 f	1	1	无
		MOV Wso, Wdo	将 Ws 中的内容送入 Wd	1	1	无
		MOV WREG, f	将 WREG 中的内容送入 f	1	1	无
		MOV.D Wns, Wd	将 W(ns):W(ns + 1) 中的双字内容送入 Wd	1	2	无
MOV.D Ws, Wnd	将 Ws 中的双字内容送入 W(nd + 1):W(nd)	1	2	无		
49	MOVPAG	MOVPAG #lit10, DSRPAG	将 10 位立即数送入 DSRPAG	1	1	无
		MOVPAG #lit8, TBLPAG	将 8 位立即数送入 TBLPAG	1	1	无
		MOVPAGW Ws, DSRPAG	将 Ws<9:0> 中的内容送入 DSRPAG	1	1	无
		MOVPAGW Ws, TBLPAG	将 Ws<7:0> 中的内容送入 TBLPAG	1	1	无
50	MOVSAC	MOVSAC Acc, Wx, Wxd, Wy, Wyd, AWB	预取操作数并保存累加器	1	1	无
51	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
52	MPY.N	MPY.N Wm*Wn, Acc, Wx, Wxd, Wy, Wyd	Wm 与 Wn 相乘并取相反数, 结果存入累加器	1	1	无
53	MSC	MSC Wm*Wm, Acc, Wx, Wxd, Wy, Wyd, AWB	相乘再从累加器中减去	1	1	OA,OB,OAB,SA,SB,SAB
54	MUL	MUL.SS Wb, Ws, Wnd	{Wnd + 1, Wnd} = 有符号 (Wb) * 有符号 (Ws)	1	1	无
		MUL.SS Wb, Ws, Acc	累加器 = 有符号 (Wb) * 有符号 (Ws)	1	1	无
		MUL.SU Wb, Ws, Wnd	{Wnd + 1, Wnd} = 有符号 (Wb) * 无符号 (Ws)	1	1	无
		MUL.SU Wb, Ws, Acc	累加器 = 有符号 (Wb) * 无符号 (Ws)	1	1	无
		MUL.SU Wb, #lit5, Acc	累加器 = 有符号 (Wb) * 无符号 (lit5)	1	1	无
		MUL.US Wb, Ws, Wnd	{Wnd + 1, Wnd} = 无符号 (Wb) * 有符号 (Ws)	1	1	无
		MUL.US Wb, Ws, Acc	累加器 = 无符号 (Wb) * 有符号 (Ws)	1	1	无
		MUL.UU Wb, Ws, Wnd	{Wnd + 1, Wnd} = 无符号 (Wb) * 无符号 (Ws)	1	1	无
		MUL.UU Wb, #lit5, Acc	累加器 = 无符号 (Wb) * 无符号 (lit5)	1	1	无
		MUL.UU Wb, Ws, Acc	累加器 = 无符号 (Wb) * 无符号 (Ws)	1	1	无
		MULW.SS Wb, Ws, Wnd	Wnd = 有符号 (Wb) * 有符号 (Ws)	1	1	无
		MULW.SU Wb, Ws, Wnd	Wnd = 有符号 (Wb) * 无符号 (Ws)	1	1	无
		MULW.US Wb, Ws, Wnd	Wnd = 无符号 (Wb) * 有符号 (Ws)	1	1	无
		MULW.UU Wb, Ws, Wnd	Wnd = 无符号 (Wb) * 无符号 (Ws)	1	1	无
		MUL.SU Wb, #lit5, Wnd	{Wnd + 1, Wnd} = 有符号 (Wb) * 无符号 (lit5)	1	1	无
		MUL.SU Wb, #lit5, Wnd	Wnd = 有符号 (Wb) * 无符号 (lit5)	1	1	无
		MUL.UU Wb, #lit5, Wnd	{Wnd + 1, Wnd} = 无符号 (Wb) * 无符号 (lit5)	1	1	无
		MUL.UU Wb, #lit5, Wnd	Wnd = 无符号 (Wb) * 无符号 (lit5)	1	1	无
		MUL f	W3:W2 = f * WREG	1	1	无

注 1: 对非 CPU SFR 执行读操作和读 - 修改 - 写操作 (例如, 位操作和逻辑运算) 时需要一个额外的指令周期。

# dsPIC33EPXXGS70X/80X 系列

表 28-2: 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 <sup>(1)</sup>	影响的状态标志
55	NEG	NEG Acc	将累加器内容求补	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	$Wd = \overline{Ws} + 1$	1	1	C,DC,N,OV,Z
56	NOP	NOP	空操作	1	1	无
		NOPR	空操作	1	1	无
57	POP	POP f	将栈顶 (Top-of-Stack, TOS) 的内容弹出到 f	1	1	无
		POP Wdo	将栈顶 (TOS) 的内容弹出到 Wdo	1	1	无
		POP.D Wnd	将栈顶 (TOS) 的内容弹出到 W(nd):W(nd + 1)	1	2	无
		POP.S	将影子寄存器的内容弹出到主寄存器	1	1	全部
58	PUSH	PUSH f	将 f 的内容压入栈顶 (TOS)	1	1	无
		PUSH Wso	将 Wso 的内容压入栈顶 (TOS)	1	1	无
		PUSH.D Wns	将 W(ns):W(ns + 1) 的内容压入栈顶 (TOS)	1	2	无
		PUSH.S	将主寄存器中的内容压入影子寄存器	1	1	无
59	PWRSV	PWRSV #lit1	进入休眠或空闲模式	1	1	WDT0,Sleep
60	RCALL	RCALL Expr	相对调用	1	4	SFA
		RCALL Wn	计算调用	1	4	SFA
61	REPEAT	REPEAT #lit15	将下一条指令重复执行 lit15 + 1 次	1	1	无
		REPEAT Wn	将下一条指令重复执行 (Wn) + 1 次	1	1	无
62	RESET	RESET	软件器件复位	1	1	无
63	RETFIE	RETFIE	从中断返回	1	6 (5)	SFA
64	RETLW	RETLW #lit10, Wn	返回并将立即数存入 Wn	1	6 (5)	SFA
65	RETURN	RETURN	从子程序返回	1	6 (5)	SFA
66	RLC	RLC f	f = 对 f 执行带进位的循环左移	1	1	C,N,Z
		RLC f, WREG	WREG = 对 f 执行带进位的循环左移	1	1	C,N,Z
		RLC Ws, Wd	Wd = 对 Ws 执行带进位的循环左移	1	1	C,N,Z
67	RLNC	RLNC f	f = 循环左移 f (不带进位)	1	1	N,Z
		RLNC f, WREG	WREG = 循环左移 f (不带进位)	1	1	N,Z
		RLNC Ws, Wd	Wd = 循环左移 Ws (不带进位)	1	1	N,Z
68	RRC	RRC f	f = 对 f 执行带进位的循环右移	1	1	C,N,Z
		RRC f, WREG	WREG = 对 f 执行带进位的循环右移	1	1	C,N,Z
		RRC Ws, Wd	Wd = 对 Ws 执行带进位的循环右移	1	1	C,N,Z
69	RRNC	RRNC f	f = 循环右移 f (不带进位)	1	1	N,Z
		RRNC f, WREG	WREG = 循环右移 f (不带进位)	1	1	N,Z
		RRNC Ws, Wd	Wd = 循环右移 Ws (不带进位)	1	1	N,Z
70	SAC	SAC Acc, #Slit4, Wdo	保存累加器内容	1	1	无
		SAC.R Acc, #Slit4, Wdo	保存舍入后的累加器内容	1	1	无
71	SE	SE Ws, Wnd	Wnd = 符号扩展后的 Ws	1	1	C,N,Z
72	SETM	SETM f	f = 0xFFFF	1	1	无
		SETM WREG	WREG = 0xFFFF	1	1	无
		SETM Ws	Ws = 0xFFFF	1	1	无
73	SFTAC	SFTAC Acc, Wn	对累加器算术移位 (Wn) 次	1	1	OA,OB,OAB,SA,SB,SAB
		SFTAC Acc, #Slit6	对累加器算术移位 Slit6 次	1	1	OA,OB,OAB,SA,SB,SAB

注 1: 对非 CPU SFR 执行读操作和读 - 修改 - 写操作 (例如, 位操作和逻辑运算) 时需要一个额外的指令周期。



# dsPIC33EPXXGS70X/80X 系列

表 28-2: 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 <sup>(1)</sup>	影响的状态标志
74	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 = 将 Wb 左移 Wns 位	1	1	N,Z
		SL Wb, #lit5, Wnd	Wnd = 将 Wb 左移 lit5 位	1	1	N,Z
75	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		
76	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 - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB Wb, Ws, Wd	Wd = Wb - Ws - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB Wb, #lit5, Wd	Wd = Wb - lit5 - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
77	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
78	SUBBR	SUBBR f	f = WREG - f - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR f, WREG	WREG = WREG - f - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR Wb, Ws, Wd	Wd = Ws - Wb - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR Wb, #lit5, Wd	Wd = lit5 - Wb - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
79	SWAP	SWAP.b Wn	Wn = 半字节交换 Wn 内容	1	1	无
		SWAP Wn	Wn = 将 Wn 的两个字节相交换	1	1	无
80	TBLRDH	TBLRDH Ws, Wd	将程序存储器中某个单元的 bit<23:16> 读入 Wd<7:0>	1	5	无
81	TBLRDL	TBLRDL Ws, Wd	将程序存储器中某个单元的 bit<15:0> 读入 Wd	1	5	无
82	TBLWTH	TBLWTH Ws, Wd	将 Ws<7:0> 写入程序存储器中某个单元的 bit<23:16>	1	2	无
83	TBLWTL	TBLWTL Ws, Wd	将 Ws 写入程序存储器中某个单元的 bit<15:0>	1	2	无
84	ULNK	ULNK	释放堆栈帧	1	1	SFA
85	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
86	ZE	ZE Ws, Wnd	Wnd = 零扩展后的 Ws	1	1	C,Z,N

注 1: 对非 CPU SFR 执行读操作和读 - 修改 - 写操作 (例如, 位操作和逻辑运算) 时需要一个额外的指令周期。

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 29.0 开发支持

一系列软件及硬件开发工具对 PIC® 单片机 (MCU) 和 dsPIC® 数字信号控制器 (DSC) 提供支持:

- 集成开发环境
  - MPLAB® X IDE 软件
- 编译器 / 汇编器 / 链接器
  - MPLAB XC 编译器
  - MPASM™ 汇编器
  - MPLINK™ 目标链接器 / MPLIB™ 目标库管理器
  - 适用于各种器件系列的 MPLAB 汇编器 / 链接器 / 库管理器
- 模拟器
  - MPLAB X SIM 软件模拟器
- 仿真器
  - MPLAB REAL ICE™ 在线仿真器
- 在线调试器 / 编程器
  - MPLAB ICD 3
  - PICKit™ 3
- 器件编程器
  - MPLAB PM3 器件编程器
- 低成本演示 / 开发板、评估工具包及入门工具包
- 第三方开发工具

## 29.1 MPLAB X 集成开发环境软件

MPLAB X IDE 是适用于 Microchip 和第三方软硬件开发工具统一的通用图形用户界面, 可以在 Windows®、Linux 和 Mac OS® X 上运行。MPLAB X IDE 是一款全新的 IDE, 它基于 NetBeans IDE, 包含许多免费的软件组件和插件, 适用于高性能的应用程序开发和调试。通过这一无缝交互的用户界面, 在不同工具之间的迁移以及从软件模拟器到硬件调试和编程工具的升级都变得极为简便。

MPLAB X IDE 具有完善的项目管理、可视化的调用图、可配置的观察窗口以及包含代码补全功能和上下文菜单的功能丰富编辑器, 因此对于新用户来说非常灵活和友好。MPLAB X IDE 支持对多个项目使用多个工具和同时调试, 因此也完全可以满足经验丰富用户的需求。

功能丰富的编辑器:

- 彩色高亮显示语法
- 智能代码补全功能, 在输入代码时提供建议和提示
- 基于用户定义规则, 代码自动格式化
- 即时解析

用户友好的可定制界面:

- 完全可定制界面: 工具栏、工具栏图标、窗口和窗口放置等
- 调用图窗口

基于项目的工作空间:

- 多个项目
- 多个工具
- 多种配置
- 同时调试会话

文件历史和错误跟踪:

- 本地文件历史功能
- 内建对 Bugzilla 缺陷跟踪系统的支持

## 29.2 MPLAB XC 编译器

MPLAB XC编译器是适用于Microchip所有8位、16位和32位MCU以及DSC器件的完全ANSI C编译器。这些编译器提供强大的集成功能以及出色的代码优化功能，且易于使用。MPLAB XC编译器可在Windows、Linux或Mac OS X上运行。

为方便进行源代码级调试，编译器提供了已针对MPLAB X IDE优化的调试信息。

MPLAB XC编译器的免费版支持所有器件和命令，没有时间或存储容量限制，且为大多数应用程序提供了充分的代码优化。

MPLAB XC编译器包含汇编器、链接器和实用程序。汇编器生成可重定位目标文件，然后通过链接器将生成的可重定位目标文件与其他可重定位目标文件或归档文件归档或链接在一起，进而生成可执行文件。MPLAB XC编译器使用汇编器来生成目标文件。汇编器具有如下突出特性：

- 支持全部器件指令集
- 支持定点和浮点数据
- 命令行接口
- 丰富的伪指令集
- 灵活的宏语言
- 与MPLAB X IDE兼容

## 29.3 MPASM 汇编器

MPASM汇编器是全功能通用宏汇编器，适用于PIC10/12/16/18 MCU。

MPASM汇编器可生成用于MPLINK目标链接器的可重定位目标文件、Intel®标准HEX文件、详细描述存储器使用状况和符号参考的MAP文件、包含源代码行及生成机器码的绝对LST文件以及用于调试的COFF文件。

MPASM汇编器具有如下特性：

- 集成在MPLAB X 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

## 29.4 MPLINK 目标链接器 / MPLIB 目标库管理器

MPLINK目标链接器组合由MPASM汇编器生成的可重定位目标文件。通过使用链接器脚本中的伪指令，它还可链接预编译库中的可重定位目标文件。

MPLIB目标库管理器管理预编译代码库文件的创建和修改。当从源文件调用库中的一段子程序时，只有包含此子程序的模块被链接到应用程序。这样可使大型库在许多不同应用中被高效地利用。

目标链接器/库管理器具有如下特性：

- 高效地连接单个的库而不是许多小文件
- 通过将相关的模块组合在一起增强代码的可维护性
- 只要列出、替换、删除和抽取模块，便可灵活地创建库

## 29.5 适用于各种器件系列的 MPLAB 汇编器、链接器和库管理器

MPLAB汇编器为PIC24和PIC32 MCU以及dsPIC DSC器件从符号汇编语言生成可重定位机器码。MPLAB XC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，或与其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特性：

- 支持整个器件指令集
- 支持定点和浮点数据
- 命令行接口
- 丰富的指令集
- 与MPLAB X IDE兼容

## 29.6 MPLAB X SIM 软件模拟器

MPLAB X SIM 软件模拟器通过在指令级对 PIC MCU 和 dsPIC DSC 进行模拟，可在 PC 主机环境下进行代码开发。对于任何给定的指令，都可以对数据区进行检查或修改，并通过一个全面的激励控制器来施加激励。可以将各寄存器记录在文件中，以便进行进一步的运行时分析。跟踪缓冲区和逻辑分析器的显示使软件模拟器还能记录和跟踪程序的执行、I/O 的动作、大部分的外设及内部寄存器。

MPLAB X SIM 软件模拟器完全支持使用 MPLAB XC 编译器以及 MPASM 和 MPLAB 汇编器的符号调试。该软件模拟器可用于在硬件实验室环境外灵活地开发和调试代码，是一款完美且经济的软件开发工具。

## 29.7 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 器件推出的新一代高速仿真器。结合 MPLAB X IDE 易于使用且功能强大的图形用户界面，该仿真器可对所有 8 位、16 位和 32 位 MCU 及 DSC 器件进行调试和编程。

该仿真器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与在线调试器系统兼容的连接器和新型抗噪声、高速低压差分信号 (LVDS) 互连电缆 (CAT5) 与目标板相连。

可通过 MPLAB X IDE 下载将来版本的固件，对该仿真器进行现场升级。在同类仿真器中，MPLAB REAL ICE 的优势十分明显：全速仿真、运行时变量观察、跟踪分析、复杂断点、逻辑探针、耐用的探针接口及较长（长达 3 米）的互连电缆。

## 29.8 MPLAB ICD 3 在线调试器系统

MPLAB ICD 3 在线调试器系统是 Microchip 成本效益最高的高速硬件调试器 / 编程器，适用于 Microchip 的闪存 DSC 和 MCU 器件。结合 MPLAB X IDE 功能强大但易于使用的图形用户界面，该调试器可对 PIC 闪存单片机和 dsPIC DSC 进行调试和编程。

MPLAB ICD 3 在线调试器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与 MPLAB ICD 2 或 MPLAB REAL ICE 系统兼容的连接器和目标板相连。MPLAB ICD 3 支持所有 MPLAB ICD 2 连接器。

## 29.9 PICKit 3 在线调试器 / 编程器

结合 MPLAB X IDE 功能强大的图形用户界面，MPLAB PICKit 3 可对 PIC 闪存单片机和 dsPIC 数字信号控制器进行调试和编程，且价位较低。MPLAB PICKit 3 通过全速 USB 接口与设计工程师的 PC 相连，并利用 Microchip 调试连接器 (RJ-11) (与 MPLAB ICD 3 和 MPLAB REAL ICE 兼容) 与目标板相连。连接器使用两个器件 I/O 引脚和复位线来实现在线调试和在线串行编程 (ICSP)。

PICKit 3 Debug Express 包括 PICKit 3、演示板和单片机、连接电缆和光盘 (内含用户指南、课程、教程、编译器 and MPLAB IDE 软件)。

## 29.10 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款符合 CE 规范的通用器件编程器，在 VDDMIN 和 VDDMAX 点对其可编程电压进行校验以确保可靠性最高。它有一个用来显示菜单和错误消息的大 LCD 显示器 (128 x 64)，以及一个支持各种封装类型的可拆卸模块化插槽装置。编程器标准配置中带有一根 ICSP 电缆。在单机模式下，MPLAB PM3 器件编程器不必与 PC 相连即可对 PIC MCU 和 dsPIC DSC 器件进行读取、校验和编程。在该模式下它还可设置代码保护。MPLAB PM3 通过 RS-232 或 USB 电缆连接到 PC 主机上。MPLAB PM3 具备高速通信能力以及优化算法，可对具有大存储器的器件进行快速编程。它还包含了 MMC 卡，用于文件存储及数据应用。

# dsPIC33EPXXGS70X/80X 系列

---

## 29.11 演示 / 开发板、评估工具包及入门工具包

有许多演示、开发和评估板可用于各种 PIC MCU 和 dsPIC DSC，实现对全功能系统的快速应用开发。大多数的演示、开发和评估板都有实验布线区，供用户添加定制电路；还有应用固件和源代码，用于检查和修改。

这些板支持多种功能部件，包括 LED、温度传感器、开关、扬声器、RS-232 接口、LCD 显示器、电位计和附加 EEPROM 存储器。

演示和开发板可用于教学环境，在实验布线区设计定制电路，从而掌握各种单片机应用。

除了 PICDEM™ 和 dsPICDEM™ 演示 / 开发板系列电路外，Microchip 还有一系列评估工具包和演示软件，适用于模拟滤波器设计、KEELOQ® 数据安全产品 IC、CAN、IrDA®、PowerSmart 电池管理、SEEVAL® 评估系统、 $\Sigma$ - $\Delta$  ADC、流速传感器，等等。

同时还提供入门工具包，其中包含体验指定器件功能所需的所有软硬件。通常提供单个应用以及调试功能，都包含在一块电路板上。

有关演示、开发和评估工具包的完整列表，请访问 Microchip 网站 ([www.microchip.com](http://www.microchip.com))。

## 29.12 第三方开发工具

Microchip 还提供一些来自第三方供应商的优秀开发工具。这些工具均经过精心挑选，功能独特，物有所值。

- SoftLog 和 CCS 等公司提供的器件编程器和量产编程器
- Gimpel 和 Trace Systems 等公司提供的软件工具
- Saleae 和 Total Phase 等公司提供的协议分析器
- MikroElektronika、Digilent® 和 Olimex 等公司提供的演示板
- EZ Web Lynx、WIZnet 和 IPLogika® 等公司提供的嵌入式以太网解决方案

## 30.0 电气特性

本章将对 dsPIC33EPXXXGS70X/80X 系列的电气特性进行概括介绍。其余信息在本文档的将来版本中提供。

下面列出了 dsPIC33EPXXXGS70X/80X 系列器件的绝对最大值。器件长时间工作在最大值条件下，其可靠性可能会受到影响。我们建议不要使器件工作在最大值甚至超过最大值的条件下。

### 绝对最大值<sup>(1)</sup>

偏置时的环境温度 .....	-40°C 至 +125°C
储存温度 .....	-65°C 至 +150°C
V <sub>DD</sub> 引脚相对于 V <sub>SS</sub> 的电压 .....	0.3V 至 +4.0V
任一非 5V 耐压引脚相对于 V <sub>SS</sub> 的电压 <sup>(3)</sup> .....	-0.3V 至 (V <sub>DD</sub> + 0.3V)
任一 5V 耐压引脚相对于 V <sub>SS</sub> 的电压（当 V <sub>DD</sub> ≥ 3.0V 时） <sup>(3)</sup> .....	-0.3V 至 +5.5V
任一 5V 耐压引脚相对于 V <sub>SS</sub> 的电压（当 V <sub>DD</sub> < 3.0V 时） <sup>(3)</sup> .....	-0.3V 至 +3.6V
流出 V <sub>SS</sub> 引脚的最大电流 .....	300 mA
流入 V <sub>DD</sub> 引脚的最大电流 <sup>(2)</sup> .....	300 mA
任一 4x I/O 引脚的最大灌电流 / 拉电流 .....	15 mA
任一 8x I/O 引脚的最大灌电流 / 拉电流 .....	25 mA
所有端口的最大灌电流 <sup>(2)</sup> .....	200 mA

**注 1:** 如果器件工作条件超过上述“绝对最大值”，可能对器件造成永久性损坏。上述数值仅是极限参数，我们建议不要使器件工作在最大值甚至超过最大值的条件下。器件长时间工作在最大值条件下，其可靠性可能受到影响。

**2:** 允许的最大电流由器件最大功耗决定（见表 30-2）。

**3:** 关于 5V 耐压的引脚，请参见“引脚图”部分。

# dsPIC33EPXXXGS70X/80X 系列

## 30.1 直流特性

表 30-1: 工作 MIPS 与电压

特性	VDD 范围 (单位: V)	温度范围 (单位: °C)	最大 MIPS
			dsPIC33EPXXXGS70X/80X 系列
—	3.0V 至 3.6V <sup>(1)</sup>	-40°C 至 +85°C	70
—	3.0V 至 3.6V <sup>(1)</sup>	-40°C 至 +125°C	60

注 1: 器件可在  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  条件下工作, 但此时模拟模块 (ADC、PGA 和比较器) 的性能可能下降。此条件下的器件功能经过测试, 但未确定特性值。关于最小和最大 BOR 值, 请参见表 30-39 中的参数 BO10。

表 30-2: 热工作条件

参数	符号	最小值	典型值	最大值	单位
工业级温度器件					
工作结温范围	TJ	-40	—	+125	°C
工作环境温度范围	TA	-40	—	+85	°C
扩展级温度器件					
工作结温范围	TJ	-40	—	+140	°C
工作环境温度范围	TA	-40	—	+125	°C
功耗: 芯片内部功耗: $P_{INT} = V_{DD} \times (I_{DD} - \Sigma I_{OH})$ I/O 引脚功耗: $I/O = \Sigma (\{V_{DD} - V_{OH}\} \times I_{OH}) + \Sigma (V_{OL} \times I_{OL})$	PD	P <sub>INT</sub> + P <sub>I/O</sub>			W
最大允许功耗	PD <sub>MAX</sub>	$(T_J - T_A) / \theta_{JA}$			W

表 30-3: 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 (12x12x1 mm)	$\theta_{JA}$	53.0	—	°C/W	1
封装热阻, 64 引脚 TQFP (10x10x1 mm)	$\theta_{JA}$	49.0	—	°C/W	1
封装热阻, 48 引脚 TQFP (7x7x1 mm)	$\theta_{JA}$	63.0	—	°C/W	1
封装热阻, 44 引脚 QFN (8x8 mm)	$\theta_{JA}$	29.0	—	°C/W	1
封装热阻, 44 引脚 TQFP (10x10x1 mm)	$\theta_{JA}$	50.0	—	°C/W	1
封装热阻, 28 引脚 QFN-S (6x6x0.9 mm)	$\theta_{JA}$	30.0	—	°C/W	1
封装热阻, 28 引脚 UQFN (6x6x0.55 mm)	$\theta_{JA}$	26.0	—	°C/W	1
封装热阻, 28 引脚 SOIC (7.50 mm)	$\theta_{JA}$	70.0	—	°C/W	1

注 1: 通过封装模拟获得结与环境的热阻值  $\theta_{JA}$ 。



# dsPIC33EPXXGS70X/80X 系列

表 30-4: 直流特性的温度和电压规范

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) <sup>(1)</sup> 工作温度 -40°C ≤ T <sub>A</sub> ≤ +85°C (工业级) -40°C ≤ T <sub>A</sub> ≤ +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
工作电压							
DC10	VDD	供电电压	3.0	—	3.6	V	
DC12	VDR	RAM 保持电压 <sup>(2)</sup>	—	—	1.95	V	+25°C, +85°C, +125°C
			—	—	2.0	V	-40°C
DC16	VPOR	确保内部上电复位信号的 V <sub>DD</sub> 启动电压	—	—	V <sub>SS</sub>	V	
DC17	SVDD	确保内部上电复位信号的 V <sub>DD</sub> 上升速率	1.0	—	—	V/ms	0V-3V/3 ms

注 1: 器件可在 V<sub>BORMIN</sub> < V<sub>DD</sub> < V<sub>DDMIN</sub> 条件下工作, 但此时模拟模块 (ADC、PGA 和比较器) 的性能可能下降。此条件下的器件功能经过测试, 但未确定特性值。关于最小和最大 BOR 值, 请参见表 30-39 中的参数 BO10。

2: 这是在总是保持 RAM 内容的情况下 V<sub>DD</sub> 可降至的电压下限。

表 30-5: 滤波电容 (CEFC) 规范

标准工作条件 (除非另外声明): 工作温度 -40°C ≤ T <sub>A</sub> ≤ +85°C (工业级) -40°C ≤ T <sub>A</sub> ≤ +125°C (扩展级)							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
	CEFC	外部滤波电容值 <sup>(1)</sup>	4.7	—	10	μF	电容必须具有较小的等效串联电阻 (<1Ω)

注 1: 当 V<sub>DD</sub> ≥ V<sub>DDMIN</sub> 时, 典型 V<sub>CAP</sub> 电压 = 1.8V。

# dsPIC33EPXXGS70X/80X 系列

表 30-6: 直流特性: 工作电流 (IDD)

直流特性			标准工作条件: 3.0V 至 3.6V (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	最大值	单位	条件		
工作电流 (IDD) <sup>(1)</sup>						
DC20d	8	13	mA	-40°C	3.3V	10 MIPS
DC20a	8	13	mA	+25°C		
DC20b	8	13	mA	+85°C		
DC20c	8	13	mA	+125°C		
DC22d	12	20	mA	-40°C	3.3V	20 MIPS
DC22a	12	20	mA	+25°C		
DC22b	12	20	mA	+85°C		
DC22c	12	20	mA	+125°C		
DC24d	19	30	mA	-40°C	3.3V	40 MIPS
DC24a	19	30	mA	+25°C		
DC24b	19	30	mA	+85°C		
DC24c	19	30	mA	+125°C		
DC25d	27	42	mA	-40°C	3.3V	60 MIPS
DC25a	27	42	mA	+25°C		
DC25b	27	42	mA	+85°C		
DC25c	27	42	mA	+125°C		
DC26d	30	46	mA	-40°C	3.3V	70 MIPS
DC26a	30	46	mA	+25°C		
DC26b	30	46	mA	+85°C		
DC27d	57	75	mA	-40°C		
DC27a	57	75	mA	+25°C	3.3V	70 MIPS (注 2)
DC27b	57	75	mA	+85°C		

注 1: IDD 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也会对电流消耗产生影响。所有 IDD 测量值的测试条件如下:

- 振荡器配置为带 PLL 的 EC 模式, OSC1 使用轨到轨的外部方波驱动 (要求 EC 时钟过冲 / 下冲 < 250 mV)
  - 在配置字中将 CLKO 配置为 I/O 输入引脚
  - 所有 I/O 引脚配置为输入且被拉到 Vss
  - MCLR = VDD, WDT 和 FSCM 被禁止
  - CPU、SRAM、程序存储器和数据存储器处于工作状态
  - 外设模块都不工作, 也不为其提供时钟 (所有定义的 PMDx 位均置 1)
  - CPU 执行 while(1) 语句
  - 禁止 JTAG
- 2: 对于此规范, 以下测试条件适用:
- 使能 APLL 时钟
  - 8 个 PWM 全部使能并且以最大速度工作 (PTCON2<2:0> = 000), PTPER = 1000h, 50% 占空比
  - 所有其他外设都被禁止 (对应的 PMDx 位置 1)

# dsPIC33EPXXGS70X/80X 系列

表 30-7: 直流特性: 空闲电流 (IDLE)

直流特性			标准工作条件: 3.0V 至 3.6V (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	最大值	单位	条件	
空闲电流 (IDLE) <sup>(1)</sup>					
DC40d	2	4	mA	-40°C	3.3V 10 MIPS
DC40a	2	4	mA	+25°C	
DC40b	2	4	mA	+85°C	
DC40c	2	4	mA	+125°C	
DC42d	3	6	mA	-40°C	3.3V 20 MIPS
DC42a	3	6	mA	+25°C	
DC42b	4	7	mA	+85°C	
DC42c	4	7	mA	+125°C	
DC44d	6	12	mA	-40°C	3.3V 40 MIPS
DC44a	6	12	mA	+25°C	
DC44b	6	12	mA	+85°C	
DC44c	6	12	mA	+125°C	
DC45d	9	17	mA	-40°C	3.3V 60 MIPS
DC45a	9	17	mA	+25°C	
DC45b	9	17	mA	+85°C	
DC45c	9	17	mA	+125°C	
DC46d	10	20	mA	-40°C	3.3V 70 MIPS
DC46a	10	20	mA	+25°C	
DC46b	10	20	mA	+85°C	

注 1: 基本空闲电流 (IDLE) 的测量条件如下:

- CPU 内核不工作, 振荡器配置为 EC 模式且外部时钟有效; OSC1 使用轨到轨的外部方波驱动 (要求 EC 时钟过冲 / 下冲 < 250 mV)
- 在配置字中将 CLKO 配置为 I/O 输入引脚
- 所有 I/O 引脚配置为输入且被拉到 Vss
- MCLR = VDD, WDT 和 FSCM 被禁止
- 外设模块都不工作, 也不为其提供时钟 (所有定义的 PMDx 位均置 1)
- NVMSIDL 位 (NVMCON<12>) = 1 (即, 在器件处于空闲模式时, 闪存稳压器设置为待机模式)
- VREGSF 位 (RCON<11>) = 0 (即, 在器件处于休眠模式时, 闪存稳压器设置为待机模式)
- 禁止 JTAG

# dsPIC33EPXXGS70X/80X 系列

表 30-8: 直流特性: 掉电电流 (IPD)

直流特性			标准工作条件: 3.0V 至 3.6V (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	最大值	单位	条件
掉电电流 (IPD) (1)				
DC60d	15	110	mA	-40°C
DC60a	20	150	mA	+25°C
DC60b	150	500	mA	+85°C
DC60c	500	1200	mA	+125°C

注 1: IPD (休眠) 电流的测量条件如下:

- CPU 内核不工作, 振荡器配置为 EC 模式且外部时钟有效; OSC1 使用轨到轨的外部方波驱动 (要求 EC 时钟过冲 / 下冲 < 250 mV)
- 在配置字中将 CLKO 配置为 I/O 输入引脚
- 所有 I/O 引脚配置为输入且被拉到 Vss
- MCLR = VDD, WDT 和 FSCM 被禁止
- 所有外设模块都被禁止 (所有 PMDx 位均置 1)
- VREGS 位 (RCON<8>) = 0 (即, 在器件处于休眠模式时, 内核稳压器设置为待机模式)
- VREGSF 位 (RCON<11>) = 0 (即, 在器件处于休眠模式时, 闪存稳压器设置为待机模式)
- 禁止 JTAG

表 30-9: 直流特性: 看门狗定时器增加电流 (ΔIWDT) (1)

直流特性			标准工作条件: 3.0V 至 3.6V (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	最大值	单位	条件
DC61d	1	10	mA	-40°C
DC61a	1	10	mA	+25°C
DC61b	2	17	mA	+85°C
DC61c	2	20	mA	+125°C

注 1: ΔIWDT 电流为当模块使能时额外消耗的电流。此电流应被加到基本 IPD 电流。所有参数均为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

表 30-10: 直流特性: 打盹电流 (IDOZE)

直流特性			标准工作条件: 3.0V 至 3.6V (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	最大值	打盹模式 时钟 分频比	单位	条件		
打盹电流 (IDOZE) <sup>(1)</sup>							
DC73a <sup>(2)</sup>	20	40	1:2	mA	-40°C	3.3V	Fosc = 140 MHz
DC73g	10	22	1:128	mA			
DC70a <sup>(2)</sup>	20	40	1:2	mA	+25°C	3.3V	Fosc = 140 MHz
DC70g	10	22	1:128	mA			
DC71a <sup>(2)</sup>	20	40	1:2	mA	+85°C	3.3V	Fosc = 140 MHz
DC71g	10	22	1:128	mA			
DC72a <sup>(2)</sup>	20	40	1:2	mA	+125°C	3.3V	Fosc = 120 MHz
DC72g	10	22	1:128	mA			

注 1: IDOZE 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也会对电流消耗产生影响。所有 IDOZE 测量值的测试条件如下:

- 振荡器配置为 EC 模式且外部时钟有效, OSC1 使用轨到轨的外部方波驱动 (要求 EC 时钟过冲 / 下冲 < 250 mV)
- 在配置字中将 CLK0 配置为 I/O 输入引脚
- 所有 I/O 引脚配置为输入且被拉到 Vss
- MCLR = VDD, WDT 和 FSCM 被禁止
- CPU、SRAM、程序存储器和数据存储器处于工作状态
- 外设模块都不工作, 也不为其提供时钟 (所有定义的 PMDx 位均置 1)
- CPU 执行 while(1) 语句
- 禁止 JTAG

2: 这些参数为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

表 30-11: 直流特性: I/O 引脚输入规范

直流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最大值	单位	条件
DI10	V <sub>IL</sub>	<b>输入低电压</b>					
DI18		所有 I/O 引脚和 $\overline{\text{MCLR}}$ 与 SDAx 和 SCLx 复用的 I/O 引脚	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	禁止 SMBus
DI19		与 SDAx 和 SCLx 复用的 I/O 引脚	V <sub>SS</sub>	—	0.8	V	使能 SMBus
DI20	V <sub>IH</sub>	<b>输入高电压</b>					
		非 5V 耐压的 I/O 引脚 <sup>(4)</sup>	0.8 V <sub>DD</sub>	—	V <sub>DD</sub>	V	
		5V 耐压的 I/O 引脚和 $\overline{\text{MCLR}}$ <sup>(4)</sup>	0.8 V <sub>DD</sub>	—	5.5	V	禁止 SMBus
		与 SDAx 和 SCLx 复用的 5V 耐压 I/O 引脚 <sup>(4)</sup>	0.8 V <sub>DD</sub>	—	5.5	V	禁止 SMBus
		与 SDAx 和 SCLx 复用的 5V 耐压 I/O 引脚 <sup>(4)</sup>	2.1	—	5.5	V	使能 SMBus
		与 SDAx 和 SCLx 复用的非 5V 耐压 I/O 引脚 <sup>(4)</sup>	0.8 V <sub>DD</sub>	—	V <sub>DD</sub>	V	禁止 SMBus
	与 SDAx 和 SCLx 复用的非 5V 耐压 I/O 引脚 <sup>(4)</sup>	2.1	—	V <sub>DD</sub>	V	使能 SMBus	
DI30	ICNPU	<b>输入电平变化通知上拉电流</b>	100	230	550	μA	V <sub>DD</sub> = 3.3V, V <sub>PIN</sub> = V <sub>SS</sub>
DI31	ICNPD	<b>输入电平变化通知下拉电流<sup>(5)</sup></b>	100	230	400	μA	V <sub>DD</sub> = 3.3V, V <sub>PIN</sub> = V <sub>DD</sub>

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

- 2:  $\overline{\text{MCLR}}$  引脚上的泄漏电流主要取决于所施加的电压。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。
- 3: 负电流定义为引脚的拉电流。
- 4: 关于 5V 耐压的 I/O 引脚, 请参见“引脚图”部分。
- 5: V<sub>IL</sub> 源 < (V<sub>SS</sub> - 0.3)。特性值, 但未经测试。
- 6: V<sub>IH</sub> 源 > (V<sub>DD</sub> + 0.3) 仅适用于非 5V 耐压的引脚。
- 7: 数字 5V 耐压引脚没有与 V<sub>DD</sub> 相连的内部上端二极管, 因此不能承受任何“正”输入注入电流。
- 8: 注入电流 ≠ 0 会影响 ADC 结果 (约 4-6 个计数)。
- 9: 只要来自所有引脚的输入注入电流的“绝对瞬时值”的和不超过规定的限制值, 就允许 I<sub>ICL</sub> 或 I<sub>ICH</sub> 条件下未排除的 I/O 引脚的任意数量和 / 或组合。特性值, 但未经测试。

# dsPIC33EPXXGS70X/80X 系列

表 30-11: 直流特性: I/O 引脚输入规范 (续)

直流特性		标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最大值	单位	条件
	I <sub>IL</sub>	输入泄漏电流 <sup>(2,3)</sup>					
DI50		5V 耐压的 I/O 引脚 <sup>(4)</sup>	-1	—	+1	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> , 引脚处于高阻态
DI51		非 5V 耐压的 I/O 引脚 <sup>(4)</sup>	-1	—	+1	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> , 引脚处于高阻态, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$
DI51a		非 5V 耐压的 I/O 引脚 <sup>(4)</sup>	-1	—	+1	μA	模拟引脚与外部参考引脚 共用, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$
DI51b		非 5V 耐压的 I/O 引脚 <sup>(4)</sup>	-1	—	+1	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> , 引脚处于高阻态, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$
DI51c		非 5V 耐压的 I/O 引脚 <sup>(4)</sup>	-1	—	+1	μA	模拟引脚与外部参考引脚 共用, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$
DI55		MCLR	-5	—	+5	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub>
DI56		OSC1	-5	—	+5	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> , XT 和 HS 模式

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

- 2: MCLR 引脚上的泄漏电流主要取决于所施加的电压。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。
- 3: 负电流定义为引脚的拉电流。
- 4: 关于 5V 耐压的 I/O 引脚, 请参见“引脚图”部分。
- 5: V<sub>IL</sub> 源 < (V<sub>SS</sub> - 0.3)。特性值, 但未经测试。
- 6: V<sub>IH</sub> 源 > (V<sub>DD</sub> + 0.3) 仅适用于非 5V 耐压的引脚。
- 7: 数字 5V 耐压引脚没有与 V<sub>DD</sub> 相连的内部上端二极管, 因此不能承受任何“正”输入注入电流。
- 8: 注入电流 ≠ 0 会影响 ADC 结果 (约 4-6 个计数)。
- 9: 只要来自所有引脚的输入注入电流的“绝对瞬时值”的和不超过规定的限制值, 就允许 I<sub>ICL</sub> 或 I<sub>ICH</sub> 条件下未排除的 I/O 引脚的任意数量和 / 或组合。特性值, 但未经测试。

# dsPIC33EPXXGS70X/80X 系列

表 30-11: 直流特性: I/O 引脚输入规范 (续)

直流特性		标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最大值	单位	条件
DI60a	I <sub>ICL</sub>	输入低注入电流	0	—	-5 <sup>(5,8)</sup>	mA	除 V <sub>DD</sub> 、V <sub>SS</sub> 、AV <sub>DD</sub> 、AV <sub>SS</sub> 、M <sub>CLR</sub> 、V <sub>CAP</sub> 和 RB7 外的所有引脚
DI60b	I <sub>ICH</sub>	输入高注入电流	0	—	+5 <sup>(6,7,8)</sup>	mA	除 V <sub>DD</sub> 、V <sub>SS</sub> 、AV <sub>DD</sub> 、AV <sub>SS</sub> 、M <sub>CLR</sub> 、V <sub>CAP</sub> 、RB7 和所有 5V 耐压引脚 <sup>(7)</sup> 外的所有引脚
DI60c	ΣI <sub>ICT</sub>	总输入注入电流 (所有 I/O 和控制引脚电流的和)	-20 <sup>(9)</sup>	—	+20 <sup>(9)</sup>	mA	来自所有 I/O 引脚的所有 ± 输入注入电流的绝对瞬时值的和 ( $ I_{ICL}  +  I_{ICH}  \leq \Sigma I_{ICT}$ )

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

- 2: M<sub>CLR</sub> 引脚上的泄漏电流主要取决于所施加的电压。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。
- 3: 负电流定义为引脚的拉电流。
- 4: 关于 5V 耐压的 I/O 引脚, 请参见“引脚图”部分。
- 5: V<sub>IL</sub> 源 < (V<sub>SS</sub> - 0.3)。特性值, 但未经测试。
- 6: V<sub>IH</sub> 源 > (V<sub>DD</sub> + 0.3) 仅适用于非 5V 耐压的引脚。
- 7: 数字 5V 耐压引脚没有与 V<sub>DD</sub> 相连的内部上端二极管, 因此不能承受任何“正”输入注入电流。
- 8: 注入电流 ≠ 0 会影响 ADC 结果 (约 4-6 个计数)。
- 9: 只要来自所有引脚的输入注入电流的“绝对瞬时值”的和不出超规定的限制值, 就允许 I<sub>ICL</sub> 或 I<sub>ICH</sub> 条件下未排除的 I/O 引脚的任意数量和 / 或组合。特性值, 但未经测试。



# dsPIC33EPXXGS70X/80X 系列

表 30-12: 直流特性: I/O 引脚输出规范

直流特性		标准工作条件: 3.0V 至 3.6V (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	符号	特性	最小值	典型值	最大值	单位	条件
DO10	VOL	输出低电压 4x 灌电流驱动引脚 (2)	—	—	0.4	V	$V_{DD} = 3.3\text{V}$ , $I_{OL} \leq 6\text{ mA}$ , $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ , $I_{OL} \leq 5\text{ mA}$ , $+85^{\circ}\text{C} < T_A \leq +125^{\circ}\text{C}$
		输出低电压 8x 灌电流驱动引脚 (3)	—	—	0.4	V	$V_{DD} = 3.3\text{V}$ , $I_{OL} \leq 12\text{ mA}$ , $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ , $I_{OL} \leq 8\text{ mA}$ , $+85^{\circ}\text{C} < T_A \leq +125^{\circ}\text{C}$
DO20	VOH	输出高电压 4x 拉电流驱动引脚 (2)	2.4	—	—	V	$I_{OH} \geq -10\text{ mA}$ , $V_{DD} = 3.3\text{V}$
		输出高电压 8x 拉电流驱动引脚 (3)	2.4	—	—	V	$I_{OH} \geq -15\text{ mA}$ , $V_{DD} = 3.3\text{V}$
DO20A	VOH1	输出高电压 4x 拉电流驱动引脚 (2)	1.5 <sup>(1)</sup>	—	—	V	$I_{OH} \geq -14\text{ mA}$ , $V_{DD} = 3.3\text{V}$
			2.0 <sup>(1)</sup>	—	—		$I_{OH} \geq -12\text{ mA}$ , $V_{DD} = 3.3\text{V}$
			3.0 <sup>(1)</sup>	—	—		$I_{OH} \geq -7\text{ mA}$ , $V_{DD} = 3.3\text{V}$
	输出高电压 8x 拉电流驱动引脚 (3)	1.5 <sup>(1)</sup>	—	—	V	$I_{OH} \geq -22\text{ mA}$ , $V_{DD} = 3.3\text{V}$	
		2.0 <sup>(1)</sup>	—	—		$I_{OH} \geq -18\text{ mA}$ , $V_{DD} = 3.3\text{V}$	
		3.0 <sup>(1)</sup>	—	—		$I_{OH} \geq -10\text{ mA}$ , $V_{DD} = 3.3\text{V}$	

注 1: 参数为特性值, 但未经测试。

2: 包括 RA0-RA2、RB0-RB1、RB9、RC1-RC2、RC9-RC10、RC12、RD7、RD8、RE4-RE5、RE8-RE9 和 RE12-RE13 引脚。

3: 包括所有不是 4x 驱动引脚的 I/O 引脚 (见注 2)。

表 30-13: 电气特性: BOR

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) (1) 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 (2)	典型值	最大值	单位	条件
BO10	VBOR	当 $V_{DD}$ 从高电压变为低电压时的 BOR 事件	2.65	—	2.95	V	$V_{DD}$ (注 2 和 3)

注 1: 器件可在  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  条件下工作, 但性能将下降。此条件下的器件功能经过测试, 但未确定特性值。模拟模块 (ADC、PGA 和比较器) 的性能可能下降。

2: 这些参数仅供设计参考, 生产时未经测试。

3: VBOR 规范与  $V_{DD}$  相关。

# dsPIC33EPXXGS70X/80X 系列

表 30-14: 直流特性: 程序存储器

直流特性			标准工作条件: 3.0V 至 3.6V (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
闪存程序存储器							
D130	EP	单元耐擦写能力	10,000	—	—	E/W	-40°C 至 +125°C
D131	VPR	读操作时的 VDD	3.0	—	3.6	V	
D132b	VPEW	自定时写的 VDD	3.0	—	3.6	V	
D134	TRETD	特性保持时间	20	—	—	年	假设没有违反其他规范, -40°C 至 +125°C
D135	IDDP	编程时的供电电流 (2)	—	10	—	mA	
D136	IPEAK	启动期间的瞬时峰值电流	—	—	150	mA	
D137a	TPE	页擦除时间	19.7	—	20.1	ms	TPE = 146893 个 FRC 周期, TA = +85°C (注 3)
D137b	TPE	页擦除时间	19.5	—	20.3	ms	TPE = 146893 个 FRC 周期, TA = +125°C (注 3)
D138a	TWW	字写周期	46.5	—	47.3	µs	TWW = 346 个 FRC 周期, TA = +85°C (注 3)
D138b	TWW	字写周期	46.0	—	47.9	µs	TWW = 346 个 FRC 周期, TA = +125°C (注 3)
D139a	TRW	行写时间	667	—	679	µs	TRW = 4965 个 FRC 周期, TA = +85°C (注 3)
D139b	TRW	行写时间	660	—	687	µs	TRW = 4965 个 FRC 周期, TA = +125°C (注 3)

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

2: 该参数为特性值, 但生产时未经测试。

3: 其他条件: FRC = 7.37 MHz, TUN<5:0> = 011111 (对于最小值), TUN<5:0> = 100000 (对于最大值)。该参数取决于 FRC 精度 (见表 30-20) 和 FRC 振荡器调节寄存器 (见寄存器 9-4) 的值。关于计算最小和最大时间的完整细节, 请参见第 5.3 节“编程操作”。

# dsPIC33EPXXXGS70X/80X 系列

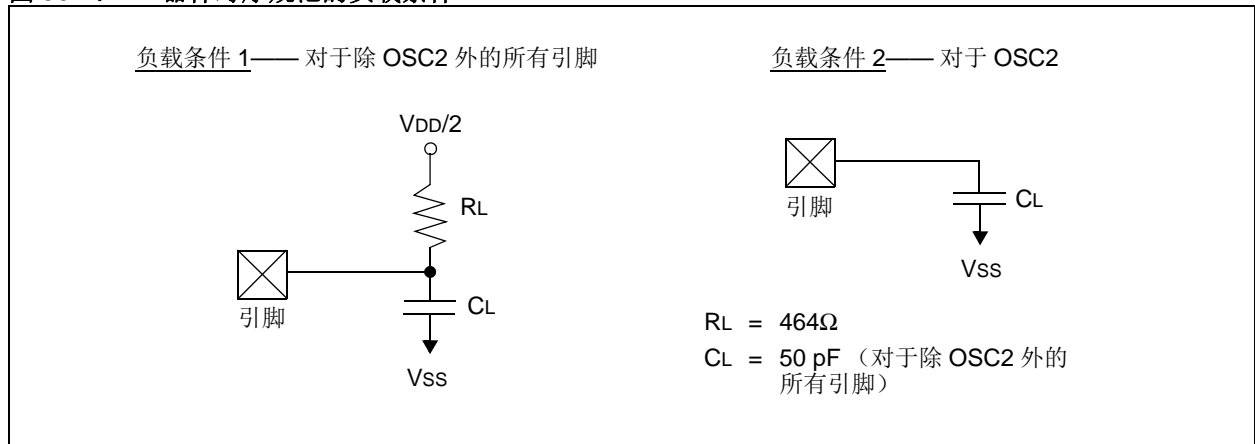
## 30.2 交流特性和时序参数

本节定义了 dsPIC33EPXXXGS70X/80X 系列的交流特性和时序参数。

**表 30-15: 温度和电压规范 —— 交流**

交流特性	标准工作条件: <b>3.0V 至 3.6V</b> (除非另外声明)
	工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级)
	$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)
	工作电压 $V_{DD}$ 范围如第 30.1 节 “直流特性” 中所述。

**图 30-1: 器件时序规范的负载条件**



**表 30-16: 输出引脚上的容性负载要求**

参数编号	符号	特性	最小值	典型值	最大值	单位	条件
DO50	Cosco	OSC2 引脚	—	—	15	pF	当外部时钟用于驱动 OSC1 时, 处于 XT 和 HS 模式下
DO56	Cio	所有 I/O 引脚和 OSC2	—	—	50	pF	EC 模式
DO58	Cb	SCLx 和 SDAx	—	—	400	pF	在 I <sup>2</sup> C 模式下

# dsPIC33EPXXGS70X/80X 系列

图 30-2: 外部时钟时序

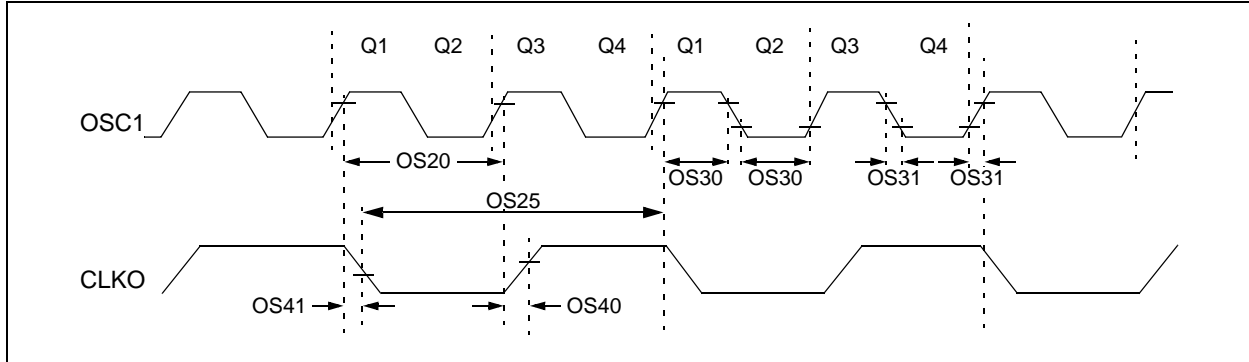


表 30-17: 外部时钟时序要求

交流特性		标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最大值	单位	条件
OS10	FIN	外部 CLKI 频率 (外部时钟仅允许运行于 EC 和 ECPLL 模式)	DC	—	60	MHz	EC
		晶振频率	3.5 10	— —	10 40	MHz MHz	XT HS
OS20	Tosc	$T_{osc} = 1/F_{osc}$	8.33	—	DC	ns	+125°C
		$T_{osc} = 1/F_{osc}$	7.14	—	DC	ns	+85°C
OS25	Tcy	指令周期 <sup>(2)</sup>	16.67	—	DC	ns	+125°C
		指令周期 <sup>(2)</sup>	14.28	—	DC	ns	+85°C
OS30	TosL, TosH	外部时钟输入 (OSC1) 的高电平或低电平时间	$0.45 \times T_{osc}$	—	$0.55 \times T_{osc}$	ns	EC
OS31	TosR, TosF	外部时钟输入 (OSC1) 的上升或下降时间	—	—	20	ns	EC
OS40	TckR	CLKO 上升时间 <sup>(3,4)</sup>	—	5.2	—	ns	
OS41	TckF	CLKO 下降时间 <sup>(3,4)</sup>	—	5.2	—	ns	
OS42	GM	外部振荡器跨导 <sup>(4)</sup>	—	12	—	mA/V	HS, V <sub>DD</sub> = 3.3V, T <sub>A</sub> = +25°C
			—	6	—	mA/V	XT, V <sub>DD</sub> = 3.3V, T <sub>A</sub> = +25°C

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

2: 指令周期 (Tcy) 等于输入振荡器时基周期的两倍。所有规定值均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 / 或电流消耗超出预期值。所有器件在测试“最小”值时, 都在 OSC1 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)。

3: 测量在 EC 模式下进行。在 OSC2 引脚上测量 CLKO 信号。

4: 该参数为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

**表 30-18: PLL 时钟时序规范**

交流特性		标准工作条件: 3.0V 至 3.6V (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
OS50	FPLLI	PLL 压控振荡器 (Voltage Controlled Oscillator, VCO) 的输入频率范围	0.8	—	8.0	MHz	ECPLL 和 XTPLL 模式
OS51	FVCO	片上 VCO 系统频率	120	—	340	MHz	
OS52	TLOCK	PLL 启动时间 (锁定时间)	0.9	1.5	3.1	ms	
OS53	DCLK	CLKO 稳定性 (抗抖动性) (2)	-3	0.5	3	%	

**注 1:** 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。这些参数仅供设计参考, 未经测试。

**2:** 该抖动规范值通过逐个时钟周期测量的方式获得。要获得应用所用的各个时基或通信时钟的实际抖动, 请使用以下公式:

$$\text{实际抖动} = \frac{DCLK}{\sqrt{\frac{FOSC}{\text{时基或通信时钟}}}}$$

例如, 如果 Fosc = 120 MHz 且 SPIx 比特率 = 10 MHz, 则实际抖动如下:

$$\text{实际抖动} = \frac{DCLK}{\sqrt{\frac{120}{10}}} = \frac{DCLK}{\sqrt{12}} = \frac{DCLK}{3.464}$$

**表 30-19: 附属 PLL 时钟时序规范**

交流特性		标准工作条件: 3.0V 至 3.6V (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
OS56	FHPOUT	片上 16 倍频 PLL CCO 频率	112	118	120	MHz	
OS57	FHPIN	片上 16 倍频 PLL 相位检测器输入频率	7.0	7.37	7.5	MHz	
OS58	Tsu	频率发生器锁定时间	—	—	10	μs	

**注 1:** 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。这些参数仅供设计参考, 生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

表 30-20: 内部 FRC 精度

交流特性		标准工作条件: 3.0V 至 3.6V (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	
<b>FRC 频率 = 7.37 MHz 时的内部 FRC 精度<sup>(1)</sup></b>							
F20a	FRC	-2	0.5	+2	%	-40°C ≤ TA ≤ -10°C	VDD = 3.0-3.6V
		-0.9	0.5	+0.9	%	-10°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
F20b	FRC	-2	1	+2	%	+85°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V

注 1: 频率在 +25°C 和 3.3V 条件下校准。TUNx 位可用于补偿温度漂移。

表 30-21: 内部 LPRC 精度

交流特性		标准工作条件: 3.0V 至 3.6V (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	
<b>频率为 32.768 kHz 时的 LPRC<sup>(1)</sup></b>							
F21a	LPRC	-30	—	+30	%	-40°C ≤ TA ≤ -10°C	VDD = 3.0-3.6V
		-20	—	+20	%	-10°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
F21b	LPRC	-30	—	+30	%	+85°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V

注 1: LPRC 频率将随 VDD 的变化而变化。

# dsPIC33EPXXGS70X/80X 系列

图 30-3: I/O 时序特性

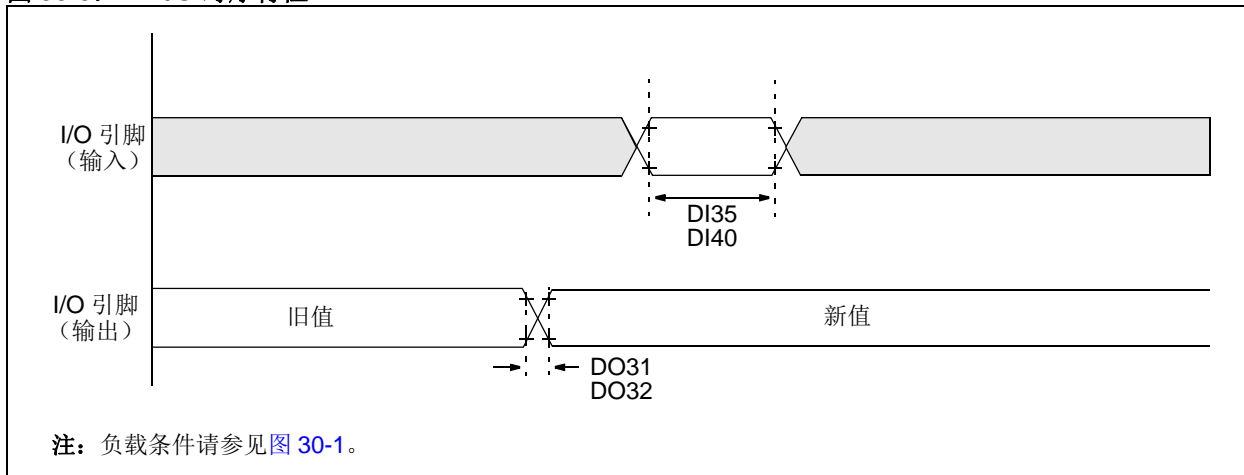
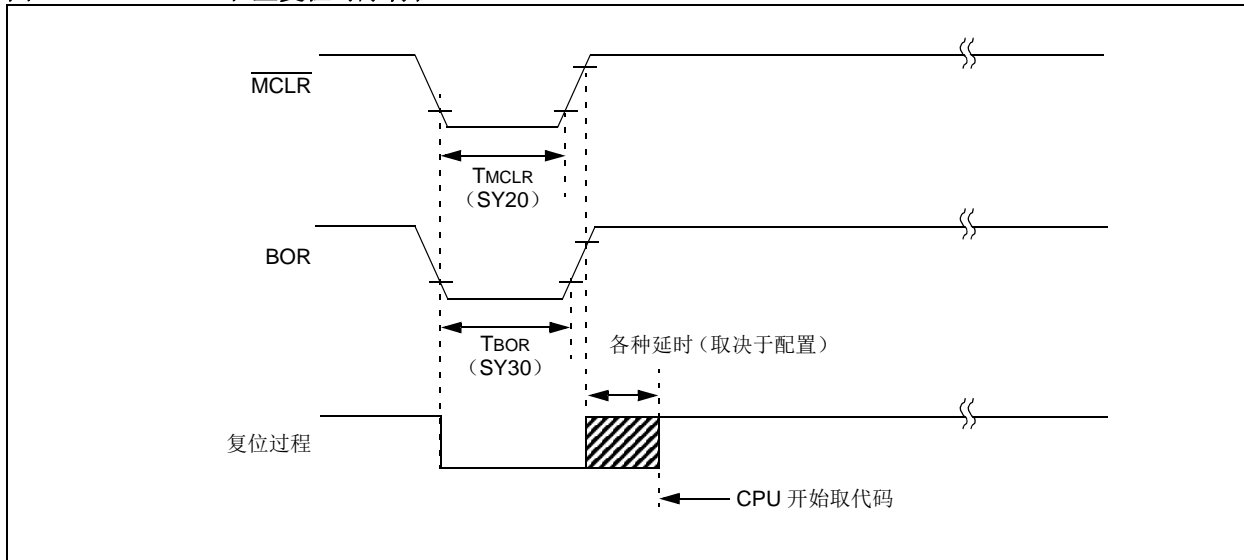


表 30-22: I/O 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最大值	单位	条件
DO31	TioR	端口输出上升时间	—	5	10	ns	
DO32	TioF	端口输出下降时间	—	5	10	ns	
DI35	TINP	INTx 引脚高电平或低电平时间 (输入)	20	—	—	ns	
DI40	TRBP	CNx 高电平或低电平时间 (输入)	2	—	—	Tcy	

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

图 30-4: BOR 和主复位时序特性



# dsPIC33EPXXGS70X/80X 系列

表 30-23: 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要求

交流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SY00	TPU	上电周期	—	400	600	μs	
SY10	TOST	振荡器起振时间	—	1024 TOSC	—	—	TOSC = OSC1 周期
SY12	TWDTPRE	看门狗定时器超时周期	0.81	—	1.22	ms	WDTPRE = 0, WDTPRSTEN = 0, 使用 F21 中给出的 LPRC 容差 (见表 30-21) (+85°C 时)
			3.25	—	4.88	ms	WDTPRE = 1, WDTPRSTEN = 0, 使用 F21 中给出的 LPRC 容差 (见表 30-21) (+85°C 时)
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	0.68	0.72	1.2	μs	
SY20	TMCLR	MCLR 脉冲宽度 (低电平)	2	—	—	μs	
SY30	TBOR	BOR 脉冲宽度 (低电平)	1	—	—	μs	
SY35	TFSCM	故障保护时钟监视器延时	—	500	900	μs	-40°C 至 +85°C
SY36	TVREG	稳压器从待机模式变为工作模式的时间	—	—	30	μs	
SY37	TOSCDFRC	FRC 振荡器起振延时	—	48	—	μs	
SY38	TOSCDLPRC	LPRC 振荡器起振延时	—	—	70	μs	

注 1: 这些参数为特性值, 但生产时未经测试。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。



# dsPIC33EPXXGS70X/80X 系列

图 30-5: Timer1-Timer5 外部时钟时序特性

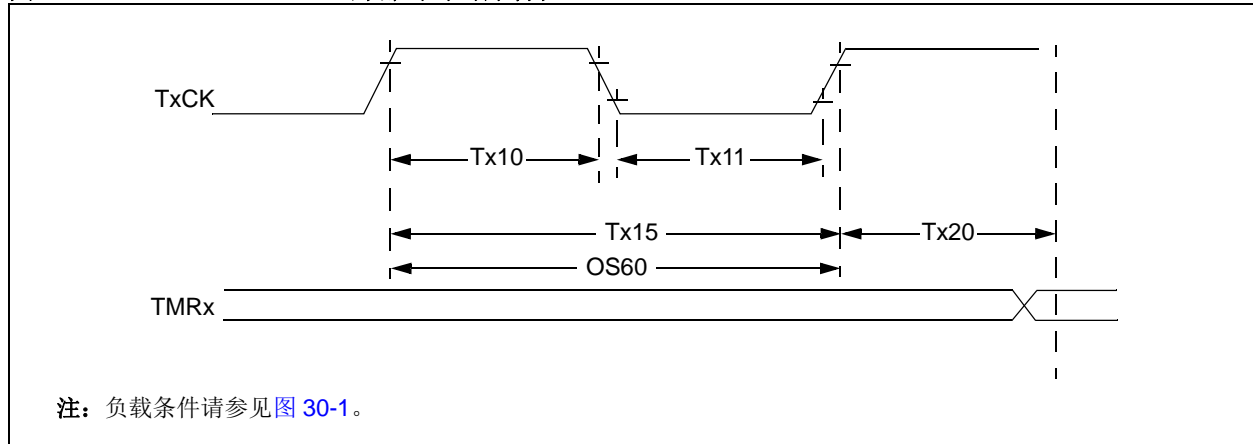


表 30-24: Timer1 外部时钟时序要求 (1)

交流特性				标准工作条件: 3.0V 至 3.6V (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 (2)		最小值	典型值	最大值	单位	条件
TA10	TtxH	T1CK 高电平时间	同步模式	取如下二者中较大值: 20 或 (Tcy + 20)/N	—	—	ns	还必须满足参数 TA15, N = 预分频值 (1、8、64 和 256)
			异步模式	35	—	—	ns	
TA11	TtxL	T1CK 低电平时间	同步模式	取如下二者中较大值: 20 或 (Tcy + 20)/N	—	—	ns	还必须满足参数 TA15, N = 预分频值 (1、8、64 和 256)
			异步模式	10	—	—	ns	
TA15	TtxP	T1CK 输入周期	同步模式	取如下二者中较大值: 40 或 (2 Tcy + 40)/N	—	—	ns	N = 预分频值 (1、8、64 和 256)
OS60	Ft1	T1CK 振荡器输入频率范围 (通过置 1 TCS 位 (T1CON<1>) 使能振荡器)		DC	—	50	kHz	
TA20	TckEXTMRL	从外部 T1CK 时钟边沿到定时器递增之间的延时		0.75 Tcy + 40	—	1.75 Tcy + 40	ns	

注 1: Timer1 属于 A 类定时器。

2: 这些参数为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

表 30-25: Timer2 和 Timer4 (B 类定时器) 外部时钟时序要求

交流特性				标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>		最小值	典型值	最大值	单位	条件
TB10	TtxH	TxCK 高电平时间	同步模式	取如下二者中较大值: 20 或 (Tcy + 20)/N	—	—	ns	还必须满足参数 TB15, N = 预分频值 (1、8、64 和 256)
TB11	TtxL	TxCK 低电平时间	同步模式	取如下二者中较大值: 20 或 (Tcy + 20)/N	—	—	ns	还必须满足参数 TB15, N = 预分频值 (1、8、64 和 256)
TB15	TtxP	TxCK 输入周期	同步模式	取如下二者中较大值: 40 或 (2 Tcy + 40)/N	—	—	ns	N = 预分频值 (1、8、64 和 256)
TB20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增之间的延时		0.75 Tcy + 40	—	1.75 Tcy + 40	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

表 30-26: Timer3 和 Timer5 (C 类定时器) 外部时钟时序要求

交流特性				标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>		最小值	典型值	最大值	单位	条件
TC10	TtxH	TxCK 高电平时间	同步	Tcy + 20	—	—	ns	还必须满足参数 TC15
TC11	TtxL	TxCK 低电平时间	同步	Tcy + 20	—	—	ns	还必须满足参数 TC15
TC15	TtxP	TxCK 输入周期	同步, 带预分频器	2 Tcy + 40	—	—	ns	N = 预分频值 (1、8、64 和 256)
TC20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增之间的延时		0.75 Tcy + 40	—	1.75 Tcy + 40	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

图 30-6: 输入捕捉 x (ICx) 时序特性

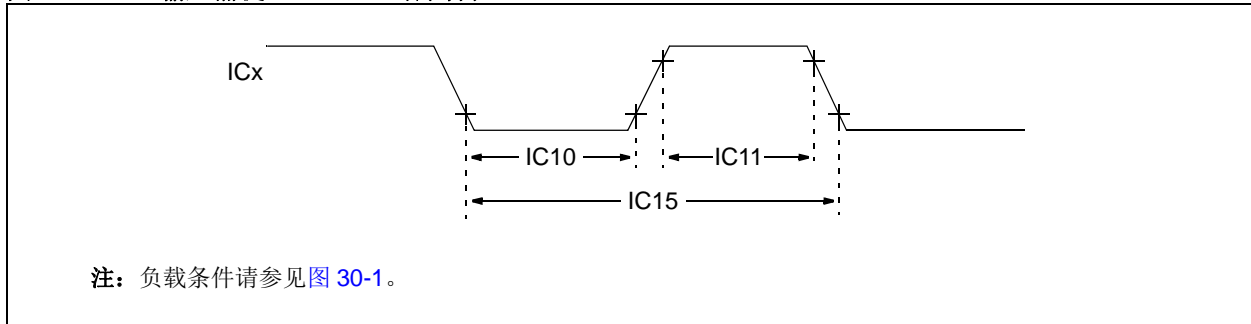


表 30-27: 输入捕捉 x 模块时序要求

交流特性		标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最小值	最大值	单位	条件
IC10	TccL	ICx 输入低电平时间	取如下二者中较大值: $12.5 + 25$ 或 $(0.5 T_{cy}/N) + 25$	—	ns	还必须满足参数 IC15
IC11	TccH	ICx 输入高电平时间	取如下二者中较大值: $12.5 + 25$ 或 $(0.5 T_{cy}/N) + 25$	—	ns	还必须满足参数 IC15
IC15	TccP	ICx 输入周期	取如下二者中较大值: $25 + 50$ 或 $(1 T_{cy}/N) + 50$	—	ns	

N = 预分频值 (1、4 和 16)

注 1: 这些参数为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

图 30-7: 输出比较 x 模块 (OCx) 时序特性

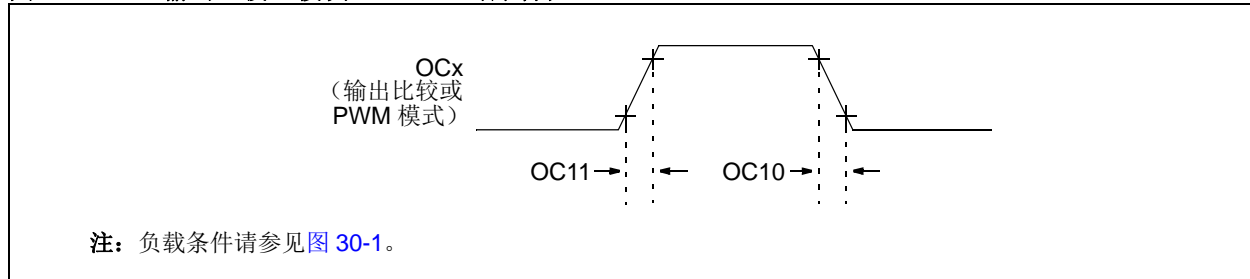


表 30-28: 输出比较 x 模块时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最小值	典型值	最大值	单位	条件
OC10	TccF	OCx 输出下降时间	—	—	—	ns	见参数 DO32
OC11	TccR	OCx 输出上升时间	—	—	—	ns	见参数 DO31

注 1: 这些参数为特性值, 但生产时未经测试。

图 30-8: OCx/PWMx 模块时序特性

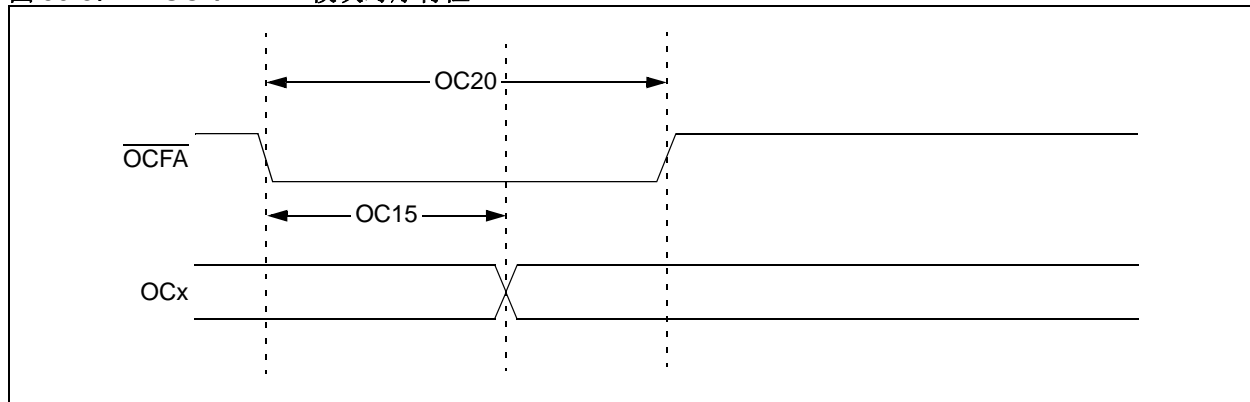


表 30-29: OCx/PWMx 模块时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最小值	典型值	最大值	单位	条件
OC15	TFD	故障输入到 PWMx I/O 发生变化的时间	—	—	$T_{CY} + 20$	ns	
OC20	TFLT	故障输入脉冲宽度	$T_{CY} + 20$	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

图 30-9: 高速 PWMx 模块故障时序特性

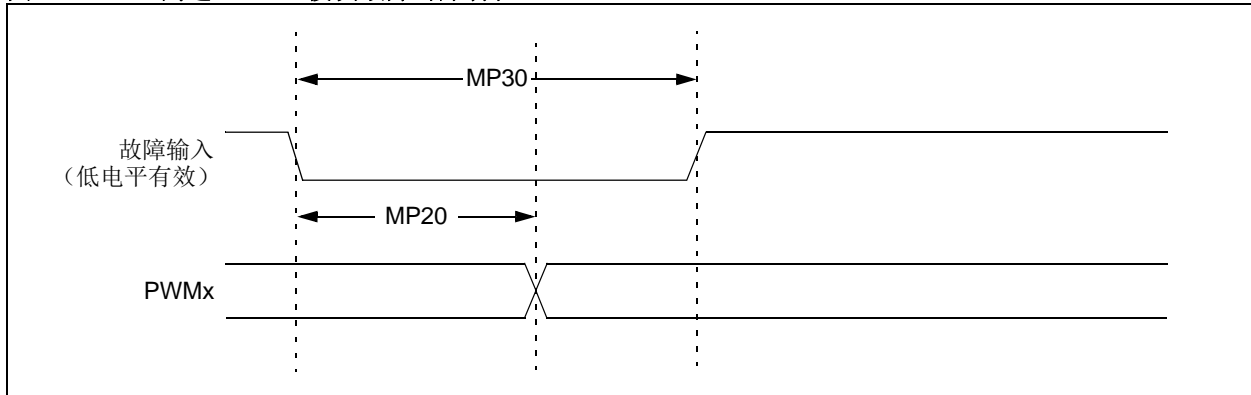


图 30-10: 高速 PWMx 模块时序特性

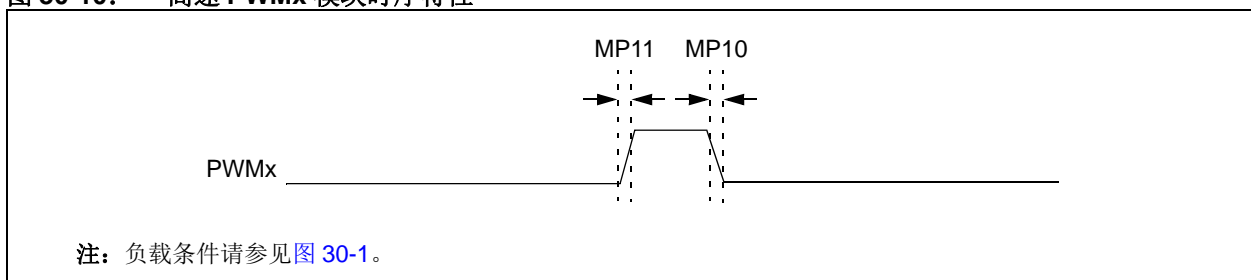


表 30-30: 高速 PWMx 模块时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最小值	典型值	最大值	单位	条件
MP10	TFPWM	PWMx 输出下降时间	—	—	—	ns	见参数 DO32
MP11	TRPWM	PWMx 输出上升时间	—	—	—	ns	见参数 DO31
MP20	TFD	故障输入 ↓ 到 PWMx I/O 发生变化的时间	—	—	15	ns	
MP30	TFH	故障输入脉冲宽度	15	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

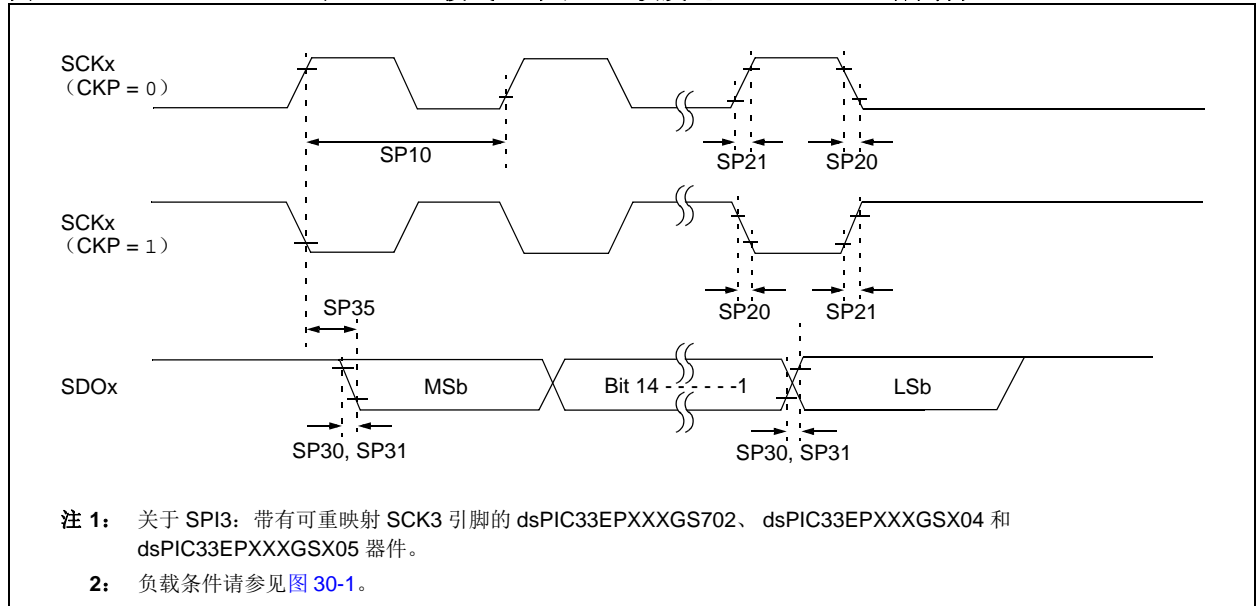
# dsPIC33EPXXGS70X/80X 系列

表 30-31: SPI1、SPI2 和 SPI3 最大数据 / 时钟速率汇总 (1)

交流特性				标准工作条件: 3.0V 至 3.6V (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)		
最大数据速率	主模式仅发送 (半双工)	主模式发送 / 接收 (全双工)	从模式发送 / 接收 (全双工)	CKE	CKP	SMP
15 MHz	表 30-32	—	—	0,1	0,1	0,1
9 MHz	—	表 30-33	—	1	0,1	1
9 MHz	—	表 30-34	—	0	0,1	1
15 MHz	—	—	表 30-35	1	0	0
11 MHz	—	—	表 30-36	1	1	0
15 MHz	—	—	表 30-37	0	1	0
11 MHz	—	—	表 30-38	0	0	0

注 1: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXGS702、dsPIC33EPXXGSX04 和 dsPIC33EPXXGSX05 器件。

图 30-11: SPI1、SPI2 和 SPI3 主模式 (半双工, 仅发送, CKE = 0) 时序特性 (1,2)



# dsPIC33EPXXGS70X/80X 系列

图 30-12: SPI1、SPI2 和 SPI3 主模式（半双工，仅发送，CKE = 1）时序特性<sup>(1,2)</sup>

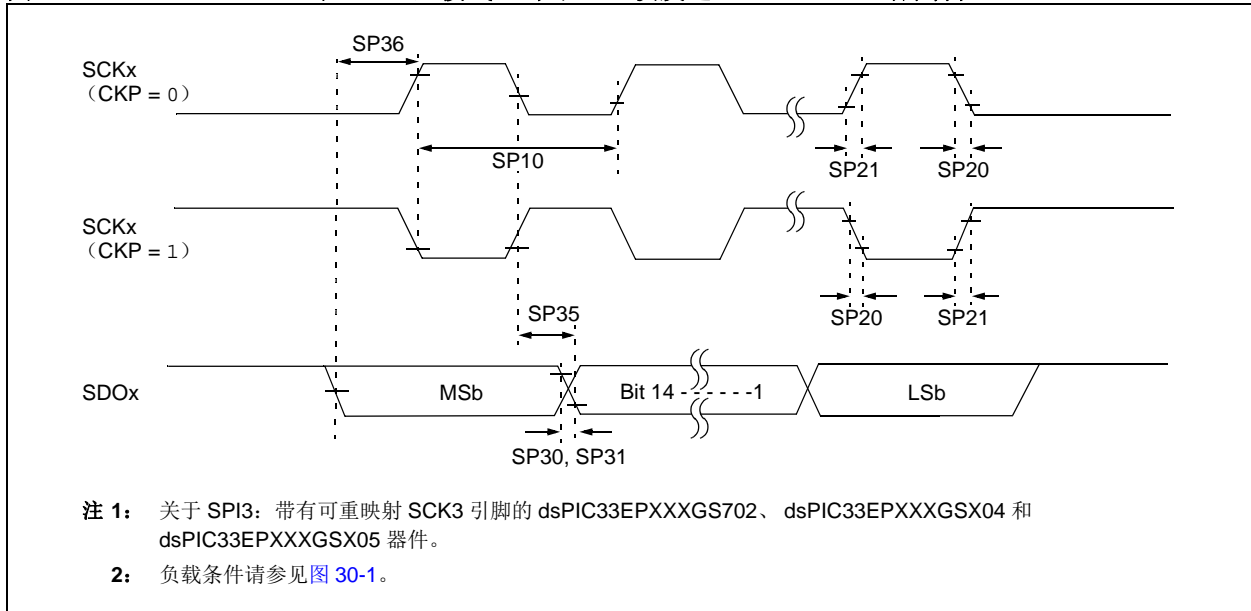


表 30-32: SPI1、SPI2 和 SPI3 主模式（半双工，仅发送）时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP10	FscP	最大 SCKx 频率	—	—	15	MHz	(注 3)
SP20	TscF	SCKx 输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP21	TscR	SCKx 输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	Tsch2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	
SP36	TdiV2scH, TdiV2scL	SDOx 数据输出建立到出现第一个 SCKx 边沿的时间	30	—	—	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 3.3V 和 +25°C 条件下的值。

3: SCKx 的最小时钟周期为 66.7 ns。因此，主模式下产生的时钟不应违反此规范。

4: 假定所有 SPIx 引脚上的负载均为 50 pF。

5: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXGS702、dsPIC33EPXXGSX04 和 dsPIC33EPXXGSX05 器件。

# dsPIC33EPXXXGS70X/80X 系列

图 30-13: SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 1, CKP = x, SMP = 1) 时序特性 (1,2)

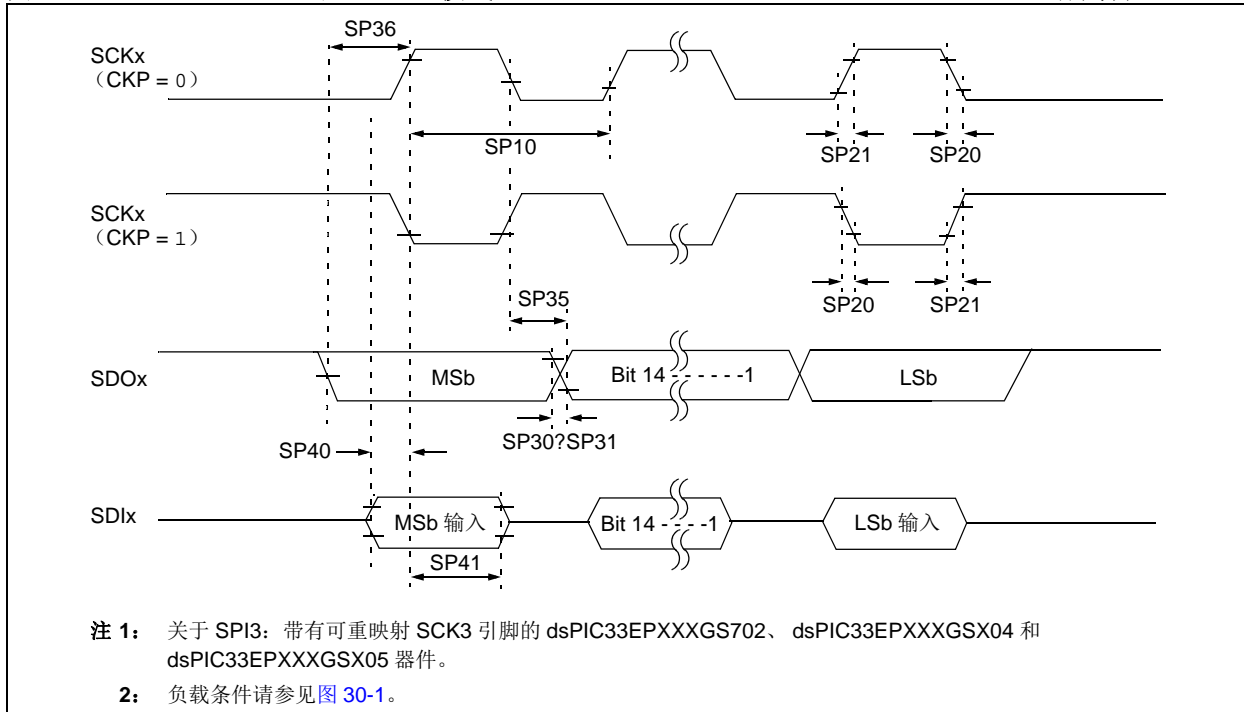


表 30-33: SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 1, CKP = x, SMP = 1) 时序要求 (5)

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	FscP	最大 SCKx 频率	—	—	9	MHz	(注 3)
SP20	TscF	SCKx 输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP21	TscR	SCKx 输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	Tsch2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2sc, TdoV2scL	SDOx 数据输出建立到出现第一个 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	30	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

注 3: SCKx 的最小时钟周期为 111 ns。主模式下产生的时钟不应违反此规范。

注 4: 假定所有 SPIx 引脚上的负载均为 50 pF。

注 5: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXXGS702、dsPIC33EPXXXGSX04 和 dsPIC33EPXXXGSX05 器件。



# dsPIC33EPXXGS70X/80X 系列

图 30-14: SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) 时序特性 (1,2)

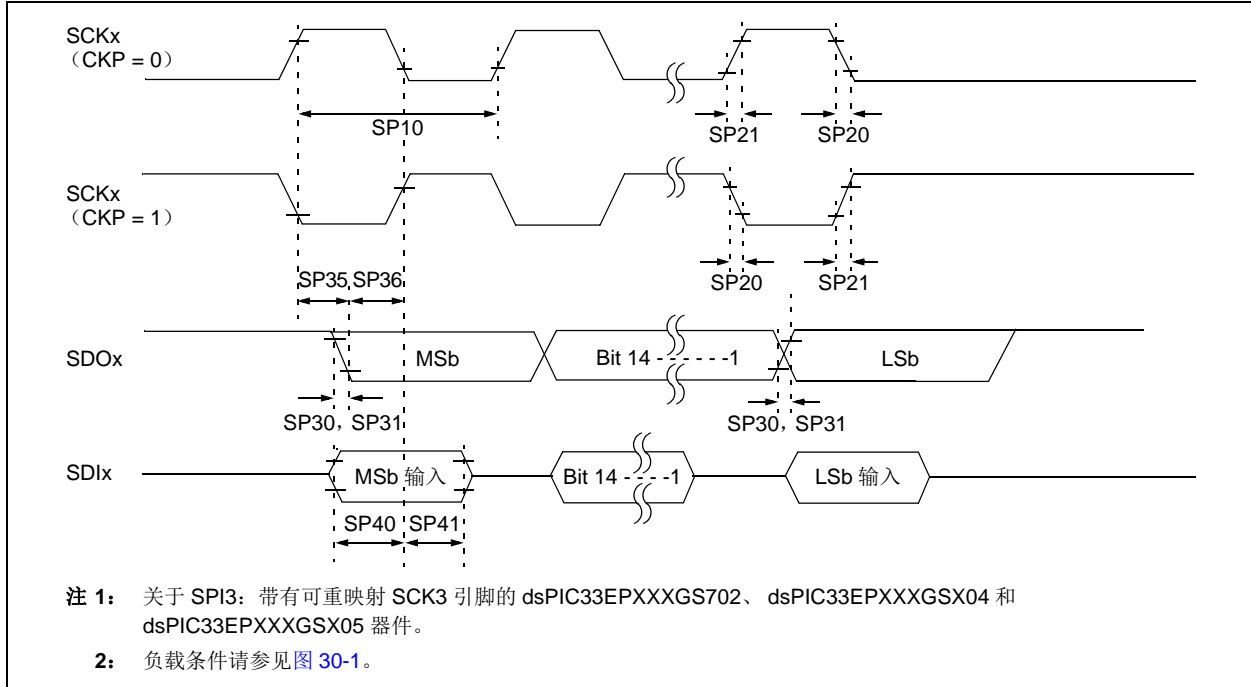


表 30-34: SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) 时序要求 (5)

交流特性		标准工作条件: 3.0V 至 3.6V (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	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	FscP	最大 SCKx 频率	—	—	9	MHz	-40°C 至 +125°C (注 3)
SP20	TscF	SCKx 输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP21	TscR	SCKx 输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	Tsch2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx 数据输出建立到出现第一个 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	30	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

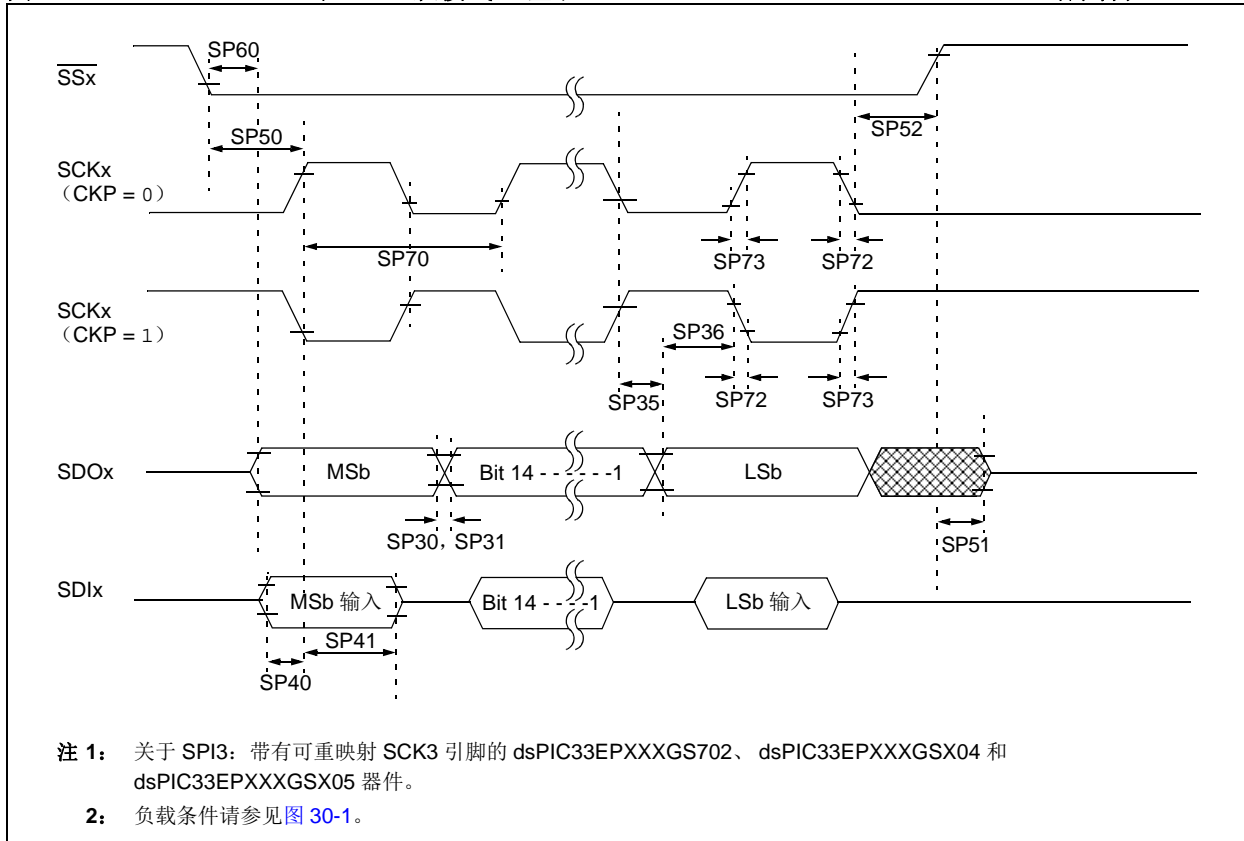
注 3: SCKx 的最小时钟周期为 111 ns。主模式下产生的时钟不应违反此规范。

注 4: 假定所有 SPIx 引脚上的负载均为 50 pF。

注 5: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXGS702、dsPIC33EPXXGSX04 和 dsPIC33EPXXGSX05 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-15: SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 1, CKP = 0, SMP = 0) 时序特性 (1,2)



# dsPIC33EPXXGS70X/80X 系列

表 30-35: SPI1、SPI2 和 SPI3 从模式（全双工，CKE = 1，CKP = 0，SMP = 0）时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCKx 输入频率	—	—	15	MHz	(注 3)
SP72	TscF	SCKx 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCKx 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx 数据输出建立到出现第一个 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	30	—	—	ns	
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	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	10	—	50	ns	(注 4)
SP52	TscH2ssH, TscL2ssH	SCKx ↑ 边沿之后 $\overline{SSx}$ 有效的 时间	1.5 Tcy + 40	—	—	ns	(注 4)
SP60	TssL2doV	$\overline{SSx}$ 边沿之后 SDOx 数据输出有效的时间	—	—	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 3.3V 和 +25°C 条件下的值。

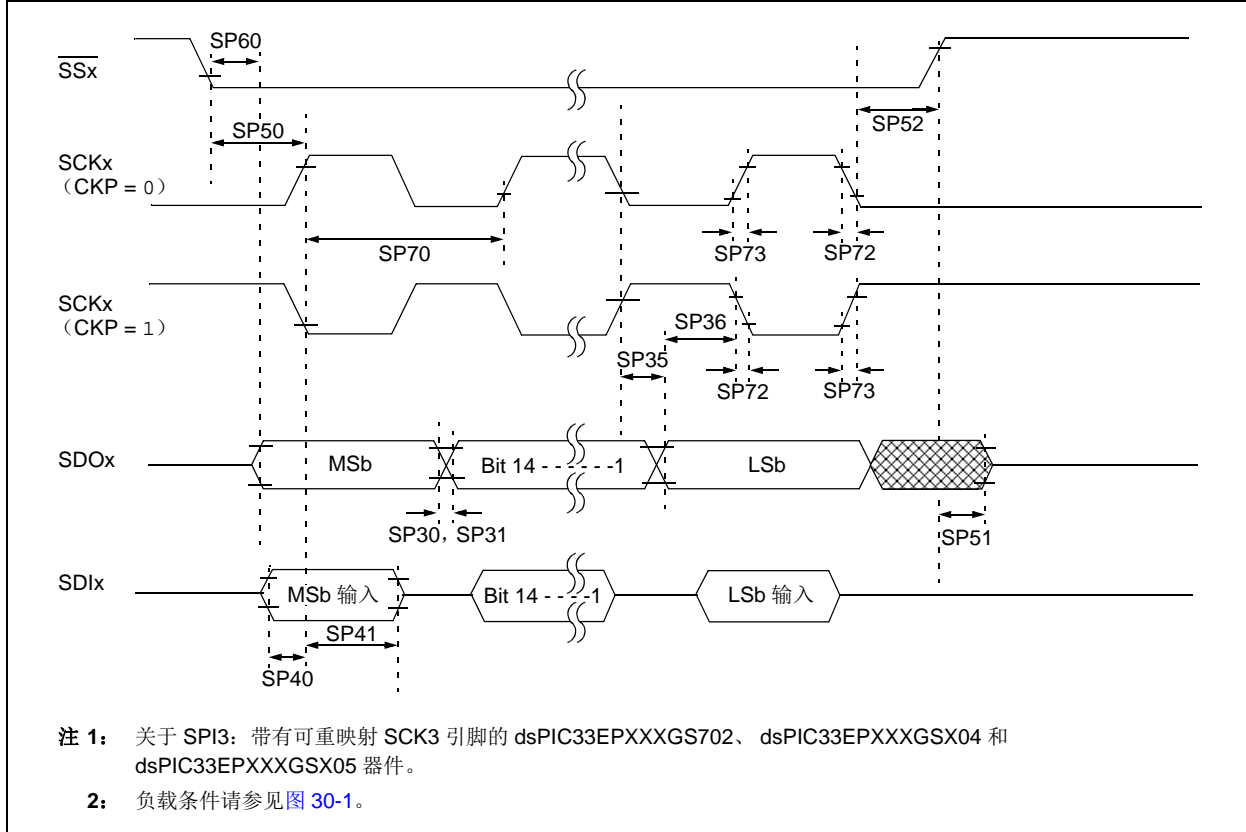
3: SCKx 的最小时钟周期为 66.7 ns。因此，主器件产生的 SCKx 时钟不应违反此规范。

4: 假定所有 SPIx 引脚上的负载均为 50 pF。

5: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXXGS702、dsPIC33EPXXXGSX04 和 dsPIC33EPXXXGSX05 器件。

# dsPIC33EPXXXGS70X/80X 系列

图 30-16: SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 1, CKP = 1, SMP = 0) 时序特性 (1,2)



# dsPIC33EPXXGS70X/80X 系列

表 30-36: SPI1、SPI2 和 SPI3 从模式（全双工，CKE = 1，CKP = 1，SMP = 0）时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCKx 输入频率	—	—	11	MHz	(注 3)
SP72	TscF	SCKx 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCKx 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx 数据输出建立到出现第一个 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	30	—	—	ns	
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	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	10	—	50	ns	(注 4)
SP52	TscH2ssH, TscL2ssH	SCKx ↑ 边沿之后 $\overline{SSx}$ 有效的 时间	1.5 Tcy + 40	—	—	ns	(注 4)
SP60	TssL2doV	$\overline{SSx}$ 边沿之后 SDOx 数据输出有效的时间	—	—	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 3.3V 和 +25°C 条件下的值。

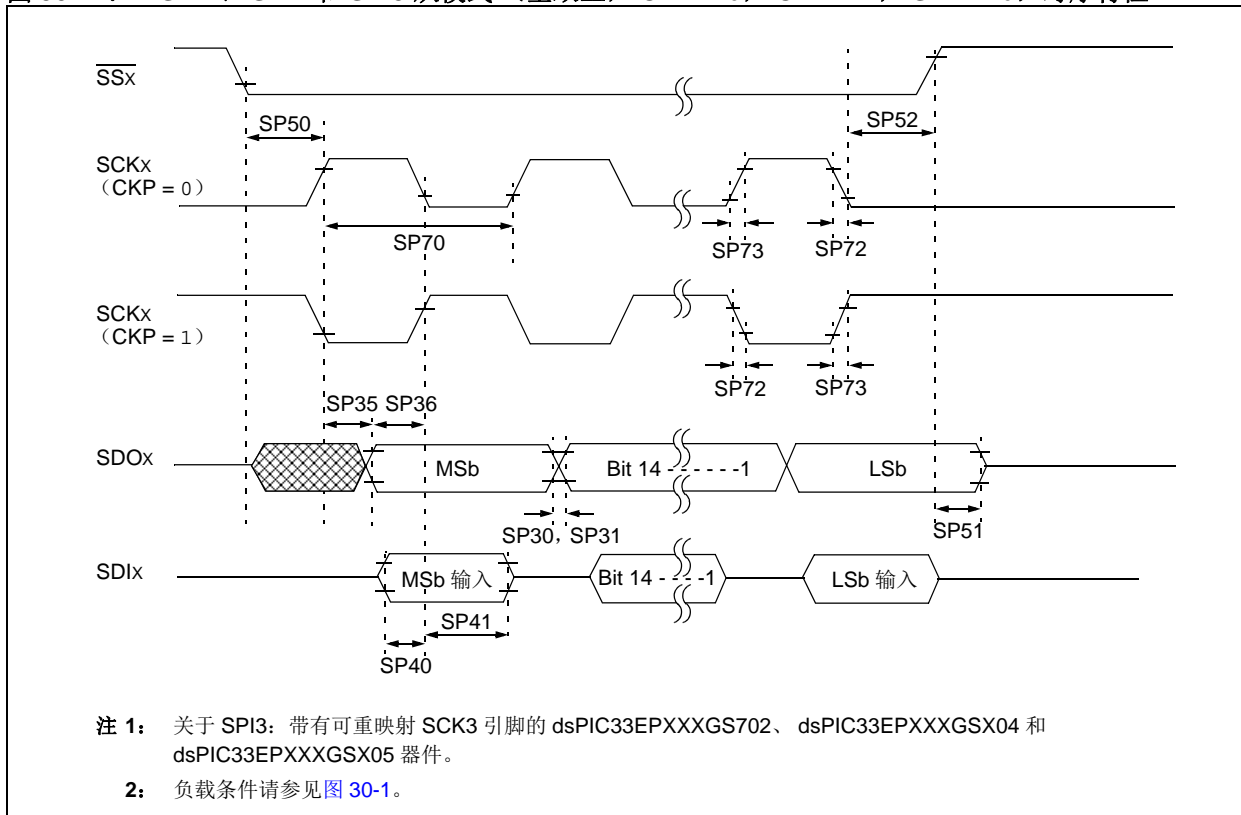
3: SCKx 的最小时钟周期为 91 ns。因此，主器件产生的 SCKx 时钟不应违反此规范。

4: 假定所有 SPIx 引脚上的负载均为 50 pF。

5: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXXGS702、dsPIC33EPXXXGSX04 和 dsPIC33EPXXXGSX05 器件。

# dsPIC33EPXXXGS70X/80X 系列

图 30-17: SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 0, CKP = 1, SMP = 0) 时序特性 (1,2)



# dsPIC33EPXXGS70X/80X 系列

表 30-37: SPI1、SPI2 和 SPI3 从模式（全双工，CKE = 0，CKP = 1，SMP = 0）时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCKx 输入频率	—	—	15	MHz	(注 3)
SP72	TscF	SCKx 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCKx 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx 数据输出建立到出现第一个 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	30	—	—	ns	
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	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	10	—	50	ns	(注 4)
SP52	TscH2ssH, TscL2ssH	SCKx ↑ 边沿之后 $\overline{SSx}$ 有效的 时间	1.5 Tcy + 40	—	—	ns	(注 4)

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 3.3V 和 +25°C 条件下的值。

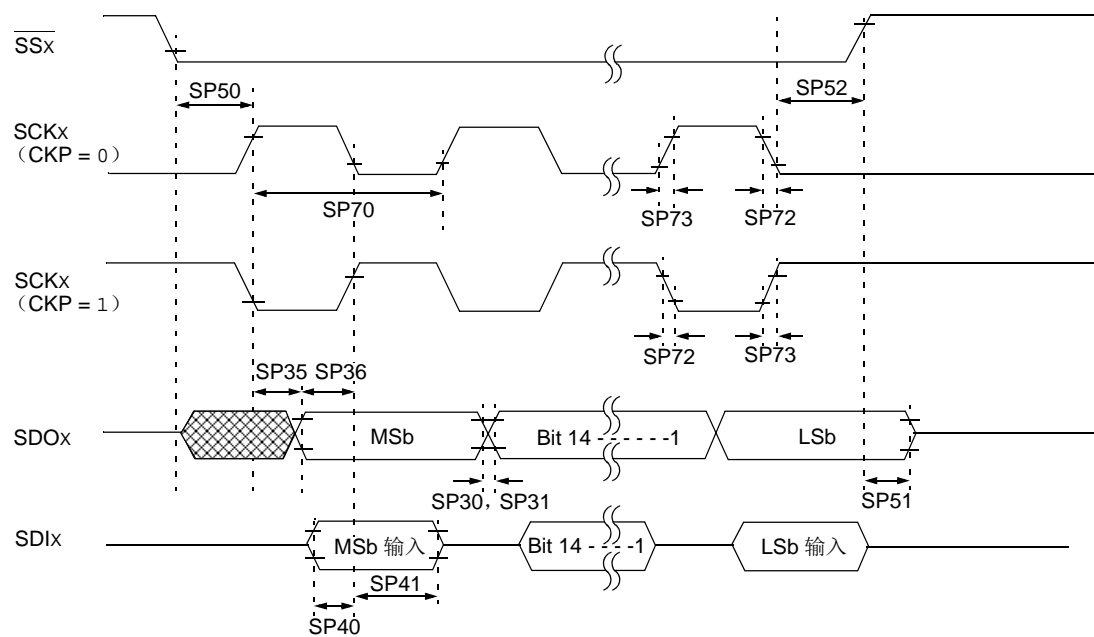
3: SCKx 的最小时钟周期为 66.7 ns。因此，主器件产生的 SCKx 时钟不应违反此规范。

4: 假定所有 SPIx 引脚上的负载均为 50 pF。

5: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXXGS702、dsPIC33EPXXXGSX04 和 dsPIC33EPXXXGSX05 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-18: SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 0, CKP = 0, SMP = 0) 时序特性 (1,2)



注 1: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXGS702、dsPIC33EPXXGSX04 和 dsPIC33EPXXGSX05 器件。

2: 负载条件请参见图 30-1。



# dsPIC33EPXXXGS70X/80X 系列

表 30-38: SPI1、SPI2 和 SPI3 从模式（全双工，CKE = 0，CKP = 0，SMP = 0）时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCKx 输入频率	—	—	11	MHz	(注 3)
SP72	TscF	SCKx 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCKx 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDOx 数据输出建立到出现第一个 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	30	—	—	ns	
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	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	10	—	50	ns	(注 4)
SP52	TscH2ssH, TscL2ssH	SCKx ↑ 边沿之后 $\overline{SSx}$ 有效的时间	1.5 Tcy + 40	—	—	ns	(注 4)

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 3.3V 和 +25°C 条件下的值。

3: SCKx 的最小时钟周期为 91 ns。因此，主器件产生的 SCKx 时钟不应违反此规范。

4: 假定所有 SPIx 引脚上的负载均为 50 pF。

5: 关于 SPI3: 带有可重映射 SCK3 引脚的 dsPIC33EPXXXGS702、dsPIC33EPXXXGSX04 和 dsPIC33EPXXXGSX05 器件。

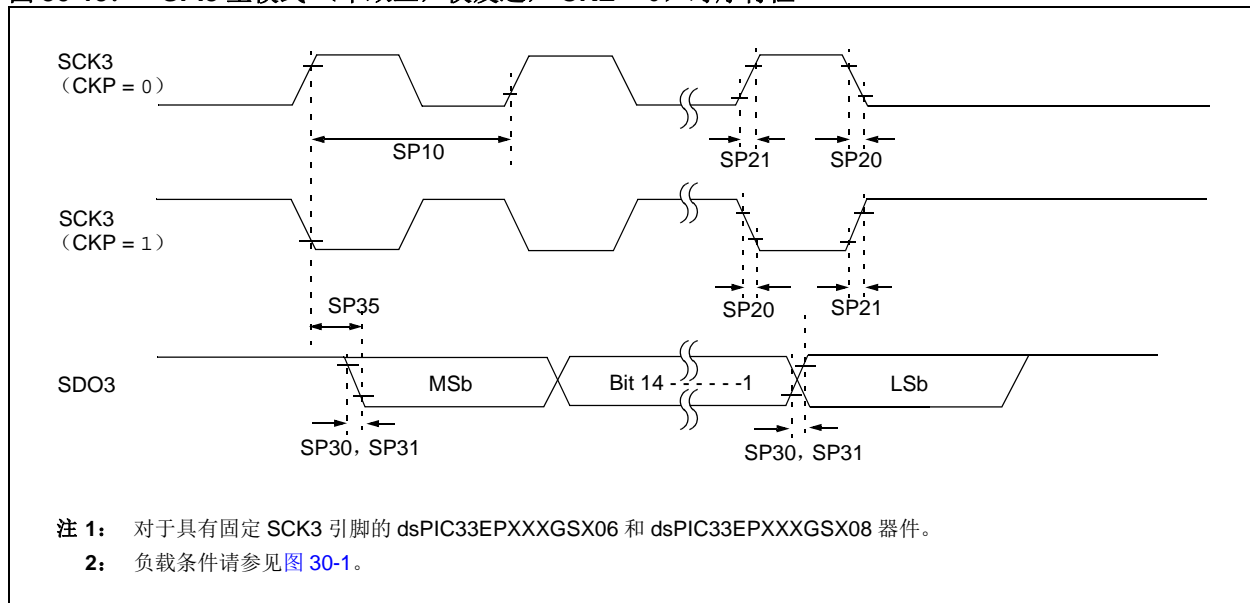
# dsPIC33EPXXGS70X/80X 系列

表 30-39: SPI3 最大数据 / 时钟速率汇总 (1)

交流特性				标准工作条件: 3.0V 至 3.6V (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)		
最大数据速率	主模式仅发送 (半双工)	主模式发送 / 接收 (全双工)	从模式发送 / 接收 (全双工)	CKE	CKP	SMP
25 MHz	表 30-40	—	—	0,1	0,1	0,1
25 MHz	—	表 30-41	—	1	0,1	1
25 MHz	—	表 30-42	—	0	0,1	1
25 MHz	—	—	表 30-43	1	0	0
25 MHz	—	—	表 30-44	1	1	0
25 MHz	—	—	表 30-45	0	1	0
25 MHz	—	—	表 30-46	0	0	0

注 1: 对于具有固定 SCK3 引脚的 dsPIC33EPXXGSX06 和 dsPIC33EPXXGSX08 器件。

图 30-19: SPI3 主模式 (半双工, 仅发送, CKE = 0) 时序特性 (1,2)



# dsPIC33EPXXGS70X/80X 系列

图 30-20: SPI3 主模式 (半双工, 仅发送, CKE = 1) 时序特性 (1,2)

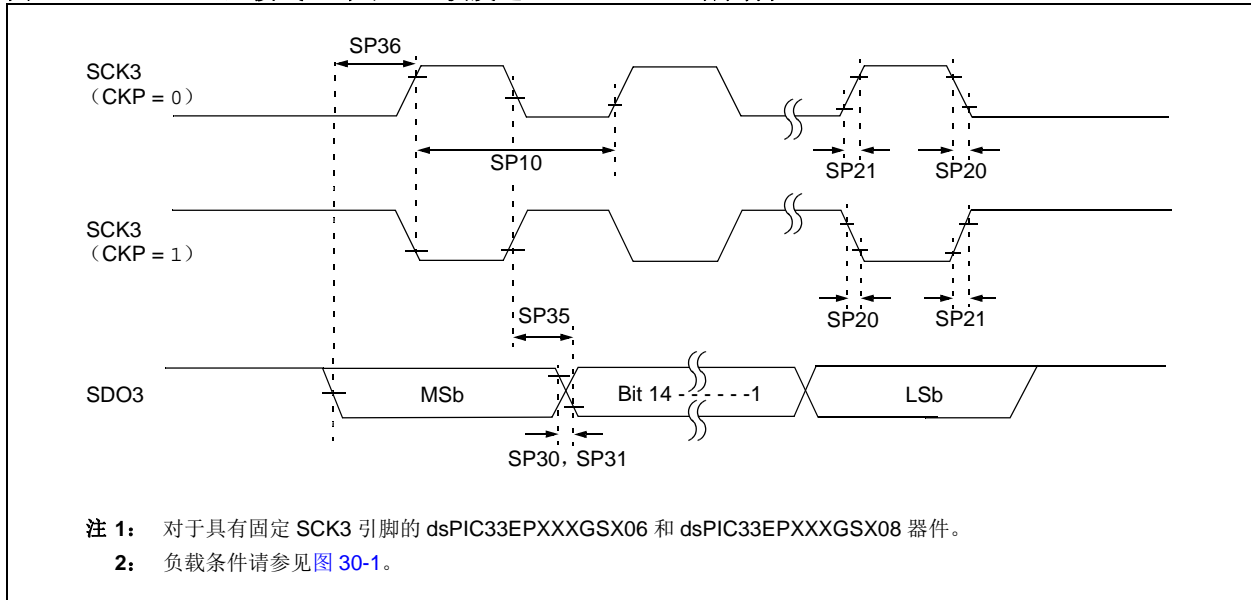


表 30-40: SPI3 主模式 (半双工, 仅发送) 时序要求 (5)

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	FscP	最大 SCK3 频率	—	—	25	MHz	(注 3)
SP20	TscF	SCK3 输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP21	TscR	SCK3 输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDO3 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDO3 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	Tsch2doV, TscL2doV	SCK3 边沿之后 SDO3 数据输出有效的时间	—	6	20	ns	
SP36	TdiV2scH, TdiV2scL	SDO3 数据输出建立到出现第一个 SCK3 边沿的时间	20	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

注 3: SCK3 的最小时钟周期为 66.7 ns。因此, 主模式下产生的时钟不应违反此规范。

注 4: 假定所有 SPI3 引脚上的负载均为 50 pF。

注 5: 对于具有固定 SCK3 引脚的 dsPIC33EPXXGSX06 和 dsPIC33EPXXGSX08 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-21: SPI3 主模式 (全双工,  $CKE = 1$ ,  $CKP = x$ ,  $SMP = 1$ ) 时序特性<sup>(1,2)</sup>

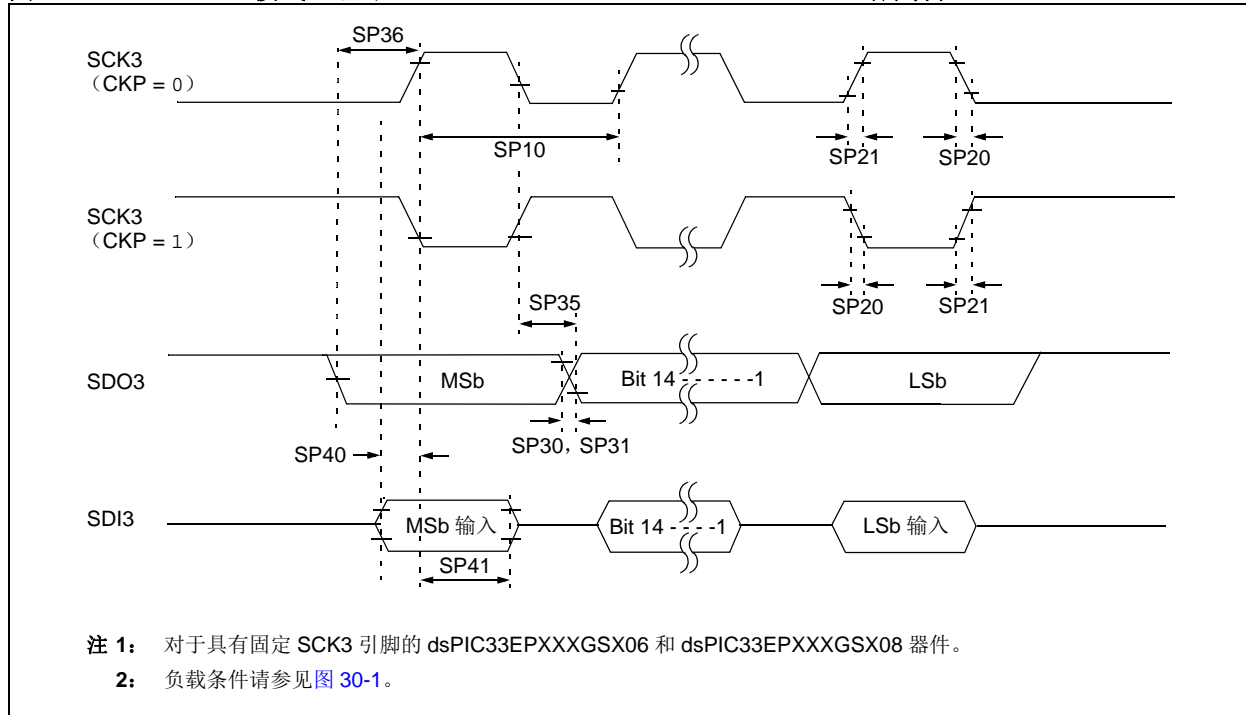


表 30-41: SPI3 主模式 (全双工,  $CKE = 1$ ,  $CKP = x$ ,  $SMP = 1$ ) 时序要求<sup>(5)</sup>

交流特性		标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP10	FscP	最大 SCK3 频率	—	—	25	MHz	(注 3)
SP20	TscF	SCK3 输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP21	TscR	SCK3 输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDO3 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDO3 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	Tsch2doV, TscL2doV	SCK3 边沿之后 SDO3 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2sc, TdoV2scL	SDO3 数据输出建立到出现第一个 SCK3 边沿的时间	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	SDI3 数据输入到 SCK3 边沿的建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDI3 数据输入到 SCK3 边沿的保持时间	15	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和  $+25^{\circ}\text{C}$  条件下的值。

注 3: SCK3 的最小时钟周期为 100 ns。主模式下产生的时钟不应违反此规范。

注 4: 假定所有 SPI3 引脚上的负载均为 50 pF。

注 5: 对于具有固定 SCK3 引脚的 dsPIC33EPXXXGSX06 和 dsPIC33EPXXXGSX08 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-22: SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) 时序特性 (1,2)

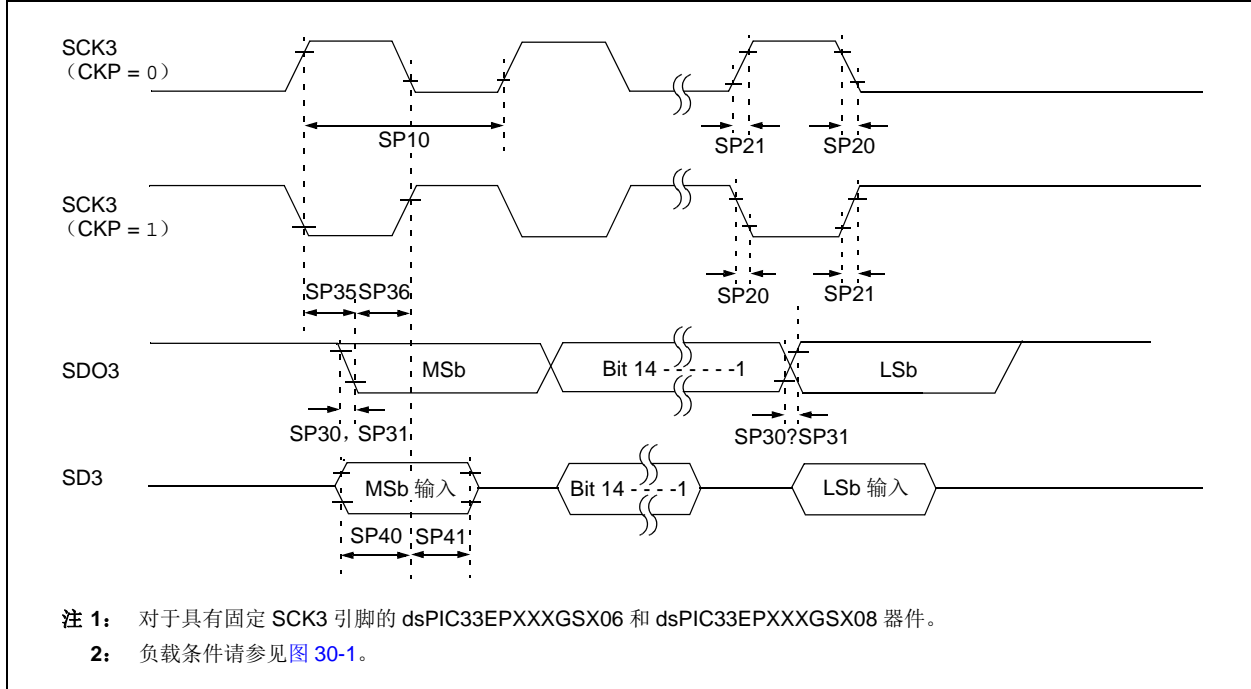


表 30-42: SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) 时序要求 (5)

交流特性		标准工作条件: 3.0V 至 3.6V (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	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	FscP	最大 SCK3 频率	—	—	25	MHz	-40°C 至 +125°C (注 3)
SP20	TscF	SCK3 输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP21	TscR	SCK3 输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDO3 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDO3 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	Tsch2doV, TscL2doV	SCK3 边沿之后 SDO3 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 数据输出建立到出现第一个 SCK3 边沿的时间	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	SDI3 数据输入到 SCK3 边沿的建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDI3 数据输入到 SCK3 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。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。

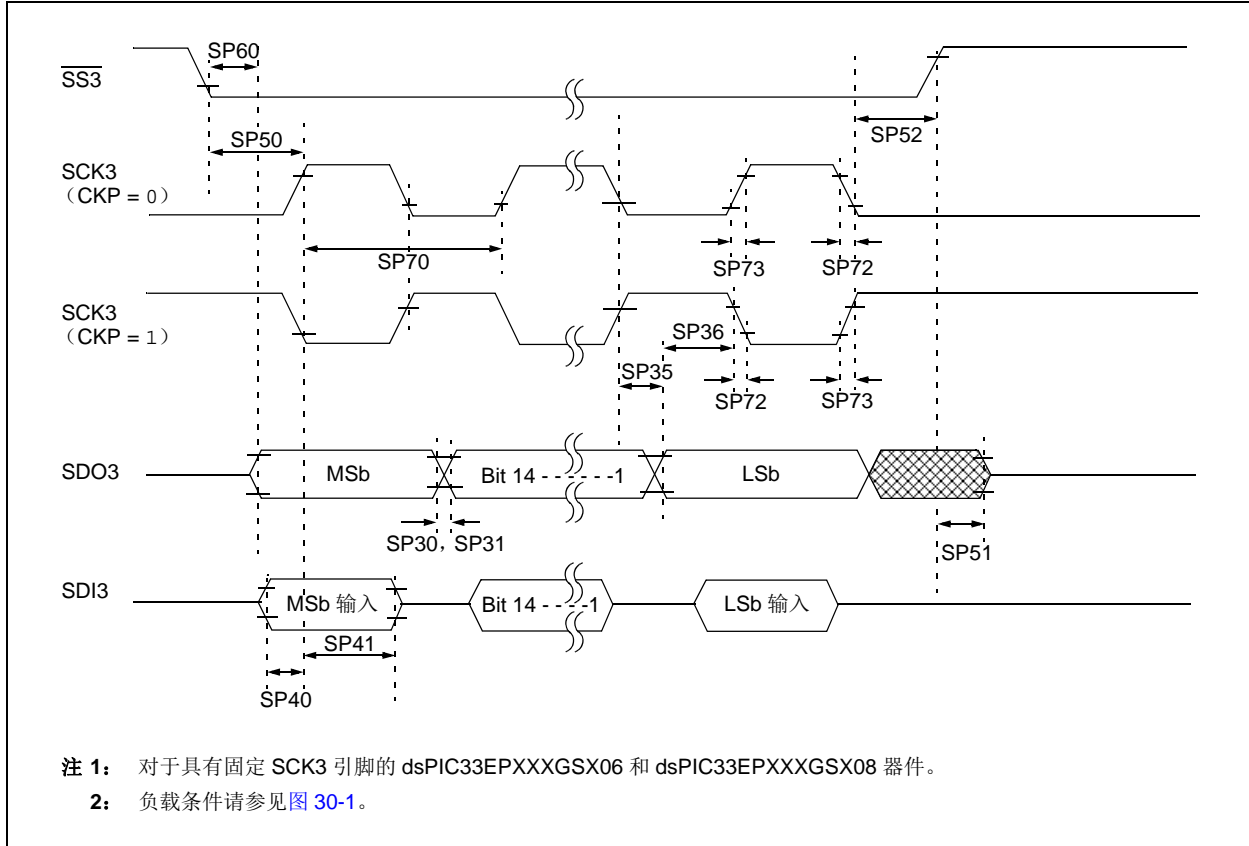
注 3: SCK3 的最小时钟周期为 100 ns。主模式下产生的时钟不应违反此规范。

注 4: 假定所有 SPI3 引脚上的负载均为 50 pF。

注 5: 对于具有固定 SCK3 引脚的 dsPIC33EPXXGSX06 和 dsPIC33EPXXGSX08 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-23: SPI3 从模式 (全双工,  $CKE = 1$ ,  $CKP = 0$ ,  $SMP = 0$ ) 时序特性 (1,2)



# dsPIC33EPXXGS70X/80X 系列

表 30-43: SPI3 从模式 (全双工, CKE = 1, CKP = 0, SMP = 0) 时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCK3 输入频率	—	—	25	MHz	(注 3)
SP72	TscF	SCK3 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCK3 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDO3 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDO3 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCK3 边沿之后 SDO3 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 数据输出建立到出现第一个 SCK3 边沿的时间	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	SDI3 数据输入到 SCK3 边沿的建立时间	20	—	—	ns	
SP41	TscH2diL, TscL2diL	SDI3 数据输入到 SCK3 边沿的保持时间	15	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SS3}$ ↓ 到 SCK3 ↑ 或 SCK3 ↓ 输入的时间	120	—	—	ns	
SP51	TssH2doZ	$\overline{SS3}$ ↑ 到 SDO3 输出高阻态的时间	10	—	50	ns	(注 4)
SP52	TscH2ssH, TscL2ssH	SCK3 边沿之后 $\overline{SS3}$ ↑ 有效的 时间	1.5 Tcy + 40	—	—	ns	(注 4)
SP60	TssL2doV	$\overline{SS3}$ 边沿之后 SDO3 数据输出有效的时间	—	—	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 3.3V 和 +25°C 条件下的值。

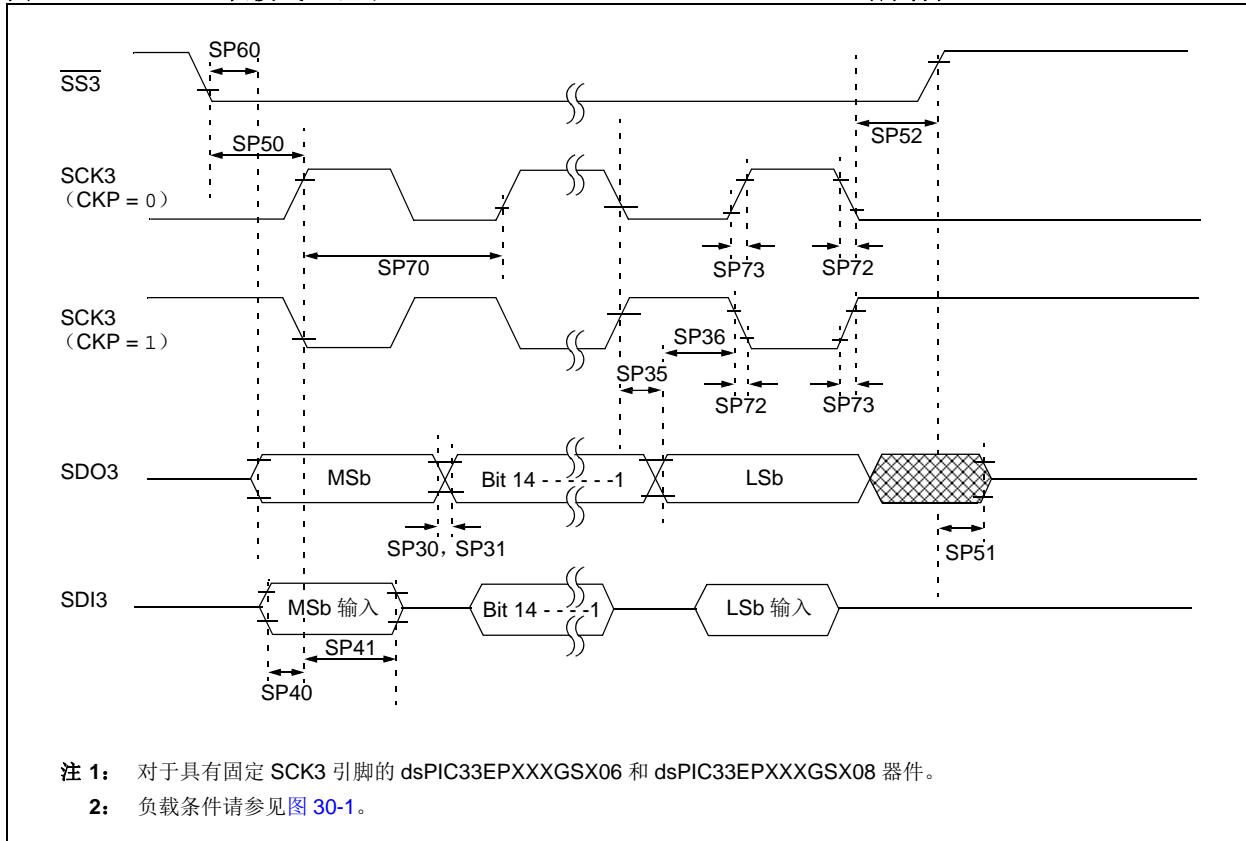
3: SCK3 的最小时钟周期为 66.7 ns。因此, 主器件产生的 SCK3 时钟不应违反此规范。

4: 假定所有 SPI3 引脚上的负载均为 50 pF。

5: 对于具有固定 SCK3 引脚的 dsPIC33EPXXXGSX06 和 dsPIC33EPXXXGSX08 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-24: SPI3 从模式 (全双工,  $\text{CKE} = 1$ ,  $\text{CKP} = 1$ ,  $\text{SMP} = 0$ ) 时序特性 (1,2)





# dsPIC33EPXXGS70X/80X 系列

表 30-44: SPI3 从模式 (全双工, CKE = 1, CKP = 1, SMP = 0) 时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCK3 输入频率	—	—	25	MHz	(注 3)
SP72	TscF	SCK3 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCK3 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDO3 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDO3 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCK3 边沿之后 SDO3 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 数据输出建立到出现第一个 SCK3 边沿的时间	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	SDI3 数据输入到 SCK3 边沿的建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDI3 数据输入到 SCK3 边沿的保持时间	15	—	—	ns	
SP50	Tssl2scH, Tssl2scL	$\overline{SS3}$ ↓ 到 SCK3 ↑ 或 SCK3 ↓ 输入的时间	120	—	—	ns	
SP51	TssH2doZ	$\overline{SS3}$ ↑ 到 SDO3 输出高阻态的时间	10	—	50	ns	(注 4)
SP52	Tsch2ssH, TscL2ssH	SCK3 边沿之后 $\overline{SS3}$ ↑ 有效的时间	1.5 Tcy + 40	—	—	ns	(注 4)
SP60	Tssl2doV	$\overline{SS3}$ 边沿之后 SDO3 数据输出有效的时间	—	—	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 3.3V 和 +25°C 条件下的值。

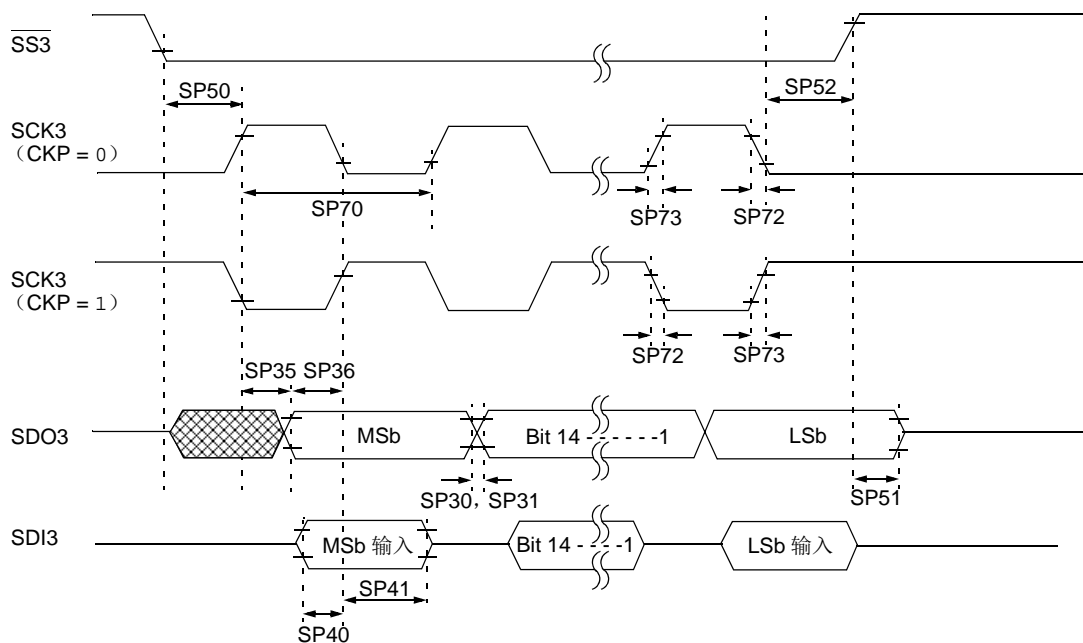
3: SCK3 的最小时钟周期为 91 ns。因此, 主器件产生的 SCK3 时钟不应违反此规范。

4: 假定所有 SPI3 引脚上的负载均为 50 pF。

5: 对于具有固定 SCK3 引脚的 dsPIC33EPXXXGSX06 和 dsPIC33EPXXXGSX08 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-25: SPI3 从模式 (全双工,  $\text{CKE} = 0$ ,  $\text{CKP} = 1$ ,  $\text{SMP} = 0$ ) 时序特性<sup>(1,2)</sup>



注 1: 对于具有固定 SCK3 引脚的 dsPIC33EPXXGSX06 和 dsPIC33EPXXGSX08 器件。

注 2: 负载条件请参见图 30-1。

# dsPIC33EPXXXGS70X/80X 系列

表 30-45: SPI3 从模式 (全双工, CKE = 0, CKP = 1, SMP = 0) 时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCK3 输入频率	—	—	25	MHz	(注 3)
SP72	TscF	SCK3 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCK3 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDO3 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDO3 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCK3 边沿之后 SDO3 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 数据输出建立到出现第一个 SCK3 边沿的时间	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	SDI3 数据输入到 SCK3 边沿的建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDI3 数据输入到 SCK3 边沿的保持时间	15	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SS3}$ ↓ 到 SCK3 ↑ 或 SCK3 ↓ 输入的时间	120	—	—	ns	
SP51	TssH2doZ	$\overline{SS3}$ ↑ 到 SDO3 输出高阻态的时间	10	—	50	ns	(注 4)
SP52	Tsch2ssH, TscL2ssH	SCK3 边沿之后 $\overline{SS3}$ ↑ 有效的 时间	1.5 Tcy + 40	—	—	ns	(注 4)

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 3.3V 和 +25°C 条件下的值。

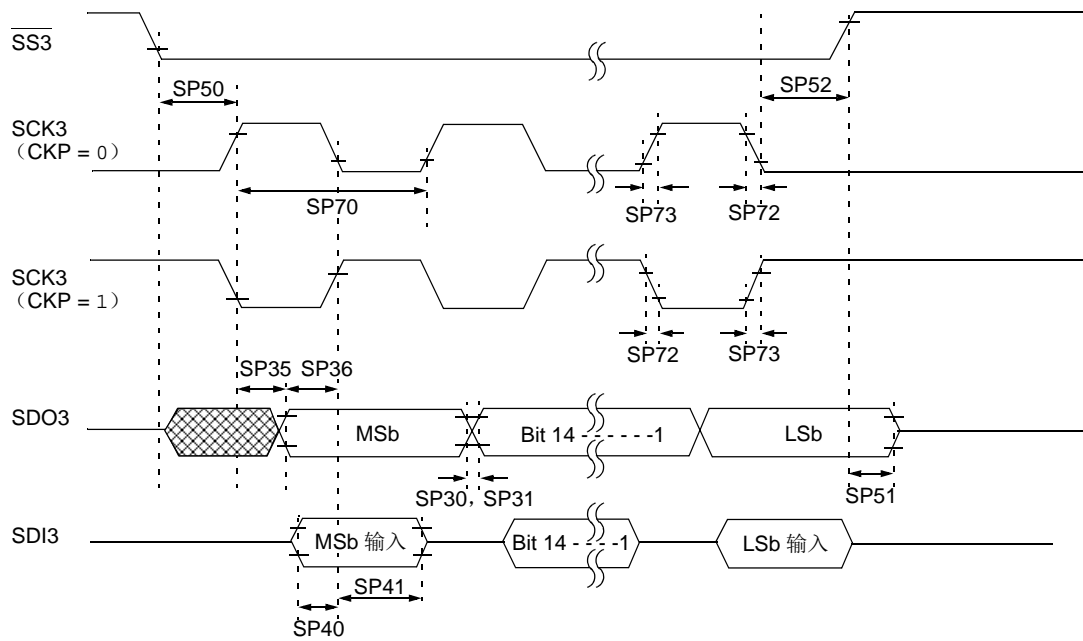
3: SCK3 的最小时钟周期为 66.7 ns。因此, 主器件产生的 SCK3 时钟不应违反此规范。

4: 假定所有 SPI3 引脚上的负载均为 50 pF。

5: 对于具有固定 SCK3 引脚的 dsPIC33EPXXXGSX06 和 dsPIC33EPXXXGSX08 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-26: SPI3 从模式 (全双工,  $\text{CKE} = 0$ ,  $\text{CKP} = 0$ ,  $\text{SMP} = 0$ ) 时序特性 (1,2)



- 注 1: 对于具有固定 SCK3 引脚的 dsPIC33EPXXGSX06 和 dsPIC33EPXXGSX08 器件。  
 注 2: 负载条件请参见图 30-1。

# dsPIC33EPXXGS70X/80X 系列

表 30-46: SPI3 从模式 (全双工, CKE = 0, CKP = 0, SMP = 0) 时序要求<sup>(5)</sup>

交流特性			标准工作条件: 3.0V 至 3.6V (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	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
SP70	FscP	最大 SCK3 输入频率	—	—	25	MHz	(注 3)
SP72	TscF	SCK3 输入下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP73	TscR	SCK3 输入上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP30	TdoF	SDO3 数据输出下降时间	—	—	—	ns	见参数 DO32 (注 4)
SP31	TdoR	SDO3 数据输出上升时间	—	—	—	ns	见参数 DO31 (注 4)
SP35	TscH2doV, TscL2doV	SCK3 边沿之后 SDO3 数据输出有效的时间	—	6	20	ns	
SP36	TdoV2scH, TdoV2scL	SDO3 数据输出建立到出现第一个 SCK3 边沿的时间	20	—	—	ns	
SP40	TdiV2scH, TdiV2scL	SDI3 数据输入到 SCK3 边沿的建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDI3 数据输入到 SCK3 边沿的保持时间	15	—	—	ns	
SP50	TssL2scH, TssL2scL	$\overline{SS3}$ ↓ 到 SCK3 ↑ 或 SCK3 ↓ 输入的时间	120	—	—	ns	
SP51	TssH2doZ	$\overline{SS3}$ ↑ 到 SDO3 输出高阻态的时间	10	—	50	ns	(注 4)
SP52	Tsch2ssH, TscL2ssH	SCK1 边沿之后 $\overline{SS3}$ ↑ 有效的	1.5 Tcy + 40	—	—	ns	(注 4)

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 3.3V 和 +25°C 条件下的值。

3: SCK3 的最小时钟周期为 91 ns。因此, 主器件产生的 SCK3 时钟不应违反此规范。

4: 假定所有 SPI3 引脚上的负载均为 50 pF。

5: 对于具有固定 SCK3 引脚的 dsPIC33EPXXXGSX06 和 dsPIC33EPXXXGSX08 器件。

# dsPIC33EPXXGS70X/80X 系列

图 30-27: I2Cx 总线启动位 / 停止位时序特性 (主模式)

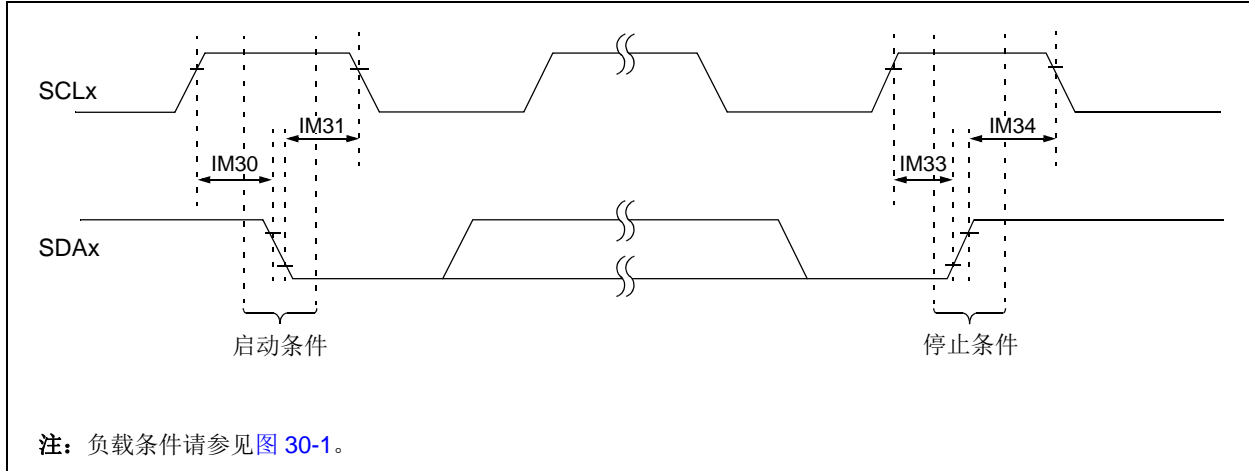
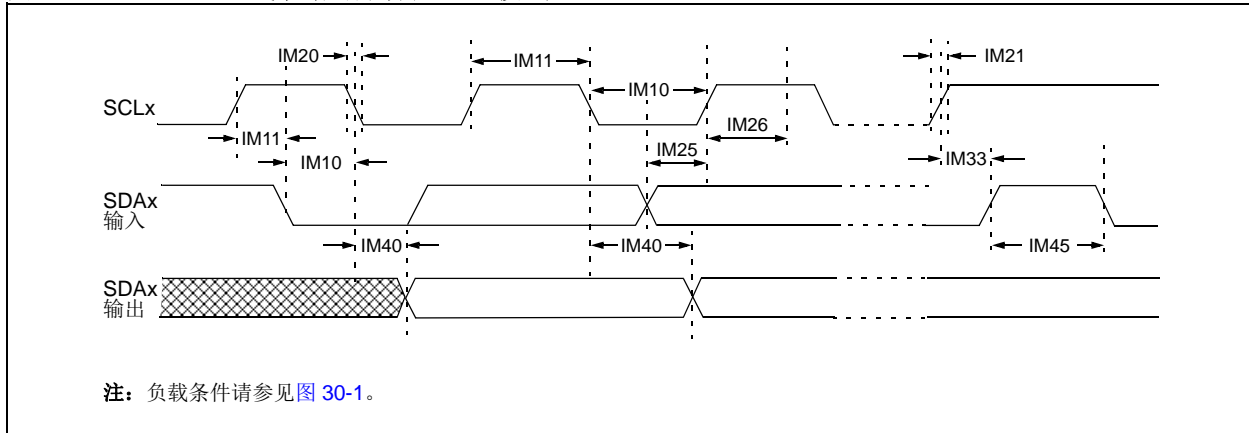


图 30-28: I2Cx 总线数据时序特性 (主模式)



# dsPIC33EPXXGS70X/80X 系列

表 30-47: I2Cx 总线数据时序要求 (主模式)

交流特性				标准工作条件: 3.0V 至 3.6V (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 <sup>(4)</sup>		最小值 <sup>(1)</sup>	最大值	单位	条件
IM10	TLO:SCL	时钟低电平时间	100 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			400 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 2)	—	μs	
IM11	THI:SCL	时钟高电平时间	100 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			400 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 2)	—	μs	
IM20	TF:SCL	SDAx 和 SCLx 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 <sup>(2)</sup>	—	100	ns	
IM21	TR:SCL	SDAx 和 SCLx 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 <sup>(2)</sup>	—	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 <sup>(2)</sup>	40	—	ns	
IM26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	μs	
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 <sup>(2)</sup>	0.2	—	μs	
IM30	TSU:STA	启动条件建立时间	100 kHz 模式	Tcy/2 (BRG + 2)	—	μs	仅与重复启动条件相关条件
			400 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 2)	—	μs	
IM31	THD:STA	启动条件保持时间	100 kHz 模式	Tcy/2 (BRG + 2)	—	μs	这个周期后产生第一个时钟脉冲
			400 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 2)	—	μs	
IM33	TSU:STO	停止条件建立时间	100 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			400 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 2)	—	μs	
IM34	THD:STO	停止条件保持时间	100 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			400 kHz 模式	Tcy/2 (BRG + 2)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 2)	—	μs	
IM40	TAA:SCL	自时钟边沿到输出有效的的时间	100 kHz 模式	—	3500	ns	
			400 kHz 模式	—	1000	ns	
			1 MHz 模式 <sup>(2)</sup>	—	400	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 <sup>(2)</sup>	0.5	—	μs	
IM50	Cb	总线容性负载	—	—	400	pF	
IM51	TPGD	脉冲干扰抑制电路 (Pulse Gobbler) 延时	—	65	—	ns	(注 3)

注 1: BRG 为 I2C 波特率发生器的值。

2: 所有 I2Cx 引脚的最大引脚电容为 10 pF (仅对于 1 MHz 模式)。

3: 该参数的典型值为 130 ns。

4: 这些参数为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

图 30-29: I2Cx 总线启动位 / 停止位时序特性 (从模式)

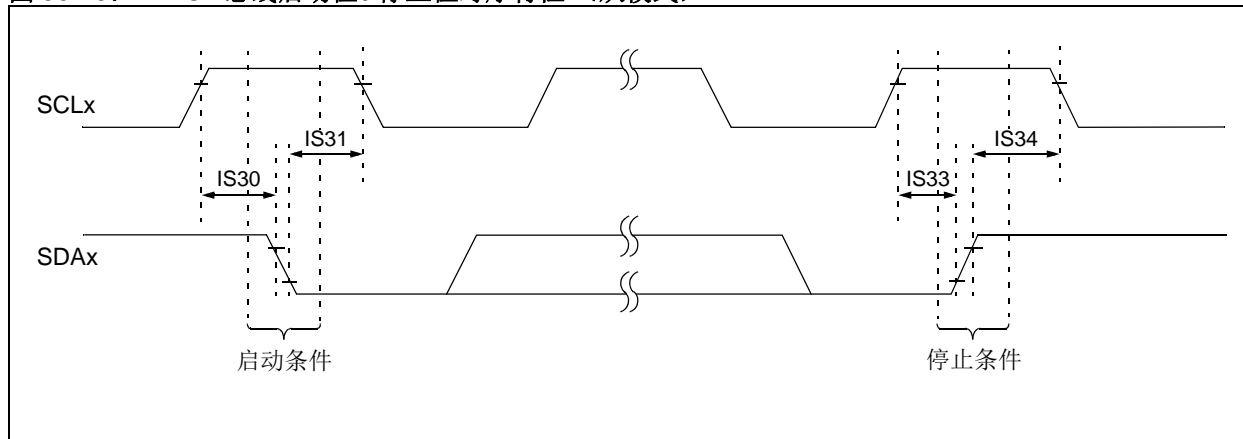
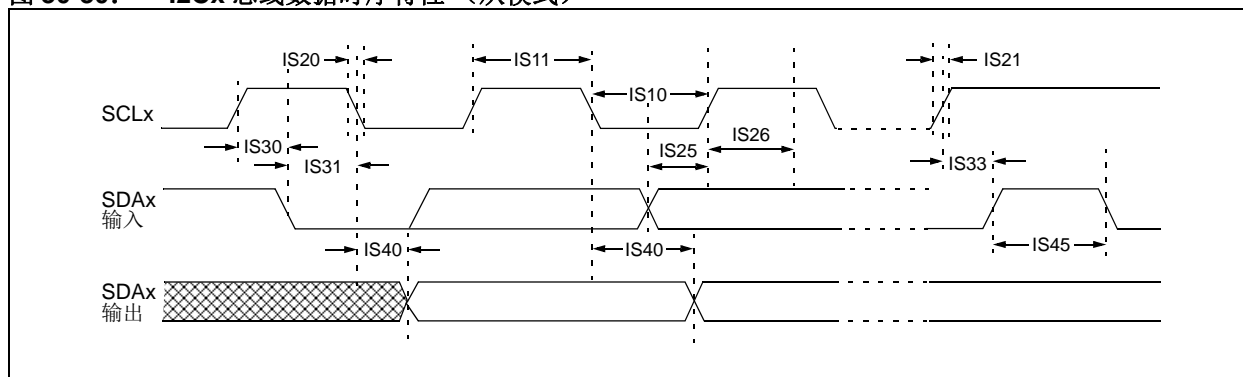


图 30-30: I2Cx 总线数据时序特性 (从模式)





# dsPIC33EPXXGS70X/80X 系列

表 30-48: I2Cx 总线数据时序要求 (从模式)

交流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(3)</sup>	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	
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 <sup>(1)</sup>	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 <sup>(1)</sup>	0.5	—	μs	
IS20	TF:SCL	SDAx 和 SCLx 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 <sup>(1)</sup>	—	100	ns	
IS21	TR:SCL	SDAx 和 SCLx 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 <sup>(1)</sup>	—	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 <sup>(1)</sup>	100	—	ns	
IS26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	μs	
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 <sup>(1)</sup>	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 <sup>(1)</sup>	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 <sup>(1)</sup>	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 <sup>(1)</sup>	0.6	—	μs	
IS34	THD:STO	停止条件保持时间	100 kHz 模式	4	—	μs	
			400 kHz 模式	0.6	—	μs	
			1 MHz 模式 <sup>(1)</sup>	0.25	—	μs	
IS40	TAA:SCL	自时钟边沿到输出有效的的时间	100 kHz 模式	0	3500	ns	
			400 kHz 模式	0	1000	ns	
			1 MHz 模式 <sup>(1)</sup>	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 <sup>(1)</sup>	0.5	—	μs	
IS50	CB	总线容性负载	—	400	pF		
IS51	TPGD	脉冲干扰抑制电路 (Pulse Gobbler) 延时	65	390	ns	(注 2)	

注 1: 所有 I2Cx 引脚的最大引脚电容为 10 pF (仅对于 1 MHz 模式)。

2: 该参数的典型值为 130 ns。

3: 这些参数为特性值, 但生产时未经测试。

# dsPIC33EPXXGS70X/80X 系列

图 30-31: CANx 模块 I/O 时序特性

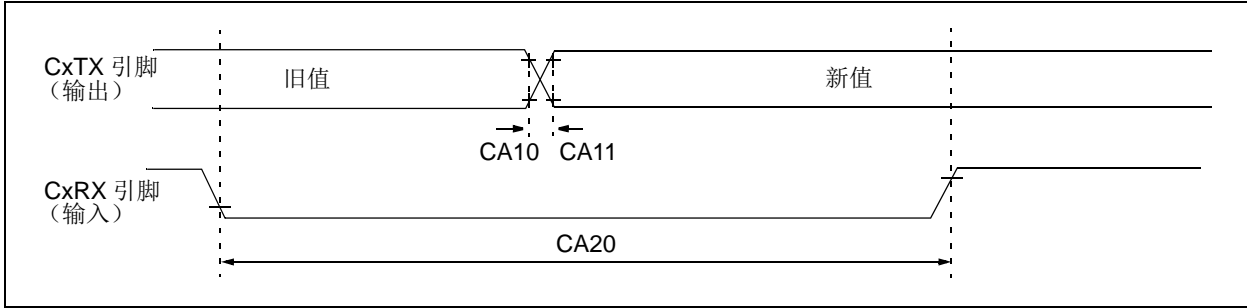


表 30-49: CANx 模块 I/O 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
CA10	TioF	端口输出下降时间	—	—	—	ns	见参数 DO32
CA11	TioR	端口输出上升时间	—	—	—	ns	见参数 DO31
CA20	TCWF	触发 CAN 唤醒滤波器的脉冲宽度	120	—	—	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 3.3V 和 +25°C 条件下的值。这些参数仅供设计参考, 未经测试。

# dsPIC33EPXXGS70X/80X 系列

图 30-32: UARTx 模块 I/O 时序特性

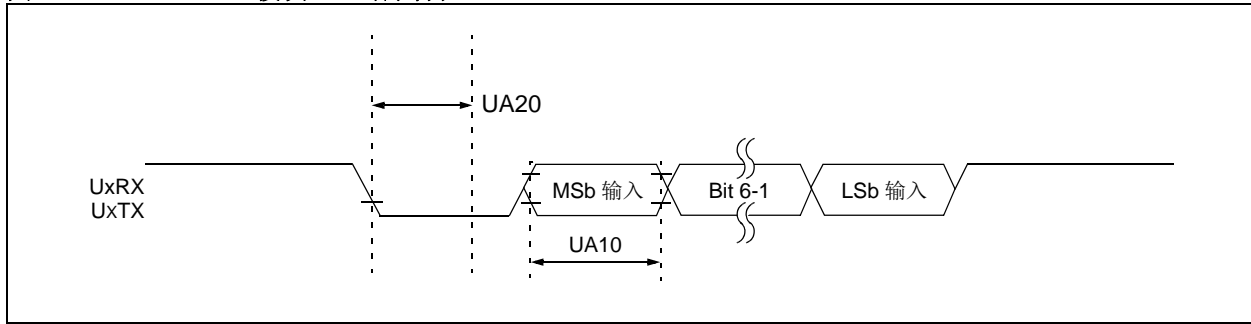


表 30-50: UARTx 模块 I/O 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$				
参数编号	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
UA10	TUABAUD	UARTx 波特率时间	66.67	—	—	ns	
UA11	FBAUD	UARTx 波特率频率	—	—	15	Mbps	
UA20	TCWF	触发 UARTx 唤醒的启动位脉冲宽度	500	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。这些参数仅供设计参考, 未经测试。

表 30-51: 模拟电流规范

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$				
参数编号	符号	特性 <sup>(1)</sup>	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
AVD01	IDD	模拟模块电流消耗	—	9	—	mA	使能以下模块时的特性数据: APLL、5 个 ADC 内核、 2 个 PGA 和 4 个模拟比较器

注 1: 这些参数为特性值, 但生产时未经测试。

注 2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 +25°C 条件下的值。这些参数仅供设计参考, 未经测试。

# dsPIC33EPXXGS70X/80X 系列

表 30-52: ADC 模块规范

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) (5) 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
<b>器件电源</b>							
AD01	AVDD	模块电源 VDD	取如下二者中较大值: VDD - 0.3 或 3.0	—	取如下二者中较小值: VDD + 0.3 或 3.6	V	总是处于 VDD 的 ±300 mV 范围内, 包括器件上电
AD02	AVSS	模块电源 Vss	Vss	—	Vss + 0.3	V	
<b>参考输入</b>							
AD06	VREFL	参考电压低电压	—	AVSS	—	V	(注 1)
AD07	VREF	绝对参考电压 (VREFH - VREFL)	2.7	—	AVDD	V	(注 3)
AD08	IREF	参考输入电流	—	5	10	μA	ADC 工作或处于待机状态
<b>模拟输入</b>							
AD12	VINH-VINL	满量程输入范围	AVSS	—	AVDD	V	
AD14	VIN	绝对输入电压	AVSS - 0.3	—	AVDD + 0.3	V	
AD17	RIN	模拟电压源的推荐阻抗	—	100	—	Ω	对于最小采样时间 (注 1)
AD66	Vbg	内部参考电压源	—	1.2	—	V	
<b>ADC 精度: 伪差分输入</b>							
AD20a	Nr	分辨率	12			位	
AD21a	INL	积分非线性误差	> -3	—	< 3	LSb	AVSS = 0V, AVDD = 3.3V
AD22a	DNL	微分非线性误差	> -1	—	< 1	LSb	AVSS = 0V, AVDD = 3.3V (注 2)
AD23a	GERR	增益误差 (专用内核)	> 0	8	< 15	LSb	AVSS = 0V, AVDD = 3.3V
		增益误差 (共用内核)	> 5	15	< 22	LSb	
AD24a	EOFF	失调误差 (专用内核)	> 0	5	< 10	LSb	AVSS = 0V, AVDD = 3.3V
		失调误差 (共用内核)	> 2	8	< 13	LSb	
AD25a	—	单调性	—	—	—	—	保证

注 1: 这些参数不是特性值, 或生产时未经测试。

2: 不会丢失编码, 限制基于特性结果。

3: 这些参数为特性值, 但生产时未经测试。

4: 采用 15 kHz 正弦波信号确定特性值。

5: ADC 模块可在 VBORMIN < VDD < VDDMIN 条件下工作, 但性能将下降。除非另外声明, 否则可确保模块功能, 但未确定特性值。

# dsPIC33EPXXGS70X/80X 系列

表 30-52: ADC 模块规范 (续)

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) <sup>(5)</sup> 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
<b>ADC 精度: 单端输入</b>							
AD20b	Nr	分辨率	12			位	
AD21b	INL	积分非线性误差	> 5	—	< 5	LSb	AVSS = 0V, AVDD = 3.3V
AD22b	DNL	微分非线性误差	> -1	—	< 1	LSb	AVSS = 0V, AVDD = 3.3V (注 2)
AD23b	GERR	增益误差 (专用内核)	> 0	8	< 15	LSb	AVSS = 0V, AVDD = 3.3V
		增益误差 (共用内核)	> 5	15	< 22	LSb	
AD24b	EOFF	失调误差 (专用内核)	> 2	9	< 15	LSb	AVSS = 0V, AVDD = 3.3V
		失调误差 (共用内核)	> 5	17	< 22	LSb	
AD25b	—	单调性	—	—	—	—	保证
<b>动态性能</b>							
AD31b	SINAD	信噪比和失真	63	—	> 65	dB	(注 3, 4)
AD34b	ENOB	有效位数	10.3	—	—	位	(注 3, 4)

注 1: 这些参数不是特性值, 或生产时未经测试。

2: 不会丢失编码, 限制基于特性结果。

3: 这些参数为特性值, 但生产时未经测试。

4: 采用 15 kHz 正弦波信号确定特性值。

5: ADC 模块可在 VBORMIN < VDD < VDDMIN 条件下工作, 但性能将下降。除非另外声明, 否则可确保模块功能, 但未确定特性值。

# dsPIC33EPXXGS70X/80X 系列

表 30-53: 模数转换时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) (2) 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	ADC 时钟周期	14.28	—	—	ns	
吞吐率							
AD51	FTP	SH0-SH3	—	—	3.25	Msp/s	70 MHz ADC 时钟, 12 位, 触发时无待处理的转换
		SH4	—	—	3.25	Msp/s	

注 1: 这些参数为特性值, 但生产时未经测试。

2: ADC 模块可在 VBORMIN < VDD < VDDMIN 条件下工作, 但性能将下降。除非另外声明, 否则可确保模块功能, 但未确定特性值。

表 30-54: 高速模拟比较器模块规范

交流 / 直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) (2) 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	备注
CM10	VIOFF	输入失调电压	-35	±5	35	mV	
CM11	VICM	输入共模电压范围 (1)	0	—	AVDD	V	
CM13	CMRR	共模抑制比	60	—	—	dB	
CM14	TRESP	大信号响应	—	15	—	ns	V- 输入保持在 AVDD/2 时 V+ 输入 100 mV 阶跃信号。延时从模拟输入引脚到 PWMx 输出引脚测得。
CM15	VHYST	输入滞后	5	10	20	mV	取决于 HYSSEL<1:0>
CM16	TON	比较器使能到输出有效的时间	—	—	1	µs	

注 1: 这些参数仅供设计参考, 生产时未经测试。

2: 比较器模块可在 VBORMIN < VDD < VDDMIN 条件下工作, 但性能将下降。除非另外声明, 否则模块功能经过测试, 但未确定特性值。

# dsPIC33EPXXGS70X/80X 系列

表 30-55: DACx 模块规范

交流 / 直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) (2) 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	备注
DA01	EXTREF	外部参考电压 (1)	1	—	AVDD	V	
DA02	CVRES	分辨率	12			位	
DA03	INL	积分非线性误差	-16	-12	0	LSB	
DA04	DNL	微分非线性误差	-1.8	±1	1.8	LSB	
DA05	EOFF	失调误差	-8	3	15	LSB	
DA06	EG	增益误差	-1.2	-0.5	0	误差	
DA07	TSET	稳定时间 (1)	—	700	—	ns	对于 10-90% 或 90-10% 的阶跃, 输出电压容差达到稳定在所需输出电压的 ±2% 范围内的时间

注 1: 这些参数仅供设计参考, 生产时未经测试。

2: DACx 模块可在 VBORMIN < VDD < VDDMIN 条件下工作, 但性能将下降。除非另外声明, 否则模块功能经过测试, 但未确定特性值。

表 30-56: DACx 输出 (DACOUTx 引脚) 规范

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) (1) 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	备注
DA11	RLOAD	阻性输出负载阻抗	10k	—	—	Ω	
DA11a	CLOAD	输出负载电容	—	—	35	pF	包括输出引脚电容
DA12	IOUT	输出电流驱动能力	—	300	—	μA	灌 / 拉电流
DA13	VRANGE	电流驱动 (300 μA) 时的输出驱动电压范围	AVSS + 250 mV	—	AVDD - 900 mV	V	
DA14	VLRANGE	降额电流驱动 (50 μA) 时的输出驱动电压范围	AVSS + 50 mV	—	AVDD - 500 mV	V	
DA15	IDD	模块使能时消耗的电流	—	—	1.3 x IOUT	μA	模块将总是消耗该电流, 即使没有负载连接到输出
DA30	VOFFSET	输入失调电压	—	±5	—	mV	

注 1: DACx 模块可在 VBORMIN < VDD < VDDMIN 条件下工作, 但性能将下降。除非另外声明, 否则模块功能经过测试, 但未确定特性值。

# dsPIC33EPXXGS70X/80X 系列

表 30-57: PGAx 模块规范

交流 / 直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) <sup>(1)</sup> 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	备注	
PA01	VIN	输入电压范围	AVSS - 0.3	—	AVDD + 0.3	V		
PA02	VCM	共模输入电压范围	AVSS	—	AVDD - 1.6	V		
PA03	Vos	输入失调电压	-10	—	10	mV		
PA04	Vos	随温度变化的输入失调电压漂移	—	±15	—	μV/°C		
PA05	RIN+	同相输入端输入阻抗	—	>1M    7 pF	—	Ω    pF		
PA06	RIN-	反相输入端输入阻抗	—	10k    7 pF	—	Ω    pF		
PA07	GERR	增益误差	-2	—	2	%	增益 = 4x, 8x	
			-3	—	3	%	增益 = 16x	
			-4	—	4	%	增益 = 32x, 64x	
PA08	LERR	增益非线性误差	—	—	0.5	%	满量程的百分比, 增益 = 16x	
PA09	IDD	电流消耗	—	2.0	—	mA	使能模块, 2V 峰-峰输出电压摆幅	
PA10a	BW	小信号带宽 (-3 dB)	G = 4x	—	10	—	MHz	
PA10b			G = 8x	—	5	—	MHz	
PA10c			G = 16x	—	2.5	—	MHz	
PA10d			G = 32x	—	1.25	—	MHz	
PA10e			G = 64x	—	0.625	—	MHz	
PA11	OST	输出容差达到稳定在最终值的 ±1% 内的时间	—	0.4	—	μs	增益 = 16x, 100 mV 输入阶跃变化	
PA12	SR	输出压摆率	—	40	—	V/μs	增益 = 16x	
PA13	TGSEL	增益选择时间	—	1	—	μs		
PA14	TON	模块开启 / 稳定时间	—	—	10	μs		

注 1: PGAx 模块可在 VBORMIN < VDD < VDDMIN 条件下工作, 但性能将下降。除非另外声明, 否则模块功能经过测试, 但未确定特性值。



# dsPIC33EPXXGS70X/80X 系列

表 30-58: 恒流源规范

直流特性				标准工作条件: 3.0V 至 3.6V (除非另外声明) <sup>(1)</sup> 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
CC01	I <sub>DD</sub>	电流消耗	—	30	—	μA	
CC02	I <sub>REG</sub>	电压开启时的电流稳定度	—	±3	—	%	
CC03	I <sub>OUT</sub>	端子处的电流输出	—	10	—	μA	

注 1: 恒流源模块可在  $V_{BORMIN} < V_{DD} < V_{DDMIN}$  条件下工作, 但性能将下降。除非另外声明, 否则模块功能经过测试, 但未确定特性值。

表 30-59: DMA 模块时序要求

交流特性		标准工作条件: 3.0V 至 3.6V (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 <sup>(1)</sup>	最大值	单位	条件
DM1	DMA 字节 / 字传输延时	1 T <sub>CY</sub> <sup>(2)</sup>	—	—	ns	

注 1: 这些参数为特性值, 但生产时未经测试。

2: 由于 DMA 传输使用 CPU 数据总线, 所以该时间依赖于总线上的其他功能。

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## 31.0 器件直流和交流特性曲线图

**注：** 以下曲线图来自有限数量样片的统计结果，仅供设计参考。此处列出的性能特性未经测试，不做任何保证。一些曲线图中列出的数据可能超出规定的工作范围（例如，超出了规定的电源范围），因此不在担保范围内。

图 31-1: VOH—4x 驱动引脚曲线

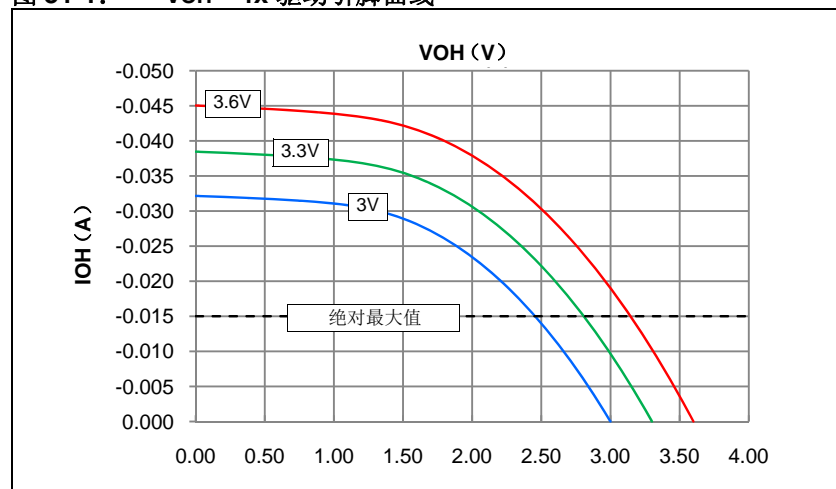


图 31-3: VOL—4x 驱动引脚曲线

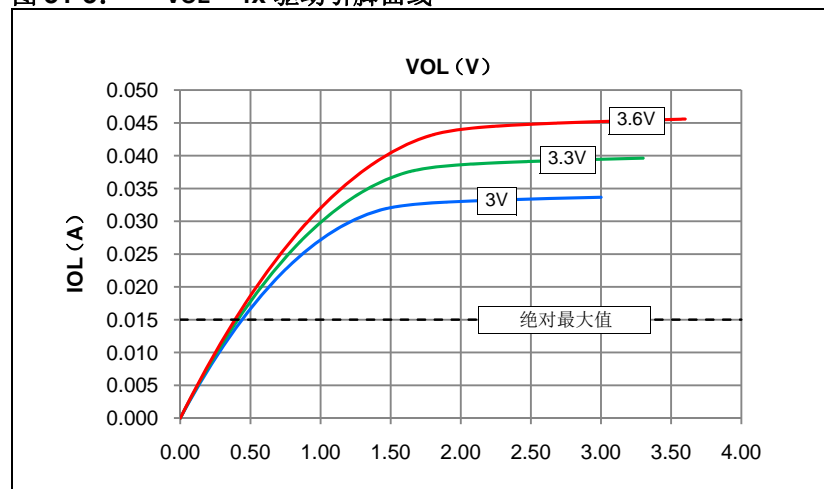


图 31-2: VOH—8x 驱动引脚曲线:

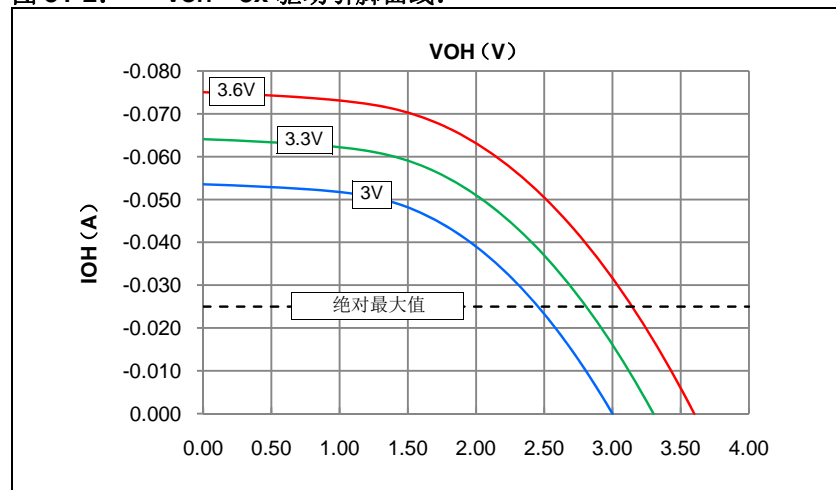


图 31-4: VOL—8x 驱动引脚曲线

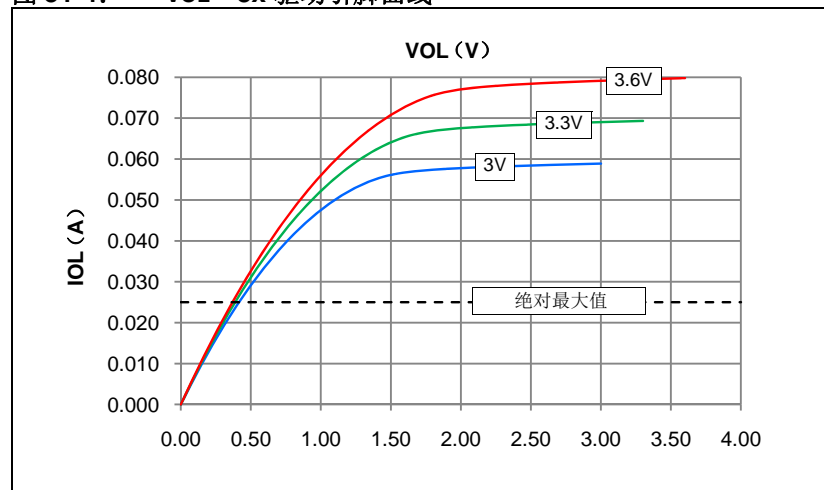


图 31-5: 典型 IPD 电流曲线 (VDD = 3.3V)

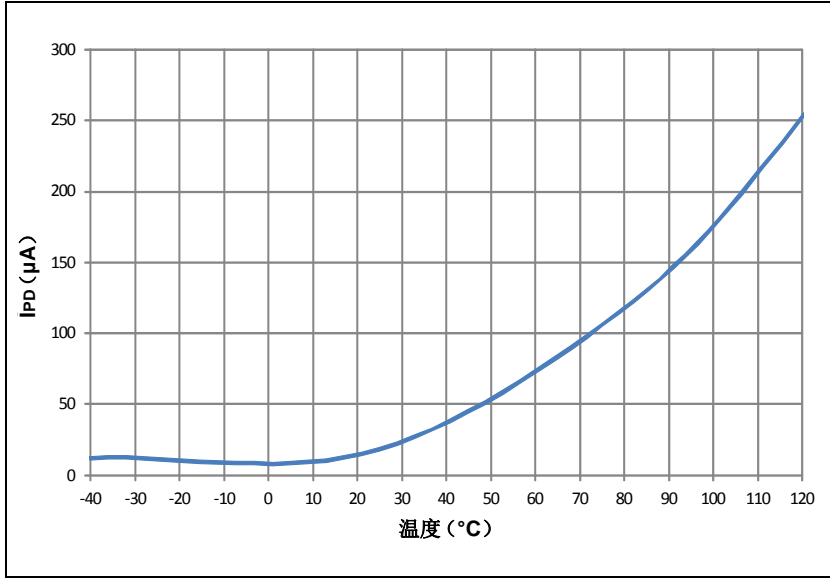


图 31-6: 典型 IDD 电流 (VDD = 3.3V, +25°C)

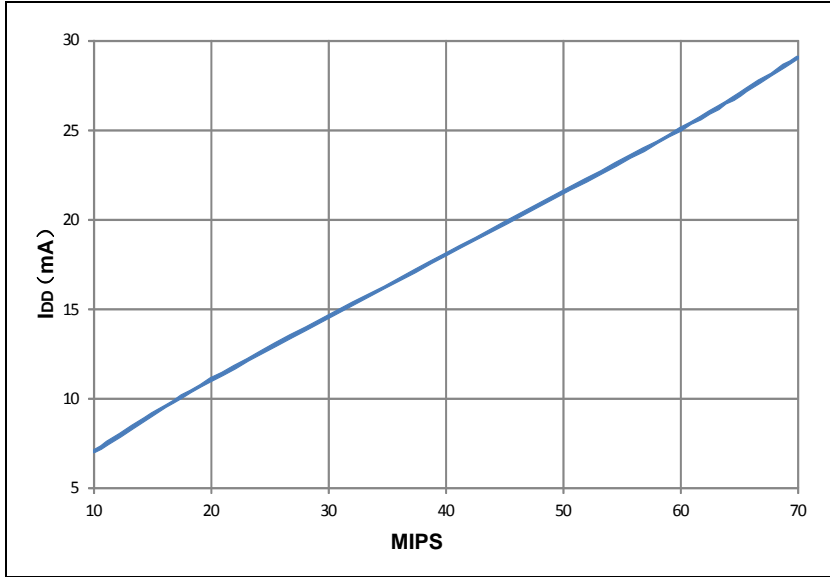


图 31-7: 典型 IDOZE 电流 (VDD = 3.3V, +25°C)

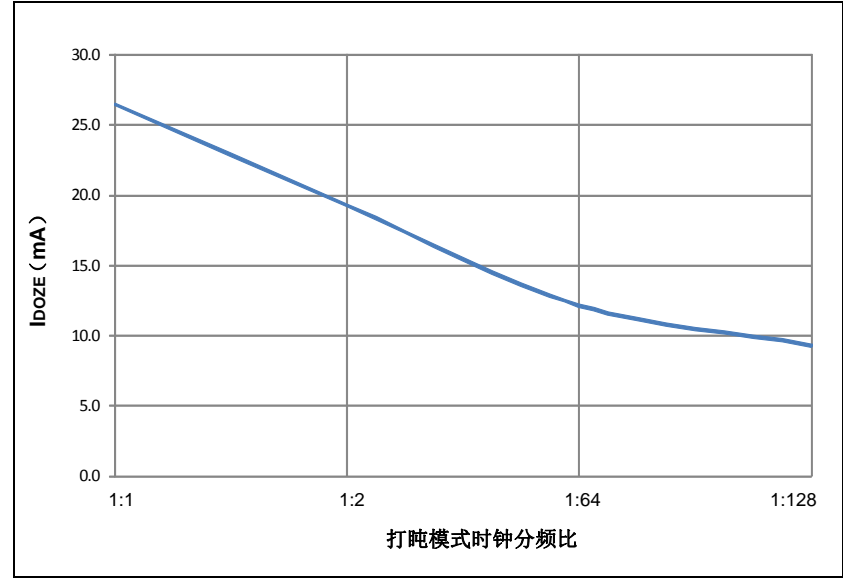


图 31-8: 典型 IIDLE 电流 (VDD = 3.3V, +25°C)

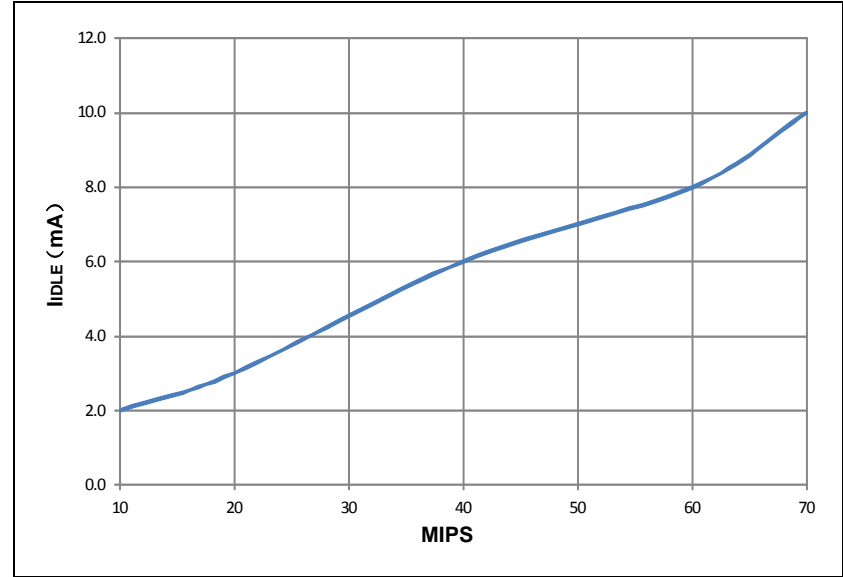


图 31-9: 典型 FRC 频率 (VDD = 3.3V)

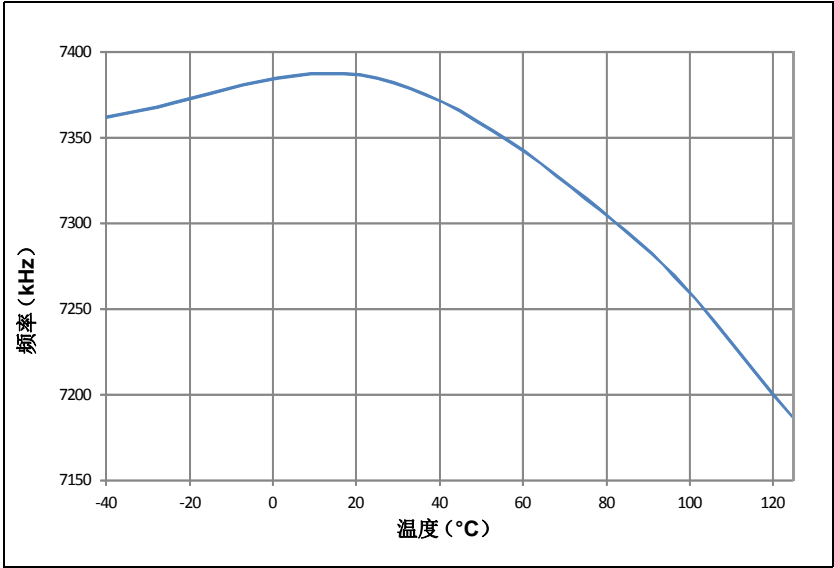
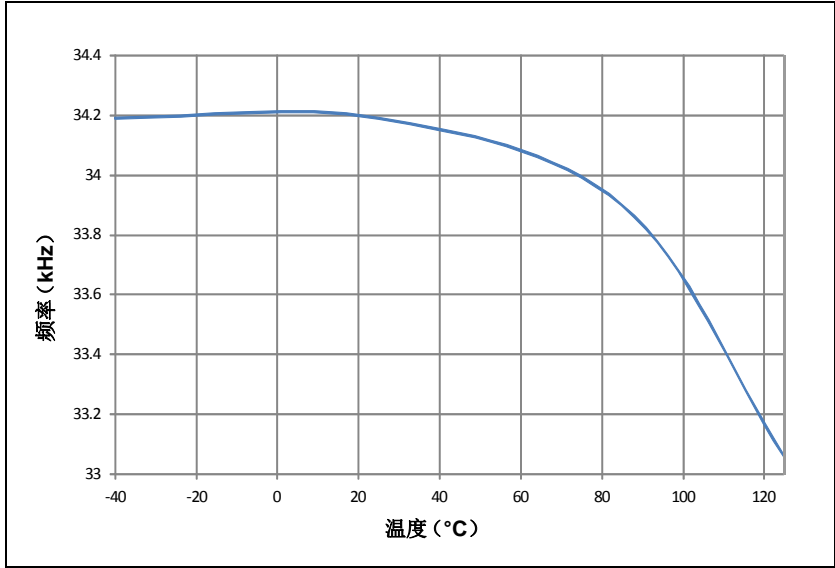


图 31-10: 典型 LPRC 频率 (VDD = 3.3V)



# dsPIC33EPXXXGS70X/80X 系列

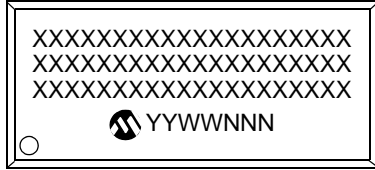
---

注:

## 32.0 封装信息

### 32.1 封装标识信息

28 引脚 SOIC (7.50 mm)



示例



28 引脚 UQFN (6x6x0.55 mm)



示例



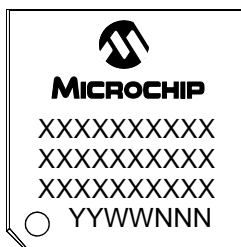
28 引脚 QFN-S (6x6x0.9 mm)



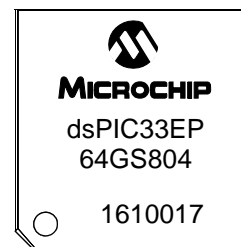
示例



44 引脚 TQFP (10x10x1 mm)



示例

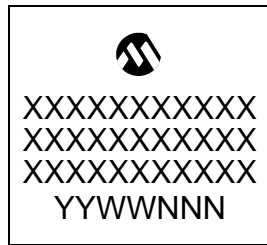


<b>图注:</b>	XX...X	客户指定信息
	Y	年份代码 (日历年的最后一位数字)
	YY	年份代码 (日历年的最后两位数字)
	WW	星期代码 (一月一日的星期代码为“01”)
	NNN	以字母数字排序的追踪代码
<b>注:</b>	Microchip 部件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户指定信息的字符数。	

# dsPIC33EPXXXGS70X/80X 系列

## 32.1 封装标识信息 (续)

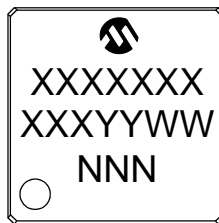
44 引脚 QFN (8x8 mm)



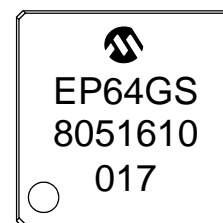
示例



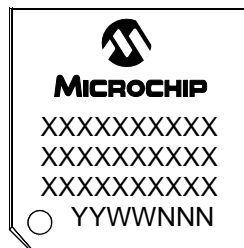
48 引脚 TQFP (7x7x1.0 mm)



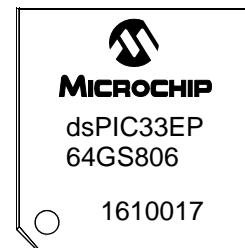
示例



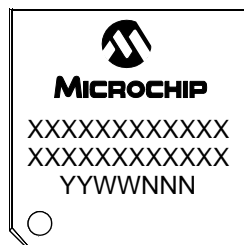
64 引脚 TQFP (10x10x1 mm)



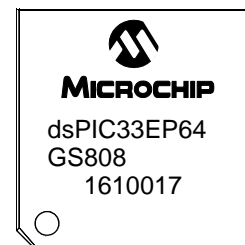
示例



80 引脚 TQFP (12x12x1 mm)



示例



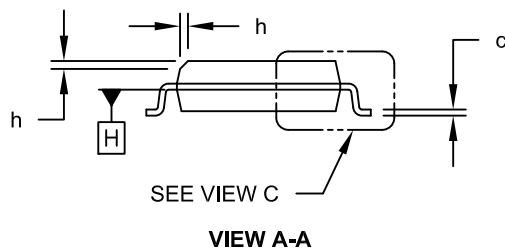
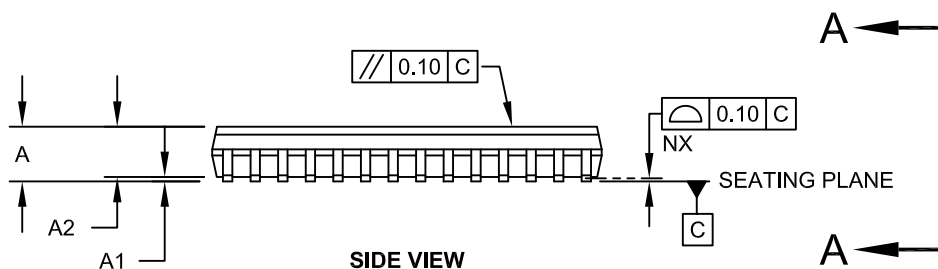
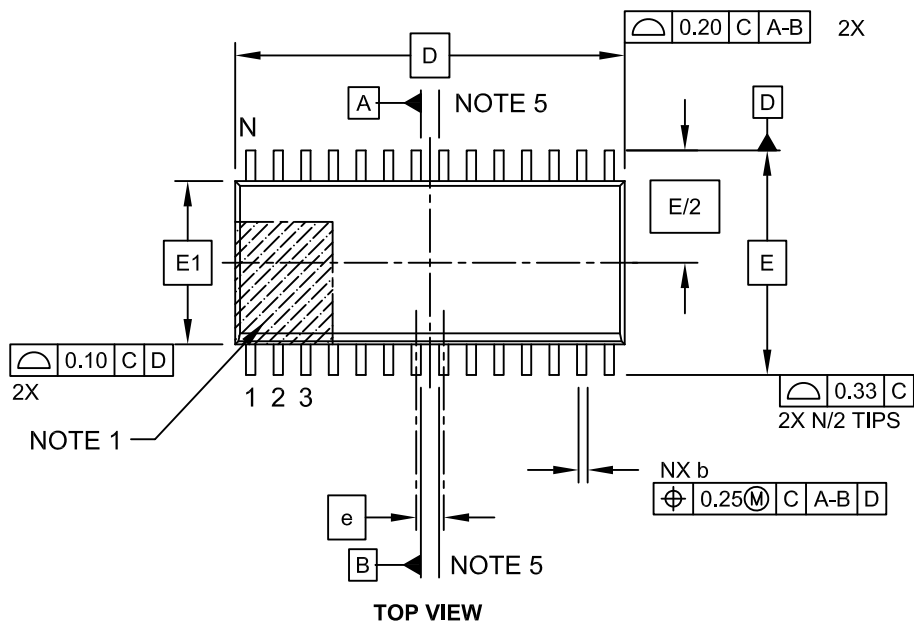


# dsPIC33EPXXGS70X/80X 系列

## 32.2 封装详细信息

### 28 引脚塑封小外形封装 (SO) —— 宽条, 主体 7.50 mm [SOIC]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

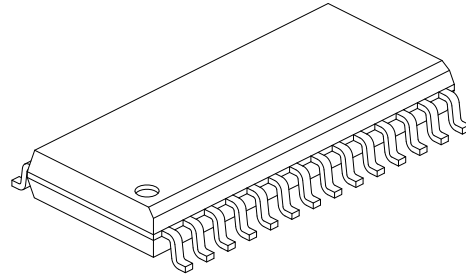
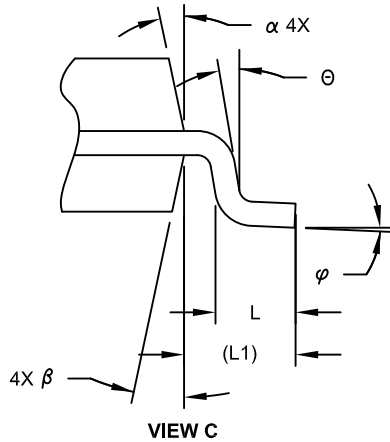


Microchip Technology Drawing C04-052C Sheet 1 of 2

# dsPIC33EPXXGS70X/80X 系列

## 28 引脚塑封小外形封装 (SO) —— 宽条, 主体 7.50 mm [SOIC]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	1.27 BSC		
Overall Height	A	-	-	2.65
Molded Package Thickness	A2	2.05	-	-
Standoff §	A1	0.10	-	0.30
Overall Width	E	10.30 BSC		
Molded Package Width	E1	7.50 BSC		
Overall Length	D	17.90 BSC		
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1	1.40 REF		
Lead Angle	θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.18	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

### Notes:

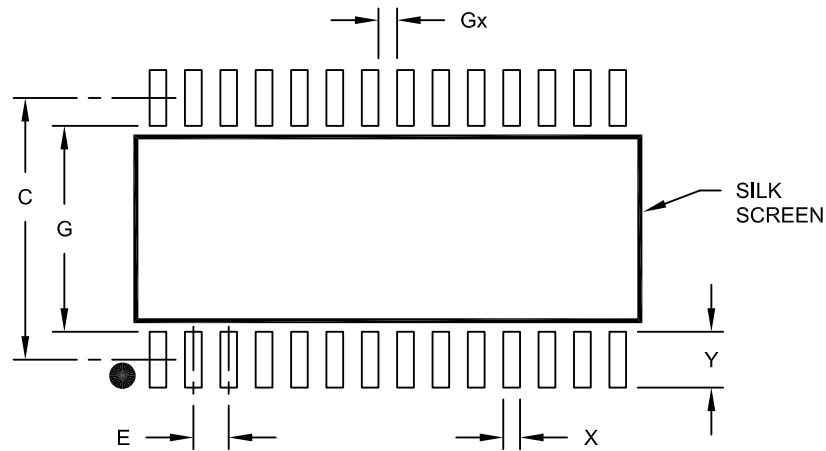
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
REF: Reference Dimension, usually without tolerance, for information purposes only.
- Datums A & B to be determined at Datum H.

Microchip Technology Drawing C04-052C Sheet 2 of 2

# dsPIC33EPXXGS70X/80X 系列

## 28 引脚塑封小外形封装 (SO) —— 宽条, 主体 7.50 mm [SOIC]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



### RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C		9.40	
Contact Pad Width (X28)	X			0.60
Contact Pad Length (X28)	Y			2.00
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.40		

#### Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

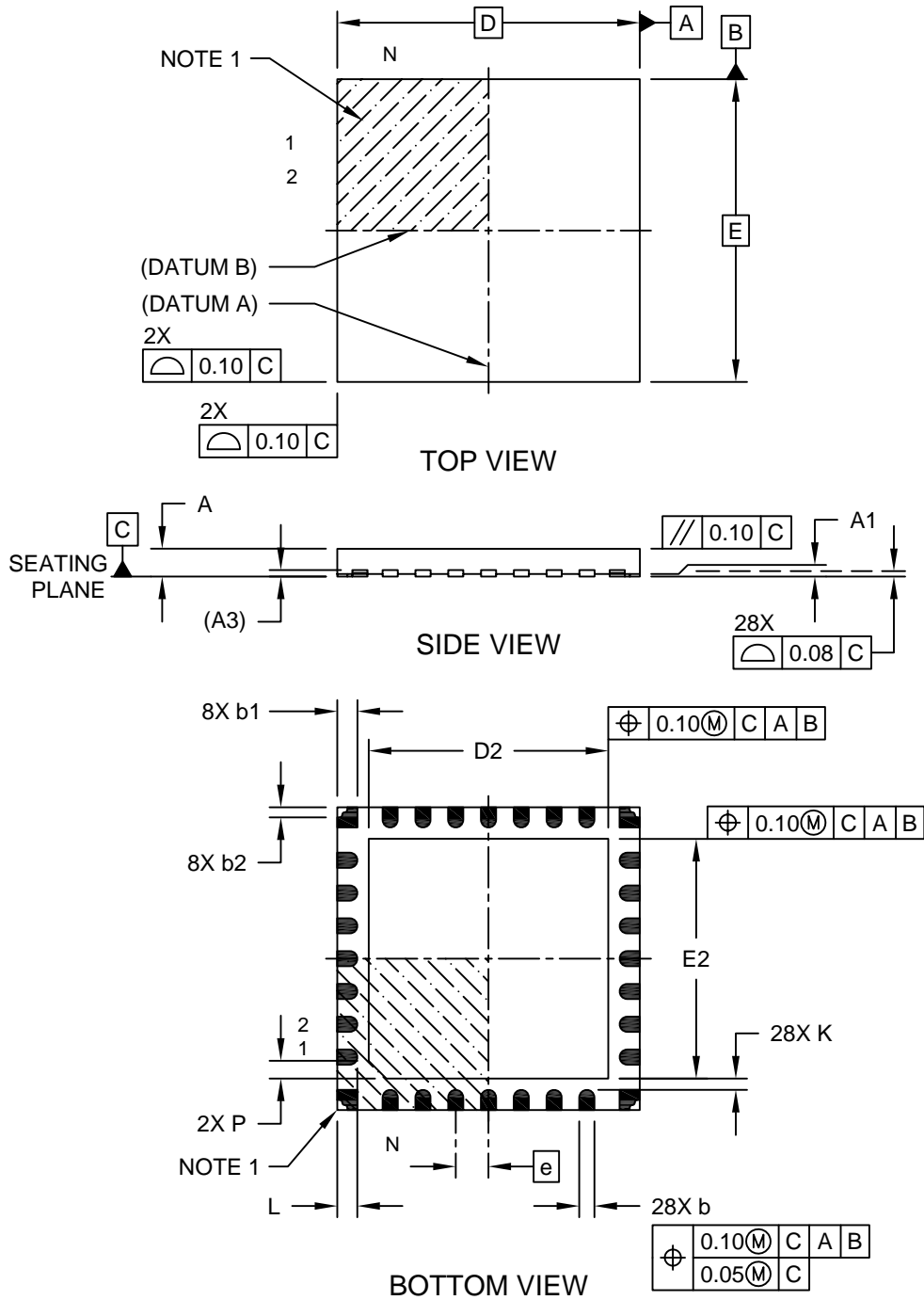
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A

# dsPIC33EPXXXGS70X/80X 系列

28 引脚塑封超薄正方扁平无脚封装 (2N) —— 主体 6x6x0.55 mm [UQFN], 带 4.65x4.65 mm 裸露焊盘和角锚点

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

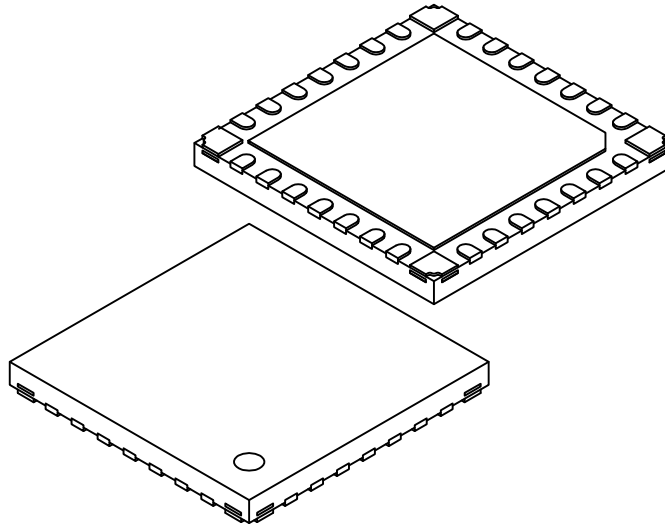


Microchip Technology Drawing C04-385B Sheet 1 of 2

# dsPIC33EPXXGS70X/80X 系列

28 引脚塑封超薄正方扁平无脚封装（2N）—— 主体 6x6x0.55 mm [UQFN]，带 4.65x4.65 mm 裸露焊盘和角锚点

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Terminals	N	28		
Pitch	e	0.65 BSC		
Overall Height	A	0.45	0.50	0.55
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3	0.127 REF		
Overall Width	E	6.00 BSC		
Exposed Pad Width	E2	4.55	4.65	4.75
Overall Length	D	6.00 BSC		
Exposed Pad Length	D2	4.55	4.65	4.75
Exposed Pad Corner Chamfer	P	-	0.35	-
Terminal Width	b	0.25	0.30	0.35
Corner Anchor Pad	b1	0.35	0.40	0.43
Corner Pad, Metal Free Zone	b2	0.15	0.20	0.25
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed-Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

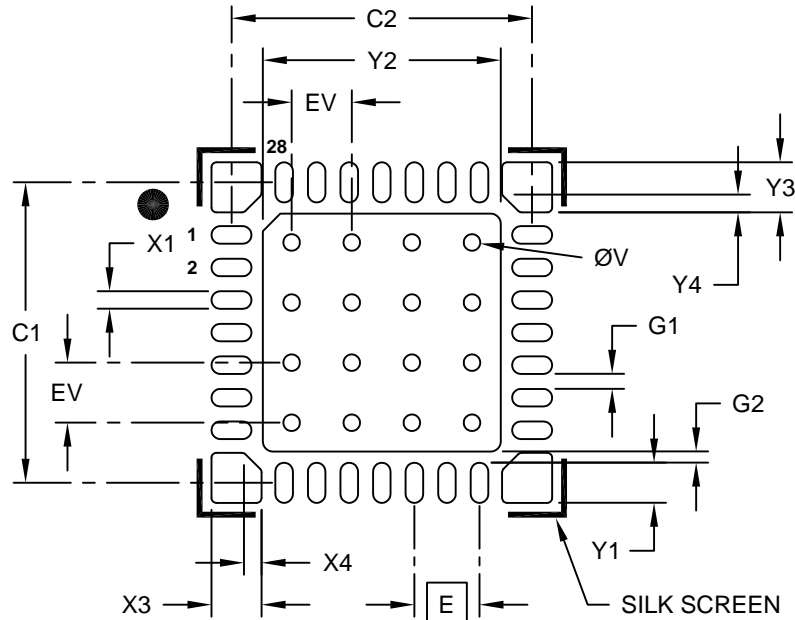
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-385B Sheet 2 of 2

# dsPIC33EPXXGS70X/80X 系列

28 引脚塑封超薄正方扁平无脚封装 (2N) —— 主体 6x6x0.55 mm [UQFN], 带 4.65x4.65 mm 裸露焊盘和角锚点

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	X2			4.75
Optional Center Pad Length	Y2			4.75
Contact Pad Spacing	C1		6.00	
Contact Pad Spacing	C2		6.00	
Contact Pad Width (X28)	X1			0.35
Contact Pad Length (X28)	Y1			0.80
Corner Anchor (X4)	X3			1.00
Corner Anchor (X4)	Y3			1.00
Corner Anchor Chamfer (X4)	X4			0.35
Corner Anchor Chamfer (X4)	Y4			0.35
Contact Pad to Pad (X28)	G1	0.20		
Contact Pad to Center Pad (X28)	G2	0.20		
Thermal Via Diameter	V		0.33	
Thermal Via Pitch	EV		1.20	

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

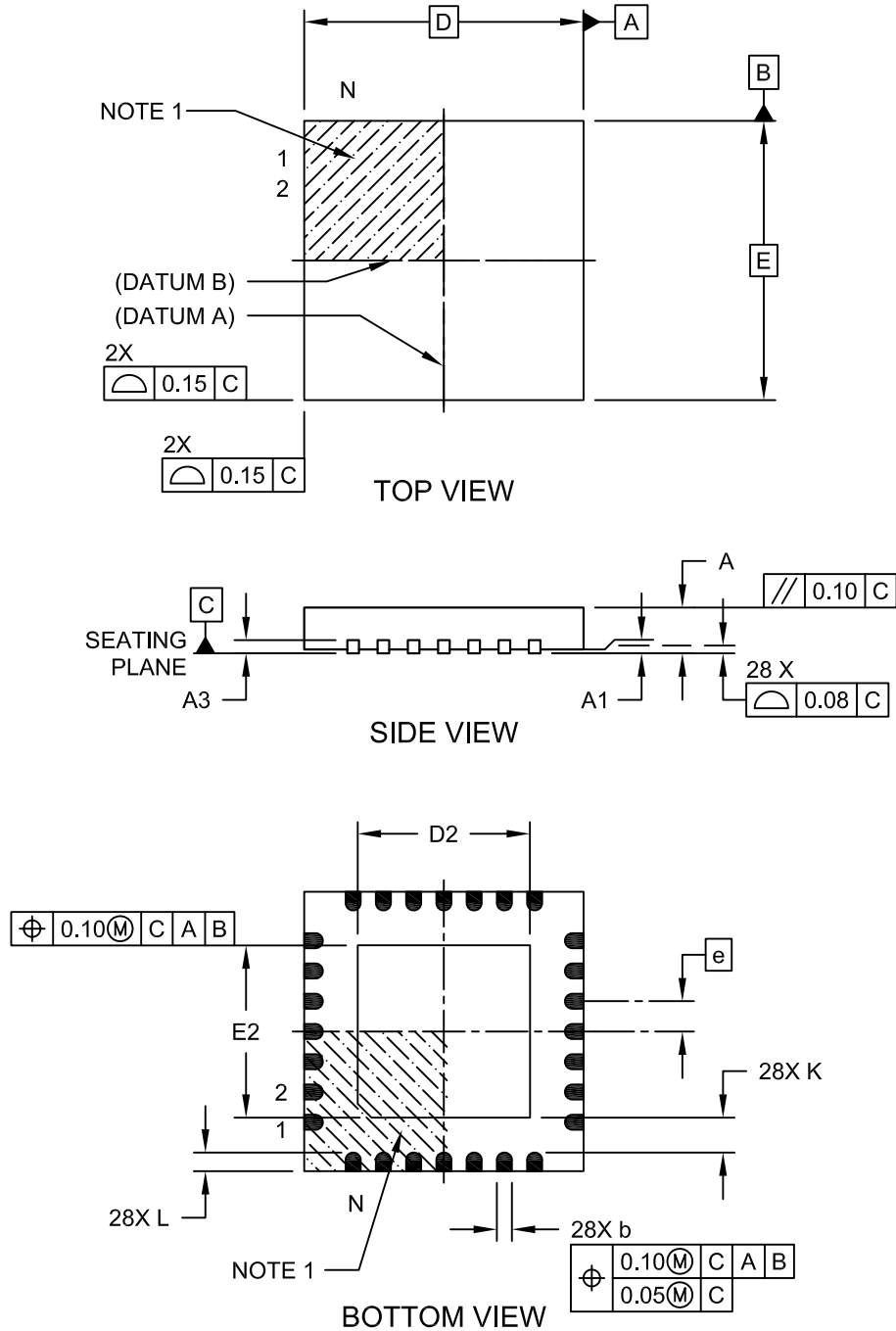
Microchip Technology Drawing C04-2385B

注: 当封装焊接到 PCB 上时, 角锚点焊盘未内部连接, 而是设计为机械特性。

# dsPIC33EPXXGS70X/80X 系列

28 引脚塑封正方扁平无脚封装 (MM) —— 主体 6x6x0.9mm [QFN-S], 端子长度为 0.40 mm

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

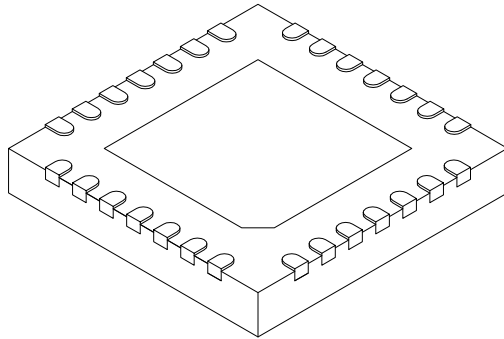


Microchip Technology Drawing C04-124C Sheet 1 of 2

# dsPIC33EPXXGS70X/80X 系列

28 引脚塑封正方扁平无脚封装（MM）—— 主体 6x6x0.9mm [QFN-S]，端子长度为 0.40 mm

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits		Units	MILLIMETERS		
			MIN	NOM	MAX
Number of Pins	N		28		
Pitch	e		0.65 BSC		
Overall Height	A	0.80	0.90	1.00	
Standoff	A1	0.00	0.02	0.05	
Terminal Thickness	A3	0.20 REF			
Overall Width	E	6.00 BSC			
Exposed Pad Width	E2	3.65	3.70	4.70	
Overall Length	D	6.00 BSC			
Exposed Pad Length	D2	3.65	3.70	4.70	
Terminal Width	b	0.23	0.30	0.35	
Terminal Length	L	0.30	0.40	0.50	
Terminal-to-Exposed Pad	K	0.20	-	-	

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

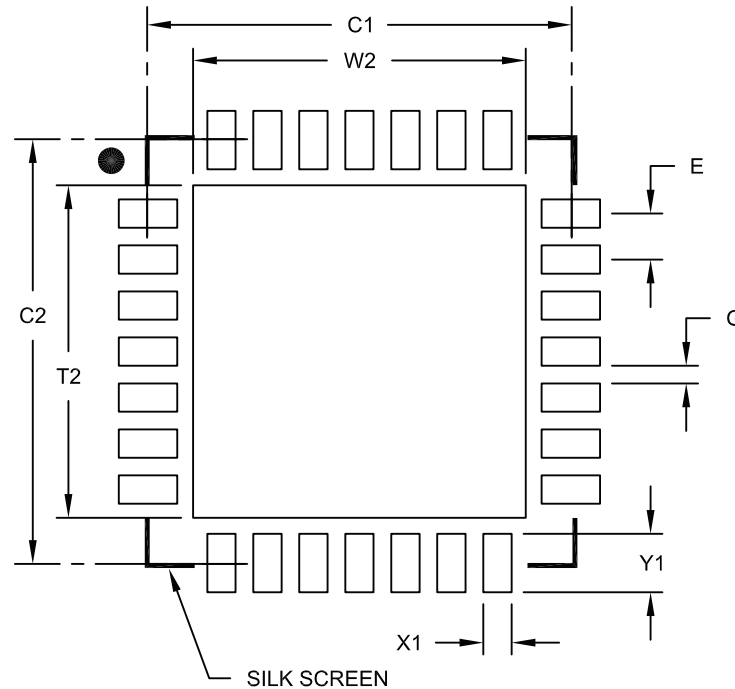
Microchip Technology Drawing C04-124C Sheet 2 of 2



# dsPIC33EPXXGS70X/80X 系列

28 引脚塑封正方扁平无脚封装 (MM) —— 主体 6x6x0.9mm [QFN-S], 触点长度为 0.40 mm

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	W2			4.70
Optional Center Pad Length	T2			4.70
Contact Pad Spacing	C1		6.00	
Contact Pad Spacing	C2		6.00	
Contact Pad Width (X28)	X1			0.40
Contact Pad Length (X28)	Y1			0.85
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

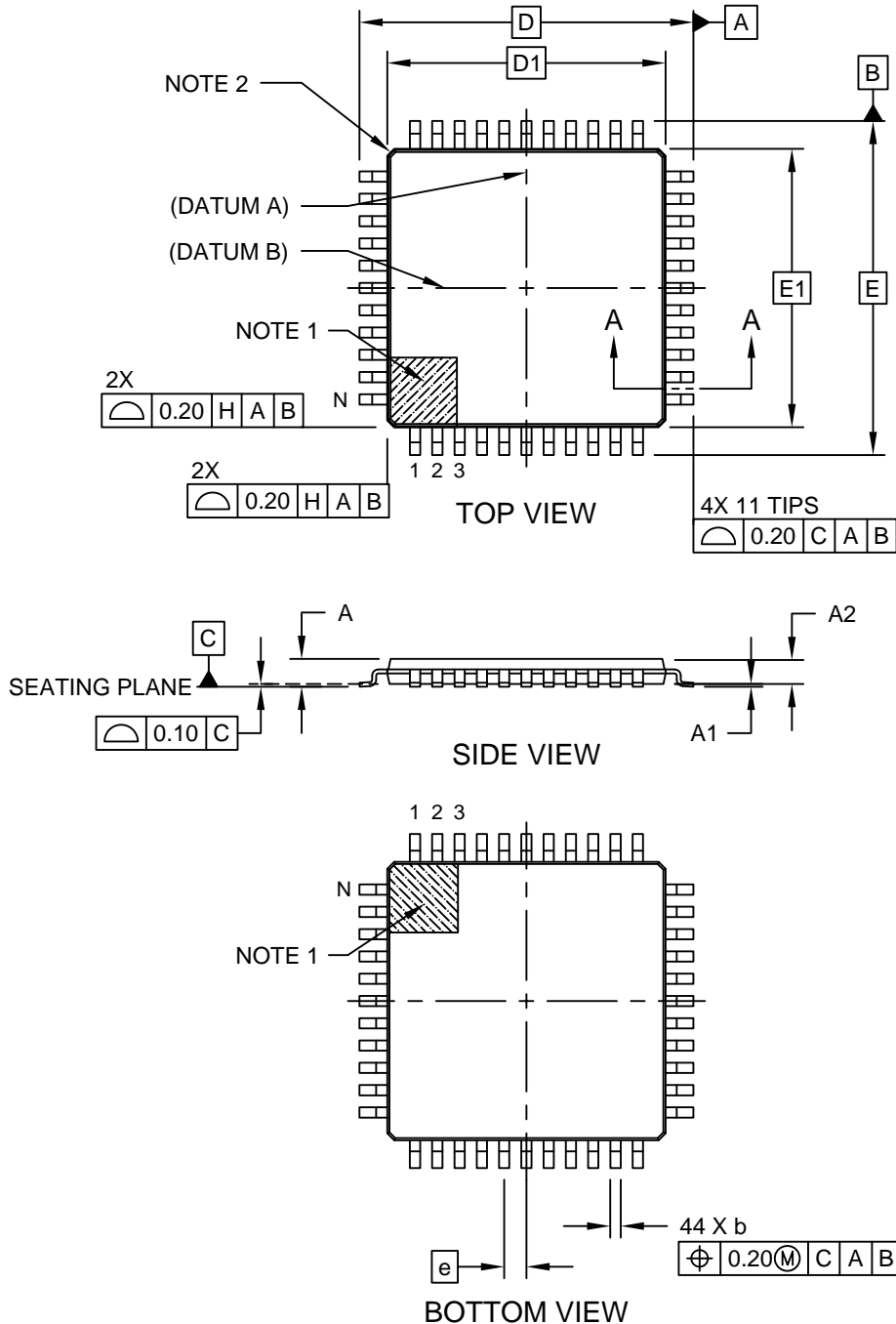
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2124A

# dsPIC33EPXXGS70X/80X 系列

## 44 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1.0 mm [TQFP]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

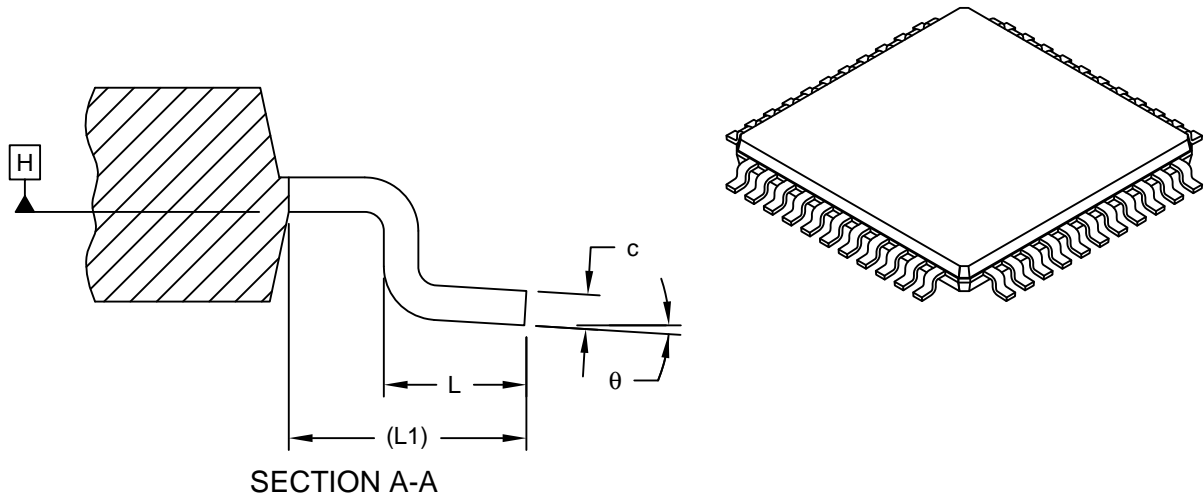


Microchip Technology Drawing C04-076C Sheet 1 of 2

# dsPIC33EPXXGS70X/80X 系列

## 44 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1.0 mm [TQFP]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	44		
Lead Pitch	e	0.80 BSC		
Overall Height	A	-	-	1.20
Standoff	A1	0.05	-	0.15
Molded Package Thickness	A2	0.95	1.00	1.05
Overall Width	E	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Width	b	0.30	0.37	0.45
Lead Thickness	c	0.09	-	0.20
Lead Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	θ	0°	3.5°	7°

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Exact shape of each corner is optional.
- Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

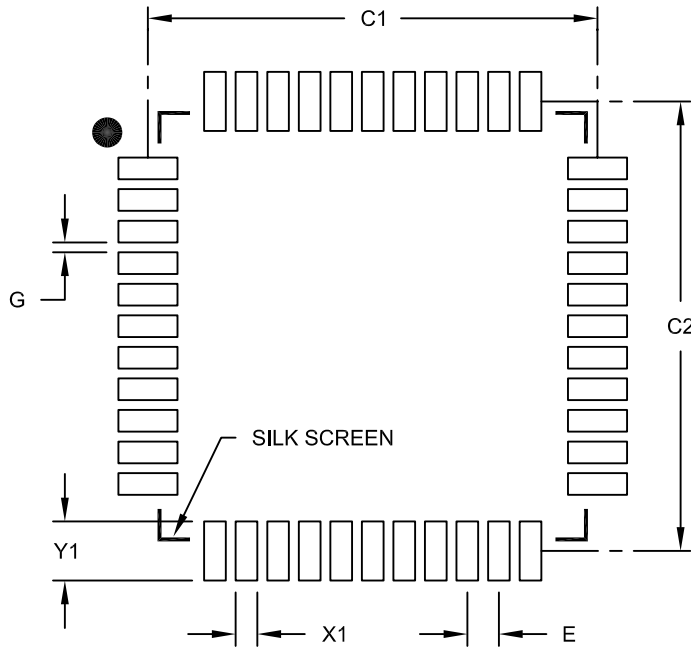
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-076C Sheet 2 of 2

# dsPIC33EPXXGS70X/80X 系列

44 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 引脚投影长度 2.00 mm [TQFP]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

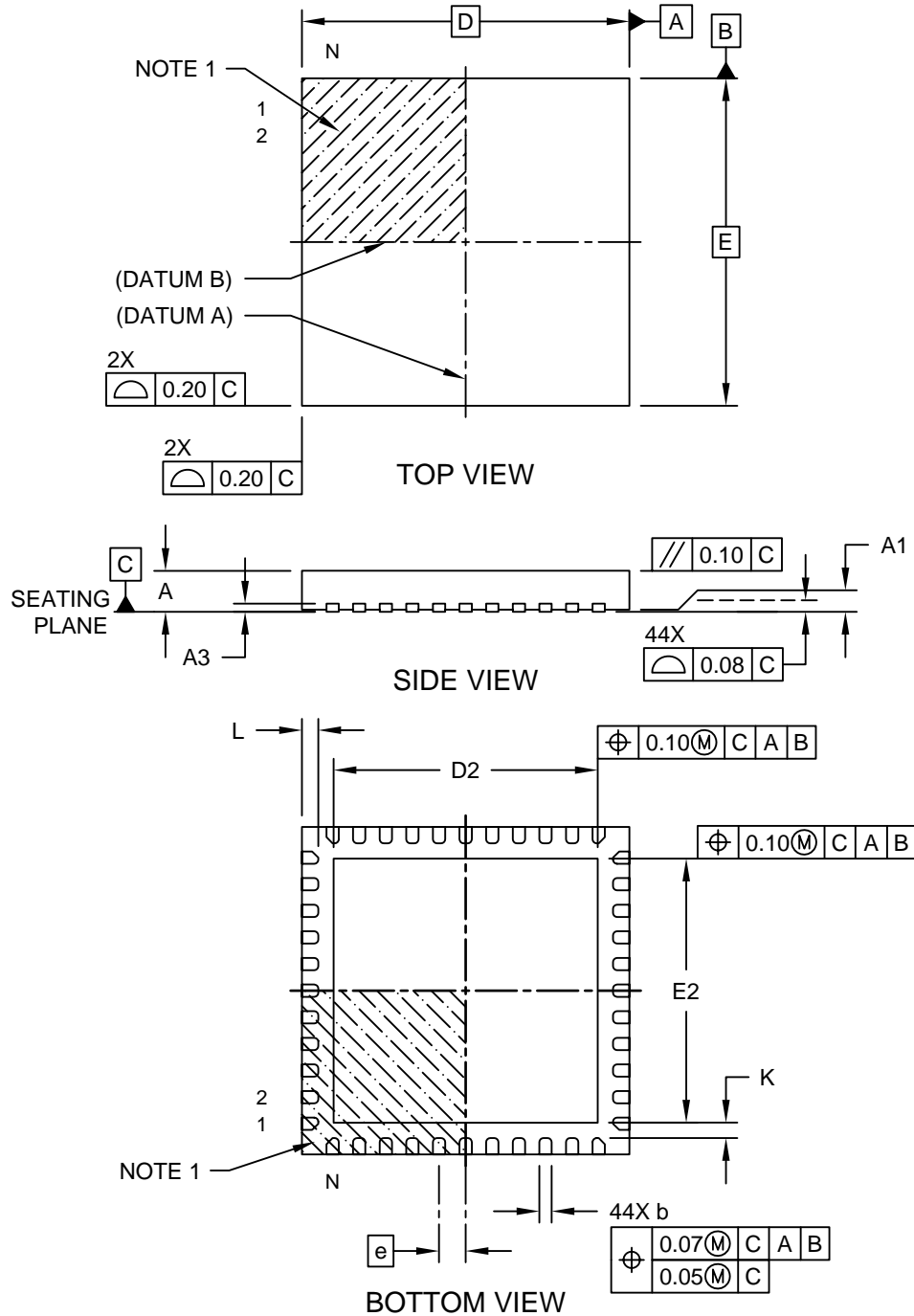
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2076B

# dsPIC33EPXXGS70X/80X 系列

## 44 引脚塑封正方扁平无脚封装 (ML) —— 主体 8x8 mm [QFN 或 VQFN]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

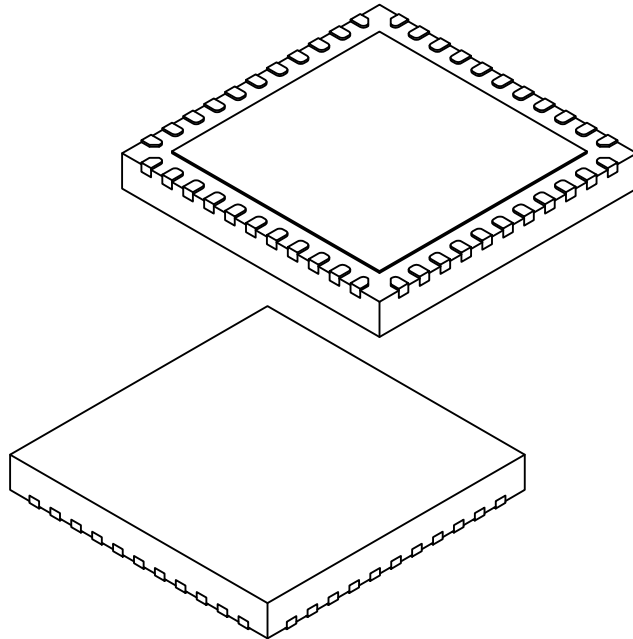


Microchip Technology Drawing C04-103D Sheet 1 of 2

# dsPIC33EPXXGS70X/80X 系列

## 44 引脚塑封正方扁平无脚封装（ML）—— 主体 8x8 mm [QFN 或 VQFN]

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	44		
Pitch	e	0.65 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3	0.20 REF		
Overall Width	E	8.00 BSC		
Exposed Pad Width	E2	6.25	6.45	6.60
Overall Length	D	8.00 BSC		
Exposed Pad Length	D2	6.25	6.45	6.60
Terminal Width	b	0.20	0.30	0.35
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed-Pad	K	0.20	-	-

### Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

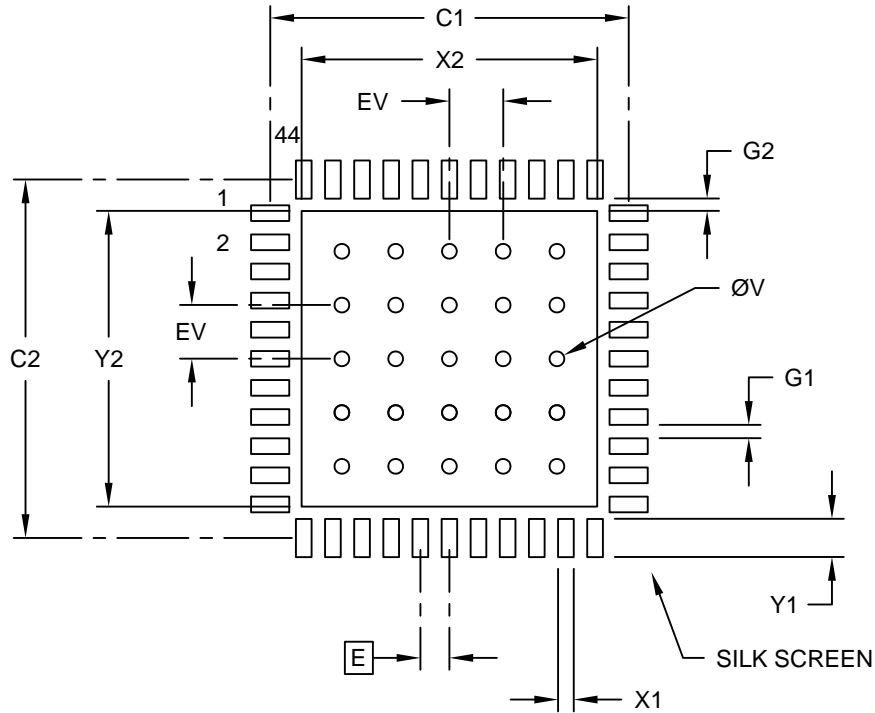
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-103D Sheet 2 of 2

# dsPIC33EPXXGS70X/80X 系列

## 44 引脚塑封正方扁平无脚封装 (ML) —— 主体 8x8 mm [QFN 或 VQFN]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



### RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	X2			6.60
Optional Center Pad Length	Y2			6.60
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Width (X44)	X1			0.35
Contact Pad Length (X44)	Y1			0.85
Contact Pad to Contact Pad (X40)	G1	0.30		
Contact Pad to Center Pad (X44)	G2	0.28		
Thermal Via Diameter	V		0.33	
Thermal Via Pitch	EV		1.20	

#### Notes:

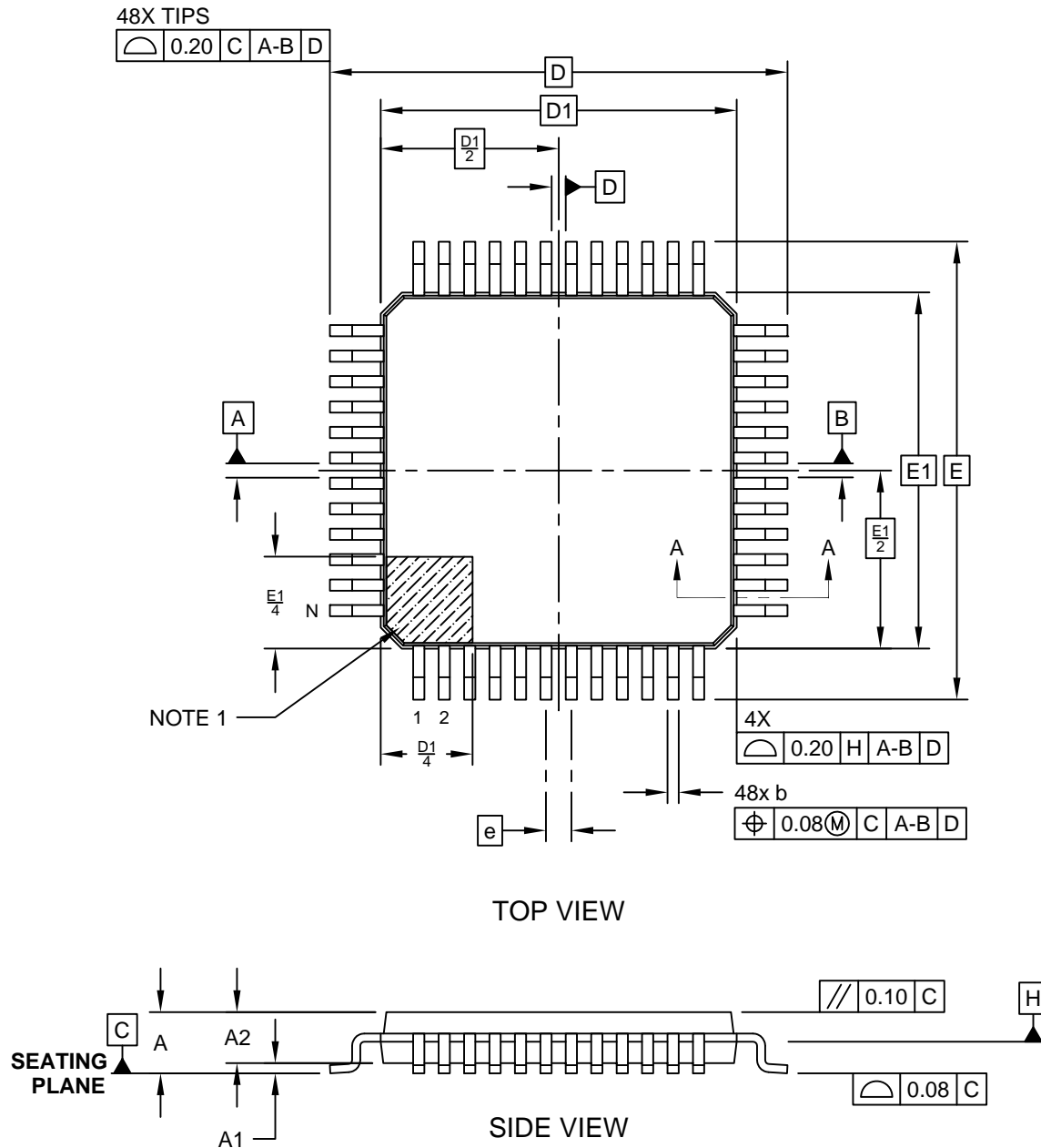
- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
- For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing No. C04-2103C

# dsPIC33EPXXGS70X/80X 系列

## 48 引脚薄型正方扁平封装 (PT) —— 主体 7x7x1.0 mm [TQFP]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



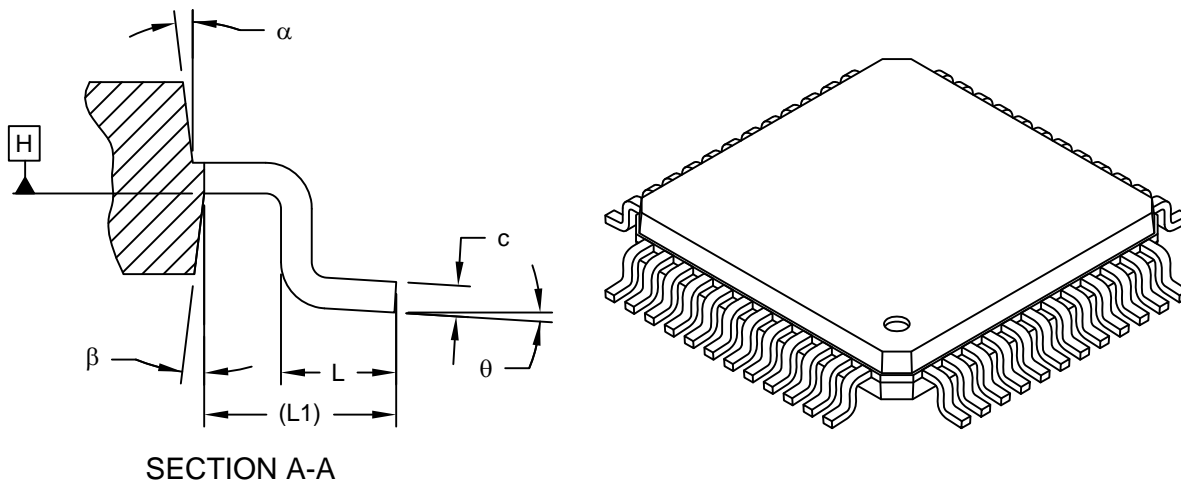
Microchip Technology Drawing C04-300-PT Rev A Sheet 1 of 2



# dsPIC33EPXXGS70X/80X 系列

## 48 引脚薄型正方扁平封装 (PT) —— 主体 7x7x1.0 mm [TQFP]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	48		
Lead Pitch	e	0.50 BSC		
Overall Height	A	-	-	1.20
Standoff	A1	0.05	-	0.15
Molded Package Thickness	A2	0.95	1.00	1.05
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	$\phi$	0°	3.5°	7°
Overall Width	E	9.00 BSC		
Overall Length	D	9.00 BSC		
Molded Package Width	E1	7.00 BSC		
Molded Package Length	D1	7.00 BSC		
Lead Thickness	c	0.09	-	0.16
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	$\alpha$	11°	12°	13°
Mold Draft Angle Bottom	$\beta$	11°	12°	13°

### Notes:

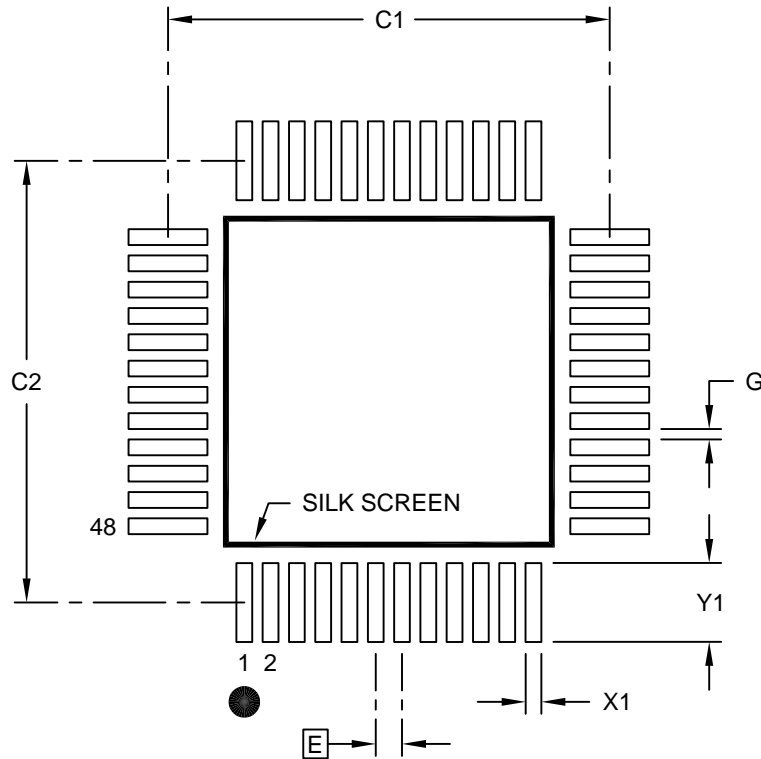
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
REF: Reference Dimension, usually without tolerance, for information purposes only.
- Datums **[A-B]** and **[D]** to be determined at center line between leads where leads exit plastic body at datum plane **[H]**

Microchip Technology Drawing C04-300-PT Rev A Sheet 2 of 2

# dsPIC33EPXXGS70X/80X 系列

## 48 引脚薄型正方扁平封装 (PT) —— 主体 7x7x1.0 mm [TQFP]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



### RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		8.40	
Contact Pad Spacing	C2		8.40	
Contact Pad Width (X48)	X1			0.30
Contact Pad Length (X48)	Y1			1.50
Distance Between Pads	G	0.20		

#### Notes:

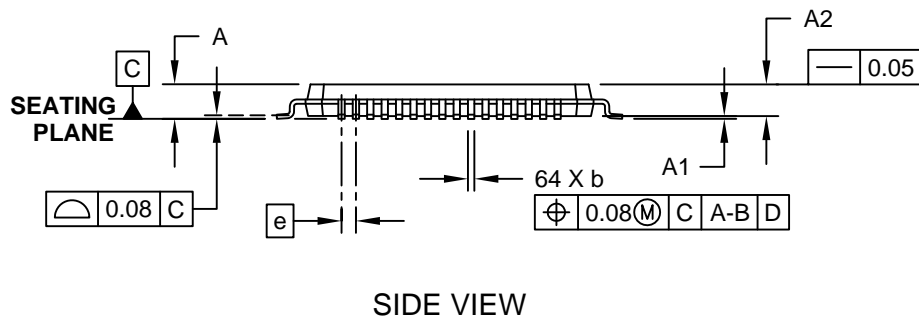
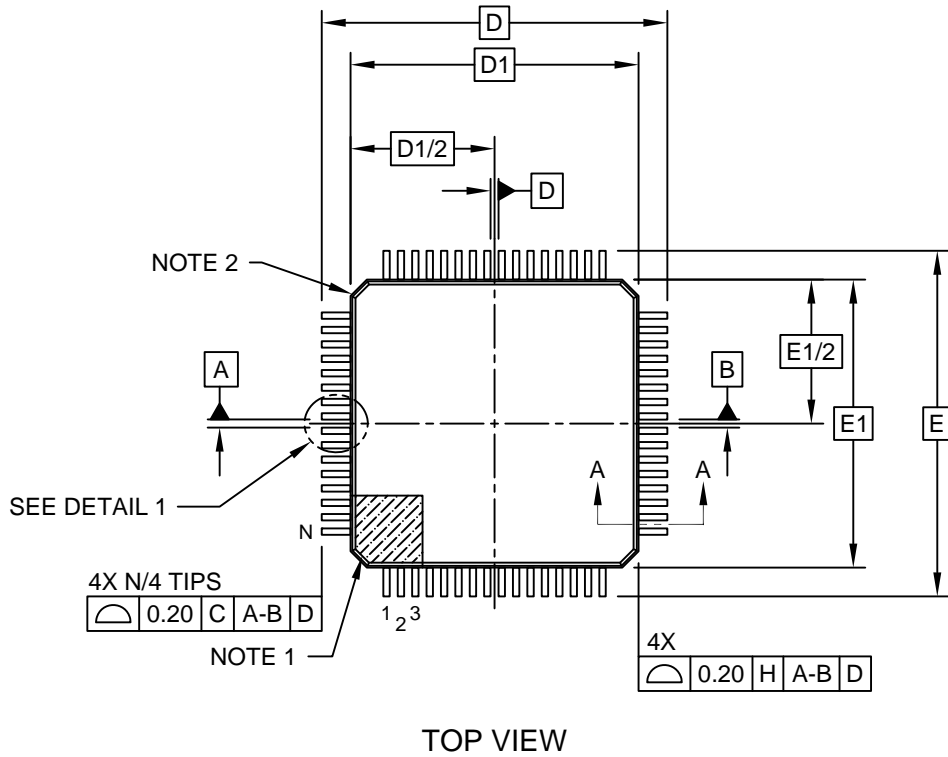
- Dimensioning and tolerancing per ASME Y14.5M  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
- For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2300-PT Rev A

# dsPIC33EPXXGS70X/80X 系列

64 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 引脚投影长度 2.00 mm [TQFP]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

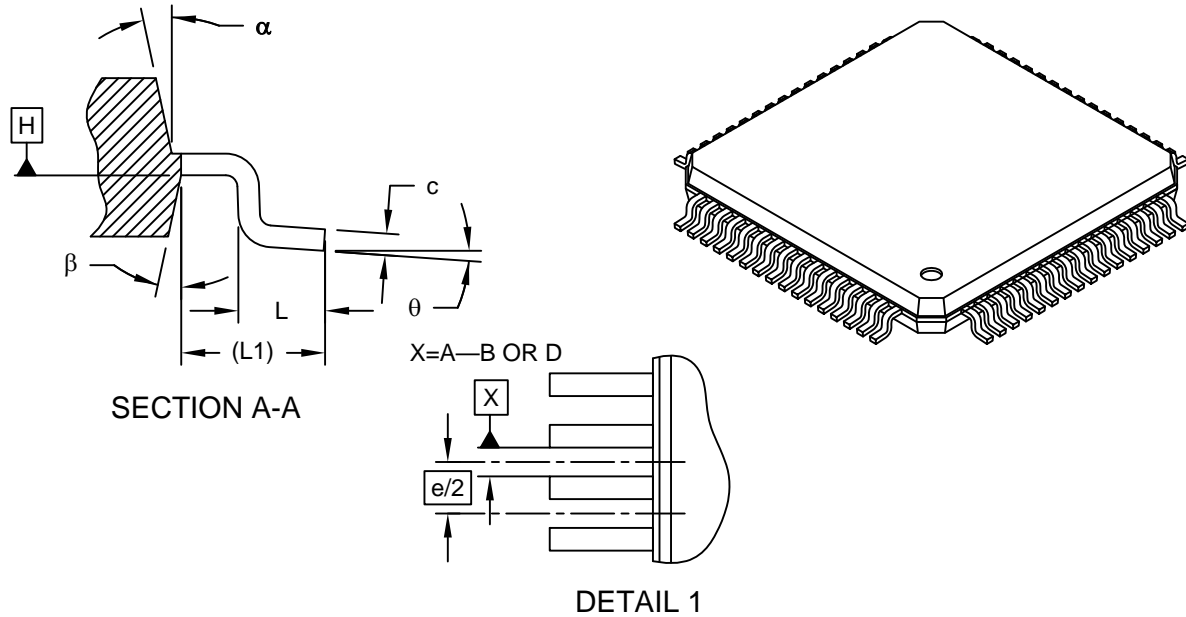


Microchip Technology Drawing C04-085C Sheet 1 of 2

# dsPIC33EPXXGS70X/80X 系列

64 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 引脚投影长度 2.00 mm [TQFP]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	64		
Lead Pitch	e	0.50 BSC		
Overall Height	A	-	-	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	-	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	$\phi$	0°	3.5°	7°
Overall Width	E	12.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Thickness	c	0.09	-	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	$\alpha$	11°	12°	13°
Mold Draft Angle Bottom	$\beta$	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

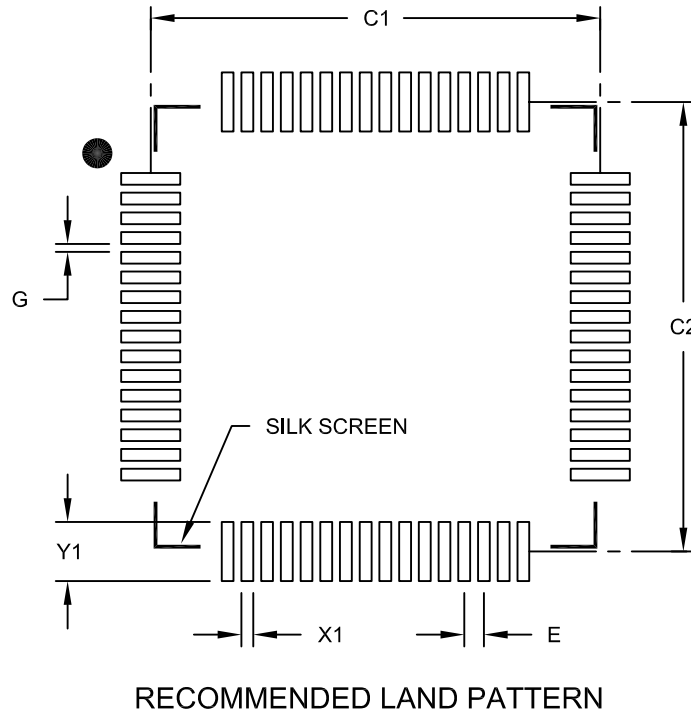
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085C Sheet 2 of 2

# dsPIC33EPXXGS70X/80X 系列

64 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 引脚投影长度 2.00 mm [TQFP]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



		Units	MILLIMETERS		
		Dimension Limits	MIN	NOM	MAX
Contact Pitch	E		0.50 BSC		
Contact Pad Spacing	C1			11.40	
Contact Pad Spacing	C2			11.40	
Contact Pad Width (X64)	X1				0.30
Contact Pad Length (X64)	Y1				1.50
Distance Between Pads	G	0.20			

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

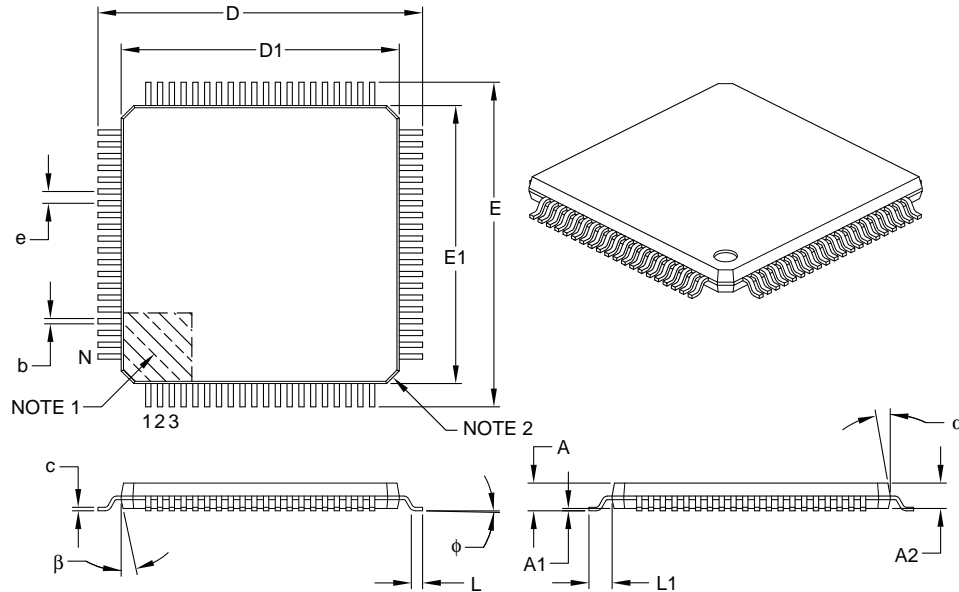
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2085B

# dsPIC33EPXXGS70X/80X 系列

80 引脚塑封薄型正方扁平封装 (PT) —— 主体 12x12x1 mm, 引脚投影长度 2.00 mm [TQFP]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	80		
Lead Pitch	e	0.50 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	φ	0°	3.5°	7°
Overall Width	E	14.00 BSC		
Overall Length	D	14.00 BSC		
Molded Package Width	E1	12.00 BSC		
Molded Package Length	D1	12.00 BSC		
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

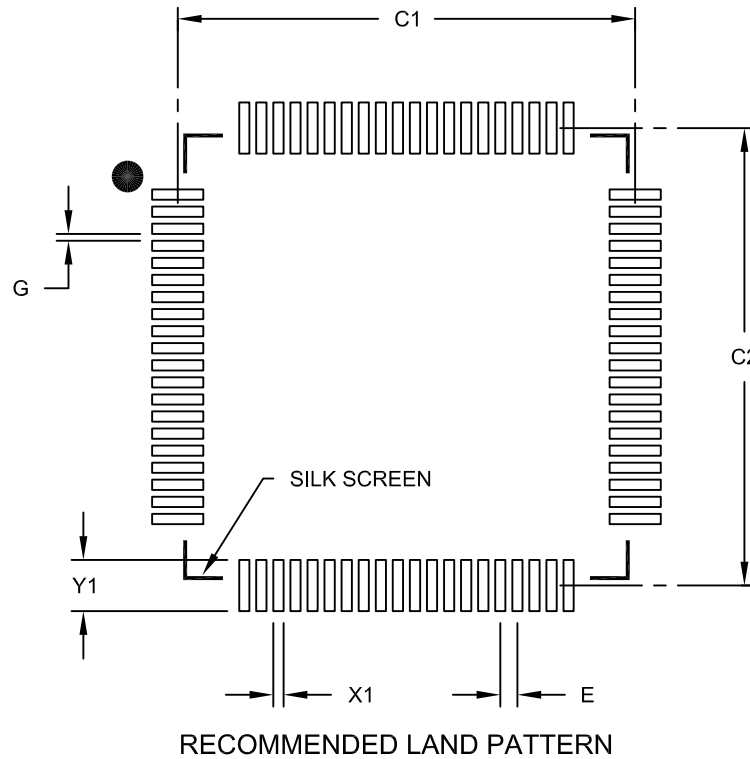
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-092B

# dsPIC33EPXXGS70X/80X 系列

80 引脚塑封薄型正方扁平封装 (PT) —— 主体 12x12x1 mm, 引脚投影长度 2.00 mm [TQFP]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		13.40	
Contact Pad Spacing	C2		13.40	
Contact Pad Width (X80)	X1			0.30
Contact Pad Length (X80)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2092B

# dsPIC33EPXXXGS70X/80X 系列

---

注:



## 附录 A： 版本历史

### 版本 A（2016 年 5 月）

这是本文档的初始版本。

### 版本 B（2017 年 1 月）

- 章节：
  - 更新了第 5.0 节 “闪存程序存储器” 中的“注 1”。
- 表：
  - 更新了第 2 页上的器件说明表。
  - 更新了表 1-1、表 4-2、表 4-11、表 7-1、表 8-1、表 11-11、表 11-13、表 17-1、表 30-3、表 30-4、表 30-6、表 30-7、表 30-8、表 30-9、表 30-10、表 30-11、表 30-52、表 30-54 和表 30-55。
  - 增加了表 11-6、表 11-7、表 11-8、表 11-9 和表 11-10。
- 图：
  - 更新了第 5 页至第 8 页上的引脚图中的引脚功能表。
  - 更新了图 4-1、图 17-1、图 18-1 和图 18-2。
- 寄存器：
  - 更新了寄存器 3-3、寄存器 16-5、寄存器 17-11、寄存器 18-1 和寄存器 19-2。
  - 增加了寄存器 11-1、寄存器 11-2、寄存器 11-3、寄存器 11-4、寄存器 11-5、寄存器 11-6、寄存器 11-7 和寄存器 11-8。

# dsPIC33EPXXXGS70X/80X 系列

---

注:

# dsPIC33EPXXGS70X/80X 系列

## 索引

### B

版本历史 .....	465
编程模型 .....	23
的寄存器说明 .....	23
变更通知客户服务 .....	475

### C

#### CAN

##### CAN 模块

报文缓冲区 .....	328
字 0 .....	328
字 1 .....	328
字 2 .....	329
字 3 .....	329
字 4 .....	330
字 5 .....	330
字 6 .....	331
字 7 .....	331

概述 .....	307
工作模式 .....	308
控制寄存器 .....	309

CAN 模块 (CAN) .....	307
--------------------	-----

#### C 编译器

MPLAB C18 .....	372
-----------------	-----

#### CLC

控制寄存器 .....	262
-------------	-----

CodeGuard 安全性 .....	347, 357
---------------------	----------

#### CPU

寄存器 .....	21
控制寄存器 .....	26
时钟系统选项 .....	105
带后分频器的 FRC 振荡器 .....	105
带 PLL 的 FRC 振荡器 (FRCPLL) .....	105
带 PLL 的主振荡器 (XTPLL、HSPLL 和 ECPLL) .....	105
低功耗 RC (LPRC) 振荡器 .....	105
快速 RC (FRC) 振荡器 .....	105
主 (XT、HS 和 EC) 振荡器 .....	105

数据空间寻址 .....	21
寻址模式 .....	21
指令集 .....	21
资源 .....	25

CPU 的特殊功能 .....	347
-----------------	-----

#### 程序存储器

复位向量 .....	36
构成 .....	36
与数据存储空间的接口 .....	58

程序地址空间 .....	31
--------------	----

表读低位字指令 (TBLRDL) .....	59
表读高位字指令 (TBLRDH) .....	59
存储器映射 (dsPIC33EP128GS70X/80X 器件, 双分区) .....	35
存储器映射 (dsPIC33EP128GS70X/80X 器件) .....	33
存储器映射 (dsPIC33EP64GS70X/80X 器件, 双分区) .....	34
存储器映射 (dsPIC33EP64GS70X/80X 器件) .....	32
构成 .....	58
使用表指令访问程序存储器中的数据 .....	59

串行外设接口。请参见 SPI。 .....	229
-----------------------	-----

串行外设接口 (SPI) .....	229
--------------------	-----

存储器构成 .....	31
-------------	----

特殊功能寄存器映射 .....	40
-----------------	----

资源 .....	39
----------	----

### D

#### DMA

##### DMA 控制器

控制寄存器 .....	92
-------------	----

DMAxCNT .....	92
---------------	----

DMAxCON .....	92
---------------	----

DMAxPAD .....	92
---------------	----

DMAxREQ .....	92
---------------	----

DMAxSTAL/H .....	92
------------------	----

DMAxSTBL/H .....	92
------------------	----

通道与外设关联 .....	90
---------------	----

支持的外设 .....	89
-------------	----

DSP 引擎 .....	30
--------------	----

打盹模式 .....	117
------------	-----

代码保护 .....	347, 357
------------	----------

#### 代码示例

PWM 写保护寄存器的解锁序列 .....	188
-----------------------	-----

PWRSAV 指令语法 .....	115
-------------------	-----

端口写 / 读操作 .....	130
-----------------	-----

电气特性 .....	375
------------	-----

交流 .....	387
----------	-----

#### 多路开关输入源

CLC1 .....	265
------------	-----

CLC2 .....	266
------------	-----

CLC3 .....	267
------------	-----

CLC4 .....	268
------------	-----

### F

封装 .....	439
----------	-----

标识 .....	439
----------	-----

详细信息 .....	441
------------	-----

复位 .....	69
----------	----

RESET 指令 (SWR) .....	69
----------------------	----

非法条件复位 (IOPUWR) .....	69
-----------------------	----

安全性 .....	69
-----------	----

非法操作码 .....	69
-------------	----

未初始化的 W 寄存器 .....	69
-------------------	----

看门狗定时器超时复位 (WDTO) .....	69
-------------------------	----

配置不匹配复位 (CM) .....	69
--------------------	----

欠压复位 (BOR) .....	69
------------------	----

上电复位 (POR) .....	69
------------------	----

陷阱冲突复位 (TRAPR) .....	69
----------------------	----

主复位 (MCLR) 引脚复位 .....	69
-----------------------	----

资源 .....	70
----------	----

### G

高速 12 位模数转换器 (ADC) .....	273
--------------------------	-----

控制寄存器 .....	276
-------------	-----

特性概述 .....	273
------------	-----

资源 .....	276
----------	-----

# dsPIC33EPXXGS70X/80X 系列

高速模拟比较器	
脉冲宽度延长器和数字逻辑	335
数模比较器 (DAC)	335
说明	334
特性概述	333
应用	335
滞后	336
资源	336
高速 PWM	
特性	187
写保护寄存器	188
资源	188
公式	
FPLLO 计算	105
Fvco 计算	105
器件工作频率	105
器件工作频率和 SPIx 时钟速度之间的关系	244
<b>H</b>	
恒流源	345
控制寄存器	346
说明	345
特性概述	345
互联网地址	475
汇编器	
MPASM 汇编器	372
<b>I</b>	
I/O 端口	125
并行 I/O (PIO)	125
寄存器映射	127
PORTA	127
PORTB	127
PORTC	128
PORTD	128
PORTE	129
控制寄存器	131
漏极开路配置	130
配置模拟 / 数字端口引脚	130
写 / 读时序	130
有用技巧	140
资源	141
I <sup>2</sup> C	245
控制寄存器	247
资源	245
I <sup>2</sup> C。请参见 I <sup>2</sup> C。	
<b>J</b>	
JTAG 边界扫描接口	347
JTAG 接口	357
寄存器	
ACLKCON (附属时钟分频比控制)	112
ADCAL0L (ADC 校准 0 的低位字)	299
ADCAL0H (ADC 校准 0 的高位字)	300
ADCAL1H (ADC 校准 1 的高位字)	301
ADCMPxCON (ADC 数字比较器 x 控制)	302
ADCMPxENH (ADC 数字比较器 x 通道使能的 高位字)	303
ADCMPxENL (ADC 数字比较器 x 通道使能的 低位字)	303
ADCON1H (ADC 控制 1 的高位字)	277
ADCON1L (ADC 控制 1 的低位字)	276
ADCON2H (ADC 控制 2 的高位字)	279
ADCON2L (ADC 控制 2 的低位字)	278
ADCON3H (ADC 控制 3 的高位字)	281
ADCON3L (ADC 控制 3 的低位字)	280
ADCON4H (ADC 控制 4 的高位字)	283
ADCON4L (ADC 控制 4 的低位字)	282
ADCON5H (ADC 控制 5 的高位字)	285
ADCON5L (ADC 控制 5 的低位字)	284
ADCORExH (专用 ADC 内核 x 控制的高位字)	287
ADCORExL (专用 ADC 内核 x 控制的低位字)	286
ADEIEH (ADC 提前中断允许的高位字)	289
ADEIEL (ADC 提前中断允许的低位字)	289
ADEISTATH (ADC 提前中断状态的高位字)	290
ADEISTATL (ADC 提前中断状态的低位字)	290
ADFLxCON (ADC 数字滤波器 x 控制)	304
ADIEH (ADC 中断允许的高位字)	293
ADIEL (ADC 中断允许的低位字)	293
ADLVLTRGH (ADC 电平敏感触发控制的 高位字)	288
ADLVLTRGL (ADC 电平敏感触发控制的 低位字)	288
ADM0D0H (ADC 输入模式控制 0 的高位字)	291
ADM0D0L (ADC 输入模式控制 0 的低位字)	291
ADM0D1L (ADC 输入模式控制 1 的低位字)	292
ADSTATH (ADC 数据就绪状态的高位字)	294
ADSTATL (ADC 数据就绪状态的低位字)	294
ADTRIGxH (ADC 通道触发 x 选择的高位字)	297
ADTRIGxL (ADC 通道触发 x 选择的低位字)	295
ALTDTRx (PWMx 备用死区)	203
ANSELx (模拟选择控制 x)	134
AUXCONx (PWMx 附属控制)	211
CHOP (PWMx 斩波时钟发生器)	196
CLCxCONH (CLCx 控制高位字)	263
CLCxCONL (CLCx 控制低位字)	262
CLCxGLSH (CLCx 门逻辑输入选择高位字)	271
CLCxGLSL (CLCx 门逻辑输入选择低位字)	269
CLCxSEL (CLCx 输入多路开关选择)	264
CLKDIV (时钟分频比)	109
CMPxCON (比较器 x 控制)	337
CMPxDAC (比较器 x DAC 控制)	339
CNENx (输入电平变化通知中断允许 x)	133
CNPDx (输入电平变化通知下拉使能 x)	134
CNPUx (输入电平变化通知上拉使能 x)	133
CORCON (内核控制)	28, 83
CTXTSTAT (CPU W 寄存器现场状态)	29
CxBUFNT1 (CANx 过滤器 0-3 缓冲区指针 1)	318
CxBUFNT2 (CANx 过滤器 4-7 缓冲区指针 2)	319
CxBUFNT3 (CANx 过滤器 8-11 缓冲区 指针 3)	319
CxBUFNT4 (CANx 过滤器 12-15 缓冲区 指针 4)	320
CxCFG1 (CANx 波特率配置 1)	316
CxCFG2 (CANx 波特率配置 2)	317
CxCTRL1 (CANx 控制 1)	309
CxCTRL2 (CANx 控制 2)	310
CxEC (CANx 发送 / 接收错误计数)	316
CxFCTRL (CANx FIFO 控制)	312
CxFEN1 (CANx 接收过滤器使能 1)	318
CxFIFO (CANx FIFO 状态)	313
CxFMSKSEL1 (CANx 过滤器 7-0 屏蔽选择 1)	322
CxFMSKSEL2 (CANx 过滤器 15-8 屏蔽选择 2)	323
CxINTE (CANx 中断允许)	315
CxINTF (CANx 中断标志)	314
CxRXFnEID (CANx 接收过滤器 n 扩展标识符)	321
CxRXFnSID (CANx 接收过滤器 n 标准标识符)	321

# dsPIC33EPXXGS70X/80X 系列

CxRXFUL1 (CANx 接收缓冲区满 1) .....	325	PMD1 (外设模块禁止控制 1) .....	118
CxRXFUL2 (CANx 接收缓冲区满 2) .....	325	PMD2 (外设模块禁止控制 2) .....	120
CxRXMnEID (CANx 接收过滤器屏蔽器 n 扩展标识符) .....	324	PMD3 (外设模块禁止控制 3) .....	121
CxRXMnSID (CANx 接收过滤器屏蔽器 n 标准标识符) .....	324	PMD4 (外设模块禁止控制 4) .....	121
CxRXOVF1 (CANx 接收缓冲区溢出 1) .....	326	PMD6 (外设模块禁止控制 6) .....	122
CxRXOVF2 (CANx 接收缓冲区溢出 2) .....	326	PMD7 (外设模块禁止控制 7) .....	123
CxTRmnCON (CANx 发送/接收缓冲区 mn 控制) .....	327	PMD8 (外设模块禁止控制 8) .....	124
CxVEC (CANx 中断编码) .....	311	PORTx (I/O PORTx) .....	131
DEVID (器件 ID) .....	354	PTCON (PWM 时基控制) .....	191
DEVREV (器件版本) .....	354	PTCON2 (PWM 时钟分频比选择) .....	192
DMALCA (上一次工作的 DMA 通道状态) .....	100	PTGADJ (PTG 调节) .....	223
DMAPPS (DMA 乒乓状态) .....	101	PTGBTE (PTG 广播触发使能) .....	218
DMAPWC (DMA 外设写冲突状态) .....	98	PTGC0LIM (PTG 计数器 0 限制) .....	221
DMARQC (DMA 请求冲突状态) .....	99	PTGC1LIM (PTG 计数器 1 限制) .....	222
DMAxCNT (DMA 通道 x 传输计数) .....	96	PTGCON (PTG 控制) .....	217
DMAxCON (DMA 通道 x 控制) .....	92	PTGCST (PTG 控制/状态) .....	215
DMAxPAD (DMA 通道 x 外设地址) .....	96	PTGHOLD (PTG 保持) .....	222
DMAxREQ (DMA 通道 x IRQ 选择) .....	93	PTGL0 (PTG 立即数 0) .....	223
DMAxSTAH (DMA 通道 x 起始地址 A, 高位字) .....	94	PTGQPTR (PTG 步阶队列指针) .....	224
DMAxSTAL (DMA 通道 x 起始地址 A, 低位字) .....	94	PTGQUEX (PTG 步阶队列 x) .....	224
DMAxSTBH (DMA 通道 x 起始地址 B, 高位字) .....	95	PTGSDLIM (PTG 步阶延时限制) .....	221
DMAxSTBL (DMA 通道 x 起始地址 B, 低位字) .....	95	PTGT0LIM (PTG Timer0 限制) .....	220
DSADRH (DMA 最近访问的 RAM 地址高位字) .....	97	PTGT1LIM (PTG Timer1 限制) .....	220
DSADRL (DMA 最近访问的 RAM 地址低位字) .....	97	PTPER (PWMx 主主控时基周期) .....	193
DTRx (PWMx 死区) .....	203	PWMCAPx (PWMx 主时基捕捉) .....	212
FCLCONx (PWMx 故障限流控制) .....	207	PWMCONx (PWMx 控制) .....	198
I2CxCONH (I2Cx 控制高位字) .....	249	PWMKEY (PWMx 保护锁定/解锁密钥) .....	197
I2CxCONL (I2Cx 控制低位字) .....	247	RCON (复位控制) .....	71
I2CxMSK (I2Cx 从模式地址掩码) .....	252	REFOCN (参考振荡器控制) .....	113
I2CxSTAT (I2Cx 状态) .....	250	RPINR0 (外设引脚选择输入 0) .....	142
ICxCON1 (输入捕捉 x 控制 1) .....	178	RPINR1 (外设引脚选择输入 1) .....	142
ICxCON2 (输入捕捉 x 控制 2) .....	179	RPINR11 (外设引脚选择输入 11) .....	145
INTCON1 (中断控制 1) .....	84	RPINR12 (外设引脚选择输入 12) .....	145
INTCON2 (中断控制 2) .....	86	RPINR13 (外设引脚选择输入 13) .....	146
INTCON3 (中断控制 3) .....	87	RPINR18 (外设引脚选择输入 18) .....	146
INTCON4 (中断控制 4) .....	87	RPINR19 (外设引脚选择输入 19) .....	147
INTTREG (中断控制和状态) .....	88	RPINR2 (外设引脚选择输入 2) .....	143
IOCONx (PWMx I/O 控制) .....	205	RPINR20 (外设引脚选择输入 20) .....	147
ISRCCON (恒流源控制) .....	346	RPINR21 (外设引脚选择输入 21) .....	148
LATx (PORTx 数据锁存器) .....	132	RPINR22 (外设引脚选择输入 22) .....	148
LEBCONx (PWMx 前沿消隐控制) .....	209	RPINR23 (外设引脚选择输入 23) .....	149
LEBDLYx (PWMx 前沿消隐延时) .....	210	RPINR26 (外设引脚选择输入 26) .....	149
LSFR (线性反馈移位) .....	114	RPINR29 (外设引脚选择输入 29) .....	150
MDC (PWMx 主控占空比) .....	197	RPINR3 (外设引脚选择输入 3) .....	143
NVMADR (非易失性存储器低位字地址) .....	67	RPINR30 (外设引脚选择输入 30) .....	150
NVMADRU (非易失性存储器高位字地址) .....	67	RPINR37 (外设引脚选择输入 37) .....	151
NVMCON (非易失性存储器 (NVM) 控制) .....	65	RPINR38 (外设引脚选择输入 38) .....	151
NVMKEY (非易失性存储器密钥) .....	68	RPINR42 (外设引脚选择输入 42) .....	152
NVMSRCADR (NVM 源数据地址) .....	68	RPINR43 (外设引脚选择输入 43) .....	152
OCxCON1 (输出比较 x 控制 1) .....	182	RPINR45 (外设引脚选择输入 45) .....	153
OCxCON2 (输出比较 x 控制 2) .....	184	RPINR46 (外设引脚选择输入 46) .....	153
ODCx (PORTx 漏极开路控制) .....	132	RPINR7 (外设引脚选择输入 7) .....	144
OSCCON (振荡器控制) .....	107	RPINR8 (外设引脚选择输入 8) .....	144
OSCTUN (FRC 振荡器调节) .....	111	RPOR0 (外设引脚选择输出 0) .....	154
PDCx (PWMx 发生器占空比) .....	200	RPOR1 (外设引脚选择输出 1) .....	154
PGAxCAL (PGAx 校准) .....	344	RPOR10 (外设引脚选择输出 10) .....	159
PGAxCON (PGAx 控制) .....	343	RPOR11 (外设引脚选择输出 11) .....	159
PHASEx (PWMx 主相移) .....	201	RPOR12 (外设引脚选择输出 12) .....	160
PLLFB (PLL 反馈分频比) .....	110	RPOR13 (外设引脚选择输出 13) .....	160
		RPOR14 (外设引脚选择输出 14) .....	161
		RPOR15 (外设引脚选择输出 15) .....	161
		RPOR16 (外设引脚选择输出 16) .....	162

# dsPIC33EPXXGS70X/80X 系列

RPOR17 (外设引脚选择输出 17) .....	162	SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 1, CKP = 0, SMP = 0) 要求 .....	403
RPOR18 (外设引脚选择输出 18) .....	163	SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 1, CKP = 1, SMP = 0) 要求 .....	405
RPOR19 (外设引脚选择输出 19) .....	163	SPI1、SPI2 和 SPI3 主模式 (半双工, 仅发送) 要求 .....	399
RPOR2 (外设引脚选择输出 2) .....	155	SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) 要求 .....	401
RPOR20 (外设引脚选择输出 20) .....	164	SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 1, CKP = x, SMP = 1) 要求 .....	400
RPOR21 (外设引脚选择输出 21) .....	164	SPI3 从模式 (全双工, CKE = 0, CKP = 0, SMP = 0) 要求 .....	421
RPOR22 (外设引脚选择输出 22) .....	165	SPI3 从模式 (全双工, CKE = 0, CKP = 1, SMP = 0) 要求 .....	419
RPOR23 (外设引脚选择输出 23) .....	165	SPI3 从模式 (全双工, CKE = 1, CKP = 0, SMP = 0) 要求 .....	415
RPOR24 (外设引脚选择输出 24) .....	166	SPI3 从模式 (全双工, CKE = 1, CKP = 1, SMP = 0) 要求 .....	417
RPOR25 (外设引脚选择输出 25) .....	166	SPI3 主模式 (半双工, 仅发送) 要求 .....	411
RPOR26 (外设引脚选择输出 26) .....	167	SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) 要求 .....	413
RPOR3 (外设引脚选择输出 3) .....	155	SPI3 主模式 (全双工, CKE = 1, CKP = x, SMP = 1) 要求 .....	412
RPOR4 (外设引脚选择输出 4) .....	156	Timer2/Timer4 外部时钟要求 .....	394
RPOR5 (外设引脚选择输出 5) .....	156	Timer3/Timer5 外部时钟要求 .....	394
RPOR6 (外设引脚选择输出 6) .....	157	Timer1 外部时钟要求 .....	393
RPOR7 (外设引脚选择输出 7) .....	157	UARTx I/O 要求 .....	427
RPOR8 (外设引脚选择输出 8) .....	158	附属 PLL 时钟 .....	389
RPOR9 (外设引脚选择输出 9) .....	158	复位、WDT、OST 和 PWRT 要求 .....	392
SDCx (PWMx 辅助占空比) .....	200	负载条件 .....	387
SEVTCMP (PWMx 特殊事件比较) .....	193	高速 PWMx 要求 .....	397
SPHASEx (PWMx 辅助相移) .....	202	模拟电流规范 .....	427
SPIxCON1H (SPIx 控制 1 高位字) .....	234	模数转换要求 .....	430
SPIxCON1L (SPIx 控制 1 低位字) .....	232	内部 FRC 精度 .....	390
SPIxCON2L (SPIx 控制 2 低位字) .....	236	内部 LPRC 精度 .....	390
SPIxIMSKH (SPIx 中断掩码高位字) .....	241	输出比较 x 要求 .....	396
SPIxIMSKL (SPIx 中断掩码低位字) .....	240	输出引脚上的容性负载要求 .....	387
SPIxSTATH (SPIx 状态高位字) .....	239	输入捕捉 x 要求 .....	395
SPIxSTATL (SPIx 状态低位字) .....	237	外部时钟要求 .....	388
SR (CPU STATUS) .....	26, 82	温度和电压规范 .....	387
SSEVTCMP (PWMx 辅助特殊事件比较) .....	196	节能特性 .....	115
STCON (PWMx 辅助主控时基控制) .....	194	时钟频率和切换 .....	115
STCON2 (PWMx 辅助时钟分频比选择) .....	195	资源 .....	117
STPER (PWMx 辅助主控时基周期) .....	195	绝对最大值 .....	375
STRIGx (PWMx 辅助触发比较值) .....	208	<b>K</b>	
T1CON (Timer1 控制) .....	171	开发支持 .....	371
TRGCONx (PWMx 触发控制) .....	204	看门狗定时器 (WDT) .....	347, 356
TRIGx (PWMx 主触发比较值) .....	206	编程注意事项 .....	356
TRISx (PORTx 数据方向控制) .....	131	勘误表 .....	10
TxCON (Timer2/4 控制) .....	175	可编程增益放大器。请参见 PGA。	
TyCON (Timer3/5 控制) .....	176	可编程增益放大器 (PGA) .....	341
UxMODE (UARTx 模式) .....	255	说明 .....	342
UxSTA (UARTx 状态和控制) .....	257	资源 .....	343
基于指令的节能模式 .....	115	客户通知服务 .....	475
空闲 .....	116	客户支持 .....	475
休眠 .....	116	可配置逻辑单元。请参见 CLC。	
交流 / 直流特性		可配置逻辑单元 (CLC) .....	259
DACx 规范 .....	431	控制器局域网。请参见 CAN。	
PGAx 规范 .....	432	框图	
高速模拟比较器规范 .....	430	16 位 Timer1 模块 .....	169
交流特性 .....	387	ADC 模块 .....	274
ADC 规范 .....	428	B 类 /C 类定时器对 (32 位定时器) .....	174
CANx I/O 要求 .....	426	CANx 模块 .....	308
DMA 模块要求 .....	433		
I/O 要求 .....	391		
I2Cx 总线数据要求 (从模式) .....	425		
I2Cx 总线数据要求 (主模式) .....	423		
OCx/PWMx 模块要求 .....	396		
PLL 时钟 .....	389		
SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 0, CKP = 0, SMP = 0) 要求 .....	409		
SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 0, CKP = 1, SMP = 0) 要求 .....	407		

# dsPIC33EPXXGS70X/80X 系列

CLCx 逻辑功能组合选项 .....	260
CLCx 模块 .....	259
CLCx 输入源选择 .....	261
CPU 内核 .....	22
DMA 控制器 .....	91
dsPIC33EP128GS70X/80X 的安全段 (双分区模式) .....	359
dsPIC33EP64GS70X/80X 的安全段 (双分区模式) .....	358
dsPIC33EPXXGS70X/80X 系列 .....	11
dsPIC33EPXXXGS70X/80X 的安全段 .....	358
I2Cx 模块 .....	246
MCLR 引脚连接 .....	16
PGAx 功能 .....	342
PGAx 模块 .....	341
PLL 模块 .....	105
PSV 的读地址生成方式 .....	49
PTG 模块 .....	214
RPn 的可重映射输出的复用 .....	138
SPIx 从一帧从模式连接 .....	244
SPIx 从一帧主模式连接 .....	244
SPIx 模块 (标准模式) .....	230
SPIx 模块 (增强型模式) .....	231
SPIx 主/从连接 (标准模式) .....	242
SPIx 主/从器件连接 (增强型缓冲区模式) .....	243
SPIx 主一帧从模式连接 .....	244
SPIx 主一帧主模式连接 .....	243
Timerx (x = 2 至 5) .....	174
U1RX 的可重映射输入 .....	136
UARTx 模块 .....	253
编程模型 .....	24
表寄存器寻址 .....	61
调用堆栈帧 .....	52
访问程序空间内数据的地址生成方式 .....	58
复位系统 .....	69
高速模拟比较器 x .....	334
高速 PWM 的简化概念 .....	190
高速 PWM 架构 .....	189
共用 ADC 内核 .....	275
共用端口结构 .....	125
恒流源 .....	345
建议的最基本连接 .....	16
交错式 PFC .....	18
看门狗定时器 (WDT) .....	356
离线式 UPS .....	20
片上稳压器连接 .....	355
输出比较 x 模块 .....	181
输入捕捉 x .....	177
外设与 DMA 控制器 .....	89
相移全桥转换器 .....	19
振荡器电路的建议布线方式 .....	17
振荡器系统 .....	104
滞后控制 .....	336
专用 ADC 内核 0-3 .....	275

## L

LPRC 振荡器 与 WDT 配合使用 .....	356
灵活的配置 .....	347
滤波电容 (CEFC) 规范 .....	377

## M

Microchip 互联网网站 .....	475
MPLAB ASM30 汇编器、链接器和库管理器 .....	372
MPLAB PM3 器件编程器 .....	373
MPLAB REAL ICE 在线仿真器系统 .....	373
MPLAB 集成开发环境软件 .....	371
MPLINK 目标链接器 /MPLIB 目标库管理器 .....	372
脉宽调制器。请参见 PWM。	
模数转换器。请参见 ADC。	
模寻址 .....	55
W 地址寄存器选择 .....	55
操作示例 .....	55
起始和结束地址 .....	55
应用 .....	56

## P

### PGA PTG

步阶命令和格式 .....	225
简介 .....	213
控制寄存器 .....	215
输出说明 .....	228

### PWM

配置寄存器映射 .....	348
配置位 .....	347
说明 .....	349

## Q

器件唯一标识符 (UID) .....	32
器件校准 .....	353
地址 .....	353
和标识 .....	353
欠压复位 (BOR) .....	347, 355
前沿消隐 (LEB) .....	187

## R

热封装特性 .....	376
热工作条件 .....	376
入门指南 .....	15
CPU 逻辑滤波电容连接 (VCAP) .....	16
ICSP 引脚 .....	17
连接要求 .....	15
目标应用 .....	18
启动时的振荡器值条件 .....	18
去耦电容 .....	15
外部振荡器引脚 .....	17
未用 I/O .....	18
主复位 (MCLR) 引脚 .....	16
软件模拟器 (MPLAB SIM) .....	373

## S

### SFR 块

000h .....	40
100h .....	40
200h .....	41
300h .....	41
400h .....	42
500h .....	42
600h .....	43
700h .....	44
800h .....	45
900h .....	45

# dsPIC33EPXXGS70X/80X 系列

A00h.....	46	SFR 空间 .....	37
B00h.....	46	构成, 对齐方式 .....	37
C00h-D00h.....	47	宽度 .....	37
E00h-F00h .....	48	数据空间 .....	
闪存程序存储器 .....	61	分页存储器方案 .....	49
RTSP 工作原理 .....	62	分页数据存储空间 (图) .....	50
操作 .....	62	扩展 X .....	52
和表指令 .....	61	输入捕捉 .....	177
控制寄存器 .....	64	控制寄存器 .....	178
双分区闪存配置 .....	63	资源 .....	177
资源 .....	63	输入电平变化通知 (ICN) .....	130
时序图 .....		算术逻辑单元 (ALU) .....	30
BOR 和主复位特性 .....	391	<b>T</b> .....	
CANx I/O .....	426	Timer2/3 和 Timer4/5 .....	173
I/O 特性 .....	391	控制寄存器 .....	175
I2Cx 总线启动位 / 停止位 (从模式) .....	424	资源 .....	173
I2Cx 总线启动位 / 停止位 (主模式) .....	422	Timer1 .....	169
I2Cx 总线数据 (从模式) .....	424	控制寄存器 .....	171
I2Cx 总线数据 (主模式) .....	422	模式设置 .....	169
OCx/PWMx 特性 .....	396	资源 .....	170
SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 0, CKP = 0, SMP = 0) .....	408	通用异步收发器。请参见 UART。	
SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 0, CKP = 1, SMP = 0) .....	406	通用异步收发器 (UART) .....	253
SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 1, CKP = 0, SMP = 0) .....	402	资源 .....	254
SPI1、SPI2 和 SPI3 从模式 (全双工, CKE = 1, CKP = 1, SMP = 0) .....	404	通用异步收发器 (UART) 模块 .....	
SPI1、SPI2 和 SPI3 主模式 (半双工, 仅发送, CKE = 0) .....	398	控制寄存器 .....	255
SPI1、SPI2 和 SPI3 主模式 (半双工, 仅发送, CKE = 1) .....	399	有用技巧 .....	254
SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) .....	401	<b>U</b> .....	
SPI1、SPI2 和 SPI3 主模式 (全双工, CKE = 1, CKP = x, SMP = 1) .....	400	UART .....	
SPI3 从模式 (全双工, CKE = 0, CKP = 0, SMP = 0) .....	420	<b>W</b> .....	
SPI3 从模式 (全双工, CKE = 0, CKP = 1, SMP = 0) .....	418	WWW, 在线支持 .....	10
SPI3 从模式 (全双工, CKE = 1, CKP = 0, SMP = 0) .....	414	WWW 地址 .....	475
SPI3 从模式 (全双工, CKE = 1, CKP = 1, SMP = 0) .....	416	外设触发信号发生器。请参见 PTG。	
SPI3 主模式 (半双工, 仅发送, CKE = 0) .....	410	外设触发信号发生器 (PTG) 模块 .....	213
SPI3 主模式 (半双工, 仅发送, CKE = 1) .....	411	外设模块禁止 (PMD) .....	117
SPI3 主模式 (全双工, CKE = 0, CKP = x, SMP = 1) .....	413	外设引脚选择 (PPS) .....	135
SPI3 主模式 (全双工, CKE = 1, CKP = x, SMP = 1) .....	412	可选择的输入源 .....	137
Timer1-Timer5 外部时钟特性 .....	393	可用外设 .....	135
UARTx I/O 特性 .....	427	可用引脚 .....	135
高速 PWMx 故障特性 .....	397	可重映射引脚的输出选择 .....	139
高速 PWMx 模块特性 .....	397	控制 .....	135
输出比较 x (OCx) 特性 .....	396	控制寄存器 .....	142
输入捕捉 x (ICx) 特性 .....	395	输出映射 .....	138
外部时钟 .....	388	输入映射 .....	136
输出比较 .....	181	位反转寻址 .....	56
控制寄存器 .....	182	示例 .....	57
资源 .....	181	实现 .....	56
数据地址空间 .....	37	序列表 (16 项) .....	57
dsPIC33EP64GS70X/80X 器件的存储器映射 .....	38	稳压器 (片上) .....	355
Near 数据空间 .....	37	<b>Y</b> .....	
		引脚 I/O 说明 (表) .....	12
		用户 OTP 存储器 .....	355
		<b>Z</b> .....	
		在节能指令执行期间的中断 .....	116
		在线串行编程 (ICSP) .....	347, 357
		在线调试器 .....	357
		在线仿真 .....	347
		振荡器 .....	
		控制寄存器 .....	107
		资源 .....	106



# dsPIC33EPXXGS70X/80X 系列

振荡器配置 .....	103
直接存储器访问。请参见 DMA。	
指令集汇总 .....	361
操作码说明中使用的符号 .....	362
概述 .....	364
指令寻址模式 .....	53
MAC 指令 .....	54
MCU 指令 .....	53
其他指令 .....	54
文件寄存器指令 .....	53
支持的基本模式 .....	53
传送指令和累加器指令 .....	54
直流和交流特性	
图表 .....	435
直流特性	
DACx 输出 (DACOUTx 引脚) 规范 .....	431
I/O 引脚输出规范 .....	385
I/O 引脚输入规范 .....	382
程序存储器 .....	386
打盹电流 (IDOZE) .....	381
掉电电流 (IPD) .....	380
工作电流 (IDD) .....	378
工作 MIPS 与电压 .....	376
恒流源规范 .....	433
看门狗定时器增加电流 ( $\Delta I_{WDT}$ ) .....	380
空闲电流 (IDLE) .....	379
欠压复位 (BOR) .....	385
温度和电压规范 .....	377
中断控制器	
备用中断向量表 (AIVT) .....	73
复位过程 .....	73
控制和状态寄存器 .....	81
INTCON1 .....	81
INTCON2 .....	81
INTCON3 .....	81
INTCON4 .....	81
INTTREG .....	81
中断向量表 (IVT) .....	73
中断向量详细信息 .....	76
资源 .....	81

# dsPIC33EPXXXGS70X/80X 系列

---

注:

## Microchip 网站

Microchip 网站 ([www.microchip.com](http://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。只要使用常用的互联网浏览器即可访问。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 (Frequently Asked Questions, FAQ)、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

## 变更通知客户服务

Microchip 的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](http://www.microchip.com)。在“支持” (Support) 下，点击“变更通知客户” (Customer Change Notification) 服务后按照注册说明完成注册。

## 客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过以下网站获得技术支持：  
<http://microchip.com/support>

# dsPIC33EPXXXGS70X/80X 系列

---

注:

# dsPIC33EPXXGS70X/80X 系列

## 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。

dsPIC 33 EP 64 GS8 04 T - I / PT XXX	
Microchip 的商标	_____
架构	_____
闪存系列	_____
程序存储器容量 (KB)	_____
产品组	_____
引脚数	_____
卷带标志 (如果适用)	_____
温度范围	_____
封装	_____
定制编号	_____

<b>架构:</b>	33 = 16 位数字信号控制器
<b>闪存系列:</b>	EP = 增强性能
<b>产品组:</b>	GS = SMPS 系列
<b>引脚数:</b>	02 = 28 引脚 04 = 44 引脚 05 = 48 引脚 06 = 64 引脚 08 = 80 引脚
<b>温度范围:</b>	I = -40°C 至 +85°C (工业级) E = -40°C 至 +125°C (扩展级)
<b>封装:</b>	ML = 塑封正方无脚封装 —— (44 引脚) 主体 8x8 mm (QFN) MM = 塑封正方无脚封装 —— (28 引脚) 主体 6x6 mm (QFN-S) 2N = 塑封正方扁平无脚封装 —— (28 引脚) 主体 6x6 mm (UQFN) PT = 塑封薄型正方扁平封装 —— (44 引脚) 主体 10x10 mm (TQFP) PT = 塑封薄型正方扁平封装 —— (48 引脚) 主体 7x7 mm (TQFP) PT = 塑封薄型正方扁平封装 —— (64 引脚) 主体 10x10 mm (TQFP) PT = 塑封薄型正方扁平封装 —— (80 引脚) 主体 12x12 mm (TQFP) SO = 塑封宽条小外形封装 —— (28 引脚) 主体 7.50 mm (SOIC)

### 示例:

dsPIC33EP64GS804-I/PT:  
dsPIC33, 增强性能,  
64 KB 程序存储器, SMPS,  
44 引脚, 工业级温度,  
TQFP 封装。

# dsPIC33EPXXXGS70X/80X 系列

---

注:

---

---

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中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字千年版权法案 (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. 的英文原版文档。

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

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-16949:2009 认证。Microchip 的 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 and 模拟产品 严格遵守公司的质量体系流程。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。

**QUALITY MANAGEMENT SYSTEM**  
**CERTIFIED BY DNV**  
**== ISO/TS 16949 ==**

## 商标

Microchip 的名称和徽标组合、Microchip 徽标、AnyRate、AVR、AVR 徽标、AVR Freaks、BeaconThings、BitCloud、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KEELOQ、KEELOQ 徽标、Kleer、LANCheck、LINK MD、maXStylus、maXTouch、MediaLB、megaAVR、MOST、MOST 徽标、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32 徽标、Prochip Designer、QTouch、RightTouch、SAM-BA、SpyNIC、SST、SST 徽标、SuperFlash、tinyAVR、UNI/O 及 XMEGA 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTouch、Precision Edge 和 Quiet-Wire 均为 Microchip Technology Inc. 在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BodyCom、chipKIT、chipKIT 徽标、CodeGuard、CryptoAuthentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherGREEN、In-Circuit Serial Programming、ICSP、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNet 徽标、Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PureSilicon、QMatrix、RightTouch 徽标、REAL ICE、Ripple Blocker、SAM-ICE、Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、ViewSpan、WiperLock、Wireless DNA 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

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

Silicon Storage Technology 为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

GestIC 为 Microchip Technology Inc. 的子公司 Microchip Technology Germany II GmbH & Co. & KG 在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2017, Microchip Technology Inc. 版权所有。

ISBN: 978-1-5224-2159-7

## 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://www.microchip.com/support>

网址: [www.microchip.com](http://www.microchip.com)

**亚特兰大 Atlanta**  
Duluth, GA  
Tel: 1-678-957-9614  
Fax: 1-678-957-1455

**奥斯汀 Austin, TX**  
Tel: 1-512-257-3370

**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**  
Novi, MI  
Tel: 1-248-848-4000

**休斯敦 Houston, TX**  
Tel: 1-281-894-5983

**印第安纳波利斯 Indianapolis**  
Noblesville, IN  
Tel: 1-317-773-8323  
Fax: 1-317-773-5453  
Tel: 1-317-536-2380

**洛杉矶 Los Angeles**  
Mission Viejo, CA  
Tel: 1-949-462-9523  
Fax: 1-949-462-9608  
Tel: 1-951-273-7800

**罗利 Raleigh, NC**  
Tel: 1-919-844-7510

**纽约 New York, NY**  
Tel: 1-631-435-6000

**圣何塞 San Jose, CA**  
Tel: 1-408-735-9110  
Tel: 1-408-436-4270

**加拿大多伦多 Toronto**  
Tel: 1-905-695-1980  
Fax: 1-905-695-2078

### 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: 852-2943-5100

Fax: 852-2401-3431

**中国 - 北京**  
Tel: 86-10-8569-7000  
Fax: 86-10-8528-2104

**中国 - 成都**  
Tel: 86-28-8665-5511  
Fax: 86-28-8665-7889

**中国 - 重庆**  
Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

**中国 - 东莞**  
Tel: 86-769-8702-9880  
**中国 - 广州**  
Tel: 86-20-8755-8029

**中国 - 杭州**  
Tel: 86-571-8792-8115  
Fax: 86-571-8792-8116

**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: 86-21-3326-8000  
Fax: 86-21-3326-8021

**中国 - 沈阳**  
Tel: 86-24-2334-2829  
Fax: 86-24-2334-2393

**中国 - 深圳**  
Tel: 86-755-8864-2200  
Fax: 86-755-8203-1760

**中国 - 武汉**  
Tel: 86-27-5980-5300  
Fax: 86-27-5980-5118

**中国 - 西安**  
Tel: 86-29-8833-7252  
Fax: 86-29-8833-7256

**中国 - 厦门**  
Tel: 86-592-238-8138  
Fax: 86-592-238-8130

**中国 - 香港特别行政区**  
Tel: 852-2943-5100  
Fax: 852-2401-3431

### 亚太地区

**中国 - 珠海**  
Tel: 86-756-321-0040  
Fax: 86-756-321-0049

**台湾地区 - 高雄**  
Tel: 886-7-213-7830

**台湾地区 - 台北**  
Tel: 886-2-2508-8600  
Fax: 886-2-2508-0102

**台湾地区 - 新竹**  
Tel: 886-3-5778-366  
Fax: 886-3-5770-955

**澳大利亚 Australia - Sydney**  
Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

**印度 India - Bangalore**  
Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

**印度 India - New Delhi**  
Tel: 91-11-4160-8631  
Fax: 91-11-4160-8632

**印度 India - Pune**  
Tel: 91-20-3019-1500

**日本 Japan - Osaka**  
Tel: 81-6-6152-7160  
Fax: 81-6-6152-9310

**日本 Japan - Tokyo**  
Tel: 81-3-6880-3770  
Fax: 81-3-6880-3771

**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

**芬兰 Finland - Espoo**  
Tel: 358-9-4520-820

**法国 France - Paris**  
Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

**德国 Germany - Garching**  
Tel: 49-8931-9700

**德国 Germany - Haan**  
Tel: 49-2129-3766400

**德国 Germany - Heilbronn**  
Tel: 49-7131-67-3636

**德国 Germany - Karlsruhe**  
Tel: 49-721-625370

**德国 Germany - Munich**  
Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

**德国 Germany - Rosenheim**  
Tel: 49-8031-354-560

**以色列 Israel - Ra'anana**  
Tel: 972-9-744-7705

**意大利 Italy - Milan**  
Tel: 39-0331-742611  
Fax: 39-0331-466781

**意大利 Italy - Padova**  
Tel: 39-049-7625286

**荷兰 Netherlands - Drunen**  
Tel: 31-416-690399  
Fax: 31-416-690340

**挪威 Norway - Trondheim**  
Tel: 47-7289-7561

**波兰 Poland - Warsaw**  
Tel: 48-22-3325737

**罗马尼亚 Romania - Bucharest**  
Tel: 40-21-407-87-50

**西班牙 Spain - Madrid**  
Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91

**瑞典 Sweden - Gothenburg**  
Tel: 46-31-704-60-40

**瑞典 Sweden - Stockholm**  
Tel: 46-8-5090-4654

**英国 UK - Wokingham**  
Tel: 44-118-921-5800  
Fax: 44-118-921-5820