

采用 XLP 技术的 20/28/44/48 引脚 通用 16 位闪存单片机

功耗管理模式

- 运行 —— CPU、闪存、SRAM 以及外设工作
- 打盹 —— CPU 时钟的速度比外设慢
- 空闲 —— CPU 不工作，闪存、SRAM 和外设工作
- 休眠 —— CPU、闪存以及外设不工作，SRAM 工作
- 深度休眠 —— CPU、闪存、SRAM 和大部分外设不工作；多种自发唤醒源
- 低功耗：
 - 运行模式电流降至 8 μ A（典型值）
 - 空闲模式电流降至 2.2 μ A（典型值）
 - 深度休眠模式电流降至 20 nA（典型值）
 - 实时时钟 / 日历电流降至 700 nA（32 kHz、1.8V 时）
 - 看门狗定时器电流降至 500 nA（1.8V 时的典型值）

高性能 CPU

- 改进型哈佛架构
- 最高运行速度可达 16 MIPS（32 MHz 时）
- 具有 4xPLL 和多个分频选项的 8 MHz 内部振荡器
- 17 位 X17 位单周期硬件乘法器
- 32 位 X16 位硬件除法器，16 位 x16 位工作寄存器阵列
- 优化的 C 编译器指令集架构

外设特性

- 硬件实时时钟和日历（Real-Time Clock and Calendar, RTCC）：
 - 提供时钟、日历和闹钟功能
 - 可在深度休眠模式下运行
 - 可使用 50/60 Hz 电源线输入作为时钟源
- 32 位可编程循环冗余校验（Cyclic Redundancy Check, CRC）发生器
- 多个串行通信模块：
 - 两个 3 线 / 4 线 SPI 模块
 - 两个支持多主器件 / 从模式的 I²C™ 模块
 - 两个支持 RS-485、RS-232、LIN/J2602 和 IrDA® 红外连接技术的 UART 模块
- 5 个带可编程预分频器的 16 位定时器 / 计数器：
 - 可以配对成 32 位定时器 / 计数器
- 使用专用定时器的 3 个 16 位捕捉输入
- 使用专用定时器的 3 个 16 位比较 / PWM 输出
- 数字 I/O 引脚上的可配置漏极开路输出
- 最多 3 个外部中断源

模拟特性

- 最多 16 路通道的 12 位模数转换器：
 - 100 ksps 转换速率
 - 休眠和空闲期间可进行转换
 - 休眠和空闲模式下基于定时器的自动采样选项
 - 自动比较时唤醒的选项
- 具有可编程输入 / 输出配置的两个轨到轨模拟比较器
- 片内参考电压
- 内部温度传感器
- 充电时间测量单元（Charge Time Measurement Unit, CTMU）：
 - 16 路通道用于电容测量
 - 分辨率低至 200 ps 的时间测量
 - 分辨率低至 1 ns 的延时 / 脉冲生成

单片机的特殊性能

- 宽工作电压范围：
 - 1.8V 至 3.6V（PIC24F 器件）
 - 2.0V 至 5.5V（PIC24FV 器件）
- 低功耗唤醒源和监视器：
 - 用于休眠 / 深度休眠的超低功耗唤醒（ULPWU）功能
 - 用于深度休眠的低功耗看门狗定时器（DSWDT）
 - 用于深度休眠的外部低功耗欠压复位（DSBOR）以及用于所有其他模式的 LPBOR
- 系统频率范围声明位：
 - 声明频率范围可优化电流消耗。
- 标准看门狗定时器（Watchdog Timer, WDT），带片上低功耗 RC 振荡器以便可靠地工作
- 可编程的高 / 低电压检测（High/Low-Voltage Detect, HLVD）
- 带有 3 个可编程跳变点的标准欠压复位（Brown-out Reset, BOR）；在休眠模式下可以禁止
- 所有 I/O 引脚上的高灌 / 拉电流（18 mA/18 mA）
- 闪存程序存储器：
 - 擦除 / 写周期：最少 10,000 次
 - 数据保存时间最短为 40 年
- 数据 EEPROM：
 - 擦除 / 写周期：最少 100,000 次
 - 数据保存时间最短为 40 年
- 故障保护时钟监视器（Fail-Safe Clock Monitor, FSCM）
- 可编程参考时钟输出
- 可在软件控制下进行自编程
- 通过 2 个引脚进行在线串行编程（In-Circuit Serial Programming™, ICSP™）和在线调试（In-Circuit Debug, ICD）

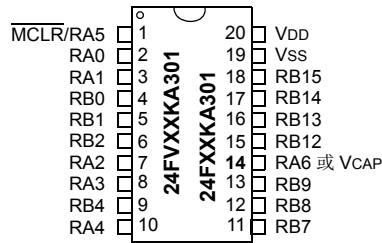
PIC24FV32KA304 系列

PIC24F 器件	引脚数	存储器			16 位 定时器	捕捉 输入	比较 /PWM 输出	UART (带有 IrDA® 接	SPI	I ² C™	12 位 A/D (通道数)	比较器	CTMU (通道数)	RTCC
		闪存 程序 (字节)	SRAM (字节)	EE 数据 (字节)										
PIC24FV16KA301/ PIC24F16KA301	20	16K	2K	512	5	3	3	2	2	2	12	3	12	有
PIC24FV32KA301/ PIC24F32KA301	20	32K	2K	512	5	3	3	2	2	2	12	3	12	有
PIC24FV16KA302/ PIC24F16KA302	28	16K	2K	512	5	3	3	2	2	2	13	3	13	有
PIC24FV32KA302/ PIC24F32KA302	28	32K	2K	512	5	3	3	2	2	2	13	3	13	有
PIC24FV16KA304/ PIC24F16KA304	44	16K	2K	512	5	3	3	2	2	2	16	3	16	有
PIC24FV32KA304/ PIC24F32KA304	44	32K	2K	512	5	3	3	2	2	2	16	3	16	有

PIC24FV32KA304 系列

引脚图

20 引脚 PDIP/SSOP/SOIC⁽¹⁾



引脚	引脚功能	
	PIC24FVXXKA301	PIC24FXXKA301
1	MCLR/VPP/RA5	MCLR/VPP/RA5
2	PGEC2/VREF+/CVREF+/AN0/C3INC/SCK2/CN2/RA0	PGEC2/VREF+/CVREF+/AN0/C3INC/SCK2/CN2/RA0
3	PGED2/CVREF-/VREF-/AN1/SDO2/CN3/RA1	PGED2/CVREF-/VREF-/AN1/SDO2/CN3/RA1
4	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/SDI2/OC2/CN4/RB0	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/SDI2/OC2/CN4/RB0
5	PGEC1/AN3/C1INC/C2INA/U2RX/OC3/CTED12/CN5/RB1	PGEC1/AN3/C1INC/C2INA/U2RX/OC3/CTED12/CN5/RB1
6	AN4/SDA2/T5CK/T4CK/U1RX/CTED13/CN6/RB2	AN4/SDA2/T5CK/T4CK/U1RX/CTED13/CN6/RB2
7	OSCI/AN13/C1INB/C2IND/CLKI/CN30/RA2	OSCI/AN13/C1INB/C2IND/CLKI/CN30/RA2
8	OSCO/AN14/C1INA/C2INC/CLKO/CN29/RA3	OSCO/AN14/C1INA/C2INC/CLKO/CN29/RA3
9	PGED3/SOSCI/AN15/U2RTS/CN1/RB4	PGED3/SOSCI/AN15/U2RTS/CN1/RB4
10	PGEC3/SOSCO/SCLKI/U2CTS/CN0/RA4	PGEC3/SOSCO/SCLKI/U2CTS/CN0/RA4
11	U1TX/C2OUT/OC1/IC1/CTED1/INT0/CN23/RB7	U1TX/INT0/CN23/RB7
12	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8
13	SDA1/T1CK/U1RTS/IC2/CTED4/CN21/RB9	SDA1/T1CK/U1RTS/IC2/CTED4/CN21/RB9
14	VCAP	C2OUT/OC1/IC1/CTED1/INT2/CN8/RA6
15	AN12/HLVDIN/SCK1/SS2/IC3/CTED2/INT2/CN14/RB12	AN12/HLVDIN/SCK1/SS2/IC3/CTED2/CN14/RB12
16	AN11/SDO1/OCFB/CTPLS/CN13/RB13	AN11/SDO1/OCFB/CTPLS/CN13/RB13
17	CVREF/AN10/C3INB/RTCC/SDI1/C1OUT/OCFA/CTED5/INT1/CN12/RB14	CVREF/AN10/C3INB/RTCC/SDI1/C1OUT/OCFA/CTED5/INT1/CN12/RB14
18	AN9/C3INA/SCL2/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15	AN9/C3INA/SCL2/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15
19	VSS/AVSS	VSS/AVSS
20	VDD/AVDD	VDD/AVDD

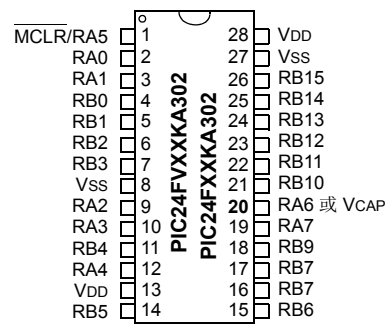
图注：加粗的引脚编号表示 PIC24FV 与 PIC24F 器件引脚功能不同。

注 1：PIC24F32KA304 器件引脚的最大电压为 3.6V，不能承受 5V 电压。

PIC24FV32KA304 系列

引脚图

28 引脚 SPDIP/SSOP/SOIC^(1,2)



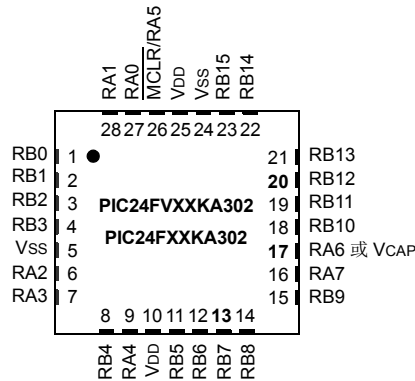
引脚	引脚功能	
	PIC24FVXXKA302	PIC24FXXKA302
1	MCLR/Vpp/RA5	MCLR/Vpp/RA5
2	VREF+/CVREF+/AN0/C3INC/CTED1/CN2/RA0	VREF+/CVREF+/AN0/C3INC/CTED1/CN2/RA0
3	CVREF-/VREF-/AN1/CN3/RA1	CVREF-/VREF-/AN1/CN3/RA1
4	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/CN4/RB0	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/CN4/RB0
5	PGEC1/AN3/C1INC/C2INA/U2RX/CTED12/CN5/RB1	PGEC1/AN3/C1INC/C2INA/U2RX/CN5/RB1
6	AN4/C1INB/C2IND/SDA2/T5CK/T4CK/U1RX/CTED13/CN6/RB2	AN4/C1INB/C2IND/SDA2/T5CK/T4CK/U1RX/CTED13/CN6/RB2
7	AN5/C1INA/C2INC/SCL2/CN7/RB3	AN5/C1INA/C2INC/SCL2/CN7/RB3
8	Vss	Vss
9	OSCI/AN13/CLKI/CN30/RA2	OSCI/AN13/CLKI/CN30/RA2
10	OSCO/AN14/CLKO/CN29/RA3	OSCO/AN14/CLKO/CN29/RA3
11	SOSCI/AN15/U2RTS/CN1/RB4	SOSCI/AN15/U2RTS/CN1/RB4
12	SOSCO/SCLKI/U2CTS/CN0/RA4	SOSCO/SCLKI/U2CTS/CN0/RA4
13	VDD	VDD
14	PGED3/ASDA ⁽¹⁾ /SCK2/CN27/RB5	PGED3/ASDA ⁽¹⁾ /SCK2/CN27/RB5
15	PGEC3/ASCL ⁽¹⁾ /SDO2/CN24/RB6	PGEC3/ASCL ⁽¹⁾ /SDO2/CN24/RB6
16	U1TX/C2OUT/OC1/INT0/CN23/RB7	U1TX/INT0/CN23/RB7
17	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8
18	SDA1/T1CK/U1RTS/IC2/CTED4/CN21/RB9	SDA1/T1CK/U1RTS/IC2/CTED4/CN21/RB9
19	SDI2/IC1/CTED3/CN9/RA7	SDI2/IC1/CTED3/CN9/RA7
20	VCAP	C2OUT/OC1/CTED1/INT2/CN8/RA6
21	PGED2/SDI1/OC3/CTED11/CN16/RB10	PGED2/SDI1/OC3/CTED11/CN16/RB10
22	PGEC2/SCK1/OC2/CTED9/CN15/RB11	PGEC2/SCK1/OC2/CTED9/CN15/RB11
23	AN12/HLVDIN/SS2/IC3/CTED2/INT2/CN14/RB12	AN12/HLVDIN/SS2/IC3/CTED2/CN14/RB12
24	AN11/SDO1/OCFB/CTPLS/CN13/RB13	AN11/SDO1/OCFB/CTPLS/CN13/RB13
25	CVREF/AN10/C3INB/RTCC/C1OUT/OCFA/CTED5/INT1/CN12/RB14	CVREF/AN10/C3INB/RTCC/C1OUT/OCFA/CTED5/INT1/CN12/RB14
26	AN9/C3INA/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15	AN9/C3INA/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15
27	Vss/AVss	Vss/AVss
28	VDD/AVDD	VDD/AVDD

图注：加粗的引脚编号表示 PIC24FV 与 PIC24F 器件引脚功能不同。
注 1：当 I2CSEL 配置位置 1 时，ASDA1 和 ASCL1 分别为 SDA1 和 SCL1 的备用复用引脚。
2：PIC24F32KA304 器件引脚的最大电压为 3.6V，不能承受 5V 电压。

PIC24FV32KA304 系列

引脚图

28 引脚 QFN^(1,2,3)



引脚	引脚功能	
	PIC24FVXXKA302	PIC24FXXKA302
1	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/CN4/RB0	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/CN4/RB0
2	PGEC1/AN3/C1INC/C2INA/U2RX/CTED12/CN5/RB1	PGEC1/AN3/C1INC/C2INA/U2RX/CTED12/CN5/RB1
3	AN4/C1INB/C2IND/SDA2/T5CK/T4CK/U1RX/CTED13/CN6/RB2	AN4/C1INB/C2IND/SDA2/T5CK/T4CK/U1RX/CTED13/CN6/RB2
4	AN5/C1INA/C2INC/SCL2/CN7/RB3	AN5/C1INA/C2INC/SCL2/CN7/RB3
5	VSS	VSS
6	OSCI/AN13/CLKI/CN30/RA2	OSCI/AN13/CLKI/CN30/RA2
7	OSCO/AN14/CLKO/CN29/RA3	OSCO/AN14/CLKO/CN29/RA3
8	SOSCI/AN15/U2RTS/CN1/RB4	SOSCI/AN15/U2RTS/CN1/RB4
9	SOSCO/SCLKI/U2CTS/CN0/RA4	SOSCO/SCLKI/U2CTS/CN0/RA4
10	VDD	VDD
11	PGED3/ASDA1 ⁽²⁾ /SCK2/CN27/RB5	PGED3/ASDA1 ⁽²⁾ /SCK2/CN27/RB5
12	PGEC3/ASCL1 ⁽²⁾ /SDO2/CN24/RB6	PGEC3/ASCL1 ⁽²⁾ /SDO2/CN24/RB6
13	U1TX/C2OUT/OC1/INT0/CN23/RB7	U1TX/INT0/CN23/RB7
14	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8
15	SDA1/T1CK/U1RTS/IC2/CTED4/CN21/RB9	SDA1/T1CK/U1RTS/IC2/CTED4/CN21/RB9
16	SDI2/IC1/CTED3/CN9/RA7	SDI2/IC1/CTED3/CN9/RA7
17	VCAP	C2OUT/OC1/CTED1/INT2/CN8/RA6
18	PGED2/SDI1/OC3/CTED11/CN16/RB10	PGED2/SDI1/OC3/CTED11/CN16/RB10
19	PGEC2/SCK1/OC2/CTED9/CN15/RB11	PGEC2/SCK1/OC2/CTED9/CN15/RB11
20	AN12/HLVDIN/SS2/IC3/CTED2/INT2/CN14/RB12	AN12/HLVDIN/SS2/IC3/CTED2/CN14/RB12
21	AN11/SDO1/OCFB/CTPLS/CN13/RB13	AN11/SDO1/OCFB/CTPLS/CN13/RB13
22	CVREF/AN10/C3INB/RTCC/C1OUT/OCFA/CTED5/INT1/CN12/RB14	CVREF/AN10/C3INB/RTCC/C1OUT/OCFA/CTED5/INT1/CN12/RB14
23	AN9/C3INA/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15	AN9/C3INA/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15
24	VSS/AVSS	VSS/AVSS
25	VDD/AVDD	VDD/AVDD
26	MCLR/VPP/RA5	MCLR/VPP/RA5
27	VREF+/CVREF+/AN0/C3INC/CTED1/CN2/RA0	VREF+/CVREF+/AN0/C3INC/CTED1/CN2/RA0
28	CVREF-/VREF-/AN1/CN3/RA1	CVREF-/VREF-/AN1/CN3/RA1

图注：加粗的引脚编号表示 PIC24FV 与 PIC24F 器件引脚功能不同。

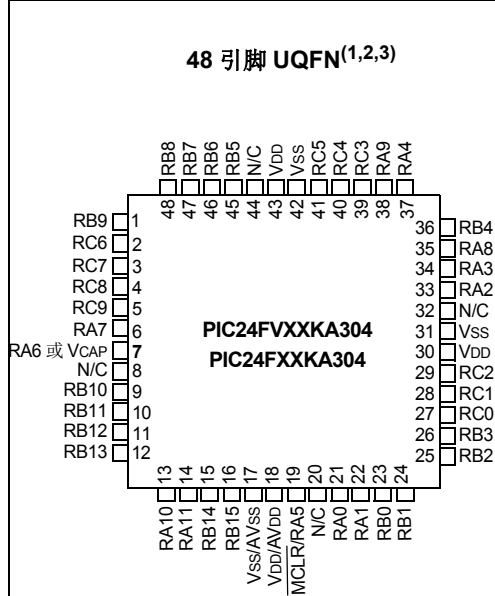
注 1：器件背面的裸露焊盘应连接到 VSS。

2：当 I2CSEL 配置位置 1 时，ASDA1 和 ASCL1 分别为 SDA1 和 SCL1 的备用复用引脚。

3：PIC24F32KA304 器件引脚的最大电压为 3.6V，不能承受 5V 电压。

PIC24FV32KA304 系列

引脚图



图注：加粗的引脚编号表示 PIC24FV 与 PIC24F 器件引脚功能不同。

- 注 1：器件背面的裸露焊盘应连接到 Vss。
- 2：当 I2CSEL 配置位置 1 时，ASDA1 和 ASCL1 分别为 SDA1 和 SCL1 的备用复用引脚。
- 3：PIC24F32KA3XX 器件引脚的最大电压为 3.6V，不能承受 5V 电压。

引脚	引脚功能	
	PIC24FVXXKA304	PIC24FXXKA304
1	SDA1/T1CK/U1RTS/CTED4/CN21/RB9	SDA1/T1CK/U1RTS/CTED4/CN21/RB9
2	U1RX/CN18/RC6	U1RX/CN18/RC6
3	U1TX/CN17/RC7	U1TX/CN17/RC7
4	OC2/CN20/RC8	OC2/CN20/RC8
5	IC2/CTED7/CN19/RC9	IC2/CTED7/CN19/RC9
6	IC1/CTED3/CN9/RA7	IC1/CTED3/CN9/RA7
7	VCAP	C2OUT/OC1/CTED1/INT2CN8/RA6
8	N/C	N/C
9	PGED2/SDI1/CTED11/CN16/RB10	PGED2/SDI1/CTED11/CN16/RB10
10	PGEC2/SCK1/CTED9/CN15/RB11	PGEC2/SCK1/CTED9/CN15/RB11
11	AN12/HLVDIN/CTED2/INT2/CN14/RB12	AN12/HLVDIN/CTED2/CN14/RB12
12	AN11/SDO1/CTPLS/CN13/RB13	AN11/SDO1/CTPLS/CN13/RB13
13	OC3/CN35/RA10	OC3/CN35/RA10
14	IC3/CTED8/CN36/RA11	IC3/CTED8/CN36/RA11
15	CVREF/AN10/C3INB/RTCC/C1OUT/OCFA/CTED5/INT1/CN12/RB14	CVREF/AN10/C3INB/RTCC/C1OUT/OCFA/CTED5/INT1/CN12/RB14
16	AN9/C3INA/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15	AN9/C3INA/T3CK/T2CK/REFO/SS1/CTED6/CN11/RB15
17	Vss/AVss	Vss/AVss
18	VDD/AVDD	VDD/AVDD
19	MCLR/RA5	MCLR/RA5
20	N/C	N/C
21	VREF+/CVREF+/AN0/C3INC/CTED1/CN2/RA0	VREF+/CVREF+/AN0/C3INC/CN2/RA0
22	CVREF-/VREF-/AN1/CN3/RA1	CVREF-/VREF-/AN1/CN3/RA1
23	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/CN4/RB0	PGED1/AN2/ULPWU/CTCMP/C1IND/C2INB/C3IND/U2TX/CN4/RB0
24	PGEC1/AN3/C1INC/C2INA/U2RX/CTED12/CN5/RB1	PGEC1/AN3/C1INC/C2INA/U2RX/CTED12/CN5/RB1
25	AN4/C1INB/C2IND/SDA2/T5CK/T4CK/CTED13/CN6/RB2	AN4/C1INB/C2IND/SDA2/T5CK/T4CK/CTED13/CN6/RB2
26	AN5/C1INA/C2INC/SCL2/CN7/RB3	AN5/C1INA/C2INC/SCL2/CN7/RB3
27	AN6/CN32/RC0	AN6/CN32/RC0
28	AN7/CN31/RC1	AN7/CN31/RC1
29	AN8/CN10/RC2	AN8/CN10/RC2
30	VDD	VDD
31	Vss	Vss
32	N/C	N/C
33	OSCI/AN13/CLKI/CN30/RA2	OSCI/AN13/CLKI/CN30/RA2
34	OSCO/AN14/CLKO/CN29/RA3	OSCO/AN14/CLKO/CN29/RA3
35	OCFB/CN33/RA8	OCFB/CN33/RA8
36	SOSCI/AN15/U2RTS/CN1/RB4	SOSCI/AN15/U2RTS/CN1/RB4
37	SOSCO/SCLKI/U2CTS/CN0/RA4	SOSCO/SCLKI/U2CTS/CN0/RA4
38	SS2/CN34/RA9	SS2/CN34/RA9
39	SDI2/CN28/RC3	SDI2/CN28/RC3
40	SDO2/CN25/RC4	SDO2/CN25/RC4
41	SCK2/CN26/RC5	SCK2/CN26/RC5
42	Vss	Vss
43	VDD	VDD
44	N/C	N/C
45	PGED3/ASDA1 ⁽²⁾ /CN27/RB5	PGED3/ASDA1 ⁽²⁾ /CN27/RB5
46	PGEC3/ASCL1 ⁽²⁾ /CN24/RB6	PGEC3/ASCL1 ⁽²⁾ /CN24/RB6
47	C2OUT/OC1/INT0/CN23/RB7	INT0/CN23/RB7
48	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8	SCL1/U1CTS/C3OUT/CTED10/CN22/RB8

PIC24FV32KA304 系列

目录

1.0	器件概述.....	11
2.0	16 位单片机入门指南	23
3.0	CPU.....	29
4.0	存储器构成	35
5.0	闪存程序存储器	57
6.0	数据 EEPROM 存储器.....	63
7.0	复位.....	69
8.0	中断控制器	75
9.0	振荡器配置	115
10.0	节能特性.....	125
11.0	I/O 端口	135
12.0	Timer1	139
13.0	Timer2/3 和 Timer4/5	141
14.0	带专用定时器的输入捕捉.....	147
15.0	带专用定时器的输出比较.....	151
16.0	串行外设接口 (SPI).....	161
17.0	I ² C™	169
18.0	通用异步收发器 (UART)	177
19.0	实时时钟和日历 (RTCC)	185
20.0	32 位可编程循环冗余校验 (CRC) 发生器.....	199
21.0	高 / 低电压检测 (HLVD)	205
22.0	带阈值检测的 12 位 A/D 转换器.....	207
23.0	比较器模块	225
24.0	比较器参考电压	229
25.0	充电时间测量单元 (CTMU)	231
26.0	特殊功能.....	239
27.0	开发支持.....	251
28.0	指令集汇总	255
29.0	电气特性.....	263
30.0	直流和交流特性图表.....	295
31.0	封装信息	325
附录 A:	版本历史	351
353Microchip	网站	353
变更通知	客户服务	359
客户支持		359
读者反馈表		360
产品标识体系		361

致 客 户

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

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

最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

<http://www.microchip.com>

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

勘误表

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

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

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

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

客户通知系统

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

PIC24FV32KA304 系列

注:

1.0 器件概述

此文档包含针对以下器件的具体信息：

- PIC24FV16KA301, PIC24F16KA301
- PIC24FV16KA302, PIC24F16KA302
- PIC24FV16KA304, PIC24F16KA304
- PIC24FV32KA301, PIC24F32KA301
- PIC24FV32KA302, PIC24F32KA302
- PIC24FV32KA304, PIC24F32KA304

PIC24FV32KA304 系列引入了 Microchip 新的超低功耗器件。该 16 位单片机系列器件具有丰富的外设功能集和增强的计算性能，它还高性能应用提供了新的移植选项，这些应用可能需要超过 8 位的平台，但无需数字信号处理器的数字处理功能。

1.1 内核特性

1.1.1 16 位架构

所有 PIC24F 器件的内核都采用了 16 位改进型哈佛架构，该架构最初是在 Microchip 的 dsPIC® 数字信号控制器中引入的。PIC24F CPU 内核提供了大量增强功能，例如：

- 16 位数据路径和 24 位地址路径，可在数据空间和程序空间之间传递数据
- 线性寻址能力最高可达 12 MB（程序空间）和 64 KB（数据空间）
- 利用内建软件堆栈支持 16 元工作寄存器阵列
- 支持整数算术运算的 17 x 17 位硬件乘法器
- 支持 32 位 /16 位除法运算的硬件
- 支持多种寻址模式并针对高级语言（如 C 语言）优化了指令集
- 工作性能最高可达 16 MIPS

1.1.2 节能技术

PIC24FV32KA304 系列的所有器件具有许多能在工作时显著降低功耗的功能。主要包括以下几项：

- **动态时钟切换：**在器件工作过程中，器件时钟可在软件控制下切换为 Timer1 时钟源或内部低功耗 RC 振荡器，允许用户把节能理念融入到软件设计中去。
- **打盹模式操作：**当那些对时间要求很高的应用（如串行通信）要求外设不间断地工作时，该模式可以适当降低 CPU 时钟速度，从而可在不丢失任一时钟脉冲的前提下进一步节约功耗。
- **基于指令的节能模式：**共有三种基于指令的节能模式：
 - 空闲模式：内核关闭，而外设仍然工作。
 - 休眠模式：内核和需要使用系统时钟的外设关闭，而使用自身时钟或由其他设备提供时钟的外设仍然工作。
 - 深度休眠模式：内核、外设（除 RTCC 和 DSWDT 外）、闪存和 SRAM 都关闭。

1.1.3 振荡器选项和特性

PIC24FV32KA304 系列器件提供五个不同的振荡器选项，使用户在开发应用软件时有更多选择。这些选项包括：

- 使用晶振或陶瓷谐振器的 2 种晶振模式。
- 提供 2 分频时钟输出选项的 2 种外部时钟模式。
- 两种内部快速振荡器（FRC）模式：一种输出标称值为 8 MHz，另一种输出标称值为 500 kHz。这些输出可在软件控制下进行分频以提供低至 31 kHz 或 2 kHz 的时钟速度。
- 锁相环（Phase Lock Loop, PLL）倍频器，可在外部振荡器模式和 8 MHz FRC 振荡器模式下使用，从而可使时钟速度最高达到 32 MHz。
- 具有固定的 31 kHz 输出的独立内部 RC 振荡器（LPRC），可为对时间要求很高的应用提供低功耗选项。

PIC24FV32KA304 系列

内部振荡器模块还为故障保护时钟监视器（FSCM）提供了一个稳定的参考源。该选项持续监视主时钟源，将之与内部振荡器提供的参考信号作比较，使控制器能将时钟源切换到内部振荡器，从而继续保持低速工作或安全地关闭应用。

1.1.4 简便移植

无论存储器容量如何，所有器件均共享同一组外设，从而可在应用扩展或改进时，方便地进行移植。

整个系列使用相同的引脚配置方案也有助于从一个器件向下一代更大型的器件移植。在具有相同引脚数的器件间移植，甚至从 20/28 引脚器件向 44/48 引脚器件的移植都是可以的。

PIC24F 系列器件的引脚同 dsPIC33 系列器件的引脚是兼容的，并与 PIC18 和 dsPIC30 器件的引脚配置方案部分兼容。这样可以将相对简单的应用顺利移植为强大而复杂的应用。

1.2 其他特性

- **通信：**PIC24FV32KA304 系列器件集成了许多串行通信外设以处理各种应用需求。提供了一个同时支持主从工作模式的 I²C™ 模块。它还包含具有内置 IrDA® 编码器 / 解码器的 UART 以及 SPI 模块。
- **实时时钟 / 日历：**此模块通过硬件实现带有闹钟功能的全功能时钟和日历，从而释放了定时器资源和程序存储空间供核心应用使用。
- **12 位 A/D 转换器：**该模块具有可编程的采集时间，从而不必在选择通道和启动转换之间等待一个采样周期，并加快了采样速度。16 级深结果缓冲区不仅可在休眠模式下减少功耗，也可在工作状态下提高吞吐量。
- **充电时间测量单元（CTMU）接口：**PIC24FV32KA304 系列器件包含新的 CTMU 接口模块，可用于电容触摸传感、接近传感，还可用于精确时间测量和脉冲生成。

1.3 系列中各产品的具体信息

PIC24FV32KA304 系列器件以 20 引脚、28 引脚、44 引脚和 48 引脚封装形式提供。图 1-1 给出了所有器件的一般框图。

这些器件在以下四个方面存在差异：

1. 闪存程序存储器（PIC24FV16KA 器件为 16 KB，PIC24FV32KA 器件为 32 KB）。
2. 可用的 I/O 引脚和端口（20 引脚器件有两个端口 18 引脚，28 引脚器件有两个端口 22 引脚，而 44/48 引脚器件有三个端口 38 个引脚）。
3. 备用 SCLx 和 SDAx 引脚仅在 28 引脚、44 引脚和 48 引脚器件上可用，在 20 引脚器件上不可用。
4. PIC24FV32KA301 系列器件提供有标准器件和高压器件。元器件编码中的“FV”指定的高压器件（如 PIC24FV32KA304）可工作在 2.0V 到 5.5V 的 VDD 范围内，还有一个为内核供电的片内稳压器。外设工作电压为 VDD。“F”指定的标准器件（如 PIC24F32KA304）可工作在 1.8V 至 3.6V 的较低 VDD 范围内。这些元器件没有内部稳压器，内核和外设直接以 VDD 运行。

该系列器件的所有其他特性都是相同的；表 1-1 对此进行了总结。

表 1-3 给出了 PIC24FV32KA304 系列器件上可用的引脚功能列表（按功能排序）。

注：	表 1-1 给出了各个外设功能所使用的引脚的位置，而没有显示同一引脚上多种功能的复用方式。数据手册中的第 3 页、第 4 页、第 5 页、第 6 页和第 7 页的引脚图提供了该信息。复用的功能按功能的优先级排序，最前面的是优先级最高的外设功能。
----	--

PIC24FV32KA304 系列

表 1-1: PIC24FV32KA304 系列器件的特性

特性	PIC24FV16KA301	PIC24FV32KA301	PIC24FV16KA302	PIC24FV32KA302	PIC24FV16KA304	PIC24FV32KA304
工作频率	DC——32 MHz					
程序存储器（字节数）	16K	32K	16K	32K	16K	32K
程序存储器（指令数）	5632	11264	5632	11264	5632	11264
数据存储器（字节数）	2048					
数据 EEPROM 存储器（字节数）	512					
中断源（软向量 / NMI 陷阱）	30（26/4）					
I/O 端口	PORTA<5:0> PORTB<15:12、9:7、4、2:0>		PORTA<7.5:0> PORTB<15:0>		PORTA<11:7,5:0> PORTB<15:0> PORTC<9:0>	
I/O 引脚总数	17		23		38	
定时器：总数（16 位）	5					
32 位（由一对 16 位定时器组成）	2					
输入捕捉通道	3					
输出比较 /PWM 通道	3					
输入电平变化通知中断	16		22		37	
串行通信：UART SPI（3 线 /4 线）	2					
I ² C™	2					
12 位模数转换器模块（输入通道）	12		13		16	
模拟比较器	3					
复位（和延时）	POR、BOR、RESET 指令、 <u>MCLR</u> 、WDT；非法操作码、 REPEAT 指令、硬件陷阱和配置字失配 （PWRT、OST 和 PLL 锁定）					
指令集	76 条基本指令和多种寻址模式					
封装	20 引脚 PDIP/SSOP/SOIC		28 引脚 SPDIP/SSOP/SOIC/QFN		44 引脚 QFN/TQFP 48 引脚 UQFN	

PIC24FV32KA304 系列

表 1-2: PIC24F32KA304 系列器件的特性

特性	PIC24F16KA301	PIC24F32KA301	PIC24F16KA302	PIC24F32KA302	PIC16F16KA304	PIC24F32KA304
工作频率	DC——32 MHz					
程序存储器（字节数）	16K	32K	16K	32K	16K	32K
程序存储器（指令数）	5632	11264	5632	11264	5632	11264
数据存储器（字节数）	2048					
数据 EEPROM 存储器（字节数）	512					
中断源（软向量 / NMI 陷阱）	30（26/4）					
I/O 端口	PORTA<6:0> PORTB<15:12、9:7、4、2:0>		PORTA<7:0> PORTB<15:0>		PORTA<11:0> PORTB<15:0> PORTC<9:0>	
I/O 引脚总数	18		24		39	
定时器：总数（16 位）	5					
32 位（由一对 16 位定时器组成）	2					
输入捕捉通道	3					
输出比较 /PWM 通道	3					
输入电平变化通知中断	17		23		38	
串行通信：UART SPI（3 线 /4 线）	2					
I ² C™	2					
12 位模数转换器模块（输入通道）	12		13		16	
模拟比较器	3					
复位（和延时）	POR、BOR、RESET 指令、 <u>MCLR</u> 、WDT；非法操作码、 REPEAT 指令、硬件陷阱和配置字失配 （PWRT、OST 和 PLL 锁定）					
指令集	76 条基本指令和多种寻址模式					
封装	20 引脚 PDIP/SSOP/SOIC		28 引脚 SPDIP/SSOP/SOIC/QFN		44 引脚 QFN/TQFP 48 引脚 UQFN	

图 1-1: PIC24FV32KA304 系列器件的一般框图

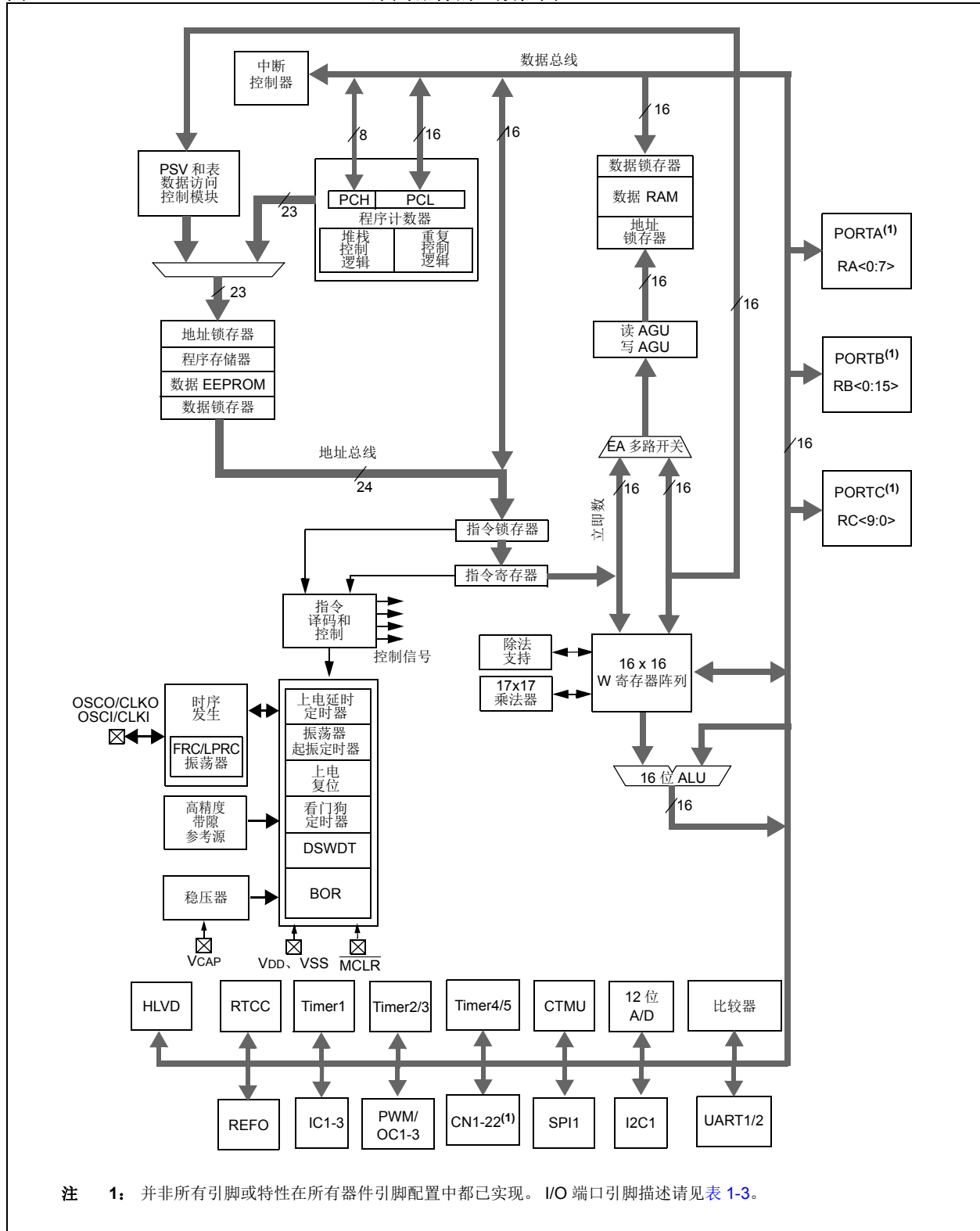


表 1-3: PIC24FV32KA304 系列器件引脚配置说明

功能	F					FV					I/O	缓冲区	说明
	引脚数					引脚数							
	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN			
AN0	2	2	27	19	21	2	2	27	19	21	I	ANA	A/D 模拟输入
AN1	3	3	28	20	22	3	3	28	20	22	I	ANA	
AN2	4	4	1	21	23	4	4	1	21	23	I	ANA	
AN3	5	5	2	22	24	5	5	2	22	24	I	ANA	
AN4	6	6	3	23	25	6	6	3	23	25	I	ANA	
AN5	—	7	4	24	26	—	7	4	24	26	I	ANA	
AN6	—	—	—	25	27	—	—	—	25	27	I	ANA	
AN7	—	—	—	26	28	—	—	—	26	28	I	ANA	
AN8	—	—	—	27	29	—	—	—	27	29	I	ANA	
AN9	18	26	23	15	16	18	26	23	15	16	I	ANA	
AN10	17	25	22	14	15	17	25	22	14	15	I	ANA	
AN11	16	24	21	11	12	16	24	21	11	12	I	ANA	
AN12	15	23	20	10	11	15	23	20	10	11	I	ANA	
AN13	7	9	6	30	33	7	9	6	30	33	I	ANA	
AN14	8	10	7	31	34	8	10	7	31	34	I	ANA	
AN15	9	11	8	33	36	9	11	8	33	36	I	ANA	
ASCL1	—	15	12	42	46	—	15	12	42	46	I/O	I ² C™	备用 I2C1 时钟输入 / 输出
ASDA1	—	14	11	41	45	—	14	11	41	45	I/O	I ² C	备用 I2C1 数据输入 / 输出
AVDD	20	28	25	17	18	20	28	25	17	18	I	ANA	A/D 电源引脚
AVss	19	27	24	16	17	19	27	24	16	17	I	ANA	
C1INA	8	7	4	24	26	8	7	4	24	26	I	ANA	比较器 1 输入 A (+)
C1INB	7	6	3	23	25	7	6	3	23	25	I	ANA	比较器 1 输入 B (-)
C1INC	5	5	2	22	24	5	5	2	22	24	I	ANA	比较器 1 输入 C (+)
C1IND	4	4	1	21	23	4	4	1	21	23	I	ANA	比较器 1 输入 D (-)
C1OUT	17	25	22	14	15	17	25	22	14	15	O	—	比较器 1 输出
C2INA	5	5	2	22	24	5	5	2	22	24	I	ANA	比较器 2 输入 A(+)
C2INB	4	4	1	21	23	4	4	1	21	23	I	ANA	比较器 2 输入 B (-)
C2INC	8	7	4	24	26	8	7	4	24	26	I	ANA	比较器 2 输入 C (+)
C2IND	7	6	3	23	25	7	6	3	23	25	I	ANA	比较器 2 输入 D (-)
C2OUT	14	20	17	7	7	11	16	13	43	47	O	—	比较器 2 输出

表 1-3: PIC24FV32KA304 系列器件引脚配置说明 (续)

功能	F					FV					I/O	缓冲区	说明
	引脚数					引脚数							
	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN			
C3INA	18	26	23	15	16	18	26	23	15	16	I	ANA	比较器 3 输入 A(+)
C3INB	17	25	22	14	15	17	25	22	14	15	I	ANA	比较器 3 输入 B (-)
C3INC	2	2	27	19	21	2	2	27	19	21	I	ANA	比较器 3 输入 C (+)
C3IND	4	4	1	21	23	4	4	1	21	23	I	ANA	比较器 3 输入 D (-)
C3OUT	12	17	14	44	48	12	17	14	44	48	O	—	比较器 3 输出
CLK I	7	9	6	30	33	7	9	6	30	33	I	ANA	主时钟输入
CLKO	8	10	7	31	34	8	10	7	31	34	O	—	系统时钟输出
CN0	10	12	9	34	37	10	12	9	34	37	I	ST	电平变化中断输入
CN1	9	11	8	33	36	9	11	8	33	36	I	ST	
CN2	2	2	27	19	21	2	2	27	19	21	I	ST	
CN3	3	3	28	20	22	3	3	28	20	22	I	ST	
CN4	4	4	1	21	23	4	4	1	21	23	I	ST	
CN5	5	5	2	22	24	5	5	2	22	24	I	ST	
CN6	6	6	3	23	25	6	6	3	23	25	I	ST	
CN7	—	7	4	24	26	—	7	4	24	26	I	ST	
CN8	14	20	17	7	7	—	—	—	—	—	I	ST	
CN9	—	19	16	6	6	—	19	16	6	6	I	ST	
CN10	—	—	—	27	29	—	—	—	27	29	I	ST	
CN11	18	26	23	15	16	18	26	23	15	16	I	ST	
CN12	17	25	22	14	15	17	25	22	14	15	I	ST	
CN13	16	24	21	11	12	16	24	21	11	12	I	ST	
CN14	15	23	20	10	11	15	23	20	10	11	I	ST	
CN15	—	22	19	9	10	—	22	19	9	10	I	ST	
CN16	—	21	18	8	9	—	21	18	8	9	I	ST	
CN17	—	—	—	3	3	—	—	—	3	3	I	ST	
CN18	—	—	—	2	2	—	—	—	2	2	I	ST	
CN19	—	—	—	5	5	—	—	—	5	5	I	ST	
CN20	—	—	—	4	4	—	—	—	4	4	I	ST	
CN21	13	18	15	1	1	13	18	15	1	1	I	ST	
CN22	12	17	14	44	48	12	17	14	44	48	I	ST	

表 1-3: PIC24FV32KA304 系列器件引脚配置说明 (续)

功能	F					FV					I/O	缓冲区	说明
	引脚数					引脚数							
	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN			
CN23	11	16	13	43	47	11	16	13	43	47	I	ST	电平变化中断输入
CN24	—	15	12	42	46	—	15	12	42	46	I	ST	
CN25	—	—	—	37	40	—	—	—	37	40	I	ST	
CN26	—	—	—	38	41	—	—	—	38	41	I	ST	
CN27	—	14	11	41	45	—	14	11	41	45	I	ST	
CN28	—	—	—	36	39	—	—	—	36	39	I	ST	
CN29	8	10	7	31	34	8	10	7	31	34	I	ST	
CN30	7	9	6	30	33	7	9	6	30	33	I	ST	
CN31	—	—	—	26	28	—	—	—	26	28	I	ST	
CN32	—	—	—	25	27	—	—	—	25	27	I	ST	
CN33	—	—	—	32	35	—	—	—	32	35	I	ST	
CN34	—	—	—	35	38	—	—	—	35	38	I	ST	
CN35	—	—	—	12	13	—	—	—	12	13	I	ST	
CN36	—	—	—	13	14	—	—	—	13	14	I	ST	
CVREF	17	25	22	14	15	17	25	22	14	15	I	ANA	比较器参考电压输出
CVREF+	2	2	27	19	21	2	2	27	19	21	I	ANA	比较器正输入参考电压
CVREF-	3	3	28	20	22	3	3	28	20	22	I	ANA	比较器负输入参考电压
CTCMP	4	4	1	21	23	4	4	1	21	23	I	ANA	CTMU 比较器输入
CTED1	14	20	17	7	7	11	2	27	19	21	I	ST	CTMU 触发边沿输入
CTED2	12	17	14	44	48	12	17	14	44	48	I	ST	
CTED3	—	21	18	8	9	—	21	18	8	9	I	ST	
CTED4	5	5	2	22	24	5	5	2	22	24	I	ST	
CTED5	6	6	3	23	25	6	6	3	23	25	I	ST	
CTED6	15	23	20	10	11	15	23	20	10	11	I	ST	
CTED7	—	19	16	6	6	—	19	16	6	6	I	ST	
CTED8	13	18	15	1	1	13	18	15	1	1	I	ST	
CTED9	17	25	22	14	15	17	25	22	14	15	I	ST	
CTED10	18	26	23	15	16	18	26	23	15	16	I	ST	
CTED11	—	—	—	5	5	—	—	—	5	5	I	ST	
CTED12	—	—	—	13	14	—	—	—	13	14	I	ST	
CTED13	—	22	19	9	10	—	22	19	9	10	I	ST	

表 1-3: PIC24FV32KA304 系列器件引脚配置说明 (续)

功能	F					FV					I/O	缓冲区	说明
	引脚数					引脚数							
	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN			
CTPLS	16	24	21	11	12	16	24	21	11	12	O	—	CTMU 脉冲输出
HLVDIN	15	23	20	10	11	15	23	20	10	11	I	ST	高 / 低电压检测输入
IC1	14	19	16	6	6	11	19	16	6	6	I	ST	输入捕捉 1 输入
IC2	13	18	15	5	5	13	18	15	5	5	I	ST	输入捕捉 2 输入
IC3	15	23	20	13	14	15	23	20	13	14	I	ST	输入捕捉 3 输入
INT0	11	16	13	43	47	11	16	13	43	47	I	ST	中断 0 输入
INT1	17	25	22	14	15	17	25	22	14	15	I	ST	中断 1 输入
INT2	14	20	17	7	7	15	23	20	10	11	I	ST	中断 2 输入
MCLR	1	1	26	18	19	1	1	26	18	19	I	ST	主复位（器件复位）输入（低电平有效）
OC1	14	20	17	7	7	11	16	13	43	47	O	—	输出比较 /PWM1 输出
OC2	4	22	19	4	4	4	22	19	4	4	O	—	输出比较 /PWM2 输出
OC3	5	21	18	12	13	5	21	18	12	13	O	—	输出比较 /PWM3 输出
OCFA	17	25	22	14	15	17	25	22	14	15	O	—	输出比较故障 A
OFCB	16	24	21	32	35	16	24	21	32	35	O	—	输出比较故障 B
OSCI	7	9	6	30	33	7	9	6	30	33	I	ANA	主振荡器输入
OSCO	8	10	7	31	34	8	10	7	31	34	O	ANA	主振荡器输出
PGEC1	5	5	2	22	24	5	5	2	22	24	I/O	ST	ICSP™ 时钟 1
PCED1	4	4	1	21	23	4	4	1	21	23	I/O	ST	ICSP 数据 1
PGEC2	2	22	19	19	10	2	22	19	19	10	I/O	ST	ICSP 时钟 2
PGED2	3	21	18	8	9	3	21	18	8	9	I/O	ST	ICSP 数据 2
PGEC3	10	15	12	42	46	10	15	12	42	46	I/O	ST	ICSP 时钟 3
PGED3	9	14	11	41	45	9	14	11	41	45	I/O	ST	ICSP 数据 3

表 1-3: PIC24FV32KA304 系列器件引脚配置说明 (续)

功能	F					FV					I/O	缓冲区	说明
	引脚数					引脚数							
	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN			
RA0	2	2	27	19	21	2	2	27	19	21	I/O	ST	PORTA 引脚
RA1	3	3	28	20	22	3	3	28	20	22	I/O	ST	
RA2	7	9	6	30	33	7	9	6	30	33	I/O	ST	
RA3	8	10	7	31	34	8	10	7	31	34	I/O	ST	
RA4	10	12	9	34	37	10	12	9	34	37	I/O	ST	
RA5	1	1	26	18	19	1	1	26	18	19	I/O	ST	
RA6	14	20	17	7	7	—	—	—	—	—	I/O	ST	
RA7	—	19	16	6	6	—	19	16	6	6	I/O	ST	
RA8	—	—	—	32	35	—	—	—	32	35	I/O	ST	
RA9	—	—	—	35	38	—	—	—	35	38	I/O	ST	
RA10	—	—	—	12	13	—	—	—	12	13	I/O	ST	
RA11	—	—	—	13	14	—	—	—	13	14	I/O	ST	
RB0	4	4	1	21	23	4	4	1	21	23	I/O	ST	PORTB 引脚
RB1	5	5	2	22	24	5	5	2	22	24	I/O	ST	
RB2	6	6	3	23	25	6	6	3	23	25	I/O	ST	
RB3	—	7	4	24	26	—	7	4	24	26	I/O	ST	
RB4	9	11	8	33	36	9	11	8	33	36	I/O	ST	
RB5	—	14	11	41	45	—	14	11	41	45	I/O	ST	
RB6	—	15	12	42	46	—	15	12	42	46	I/O	ST	
RB7	11	16	13	43	47	11	16	13	43	47	I/O	ST	
RB8	12	17	14	44	48	12	17	14	44	48	I/O	ST	
RB9	13	18	15	1	1	13	18	15	1	1	I/O	ST	
RB10	—	21	18	8	9	—	21	18	8	9	I/O	ST	
RB11	—	22	19	9	10	—	22	19	9	10	I/O	ST	
RB12	15	23	20	10	11	15	23	20	10	11	I/O	ST	
RB13	16	24	21	11	12	16	24	21	11	12	I/O	ST	
RB14	17	25	22	14	15	17	25	22	14	15	I/O	ST	
RB15	18	26	23	15	16	18	26	23	15	16	I/O	ST	

表 1-3: PIC24FV32KA304 系列器件引脚配置说明 (续)

功能	F					FV					I/O	缓冲区	说明
	引脚数					引脚数							
	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN			
RC0	—	—	—	25	27	—	—	—	25	27	I/O	ST	PORTC 引脚
RC1	—	—	—	26	28	—	—	—	26	28	I/O	ST	
RC2	—	—	—	27	29	—	—	—	27	29	I/O	ST	
RC3	—	—	—	36	39	—	—	—	36	39	I/O	ST	
RC4	—	—	—	37	40	—	—	—	37	40	I/O	ST	
RC5	—	—	—	38	41	—	—	—	38	41	I/O	ST	
RC6	—	—	—	2	2	—	—	—	2	2	I/O	ST	
RC7	—	—	—	3	3	—	—	—	3	3	I/O	ST	
RC8	—	—	—	4	4	—	—	—	4	4	I/O	ST	
RC9	—	—	—	5	5	—	—	—	5	5	I/O	ST	
REFO	18	26	23	15	16	18	26	23	15	16	O	—	参考时钟输出
RTCC	17	25	22	14	15	17	25	22	14	15	O	—	实时时钟 / 日历输出
SCK1	15	22	19	9	10	15	22	19	9	10	I/O	ST	SPI1 串行输入 / 输出时钟
SCK2	2	14	11	38	41	2	14	11	38	41	I/O	ST	SPI2 串行输入 / 输出时钟
SCL1	12	17	14	44	48	12	17	14	44	48	I/O	I ² C	I2C1 时钟输入 / 输出
SCL2	18	7	4	24	26	18	7	4	24	26	I/O	I ² C	I2C2 时钟输入 / 输出
SCLKI	10	12	9	34	37	10	12	9	34	37	I	ST	数字辅助时钟输入
SDA1	13	18	15	1	1	13	18	15	1	1	I/O	I ² C	I2C1 数据输入 / 输出
SDA2	6	6	3	23	25	6	6	3	23	25	I/O	I ² C	I2C2 数据输入 / 输出
SDI1	17	21	18	8	9	17	21	18	8	9	I	ST	SPI1 串行数据输入
SDI2	4	19	16	36	39	4	19	16	36	39	I	ST	SPI2 串行数据输入
SDO1	16	24	21	11	12	16	24	21	11	12	O	—	SPI1 串行数据输出
SDO2	3	15	12	37	40	3	15	12	37	40	O	—	SPI2 串行数据输出
SOSCI	9	11	8	33	36	9	11	8	33	36	I	ANA	辅助振荡器输入
SOSCO	10	12	9	34	37	10	12	9	34	37	O	ANA	辅助振荡器输出
SS1	18	26	23	15	16	18	26	23	15	16	O	—	SPI1 从选择
SS2	15	23	20	35	38	15	23	20	35	38	O	—	SPI2 从选择

表 1-3: PIC24FV32KA304 系列器件引脚配置说明 (续)

功能	F					FV					I/O	缓冲区	说明
	引脚数					引脚数							
	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN	20 引脚 PDIP/SSOP/ SOIC	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN	44 引脚 QFN/TQFP	48 引脚 UQFN			
T1CK	13	18	15	1	1	13	18	15	1	1	I	ST	Timer1 时钟
T2CK	18	26	23	15	16	18	26	23	15	16	I	ST	Timer2 时钟
T3CK	18	26	23	15	16	18	26	23	15	16	I	ST	Timer3 时钟
T4CK	6	6	3	23	25	6	6	3	23	25	I	ST	Timer4 时钟
T5CK	6	6	3	23	25	6	6	3	23	25	I	ST	Timer5 时钟
U1CTS	12	17	14	44	48	12	17	14	44	48	I	ST	UART1 允许发送输入
U1RTS	13	18	15	1	1	13	18	15	1	1	O	—	UART1 请求发送输出
U1RX	6	6	3	2	2	6	6	3	2	2	I	ST	UART1 接收
U1TX	11	16	13	3	3	11	16	13	3	3	O	—	UART1 发送
U2CTS	10	12	9	34	37	10	12	9	34	37	I	ST	UART2 允许发送输入
U2RTS	9	11	8	33	36	9	11	8	33	36	O	—	UART2 请求发送输出
U2RX	5	5	2	22	24	5	5	2	22	24	I	ST	UART2 接收
U2TX	4	4	1	21	23	4	4	1	21	23	O	—	UART2 发送
ULPWU	4	4	1	21	23	4	4	1	21	23	I	ANA	超低功耗唤醒输入
VCAP	—	—	—	—	—	14	20	17	7	7	P	—	内核功耗
VDD	20	28,13	25,10	17,28,40	18,30,43	20	28,13	25,10	17,28,40	18,30,43	P	—	器件数字电源电压
VREF+	2	2	27	19	21	2	2	27	19	21	I	ANA	A/D 参考电压输入 (+)
VREF-	3	3	28	20	22	3	3	28	20	22	I	ANA	A/D 参考电压输入 (-)
VSS	19	27,8	24,5	16,29,39	17,31,42	19	27,8	24,5	16,29,39	17,31,42	P	—	器件数字地回路

2.0 16 位单片机入门指南

2.1 基本连接要求

在着手使用 16 位单片机 PIC24FV32KA304 系列进行开发之前，要求注意器件引脚的最少连接。

以下引脚必须始终处于连接状态：

- 所有 VDD 和 VSS 引脚（见第 2.2 节“电源引脚”）
- 所有 AVDD 和 AVSS 引脚，而与是否使用模拟器件功能无关（见第 2.2 节“电源引脚”）
- MCLR 引脚（见第 2.3 节“主复位（MCLR）引脚”）
- VCAP 引脚（见第 2.4 节“稳压器引脚（VCAP）”）

如果在最终应用中使用了以下引脚，那么也必须连接这些引脚：

- 用于在线串行编程（ICSP™）和调试的 PGCx/PGEDx 引脚（见第 2.5 节“ICSP 引脚”）
- 使用外部振荡器源时用到的 OSCI 和 OSCO 引脚（见第 2.6 节“外部振荡器引脚”）

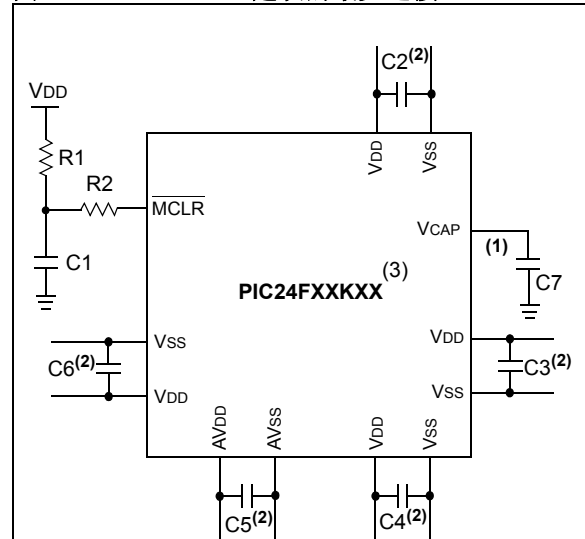
此外，可能需要以下引脚：

- 实现模拟模块的外部参考电压时使用的 VREF+/VREF- 引脚

注： 无论是否使用任何模拟模块，AVDD 和 AVSS 必须始终连接。

图 2-1 中给出了必需的最少连接。

图 2-1: 建议的最少连接



要点（所有值都是建议值）：

C1 至 C6: 0.1 μ F、20V 的陶瓷电容

C7: 10 μ F、16V 的钽电容或陶瓷电容

R1: 10 k Ω

R2: 100 Ω 至 470 Ω

- 注**
- 1: 请参见第 2.4 节“稳压器引脚（VCAP）”了解 VCAP 引脚连接说明。
 - 2: 所给出的示例针对的是具有 5 对 VDD/VSS 和 AVDD/AVSS 引脚的 PIC24F 器件。其他器件所具有的引脚对数可能有所不同；应相应调整去耦电容数。
 - 3: 一些 PIC24F K 器件没有稳压器。

PIC24FV32KA304 系列

2.2 电源引脚

2.2.1 去耦电容

需要在每对电源引脚（如 VDD 与 VSS 以及 AVDD 与 AVSS）上使用去耦电容。

使用去耦电容时，需考虑以下条件：

- **电容的值和类型：**建议使用 $0.1\ \mu\text{F}$ （ $100\ \text{nF}$ ）、 $10\text{-}20\text{V}$ 的电容。应使用谐振频率在 $200\ \text{MHz}$ 及更高范围内的低 ESR 电容。建议使用陶瓷电容。
- **印刷电路板上的放置：**去耦电容应尽可能靠近相应的引脚放置。建议将电容放在电路板上与器件相同的一侧。如果空间有限，可使用过孔将电容放到 PCB 的另一层上；但是，需要确保从引脚到电容的走线长度不超过 0.25 英寸（ $6\ \text{mm}$ ）。
- **处理高频噪声：**如果电路板产生高达几十兆赫兹的高频噪声，请在上述去耦电容旁并联一个陶瓷类型的辅助电容。辅助电容值的范围为 $0.001\ \mu\text{F}$ 至 $0.01\ \mu\text{F}$ 。请将这个辅助电容靠近每个主去耦电容放置。在高速电路设计中，请考虑在尽可能靠近电源和接地引脚的地方布置十对电容（例如，一个 $0.1\ \mu\text{F}$ 的电容与一个 $0.001\ \mu\text{F}$ 的电容并联构成一对）。
- **最大程度提高性能：**在从电源电路开始布置电路板时，请首先连接电源并将走线返回到去耦电容，然后再连接到器件引脚。这可以确保去耦电容在电源链中处于第一位。保持电容和电源引脚之间走线长度尽可能短也同样重要，因为这可以减少 PCB 的走线感抗。

2.2.2 槽路电容

在电源走线长度大于 6 英寸的电路板上，建议对包括单片机在内的集成电路使用槽路电容，以提供本地电源。槽路电容的值应根据以下因素确定：连接电源和器件的走线的电阻值以及应用中器件消耗的最大电流。也就是说，选择槽路电容使之满足器件的可接受的电压骤降要求。典型值范围为 $4.7\ \mu\text{F}$ 至 $47\ \mu\text{F}$ 。

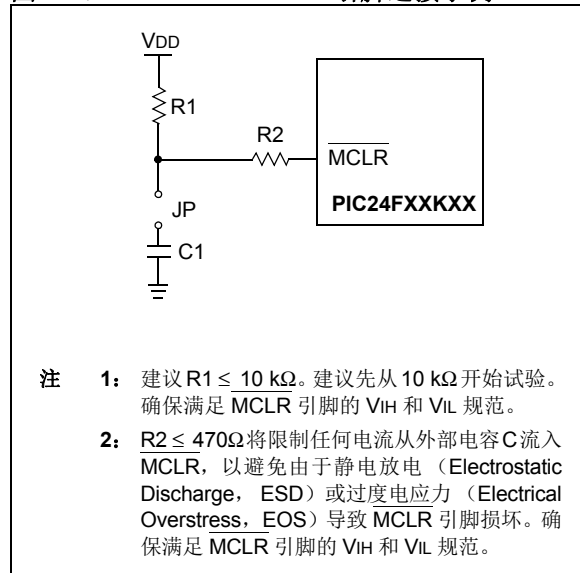
2.3 主复位（MCLR）引脚

MCLR 引脚提供两个特定的器件功能：器件复位及器件编程与调试。如果在最终应用中不需要编程和调试，则只需直接将该引脚连接到 VDD。为帮助增加应用的电阻以避免因电压骤降而意外复位，可能需要添加其他元件。图 2-1 给出了典型的配置。可根据应用的要求实现其他电路设计。

在编程和调试期间，必须考虑可以添加到引脚的电阻和电容。器件编程器和调试器驱动 MCLR 引脚。因此，不允许对特殊电压电平（ V_{IH} 和 V_{IL} ）以及快速信号变化造成不良影响。这就需要根据应用和 PCB 的要求调整 R1 和 C1 的具体值。例如，建议在编程和调试操作期间，通过使用跳线将电容 C1 和 MCLR 引脚隔离（图 2-2）。正常的运行时操作不需要这样的跳线。

与 MCLR 引脚相关的任何元件应放置在距离该引脚 0.25 英寸（ $6\ \text{mm}$ ）的范围内。

图 2-2: MCLR 引脚连接示例



2.4 稳压器引脚 (VCAP)

注： 本节仅适用于具有片上稳压器的PIC24F K器件。

一些 PIC24F K 器件具有内部稳压器。这些器件的稳压器输出连接到 VCAP 引脚。在具有稳压器的 PIC24F K 器件上，VCAP 引脚上需要连接一个低 ESR ($< 5\Omega$) 的电容器来稳定稳压器输出。VCAP 引脚不能连接到 VDD，并且必须使用一个 $10\ \mu\text{F}$ 的电容器接地。电容类型可以是陶瓷电容或者钽电容。表 2-1 例举了一些可采用的电容。也可以使用具有同等规格的其他电容。

设计人员可以使用图 2-3 来计算备选电容的 ESR。

此电容应靠近 VCAP 放置。建议走线长度不超过 0.25 英寸 (6 mm)。其他信息请参见第 29.0 节“电气特性”。

有关 VDD 和 VDDCORE 的信息，请参见第 29.0 节“电气特性”。

图 2-3: 建议 VCAP 的频率与 ESR 性能关系

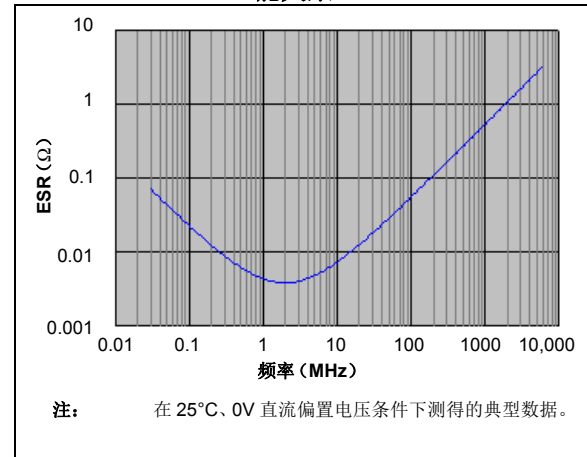


表 2-1: 适合的同规格电容

制造商	部件编号	标称电容	基本公差	额定电压	温度范围
TDK	C3216X7R1C106K	$10\ \mu\text{F}$	$\pm 10\%$	16V	-55 至 125°C
TDK	C3216X5R1C106K	$10\ \mu\text{F}$	$\pm 10\%$	16V	-55 至 85°C
Panasonic	ECJ-3YX1C106K	$10\ \mu\text{F}$	$\pm 10\%$	16V	-55 至 125°C
Panasonic	ECJ-4YB1C106K	$10\ \mu\text{F}$	$\pm 10\%$	16V	-55 至 85°C
Murata	GRM32DR71C106KA01L	$10\ \mu\text{F}$	$\pm 10\%$	16V	-55 至 125°C
Murata	GRM31CR61C106KC31L	$10\ \mu\text{F}$	$\pm 10\%$	16V	-55 至 85°C

PIC24FV32KA304 系列

2.4.1 采用陶瓷电容时的注意事项

近年来，大容量、低电压的表面贴装式陶瓷电容极具成本效益，其容值最高可达数十微法。低 ESR、较小的物理尺寸和其他特性使得陶瓷电容在许多类型的应用中非常抢眼。

陶瓷电容适合用于本系列单片机的内部稳压器。但是，选择电容时需谨慎，应确保在应用的目标工作范围内维持足够的电容。

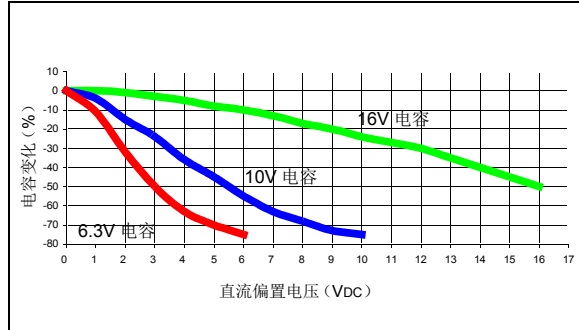
典型值为 $10\ \mu\text{F}$ 的低成本陶瓷电容具有 X5R、X7R 和 Y5V 这三种介电特性（也可使用其他类型，但是不太常见）。这些类型电容的初始公差规范通常指定为 $\pm 10\%$ 到 $\pm 20\%$ （X5R 和 X7R）或 $-20\%/+80\%$ （Y5V）。然而，这些电容在应用电路中提供的有效电容还将根据其他因素（如施加的直流偏置电压和温度）而变化。因此，电容在电路中的总公差要比初始公差规范宽的多。

X5R 和 X7R 电容通常会展示出令人满意的温度稳定性（例如：在较大的温度范围内为 $\pm 15\%$ ，具体规范请查看制造商数据手册）。但是，Y5V 电容在温度变化时的电容变化范围通常很糟糕，为 $+22\%/-82\%$ 。因此，标称值为 $10\ \mu\text{F}$ 的 Y5V 类型电容可能无法提供足够的总电容，无法满足内部稳压器稳定性和瞬态响应要求。因此，如果应用必须在宽温度范围内工作，不建议将 Y5V 电容用于内部稳压器。

除了温度的影响外，电容值较大的陶瓷电容的有效电容还可能根据施加到电容的直流电压量而有较大的变化。这一影响可能非常大，但却往往被人们忽视，或常常在文档中找不到相关说明。

典型的直流偏置电压与 X7R 类型电容的电容值关系如图 2-4 所示。

图 2-4: 直流偏置电压与电容特性



选择要用于内部稳压器的陶瓷电容时，建议选择额定电压较高的电容，以使工作电压占电容最大电压的一小部分。例如，对于内核电压为 3.3V 或 2.5V 的应用，选择额定值为 16V 的陶瓷电容。建议的电容在表 2-1 中给出。

2.5 ICSP 引脚

PGC 和 PGD 引脚用于在线串行编程（ICSP™）和调试。建议保持器件上的 ICSP 连接器和 ICSP 引脚之间的走线长度尽可能短。如果预期 ICSP 连接器上会发生 ESD 事件，建议串联一个电阻，且电阻值在几十欧姆范围内，不要超过 100Ω 。

建议不要在 PGC 和 PGD 引脚上使用上拉电阻、串联二极管和电容，因为它们会干扰编程器 / 调试器与器件之间的通信。如果应用需要这类分立元件，应在编程和调试期间将它们从电路中除去。或者，参阅相关器件闪存编程规范中的交流 / 直流特性和时序要求信息，以了解关于容性负载限制以及引脚高输入电压（ V_{IH} ）和低输入电压（ V_{IL} ）要求的信息。

对于器件仿真，请确保在软件中设置的“通信通道选择”信息（即，PGCx/PGDx 引脚）与 ICSP 到 Microchip 调试器 / 仿真器工具的物理连接一致。

关于可用 Microchip 开发工具连接要求的更多信息，请参见第 27.0 节“开发支持”。

2.6 外部振荡器引脚

许多单片机都至少有两个振荡器选项：高频主振荡器和低频辅助振荡器（详情请参见第9.0节“振荡器配置”）。

振荡器电路应放到电路板上与器件相同的一侧。将振荡器电路靠近相关振荡器引脚放置，且电路元件与引脚之间的距离不要超过 0.5 英寸（12 mm）。负载电容应在电路板的同一侧挨着振荡器本身放置。

在振荡器电路周围设置接地敷铜区，将其与周围电路隔离。接地敷铜区应直接连接到 MCU 地。不要在接地敷铜区内使用任何信号线或电源线。而且，如果使用双面的电路板，请避免在电路板上放置晶振的另一面布线。

图 2-5 给出了建议的电路板布局。可以使用包含完整振荡器引脚的单面布局来处理直插式封装器件。对于引脚排列紧密的器件，单面布局则可能无法始终完全包含所有的引脚和元件。适当的解决方案是将含有保护走线的部分连接到反面的接地层。在所有情况下，保护走线必须返回到地。

在规划应用的走线和 I/O 分配过程中，确保临近的端口引脚和离振荡器非常近的其他信号状态良好（即，没有很高的频率、陡升和陡降以及其他类似的噪声）。

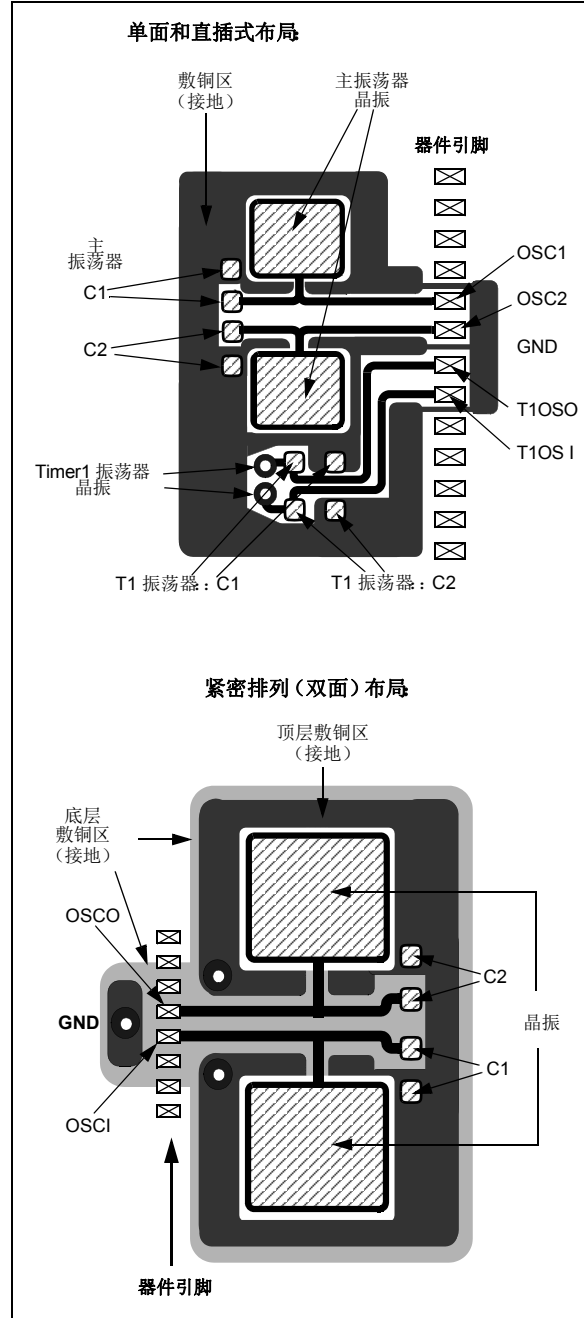
关于振荡器电路的其他信息和设计指导，请参考我公司网站（www.microchip.com）上提供的以下应用笔记：

- AN826, “Crystal Oscillator Basics and Crystal Selection for *rfPIC™* and *PICmicro®* Devices”
- AN849, “Basic *PICmicro®* Oscillator Design”
- AN943, “Practical *PICmicro®* Oscillator Analysis and Design”
- AN949, “Making Your Oscillator Work”

2.7 未使用的 I/O

未使用的 I/O 引脚应配置为输出，并驱动为逻辑低电平状态。或者，使用一个 1 kΩ 至 10 kΩ 电阻将未使用的引脚连接至 Vss，并驱动输出为逻辑低电平。

图 2-5: 建议的振荡器电路布局



PIC24FV32KA304 系列

注:

3.0 CPU

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解 CPU 的更多信息，请参见《PIC24F 系列参考手册》中的第 2 章“CPU”（DS39703A_CN）。

PIC24F CPU 采用 16 位（数据）的改进型哈佛架构，具有增强的指令集和带有长度可变的操作码字段的 24 位指令字。程序计数器（Program Counter, PC）为 23 位宽，可以寻址最大 4M 指令字的用户程序存储空间。单周期指令预取机制可帮助维持吞吐量，并使指令的执行具有预测性。除了改变程序流的指令、双字传送（MOV.D）指令和表指令以外，所有指令都在单个周期内执行。使用 REPEAT 指令可以支持无开销的程序循环结构，在任何时候都可被中断。

PIC24F 器件的编程模型中有 16 个 16 位的工作寄存器。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）用作软件堆栈指针（Software Stack Pointer, SSP），用于中断和调用。

可以选择将数据存储空间的高 32 KB 映射到由 8 位程序空间可视性页地址（Program Space Visibility Page Address, PSVPAG）寄存器定义的程序存储器或数据 EEPROM 存储器的任何 16K 字边界内的程序空间内。程序空间到数据空间的映射功能让任何指令都能像访问数据空间一样访问程序空间。

指令集架构（Instruction Set Architecture, ISA）与 PIC18 的相比有了显著的增强，但仍保持了一定程度的向后兼容性。该架构直接或通过简单的宏支持所有的 PIC18 指令和寻址模式。对编译器执行效率的需求促使了对 ISA 的许多改进。

内核支持固有（无操作数）寻址、相对寻址、立即数寻址、存储器直接寻址模式及其他 3 组寻址模式。所有模式都支持寄存器直接寻址和各种寄存器间接寻址模式。每组都提供最多 7 种寻址模式。指令根据其功能要求，与预定义的寻址模式相关联。

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

包含了一个高速 17 位 x 17 位乘法器，显著提高了内核的运算能力和吞吐量。此乘法器支持有符号、无符号和混合模式的 16 位 x 16 位或 8 位 x 8 位整数乘法。所有的乘法指令都在单周期内执行。

已对 16 位 ALU 进行了改进使其具备一个支持整数除法的硬件，该硬件支持迭代的不可撤销的除法算法。它可以和 REPEAT 指令循环机制和迭代除法指令（可选）一起工作，支持 32 位（或 16 位）除以 16 位有符号和无符号整数的除法运算。所有除法运算都需要 19 个周期来完成，但可以在任何周期边界被中断。

PIC24F 器件具有向量异常机制，具有最多 8 个不可屏蔽的陷阱源和 118 个中断源。可以为每个中断源分配 7 个优先级之一。

CPU 的框图如图 3-1 所示。

3.1 编程模型

图 3-2 中所示为 PIC24F 器件的编程模型。编程模型中的所有寄存器都是存储器映射的，并且可以由指令直接控制。

表 3-1 中提供了对每个寄存器的描述。所有与编程模型相关联的寄存器都是存储器映射的。

PIC24FV32KA304 系列

图 3-1: PIC24F CPU 内核框图

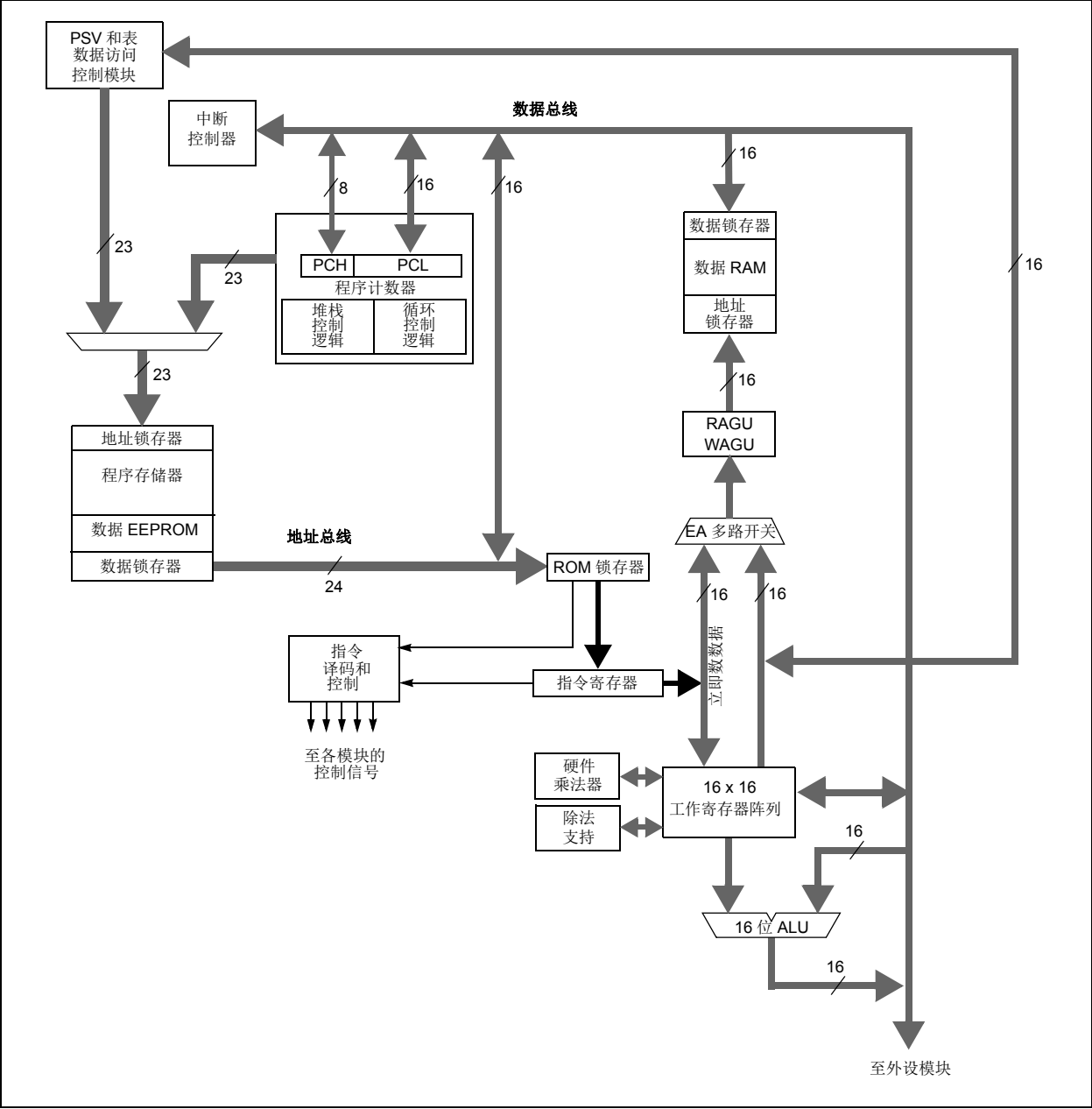
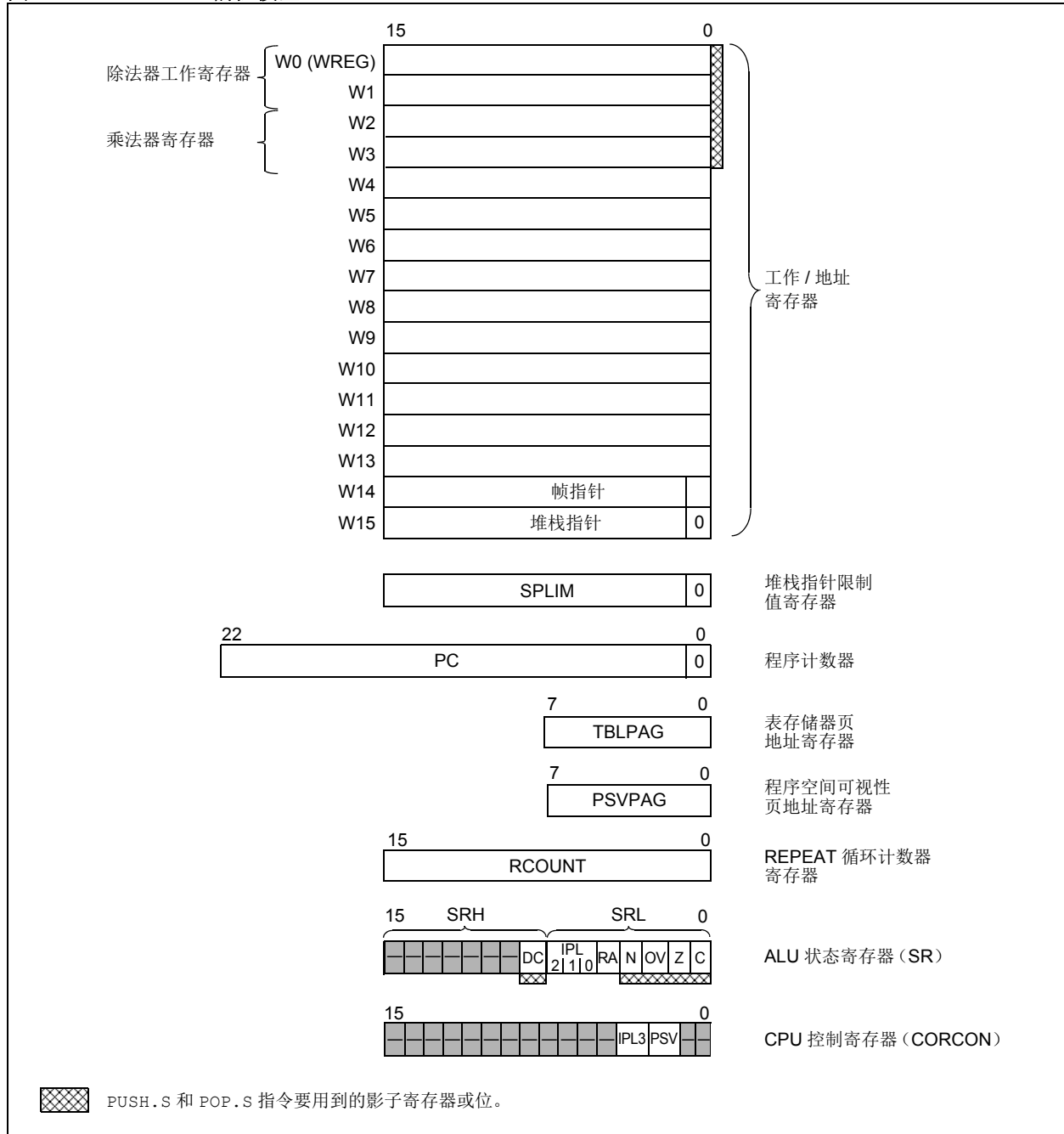


表 3-1: CPU 内核寄存器

寄存器名称	说明
W0 到 W15	工作寄存器阵列
PC	23 位程序计数器
SR	ALU 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
PSVPAG	程序空间可视性页地址寄存器
RCOUNT	Repeat 循环计数器寄存器
CORCON	CPU 控制寄存器

图 3-2: 编程模型



PIC24FV32KA304 系列

3.2 CPU 控制寄存器

寄存器 3-1: SR: ALU 状态寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HSC
—	—	—	—	—	—	—	DC
bit 15							bit 8

R/W-0, HSC ⁽¹⁾	R/W-0, HSC ⁽¹⁾	R/W-0, HSC ⁽¹⁾	R-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC
IPL2 ⁽²⁾	IPL1 ⁽²⁾	IPL0 ⁽²⁾	RA	N	OV	Z	C
bit 7							bit 0

图注:	HSC = 可由硬件置 1/ 清零的位						
R = 可读位	W = 可写位		U = 未实现位, 读为 0				
-n = 上电复位时的值	1 = 置 1		0 = 清零		x = 未知		

bit 15-9	未实现: 读为 0
bit 8	DC: 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) 发生了向高位的进位 0 = 结果的第 4 个低位或第 8 个低位未发生向高位的进位
bit 7-5	IPL<2:0>: CPU 中断优先级状态位 ^(1,2) 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	RA: REPEAT 循环活动位 1 = 正在进行 REPEAT 循环 0 = 未进行 REPEAT 循环
bit 3	N: ALU 负标志位 1 = 结果为负 0 = 结果为非负 (零或正值)
bit 2	OV: ALU 溢出标志位 1 = 在本次算术运算中有符号 (二进制补码) 运算发生了溢出 0 = 未发生溢出
bit 1	Z: 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经将该位清零 (即运算结果非零)
bit 0	C: ALU 进位 / 借位标志位 1 = 结果的最高位 (MSb) 发生了进位 0 = 结果的最高位 (MSb) 未发生进位

- 注 1: 当 NSTDIS (INTCON1<15>) = 1 时 IPLx 状态位为只读。
- 2: IPL<2:0> 状态位与 IPL3 位 (CORCON<3>) 共同决定 CPU 的中断优先级 (Interrupt Priority Level, IPL)。如果 IPL3 = 1, 则括号中的值表示 IPL。

寄存器 3-2: CORCON: CPU 控制寄存器

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/C-0, HSC	R/W-0	U-0	U-0
—	—	—	—	IPL3 ⁽¹⁾	PSV	—	—
bit 7							bit 0

图注:	HSC = 可由硬件置 1/ 清零的位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-4 **未实现:** 读为 0
- bit 3 **IPL3:** CPU 中断优先级状态位 ⁽¹⁾
 1 = CPU 中断优先级大于 7
 0 = CPU 中断优先级为 7 或更小
- bit 2 **PSV:** 数据空间中的程序空间可视性使能位
 1 = 程序空间在数据空间中可视
 0 = 程序空间在数据空间中不可视
- bit 1-0 **未实现:** 读为 0

注 1: 当 IPL3 = 1 时禁止用户中断。

3.3 算术逻辑单元 (ALU)

PIC24F ALU 为 16 位宽, 并能进行加法、减法、移位和逻辑运算。除非另外声明, 否则算术运算一般采用二进制补码。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和半借位位。

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

PIC24F CPU 融入了对乘法和除法的硬件支持。它带有专用的硬件乘法器以及支持 16 位除数除法的硬件。

3.3.1 乘法器

ALU 包含一个高速的 17 位 x 17 位乘法器。它支持几种乘法模式下的无符号、有符号或混合符号运算:

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

PIC24FV32KA304 系列

3.3.2 除法器

除法模块支持具有下列数据长度的 32 位/16 位和 16 位/16 位的有符号和无符号整数除法运算：

- 1. 32 位有符号数 /16 位有符号数
- 2. 32 位无符号数 /16 位无符号数
- 3. 16 位有符号数 /16 位有符号数
- 4. 16 位无符号数 /16 位无符号数

所有除法指令的商都被放在 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临近的 W 寄存器构成的对（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。

3.3.3 支持多位移位

PIC24F ALU 支持单位和单周期多位算术和逻辑移位操作。由一个移位寄存器模块实现多位移位，在单个周期内最多可将数据算术右移或左移 15 位。所有的多位移位指令仅支持源操作数和目标结果的寄存器直接寻址模式。

表 3-2 中汇总了所有使用移位操作的指令。

表 3-2: 使用单位和多位移位操作的指令

指令	说明
ASR	将源寄存器算术右移一位或多位。
SL	将源寄存器左移一位或多位。
LSR	将源寄存器逻辑右移一位或多位。

PIC24FV32KA304 系列

4.0 存储器构成

作为哈佛架构器件，PIC24F 单片机具有独立的程序和数据存储空间以及独立的程序和数据总线。此架构还允许在代码执行过程中直接通过数据空间访问程序空间。

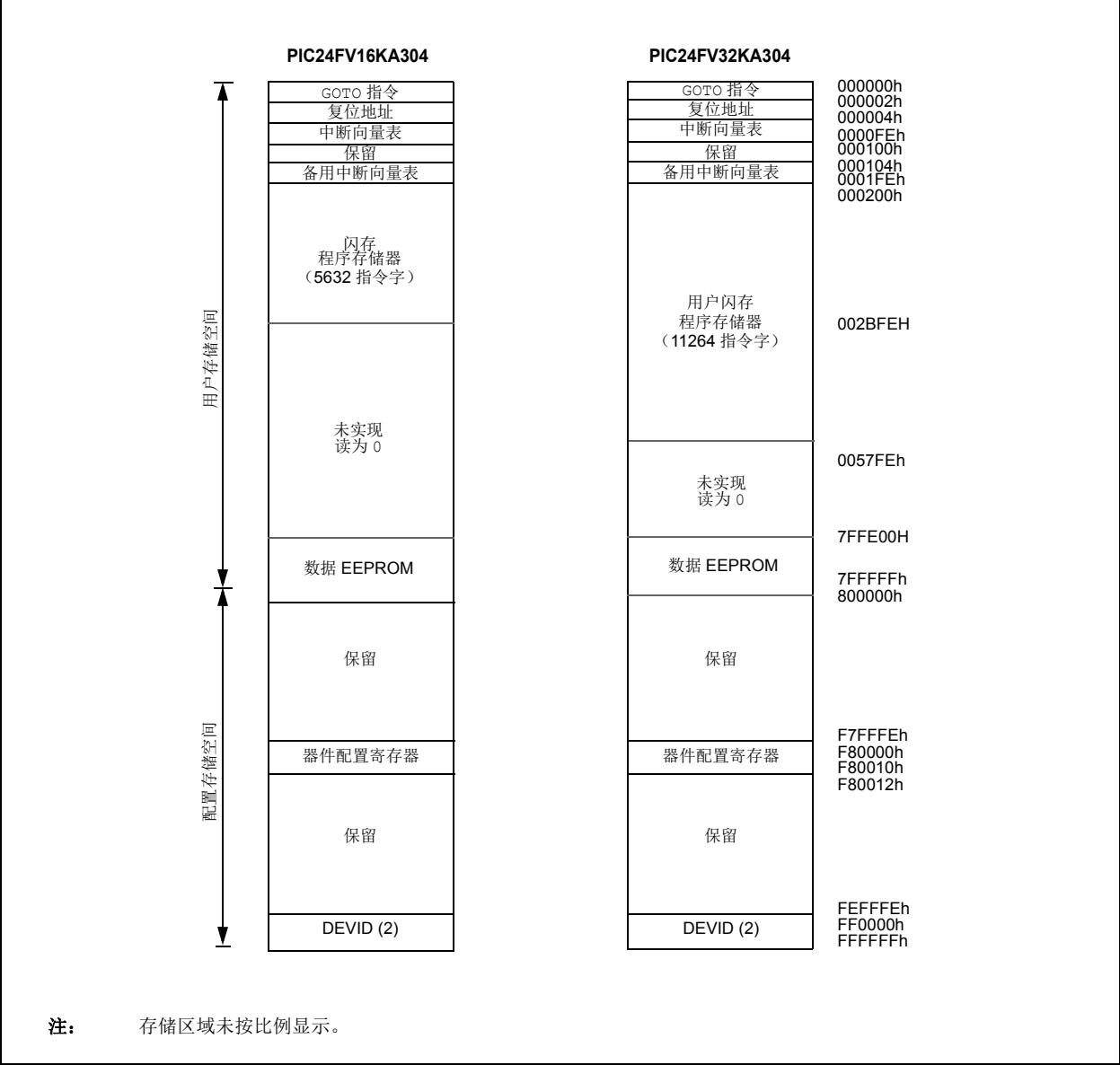
4.1 程序地址空间

PIC24FV32KA304 系列器件的程序地址存储空间可存储 4M 指令字。可通过由程序执行过程中 23 位程序计数器（Program Counter，PC）或第 4.3 节“程序存储空间与数据存储空间的接口”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。

用户只能访问程序存储空间的低半地址部分（地址范围为 000000h 至 7FFFFFFh）。使用 TBLRD/TBLWT 指令时，情况有所不同，这两条指令使用 TBLPAG<7> 位以允许访问配置存储空间中的配置位和器件 ID。

图 4-1 给出了 PIC24FV32KA304 系列器件的存储器映射情况。

图 4-1： PIC24FV32KA304 系列器件的程序存储空间映射



4.1.1 程序存储器构成

程序存储空间由可字寻址的块构成。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没有实现。低位字的地址总是偶数，而高位字的地址为奇数，如图 4-2 所示。

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。这种对齐方式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。

4.1.2 硬存储器向量

所有 PIC24F 器件中从 00000h 到 000200h 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开始处。用户可在 000000h 地址编写一条 GOTO 指令以将代码的实际起始地址设置为 000002h。

PIC24F 器件也具有 2 个中断向量表，地址范围分别为 000004h 至 0000FFh 和 000104h 至 0001FFh。这两个中断向量表允许使用不同的中断服务程序（Interrupt Service Routines, ISR）处理每个器件中断源。第 8.1 节“中断向量表（IVT）”提供了有关中断向量表（Interrupt Vector Table, IVT）的更多详细信息。

4.1.3 数据 EEPROM

在 PIC24FV32KA304 系列器件中，数据 EEPROM 映射到用户程序存储空间的顶部，起始于地址 7FFE00 并扩大到地址 7FFFFF。

数据 EEPROM 存储器的宽度为 16 位，深度为 256 个字。类似于用户代码存储器，可使用表读和表写操作访问该存储器。

4.1.4 器件配置字

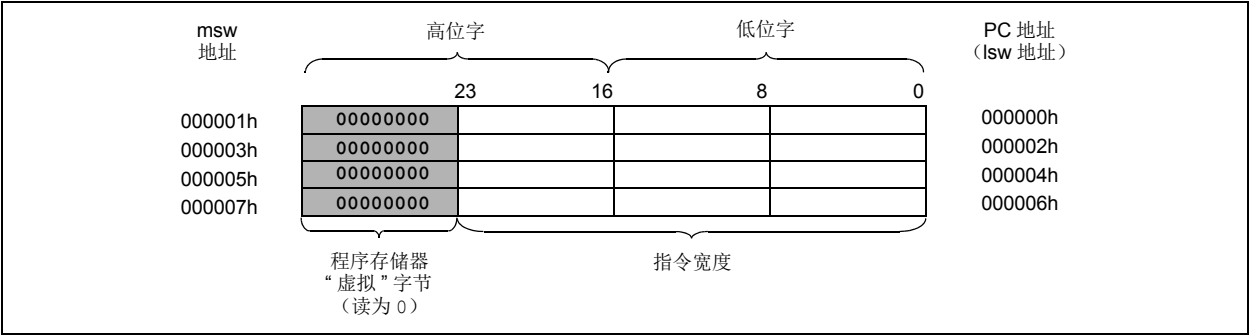
PIC24FV32KA304 系列器件的器件配置字的地址如表 4-1 所示。图 4-1 给出了器件配置字在存储器映射中的位置。

有关器件配置字的更多信息，请参见第 26.0 节“特殊功能”。

表 4-1: PIC24FV32KA304 系列器件的器件配置字

配置字	配置字地址
FBS	F80000
FGS	F80004
FOSCSEL	F80006
FOSC	F80008
FWDT	F8000A
FPOR	F8000C
FICD	F8000E
FDS	F80010

图 4-2: 程序存储器构成



PIC24FV32KA304 系列

4.2 数据地址空间

PIC24F 内核具有一个 16 位宽的数据存储空间，可将其作为一个线性空间寻址。使用两个地址发生单元（Address Generation Unit，AGU）对数据空间进行寻址，分别用于读 / 写操作。图 4-3 给出了数据存储空间的映射。

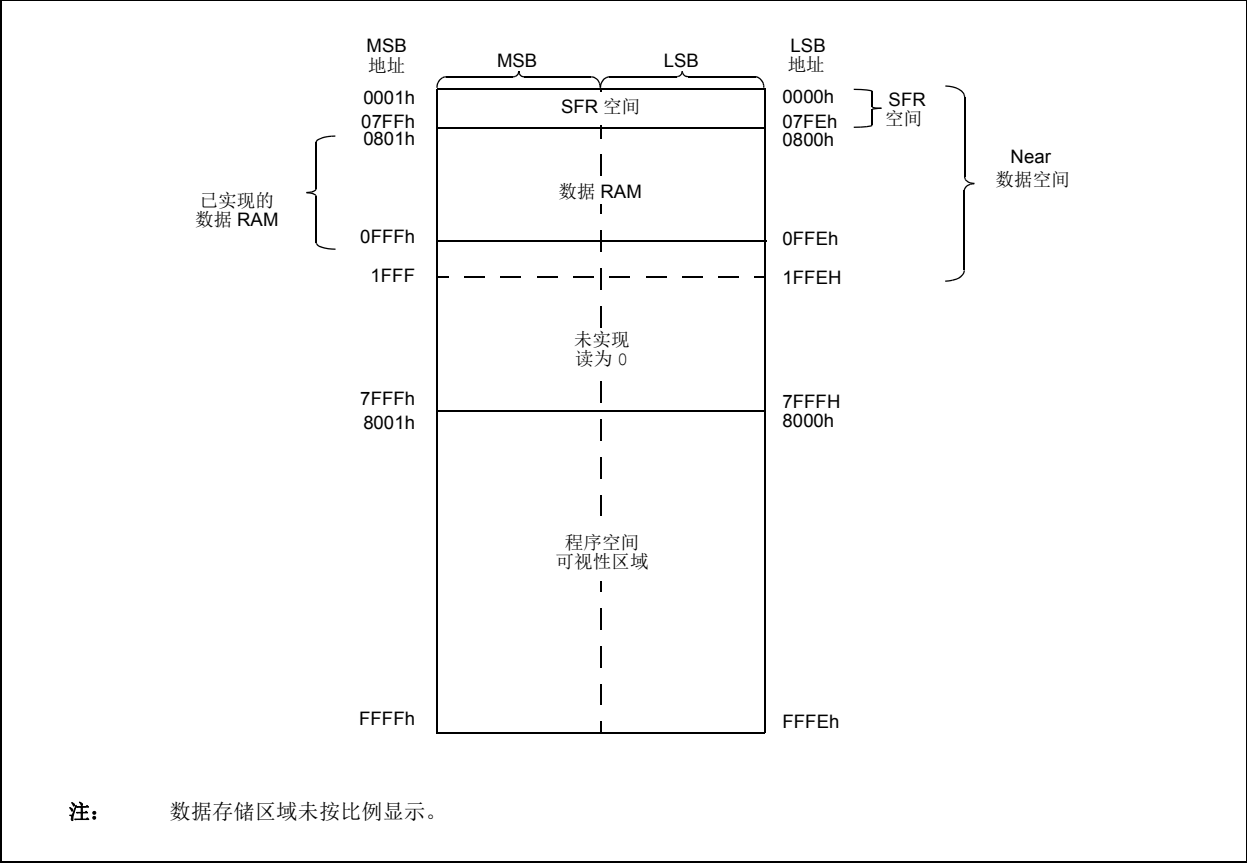
数据存储空间中的所有有效地址（Effective Addresses，EA）均为 16 位宽且指向数据空间中的字节。这种构成方式使得数据空间的地址范围为 64 KB 或 32K 字。数据存储空间的下半部分（即，EA<15> = 0 时）用于实现的存储器地址，而上半部分（EA<15> = 1）保留用于程序空间可视性（Program Space Visibility，PSV）区域（见第 4.3.3 节“使用程序空间可视性从程序存储器读数据”）。

PIC24FV32KA304 系列器件实现了总的 1024 字数据存储单元。如果 EA 指向该区域外部的存储单元，将返回一个全零的字或字节。

4.2.1 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，但所有数据空间的 EA 都被解析为字节。每个字的低字节（Least Significant Byte，LSB）部分为偶地址，而高字节（Most Significant Byte，MSB）部分则为奇地址。

图 4-3: PIC24FV32KA304 系列器件的数据存储空间映射



4.2.2 数据存储器和对齐方式

为了与先前的 PIC® MCU 器件兼容和提高数据存储空间的使用效率，PIC24F 指令集既支持字操作，也支持字节操作。字节访问会在内部对按字对齐的存储空间的所有有效地址（EA）进行计算调整。例如，对于执行后修改寄存器间接寻址模式 [Ws++] 的结果，字节操作时，内核将其识别为值 Ws + 1，而字操作时，内核将其识别为 Ws + 2。

使用任何有效地址的低字节决定要选择哪个字节，数据字节读操作将读取包含此字节的整个字。选中的字节将被放在数据总线的低字节处。也就是说，数据存储器和寄存器被组织为两个并行的字节宽实体，它们共享（字）地址译码，而写入线相互独立。数据字节写操作仅写入存储阵列和寄存器中与字节地址匹配的相应部分。

所有字访问都必须按偶地址对齐。不支持不对齐的字数据取操作。因此当混合字节和字操作或从 8 位 MCU 代码移植时，必须要小心。若试图进行这种不对齐的读或写操作，则会产生地址错误陷阱。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。无论是哪种情况都将发生陷阱，从而允许系统和 / 或用户检查地址错误发生之前的机器状态。

所有装入任意 W 寄存器的字节都将装入 W 寄存器的低字节。W 寄存器的高字节不变。

提供了一条符号扩展指令（SE），允许用户把 8 位有符号数据转换为 16 位有符号值。或者，对于 16 位无符号数据，用户可以清零任何 W 寄存器的 MSB，方法是在相应的地址处执行一条零扩展（ZE）指令。

尽管大多数指令能够对字或字节大小的数据进行操作，但需要注意的是，某些指令只对字大小的数据进行操作。

4.2.3 NEAR 数据空间

0000h 和 1FFFh 之间的 8 KB 的区域被称为 Near 数据空间（Near Data Space，NDS）。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地址单元。剩余的数据空间通过间接寻址访问。此外，还可以使用 MOV 指令寻址整个数据空间，支持使用 16 位地址字段的存储器直接寻址（Memory Direct Addressing，MDA）模式。对于 PIC24FV32KA304 系列器件，整个已实现的数据存储器位于 Near 数据空间。

4.2.4 SFR 空间

Near 数据空间的前 2 KB（0000h 至 07FFh）主要被特殊功能寄存器（Special Function Register，SFR）占用。PIC24F 的内核和外设模块使用这些寄存器来控制器件的工作。

SFR 分布在受其控制的大量模块中，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。表 4-2 给出了 SFR 空间的布局表，显示了实际实现 SFR 的位置。每个已实现的区域表示一个 32 字节区域，其中至少有一个地址实现为 SFR。表 4-3 至表 4-25 给出了已实现的 SFR 及其地址的完整列表。

表 4-2: SFR 数据空间的已实现区域

SFR 空间地址								
	xx00	xx20	xx40	xx60	xx80	xxA0	xxC0	xxE0
000h	内核			ICN	中断			—
100h	定时器		捕捉	—	比较	—	—	—
200h	I ² C™	UART	SPI		—	—	I/O	
300h	A/D/CMTU				—	—	—	—
400h	—	—	—	—	—	—	—	—
500h	—	—	—	—	—	—	—	—
600h	—	RTC/ 比较器	CRC	—	—			
700h	—	—	系统 /DS/HLVD	NVM/PMD	—	—	—	—

图注： — = 此存储块中未实现的 SFR。

表 4-3: CPU 内核寄存器映射

寄存器名称	起始地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
WREG0	0000	WREG0																0000
WREG1	0002	WREG1																0000
WREG2	0004	WREG2																0000
WREG3	0006	WREG3																0000
WREG4	0008	WREG4																0000
WREG5	000A	WREG5																0000
WREG6	000C	WREG6																0000
WREG7	000E	WREG7																0000
WREG8	0010	WREG8																0000
WREG9	0012	WREG9																0000
WREG10	0014	WREG10																0000
WREG11	0016	WREG11																0000
WREG12	0018	WREG12																0000
WREG13	001A	WREG13																0000
WREG14	001C	WREG14																0000
WREG15	001E	WREG15																0000
SPLIM	0020	SPLIM																xxxx
PCL	002E	PCL																0000
PCH	0030	—	—	—	—	—	—	—	—	—	PCH							0000
TBLPAG	0032	—	—	—	—	—	—	—	—	TBLPAG								0000
PSVPAG	0034	—	—	—	—	—	—	—	—	PSVPAG								0000
RCOUNT	0036	RCOUNT																xxxxxx
SR	0042	—	—	—	—	—	—	—	DC	IPL2	IPL1	IPL0	RA	无	OV	Z	C	0000
CORCON	0044	—	—	—	—	—	—	—	—	—	—	—	—	IPL3	PSV	—	—	0000
DISICNT	0052	—	—	DISICNT													xxxx	

图注： — = 未实现，读为 0。复位值以十六进制格式显示。

表 4-4: ICN 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CNPD1	0056	CN15PDE ⁽¹⁾	CN14PDE	CN13PDE	CN12PDE	CN11PDE	CN10PDE ^(1,2)	CN9PDE ⁽¹⁾	CN8PDE ⁽³⁾	CN7PDE ⁽¹⁾	CN6PDE	CN5PDE	CN4PDE	CN3PDE	CN2PDE	CN1PDE	CN0PDE	0000
CNPD2	0058	CN31PDE ^(1,2)	CN30PDE	CN29PDE	CN28PDE ^(1,2)	CN27PDE ⁽¹⁾	CN26PDE ^(1,2)	CN25PDE ^(1,2)	CN24PDE ⁽¹⁾	CN23PDE	CN22PDE	CN21PDE	CN20PDE ^(1,2)	CN19PDE ^(1,2)	CN18PDE ^(1,2)	CN17PDE ^(1,2)	CN16PDE ⁽¹⁾	0000
CNPD3	005A	—	—	—	—	—	—	—	—	—	—	—	CN36PDE ^(1,2)	CN35PDE ^(1,2)	CN34PDE ^(1,2)	CN33PDE ^(1,2)	CN32PDE ^(1,2)	0000
CNEN1	0062	CN15IE ⁽¹⁾	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE ^(1,2)	CN9IE ⁽¹⁾	CN8IE ⁽³⁾	CN7IE ⁽¹⁾	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0064	CN31IE ^(1,2)	CN30IE	CN29IE	CN28IE ^(1,2)	CN27IE ⁽¹⁾	CN26IE ^(1,2)	CN25IE ^(1,2)	CN24IE ⁽¹⁾	CN23IE	CN22IE	CN21IE	CN20IE ^(1,2)	CN19IE ^(1,2)	CN18IE ^(1,2)	CN17IE ^(1,2)	CN16IE ⁽¹⁾	0000
CNEN3	0066	—	—	—	—	—	—	—	—	—	—	—	CN36IE ^(1,2)	CN35IE ^(1,2)	CN34IE ^(1,2)	CN33IE ^(1,2)	CN32IE ^(1,2)	0000
CNPU1	006E	CN15PUE ⁽¹⁾	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE ^(1,2)	CN9PUE ⁽¹⁾	CN8PUE ⁽³⁾	CN7PUE ⁽¹⁾	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	0070	CN31PUE ^(1,2)	CN30PUE	CN29PUE	CN28PUE ^(1,2)	CN27PUE ⁽¹⁾	CN26PUE ^(1,2)	CN25PUE ^(1,2)	CN24PUE ⁽¹⁾	CN23PUE	CN22PUE	CN21PUE	CN20PUE ^(1,2)	CN19PUE ^(1,2)	CN18PUE ^(1,2)	CN17PUE ^(1,2)	CN16PUE ⁽¹⁾	0000
CNPU3	0072	—	—	—	—	—	—	—	—	—	—	—	CN36PUE ^(1,2)	CN35PUE ^(1,2)	CN34PUE ^(1,2)	CN33PUE ^(1,2)	CN32PUE ^(1,2)	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

- 注
- 1: 这些位在 20 引脚器件上未实现。
 - 2: 这些位在 28 引脚器件上未实现。
 - 3: 这些位在 FV 器件上未实现。

表 4-5: 中断控制器寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
INTCON1	0080	NSTDIS	—	—	—	—	—	—	—	—	—	—	MATHERR	ADDRERR	STKERR	OSCFail	—	0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	—	—	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	NVMIF	—	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPF1IF	T3IF	T2IF	OC2IF	IC2IF	—	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	—	OC3IF	—	—	—	—	INT1IF	CNIF	CMIF	MI2C1IF	SI2C1IF	0000
IFS2	0088	—	—	—	—	—	—	—	—	—	—	IC3IF	—	—	—	SPI2IF	SPF2IF	0000
IFS3	008A	—	RTCIF	—	—	—	—	—	—	—	—	—	—	—	MI2C2IF	SI2C2IF	—	0000
IFS4	008C	—	—	CTMUIF	—	—	—	—	HLVDIF	—	—	—	—	CRCIF	U2ERIF	U1ERIF	—	0000
IFS5	008E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ULPWUIF	0000
IEC0	0094	NVMIE	—	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPF1IE	T3IE	T2IE	OC2IE	IC2IE	—	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	—	OC3IE	—	—	—	—	INT1IE	CNIE	CMIE	MI2C1IE	SI2C1IE	0000
IEC2	0098	—	—	—	—	—	—	—	—	—	—	IC3IE	—	—	—	SPI2IE	SPF2IE	0000
IEC3	009A	—	RTCIE	—	—	—	—	—	—	—	—	—	—	—	MI2C2IE	SI2C2IE	—	0000
IEC4	009C	—	—	CTMUIE	—	—	—	—	HLVDIE	—	—	—	—	CRCIE	U2ERIE	U1ERIE	—	0000
IEC5	009E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ULPWUIE	0000
IPC0	00A4	—	T1IP2	T1IP1	T1IP0	—	OC1IP2	OC1IP1	OC1IP0	—	IC1IP2	IC1IP1	IC1IP0	—	INT0IP2	INT0IP1	INT0IP0	4444
IPC1	00A6	—	T2IP2	T2IP1	T2IP0	—	OC2IP2	OC2IP1	OC2IP0	—	IC2IP2	IC2IP1	IC2IP0	—	—	—	—	4444
IPC2	00A8	—	U1RXIP2	U1RXIP1	U1RXIP0	—	SPI1IP2	SPI1IP1	SPI1IP0	—	SPF1IP2	SPF1IP1	SPF1IP0	—	T3IP2	T3IP1	T3IP0	4444
IPC3	00AA	—	NVMIP2	NVMIP1	NVMIP0	—	—	—	—	—	AD1IP2	AD1IP1	AD1IP0	—	U1TXIP2	U1TXIP1	U1TXIP0	4044
IPC4	00AC	—	CNIP2	CNIP1	CNIP0	—	CMIP2	CMIP1	CMIP0	—	MI2C1P2	MI2C1P1	MI2C1P0	—	SI2C1P2	SI2C1P1	SI2C1P0	4444
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	INT1IP2	INT1IP1	INT1IP0	0004
IPC6	00B0	—	T4IP2	T4IP1	T4IP0	—	—	—	—	—	OC3IP2	OC3IP1	OC3IP0	—	—	—	—	4040
IPC7	00B2	—	U2TXIP2	U2TXIP1	U2TXIP0	—	U2RXIP2	U2RXIP1	U2RXIP0	—	INT2IP2	INT2IP1	INT2IP0	—	T5IP2	T5IP1	T5IP0	4440
IPC8	00B4	—	—	—	—	—	—	—	—	—	SPI2IP2	SPI2IP1	SPI2IP0	—	SPF2IP2	SPF2IP1	SPF2IP0	0044
IPC9	00B6	—	—	—	—	—	—	—	—	—	IC3IP2	IC3IP1	IC3IP0	—	—	—	—	0040
IPC12	00BC	—	—	—	—	—	MI2C2IP2	MI2C2IP1	MI2C2IP0	—	SI2C2IP2	SI2C2IP1	SI2C2IP0	—	—	—	—	0440
IPC15	00C2	—	—	—	—	—	RTCIP2	RTCIP1	RTCIP0	—	—	—	—	—	—	—	—	0400
IPC16	00C4	—	CRCIP2	CRCIP1	CRCIP0	—	U2ERIP2	U2ERIP1	U2ERIP0	—	U1ERIP2	U1ERIP1	U1ERIP0	—	—	—	—	4440
IPC18	00C8	—	—	—	—	—	—	—	—	—	—	—	—	—	HLVDIP2	HLVDIP1	HLVDIP0	0004
IPC19	00CA	—	—	—	—	—	—	—	—	—	CTMUIP2	CTMUIP1	CTMUIP0	—	—	—	—	0040
IPC20	00CC	—	—	—	—	—	—	—	—	—	—	—	—	—	ULPWUIP2	ULPWUIP1	ULPWUIP0	0000
INTTREG	00E0	CPUIRQ	—	VHOLD	—	ILR3	ILR2	ILR1	ILR0	—	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-6: 定时器寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TMR1	0100	TMR1																0000
PR1	0102	PR1																FFFF
T1CON	0104	TON	—	TSIDL	—	—	—	T1ECS1	T1ECS0	—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—	0000
TMR2	0106	TMR2																0000
TMR3HLD	0108	TMR3HLD																0000
TMR3	010A	TMR3																0000
PR2	010C	PR2																0000
PR3	010E	PR3																FFFF
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	T32	—	TCS	—	FFFF
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	—	TCS	—	0000
TMR4	0114	TMR4																0000
TMR5HLD	0116	TMR5HLD																0000
TMR5	0118	TMR5																0000
PR4	011A	PR4																FFFF
PR5	011C	PR5																FFFF
T4CON	011E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	T45	—	TCS	—	0000
T5CON	0120	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	—	TCS	—	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-7: 输入捕捉寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
IC1CON1	0140	—	—	ICSIDL	ICTSEL2	ICTSEL1	ICTSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC1CON2	0142	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC1BUF	0144	IC1BUF																0000
IC1TMR	0146	IC1TMR																xxxx
IC2CON1	0148	—	—	ICSIDL	IC2TSEL2	IC2TSEL1	IC2TSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC2CON2	014A	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC2BUF	014C	IC2BUF																0000
IC2TMR	014E	IC2TMR																xxxx
IC3CON1	0150	—	—	ICSIDL	IC3TSEL2	IC3TSEL1	IC3TSEL0	—	—	—	IC11	IC10	ICOV	ICBNE	ICM2	ICM1	ICM0	0000
IC3CON2	0152	—	—	—	—	—	—	—	IC32	ICTRIG	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000D
IC3BUF	0154	IC3BUF																0000
IC3TMR	0156	IC3TMR																xxxx

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-8: 输出比较寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
OC1CON1	0190	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC1CON2	0192	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIG-STAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC1RS	0194	OC1RS																0000
OC1R	0196	OC1R																0000
OC1TMR	0198	OC1TMR																xxxx
OC2CON1	019A	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC2CON2	019C	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIG-STAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC2RS	019E	OC2RS																0000
OC2R	01A0	OC2R																0000
OC2TMR	01A2	OC2TMR																xxxx
OC3CON1	01A4	—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1	ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2	OCM1	OCM0	0000
OC3CON2	01A6	FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1	DCB0	OC32	OCTRIG	TRIG-STAT	OCTRIS	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0	000C
OC3RS	01A8	OC3RS																0000
OC3R	01AA	OC3R																0000
OC3TMR	01AC	OC3TMR																xxxx

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-9: I2Cx 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
I2C1RCV	0200	—	—	—	—	—	—	—	—	I2CRCV								0000
I2C1TRN	0202	—	—	—	—	—	—	—	—	I2CTRN								00FF
I2C1BRG	0204	—	—	—	—	—	—	—	—	I2CBRG								0000
I2C1CON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C1STAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D/ $\bar{A}$	P	S	R/ $\bar{W}$	RBF	TBF	0000
I2C1ADD	020A	—	—	—	—	—	—	I2CADD										0000
I2C1MSK	020C	—	—	—	—	—	—	AMSK9	AMSK8	AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0	0000
I2C2RCV	0210	—	—	—	—	—	—	—	—	I2CRCV								0000
I2C2TRN	0212	—	—	—	—	—	—	—	—	I2CTRN								00FF
I2C2BRG	0214	—	—	—	—	—	—	—	—	I2CBRG								0000
I2C2CON	0216	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C2STAT	0218	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D/ $\bar{A}$	P	S	R/ $\bar{W}$	RBF	TBF	0000
I2C2ADD	021A	—	—	—	—	—	—	I2CADD										0000
I2C2MSK	021C	—	—	—	—	—	—	AMSK9	AMSK8	AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-10: UARTx 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
U1MODE	0220	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U1TXREG	0224	—	—	—	—	—	—	—	U1TXREG									xxxx
U1RXREG	0226	—	—	—	—	—	—	—	U1RXREG									0000
U1BRG	0228	BRG																0000
U2MODE	0230	UARTEN	—	USIDL	IREN	RTSMD	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U2STA	0232	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U2TXREG	0234	—	—	—	—	—	—	—	U2TXREG									xxxx
U2RXREG	0236	—	—	—	—	—	—	—	U2RXREG									0000
U2BRG	0238	BRG																0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-11: SPIx 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	SPIBEC2	SPIBEC1	SPIBEC0	SRMPT	SPIROV	SR1MPT	SISEL2	SISEL1	SISEL0	SPITBF	SPIRBF	0000
SPI1CON1	0242	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000
SPI1CON2	0244	FRMEN	SPIFSD	SPIFPOL	—	—	—	—	—	—	—	—	—	—	—	SPIFE	SPIBEN	0000
SPI1BUF	0248	SPI1BUF																0000
SPI2STAT	0260	SPIEN	—	SPISIDL	—	—	SPIBEC2	SPIBEC1	SPIBEC0	SRMPT	SPIROV	SRXMPT	SISEL2	SISEL1	SISEL0	SPITBF	SPIRBF	0000
SPI2CON1	0262	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000
SPI2CON2	0264	FRMEN	SPIFSD	SPIFPOL	—	—	—	—	—	—	—	—	—	—	—	SPIFE	SPIBEN	0000
SPI2BUF	0268	SPI2BUF																0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-12: PORTA 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11 ^(2,3)	Bit 10 ^(2,3)	Bit 9 ^(2,3)	Bit 8 ^(2,3)	Bit 7 ⁽²⁾	Bit 6 ⁽⁴⁾	Bit 5 ⁽¹⁾	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISA	02C0	—	—	—	—	TRISA11	TRISA10	TRISA9	TRISA8	TRISA7	TRISA6	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	00DF
PORTA	02C2	—	—	—	—	RA11	RA10	RA9	RA8	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	—	—	—	—	LATA11	LATA10	LATA9	LATA8	LATA7	LATA6	—	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA	02C6	—	—	—	—	ODA11	ODA10	ODA9	ODA8	ODA7	ODA6	—	ODA4	ODA3	ODA2	ODA1	ODA0	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

- 注
- 1: 该位仅当 MCLRE = 1 时可用。
 - 2: 这些位在 20 引脚器件上未实现。
 - 3: 这些位在 28 引脚器件上未实现。
 - 4: 这些位在 FV 器件上未实现。

表 4-13: PORTB 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11 ⁽¹⁾	Bit 10 ⁽¹⁾	Bit 9	Bit 8	Bit 7	Bit 6 ⁽¹⁾	Bit 5 ⁽¹⁾	Bit 4	Bit 3 ⁽¹⁾	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISB	02C8	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	FFFF
PORTB	02CA	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
LATB	02CC	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx
ODCB	02CE	ODB15	ODB14	ODB13	ODB12	ODB11	ODB10	ODB9	ODB8	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

- 注
- 1: 这些位在 20 引脚器件上未实现。

表 4-14: PORTC 寄存器映射⁽¹⁾

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISC	02D0	—	—	—	—	—	—	TRISC9	TRISC8	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	03FF
PORTC	02D2	—	—	—	—	—	—	RC9	RC8	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx
LATC	02D4	—	—	—	—	—	—	LATC9	LATC8	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	xxxx
ODCC	02D6	—	—	—	—	—	—	ODC9	ODC8	ODC7	ODC6	ODC5	ODC4	ODC3	ODC2	ODC1	ODC0	0000

图注: — = 未实现，读为 0。复位值以十六进制格式显示。

注 1: PORTC 在 20 引脚器件和 28 引脚器件上未实现。

表 4-15: 焊盘配置寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
PADCFG1	02FC	—	—	—	—	—	—	—	—	—	—	SMBUSDEL2	SMBUSDEL1	—	—	—	—	0000

图注: — = 未实现，读为 0。复位值以十六进制格式显示。

表 4-16: A/D 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ADC1BUF0	0300	ADC1BUF0																xxxx
ADC1BUF1	0302	ADC1BUF1																xxxx
ADC1BUF2	0304	ADC1BUF2																xxxx
ADC1BUF3	0306	ADC1BUF3																xxxx
ADC1BUF4	0308	ADC1BUF4																xxxx
ADC1BUF5	030A	ADC1BUF5																xxxx
ADC1BUF6	030C	ADC1BUF6																xxxx
ADC1BUF7	030E	ADC1BUF7																xxxx
ADC1BUF8	0310	ADC1BUF8																xxxx
ADC1BUF9	0312	ADC1BUF9																xxxx
ADC1BUF10	0314	ADC1BUF10																xxxx
ADC1BUF11	0316	ADC1BUF11																xxxx
ADC1BUF12	0318	ADC1BUF12																xxxx
ADC1BUF13	031A	ADC1BUF13																xxxx
ADC1BUF14	031C	ADC1BUF14																xxxx
ADC1BUF15	031E	ADC1BUF15																xxxx
ADC1BUF16	0320	ADC1BUF16																xxxx
ADC1BUF17	0322	ADC1BUF17																xxxx
AD1CON1	0340	ADON	—	ADSIDL	—	—	MODE12	FORM1	FORM0	SSRC3	SSRC2	SSRC1	SSRC0	—	ASAM	SAMP	DONE	0000
AD1CON2	0342	PVCFG1	PVCFG0	NVCFG0	OFFCAL	BUFREGEN	CSCNA	—	—	BUFS	SMPI4	SMPI3	SMPI2	SMPI1	SMPI0	BUFM	ALTS	0000
AD1CON3	0344	ADRC	EXTSAM	—	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0	ADCS7	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0	0000
AD1CHS	0348	CH0NB2	CH0NB1	CH0NB0	CH0SB4	CH0SB3	CH0SB2	CH0SB1	CH0SB0	CH0NA2	CH0NA1	CH0NA0	CH0SA4	CH0SA3	CH0SA2	CH0SA1	CH0SA0	0000
AD1CSSH	034E	—	CSSL30	CSSL29	CSSL28	CSSL27	CSSL26	—	—	—	—	—	—	—	—	CSSL17	CSSL16	0000
AD1CSSL	0350	CSSL15	CSSL14	CSSL13	CSSL12	CSSL11	CSSL10	CSSL9	CSSL8	CSSL7	CSSL6	CSSL5	CSSL4	CSSL3	CSSL2	CSSL1	CSSL0	0000
AD1CON5	0354	ASEN	LPEN	CTMUREQ	BGREQ	r	—	ASINT1	ASINT0	—	—	—	—	WM1	WM0	CM1	CM0	0000
AD1CHITH	0356	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CHH17	CHH16	0000
AD1CHITL	0358	CHH15	CHH14	CHH13	CHH12	CHH11	CHH10	CHH9	CHH8	CHH7	CHH6	CHH5	CHH4	CHH3	CHH2	CHH1	CHH0	0000

图注: — = 未实现, 读为 0。r = 保留。复位值以十六进制格式显示。

表 4-17: CTMU 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CTMUCON1	035A	CTMUEN	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG	—	—	—	—	—	—	—	—	0000
CTMUCON2	035C	EDG1MOD	EDG1POL	EDG1SEL3	EDG1SEL2	EDG1SEL1	EDG1SEL0	EDG2STAT	EDG1STAT	EDG2EMOD	EDG2POL	EDG2SEL3	EDG2SEL2	EDG2SEL1	EDG2SEL0	—	—	0000
CTMUICON	035E	ITRIM5	ITRIM4	ITRIM3	ITRIM2	ITRIM1	ITRIM0	IRNG1	IRNG0	—	—	—	—	—	—	—	—	0000
AD1CTMUENH	0360	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CTMEN17	CTMEN16	0000
AD1CTMUENL	0362	CTMEN15	CTMEN14	CTMEN13	CTMEN12	CTMEN11	CTMEN10	CTMEN9	CTMEN8	CTMEN7	CTMEN6	CTMEN5	CTMEN4	CTMEN3	CTMEN2	CTMEN1	CTMEN0	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-18: 模拟选择寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ANSA	04E0	—	—	—	—	—	—	—	—	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	000F
ANSB	04E2	ANSB15	ANSB14	ANSB13	ANSB12	—	—	—	—	—	—	—	ANSB4	ANSB3 ⁽¹⁾	ANSB2	ANSB1	ANSB0	F01F
ANSC	04E4	—	—	—	—	—	—	—	—	—	—	—	—	ANSC2 ^(1,2)	ANSC1 ^(1,2)	ANSC0 ^(1,2)	0007	

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

注 1: 这些位在 20 引脚器件上未实现。

注 2: 这些位在 28 引脚器件上未实现。

表 4-19: 实时时钟和日历寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ALRMVAL	0620	ALRMVAL																xxxx
ALCFGRPT	0622	ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	ALRMPTR1	ALRMPTR0	ARPT7	ARPT6	ARPT5	ARPT4	ARPT3	ARPT2	ARPT1	ARPT0	0000
RTCVAL	0624	RTCVAL																xxxx
RCFGCAL	0626	RTCEN	—	RTCWREN	RTCSYNC	HALFSEC	RTCOC	RTCPTR1	RTCPTR0	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	0000
RTCPWC	0628	PWCEN	PWCPOL	PWCCPRE	PWCSPRE	RTCCLK1	RTCCLK0	RTCCOUT1	RTCCOUT0	—	—	—	—	—	—	—	—	xxxx

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-20: 三比较器寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CMSTAT	0630	CMIDL	—	—	—	—	C3EVT	C2EVT	C1EVT	—	—	—	—	—	C3OUT	C2OUT	C1OUT	xxxx
CVRCON	0632	—	—	—	—	—	—	—	—	CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0	0000
CM1CON	0634	CON	COE	CPOL	CLPWR	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	xxxx
CM2CON	0636	CON	COE	CPOL	CLPWR	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	0000
CM3CON	0638	CON	COE	CPOL	CLPWR	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-21: CRC 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CRCCON1	0640	CRCEN	—	CSIDL	VWORD4	VWORD3	VWORD2	VWORD1	VWORD0	CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	—	—	—	0000
CRCCON2	0642	—	—	—	DWIDTHx4	DWIDTHx3	DWIDTHx2	DWIDTHx1	DWIDTHx0	—	—	—	PLEN4	PLEN3	PLEN2	PLEN1	PLEN0	0000
CRCXORL	0644	X15	X14	X13	X12	X11	X10	X9	X8	X7	X6	X5	X4	X3	X2	X1	—	0000
CRCXORH	0646	X31	X30	X29	X28	X27	X26	X25	X24	X23	X22	X21	X20	X19	X18	X17	X16	0000
CRCDATL	0648	CRCDATL																xxxx
CRCDATH	064A	CRCDATH																xxxx
CRCWDATL	064C	CRCWDATL																xxxx
CRCWDATH	064E	CRCWDATH																xxxx

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-22: 时钟控制寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
RCON	0740	TRAPR	IOPUWR	SBOREN	RETEN	—	DPSLP	CM	PMSLP	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	(注 1)
OSCCON	0742	—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0	CLKLOCK	—	LOCK	—	CF	SOSCDRV	SOSCEN	OSWEN	(注 2)
CLKDIV	0744	ROI	DOZE2	DOZE1	DOZE0	DOZEN	RCDIV2	RCDIV1	RCDIV0	—	—	—	—	—	—	—	—	3140
OSCTUN	0748	—	—	—	—	—	—	—	—	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	0000
REFOCON	074E	ROEN	—	ROSSLP	ROSEL	RODIV3	RODIV2	RODIV1	RODIV0	—	—	—	—	—	—	—	—	0000
HLVDCON	0756	HLVDEN	—	HLSIDL	—	—	—	—	—	VDIR	BGVST	IRVST	—	HLVDL3	HLVDL2	HLVDL1	HLVDL0	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

- 注 1: RCON 寄存器的复位值取决于复位类型。
 注 2: OSCCON 寄存器的复位值取决于配置位及复位类型。

表 4-23: 深度休眠寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
DSCON	0758	DSEN	—	—	—	—	—	—	RTCCWDIS	—	—	—	—	—	ULPWDIS	DSBOR	RELEASE	0000
DSWAKE	075A	—	—	—	—	—	—	—	DSINT0	DSFLT	—	—	DSWDT	DSRTCC	DSMCLR	—	DSPOR	0000
DSGPR0 ⁽¹⁾	075C	DSGPR0																0000
DSGPR1 ⁽¹⁾	075E	DSGPR1																0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

- 注 1: 深度休眠寄存器 DSGPR0 和 DSGPR1 仅在发生 V_{DD} POR 事件时复位。

表 4-24: NVM 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态 ⁽¹⁾
NVMCON	0760	WR	WREN	WRERR	PGMONLY	—	—	—	—	—	ERASE	NVMOP5	NVMOP4	NVMOP3	NVMOP2	NVMOP1	NVMOP0	0000
NVMKEY	0766	—	—	—	—	—	—	—	—	NVMKEY								0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。
注 1: 显示的复位值仅适用于上电复位。其他复位状态的值取决于复位时存储器写操作或擦除操作的状态。

表 4-25: 超低功耗唤醒寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ULPWCON	0768	ULPEN	—	ULPSIDL	—	—	—	—	ULPSINK	—	—	—	—	—	—	—	—	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

表 4-26: PMD 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	—	—	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	—	ADC1MD	0000
PMD2	0772	—	—	—	—	—	IC3MD	IC2MD	IC1MD	—	—	—	—	—	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	—	—	—	—	—	CMPMD	RTCCMD	—	CRCPMD	—	—	—	—	—	I2C2MD	—	0000
PMD4	0776	—	—	—	—	—	—	—	—	ULPWUMD	—	—	EEMD	REFOMD	CTMUMD	HLVDMD	—	0000

图注: — = 未实现, 读为 0。复位值以十六进制格式显示。

4.2.5 软件堆栈

除了用作工作寄存器外，PIC24F 器件中的 W15 寄存器也可用作软件堆栈指针。此指针总是指向第一个可用的空字，从低地址向高地址方向增长。它在弹出堆栈之前递减，而在压入堆栈后递增，如图 4-4 所示。

注意，对于执行任何 CALL 指令时的 PC 压栈操作，在压入堆栈之前，PC 的 MSB 要进行零扩展，从而确保 MSB 始终清零。

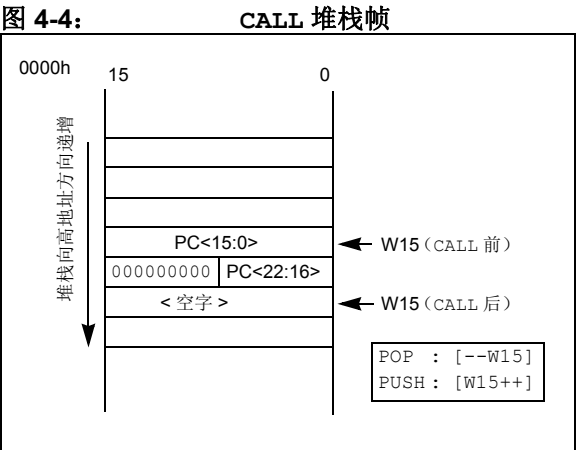
注： 在异常处理期间，在 PC 压入堆栈之前，要
先将 PC 的 MSB 与 SRL 寄存器组合在一起。

堆栈指针限制值寄存器（Stack Pointer Limit Value, SPLIM）与堆栈指针相关联，它设置堆栈上边界的地址。SPLIM 在复位时不会被初始化。与堆栈指针一样，SPLIM<0> 被强制为 0，因为所有的堆栈操作必须是字对齐的。每当使用 W15 作为源指针或目标指针产生有效地址时，有效地址会与 SPLIM 中的值进行比较。如果堆栈指针（W15）的内容与 SPLIM 寄存器的内容相等，则会执行压栈操作而不产生堆栈错误陷阱。但在随后的压栈操作时将会产生堆栈错误陷阱。

因此，例如如果想要在堆栈增长到超过 RAM 中的地址 0DF6 时产生堆栈错误陷阱，则需用值 0DF4 来初始化 SPLIM。

同理，当堆栈指针地址小于 0800h 时，将产生堆栈指针下溢（堆栈错误）陷阱。这可防止堆栈进入特殊功能寄存器（SFR）空间。

注： 在对 SPLIM 寄存器进行写操作后，不应紧
跟使用 W15 进行间接读操作的指令。



4.3 程序存储空间与数据存储空间的接口

PIC24F 架构采用 24 位宽的程序空间和 16 位宽的数据空间。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，PIC24F 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新映射到数据空间 PSV

表指令允许应用程序读写程序存储器中的一小块区域。这一功能对于访问需要随时更新的数据来说非常理想。也可通过表操作访问一个程序字的所有字节。重映射方式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进行查找。这一方式只能访问程序字的低位字（lsw）。

4.3.1 对程序空间进行寻址

由于数据和程序空间的地址范围分别为 16 位和 24 位，因此需要一个从 16 位数据寄存器创建一个 23 位或 24 位程序地址的方法。方法取决于所采用的接口方式。

对于表操作，使用 8 位的表存储器页地址寄存器（TBLPAG）定义程序空间中一个 32K 字的区域。TBLPAG 寄存器的 8 位与 16 位 EA 组合形成了一个完整的 24 位程序空间地址。在这种地址形式下，TBLPAG 的最高位（MSb）用来决定操作是发生在用户存储区（TBLPAG<7> = 0）中还是配置存储区（TBLPAG<7> = 1）中。

对于重映射操作，使用 8 位的程序空间可视性页地址寄存器（PSVPAG）定义程序空间中的 16K 字页。当 EA 的 MSb 为 1 时，PSVPAG 与低 15 位组合形成了 23 位程序空间地址。与表操作不同，重映射操作被严格限制在用户存储区中。

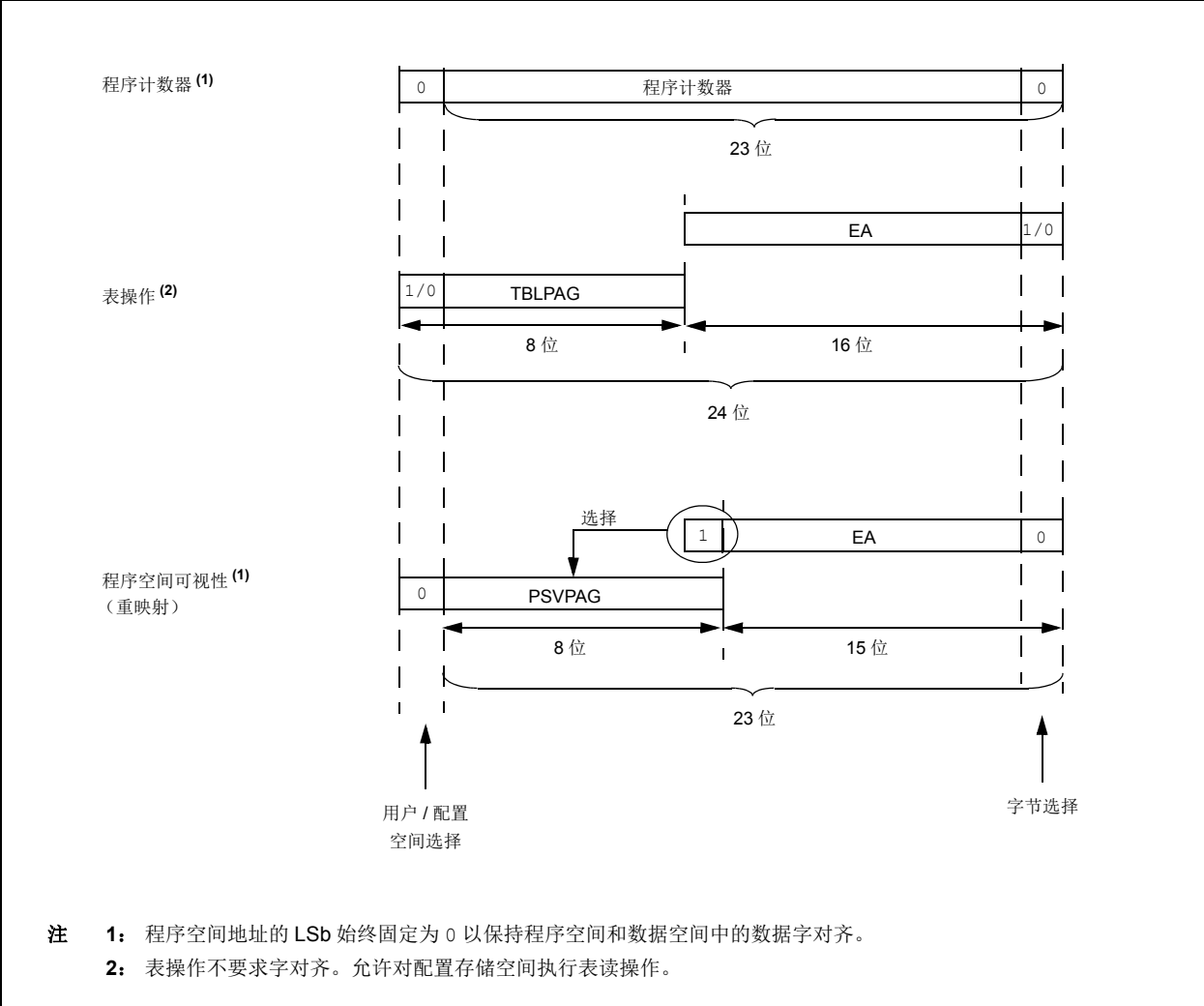
表 4-27 和图 4-5 显示了如何从数据 EA 创建程序 EA 来进行表操作和重映射访问。本文中 P<23:0> 指的是程序空间字，而 D<15:0> 指的是数据空间字。

表 4-27: 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>
通过指令访问 (代码执行)	用户	0	PC<22:1>			0
		0xx xxxx xxxx xxxx xxxx xxx0				
TBLRD/TBLWT (读 / 写字节或字)	用户	TBLPAG<7:0>		数据 EA<15:0>		
		0xxx xxxx		xxxx xxxx xxxx xxxx		
	配置	TBLPAG<7:0>		数据 EA<15:0>		
		1xxx xxxx		xxxx xxxx xxxx xxxx		
程序空间可视性 (块重映射 / 读)	用户	0	PSVPAG<7:0> ⁽²⁾		数据 EA<14:0> ⁽¹⁾	
		0	xxxx xxxx		xxx xxxx xxxx xxxx	

注 1: 在这种情况下, 数据 EA<15> 始终为 1, 但并不用它来计算程序空间地址。地址的 bit 15 为 PSVPAG<0>。
2: 在 PIC24FV32KA304 系列器件上, PSVPAG 只可以有两个值 (00 用于访问程序存储器, 而 FF 用于访问 EEPROM)。

图 4-5: 访问程序空间内的数据的地址生成方式



PIC24FV32KA304 系列

4.3.2 使用表指令访问程序存储器和数据 EEPROM 存储器中的数据

TBLRD L和TBLWT L指令提供了读或写程序空间内任何地址的低位字的直接方法，无需通过数据空间。它还提供了读或写数据 EEPROM 存储器中任何地址的字的直接方法。TBLRD H和TBLWT H指令是可以把一个程序空间字的高 8 位作为数据读写的惟一方法。

注： 访问数据 EEPROM 存储器时不使用
TBLRD H和TBLWT H指令。

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被直接映射到数据空间地址中。于是，程序存储器可以被看作是两个 16 位字宽的地址空间，它们并排放置，具有相同的地址范围。TBLRD L和TBLWT L访问包含数据低位字的空间，而TBLRD H和TBLWT H访问包含高数据字节的空间。

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

1. TBLRD L（表读低位字）：在字模式下，该指令将程序空间地址的低位字（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。

2. TBLRD H（表读高位字）：在字模式下，该指令将程序地址的整个高位字（P<23:16>）映射到数据地址中。注意，D<15:8>为“虚拟”字节，始终为 0。

在字节模式下，程序字的高字节或低字节被映射到数据地址的 D<7:0> 中，这和 TBLRD L指令相同。注意，当选择高位“虚拟”字节（字节选择位 = 1）时，数据将始终为 0。

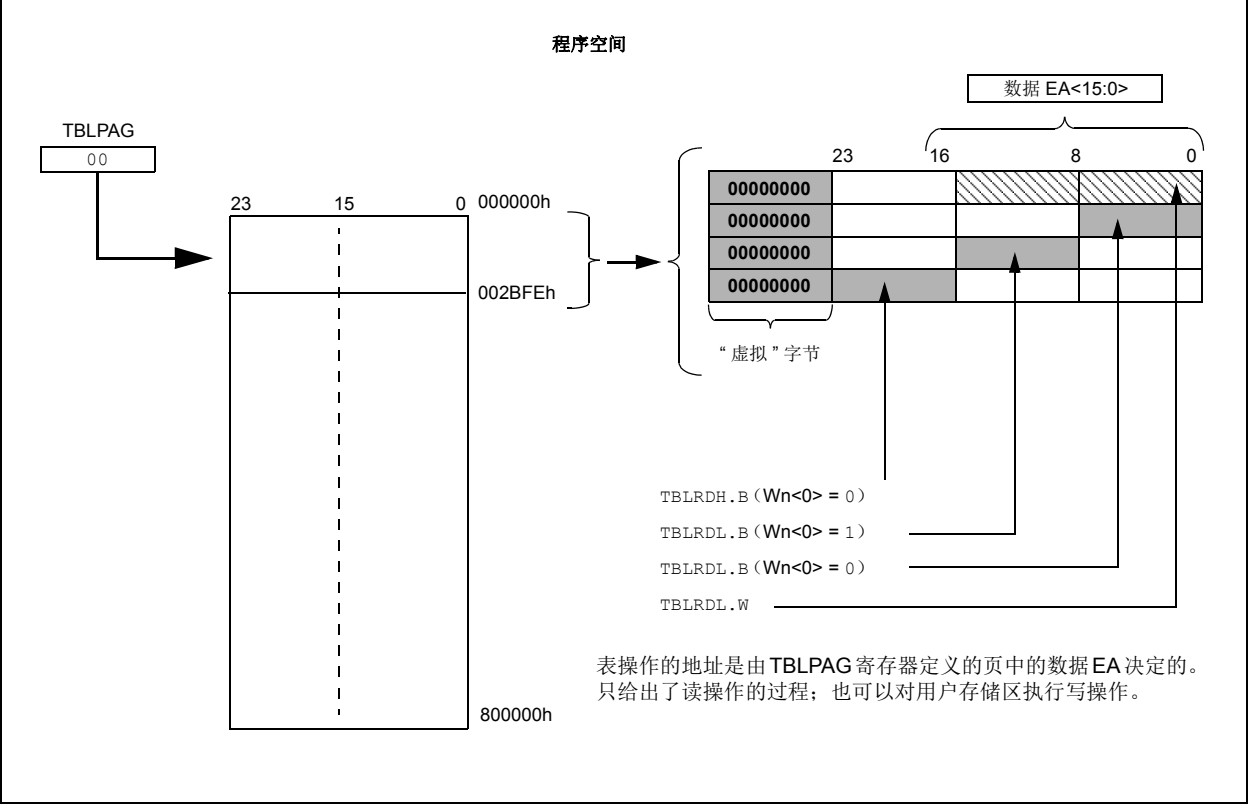
表指令 TBLWTH 和 TBLWTL 以类似的方式向程序地址空间写入各字节或字。**第 5.0 节“闪存程序存储器”**对这两条指令的详细操作给出了说明。

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存储器页地址寄存器（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，表页位于配置空间中。

注： 仅表读操作可在配置存储空间内执行，且只能在实现的区域中执行，如器件 ID 所在的空间。不允许表写操作。

图 4-6： 使用表指令访问程序存储器



4.3.3 使用程序空间可视性从程序存储器读数据

可选择将数据空间的高 32 KB 映射到程序空间中的任一 16K 字页 (PIC24FV16KA3XX 器件) 和 32K 字页 (PIC24FV32KA3XX 器件) 中。这提供了通过数据空间对存储的常量数据的透明访问, 而无需使用特殊指令 (即 TBLRD L/H 指令)。

当数据空间 EA 的 MSb 为 1 且通过将 CPU 控制寄存器中的 PSV 位 (CORCON<2>) 置 1 使能 PSV 时, 就可以通过数据空间访问程序空间。要映射到数据空间的程序存储空间的存储单元由程序空间可视性地址寄存器 (PSVPAG) 决定。该 8 位寄存器定义程序空间中 256 个可能的 16K 字页中的一个。实际上, PSVPAG 用作程序存储器地址的高 8 位, EA 的 15 位用作低位。

对于每个程序存储字, PC 都将递增 2, 数据空间地址的低 15 位直接映射到对应程序空间地址的低 15 位。

从该区域读取数据的指令, 需要增加一个额外的指令周期, 因为这类指令需要对程序存储器进行两次数据取操作。

尽管每个数据空间地址 (8000h 和更高地址) 直接映射到对应的程序存储器地址 (见图 4-7), 但只有 24 位程序字的低 16 位用于存储数据。任何用于存储数据的程序空间存储单元的高 8 位应编程为 1111 1111 或 0000 0000, 强制执行一条 NOP 指令, 从而避免了可能出现意外执行这一区域内代码的情况。

注: 在进行表读/写期间, 暂时禁止 PSV 访问。

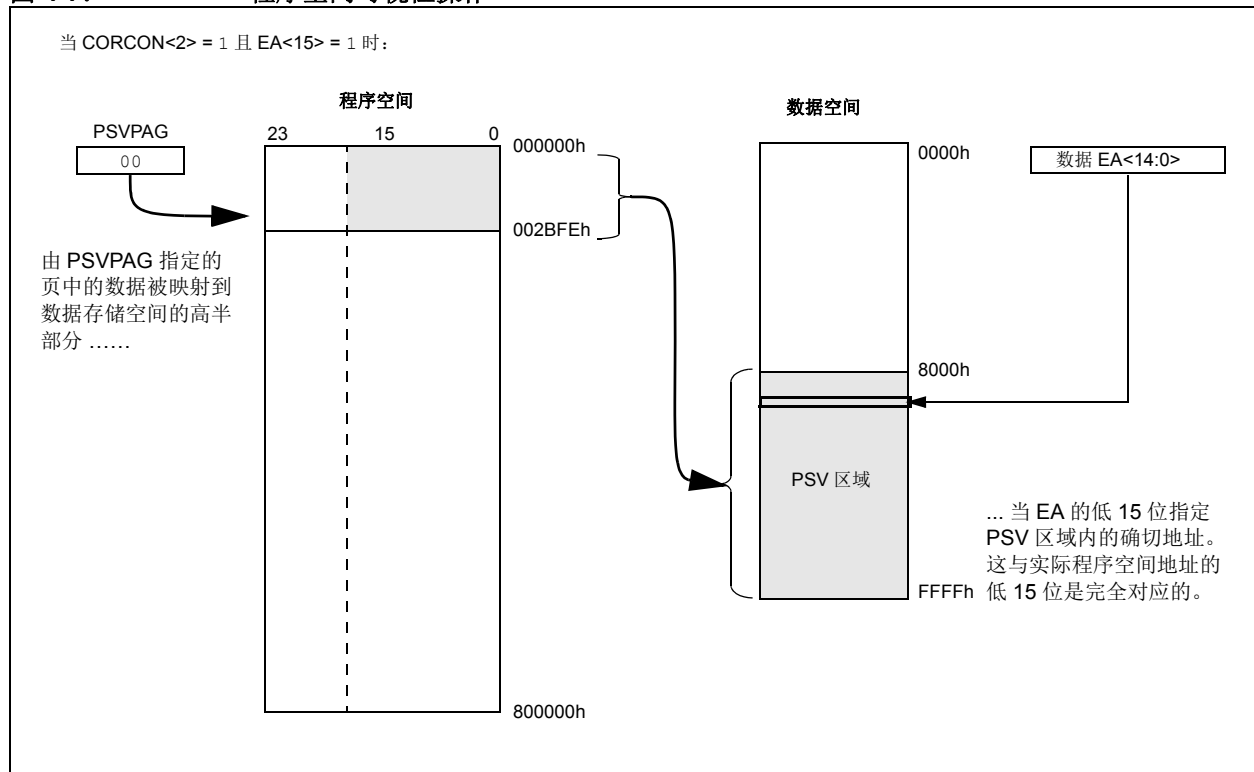
对于使用 PSV 而又在 REPEAT 循环之外执行的指令, MOV 和 MOV.D 指令除了规定的执行时间之外, 还需要一个额外的指令周期。其他所有的指令, 除了规定的指令执行时间之外, 还需要两个额外的指令周期。

对于使用 PSV 而又在 REPEAT 循环内执行的指令, 下列情况下, 除了规定的指令执行时间之外, 还需要两个额外的指令周期:

- 在第一次迭代中执行的指令
- 在最后一次迭代中执行的指令
- 由于中断而退出循环之前执行的指令
- 中断得到处理后再次进入循环时执行的指令

REPEAT 循环的所有其他各次迭代, 都允许使用 PSV 访问数据的指令在一个周期内执行。

图 4-7: 程序空间可视性操作



PIC24FV32KA304 系列

注:

5.0 闪存程序存储器

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解闪存编程的更多信息，请参见《PIC24F 系列参考手册》中的第 4 章 “程序存储器” (DS39715A_CN)。

PIC24FV32KA304 系列器件包含内部闪存程序存储器，用于存储和执行应用代码。在 VDD 大于 1.8V 的条件下工作时，该存储器是可读写并可擦除的。

闪存有以下三种编程方式：

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

ICSP 允许在最终应用电路中对 PIC24FV32KA304 器件进行串行编程。只需五根线就可以完成编程，分别是编程时钟线、编程数据线（分别命名为 PGCEx 和 PGEDx）、电源线（VDD）、接地线（VSS）和进入主复位 / 编程模式的控制电压线（MCLR/VPP）。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单片机进行编程，从而可以使用最新版本的固件或者定制固件进行编程。

运行时自编程（RTSP）是通过 TBLRD（表读）和 TBLWT（表写）指令完成的。使用 RTSP，用户可以一次写入一块 32 条指令（96 字节）的程序存储器数据，也可以一次擦除一块 32、64 和 128 条指令（96、192 和 384 字节）的程序存储区域。

NVMOP<1:0>（NVMCON<1:0>）位确定擦除块的大小。

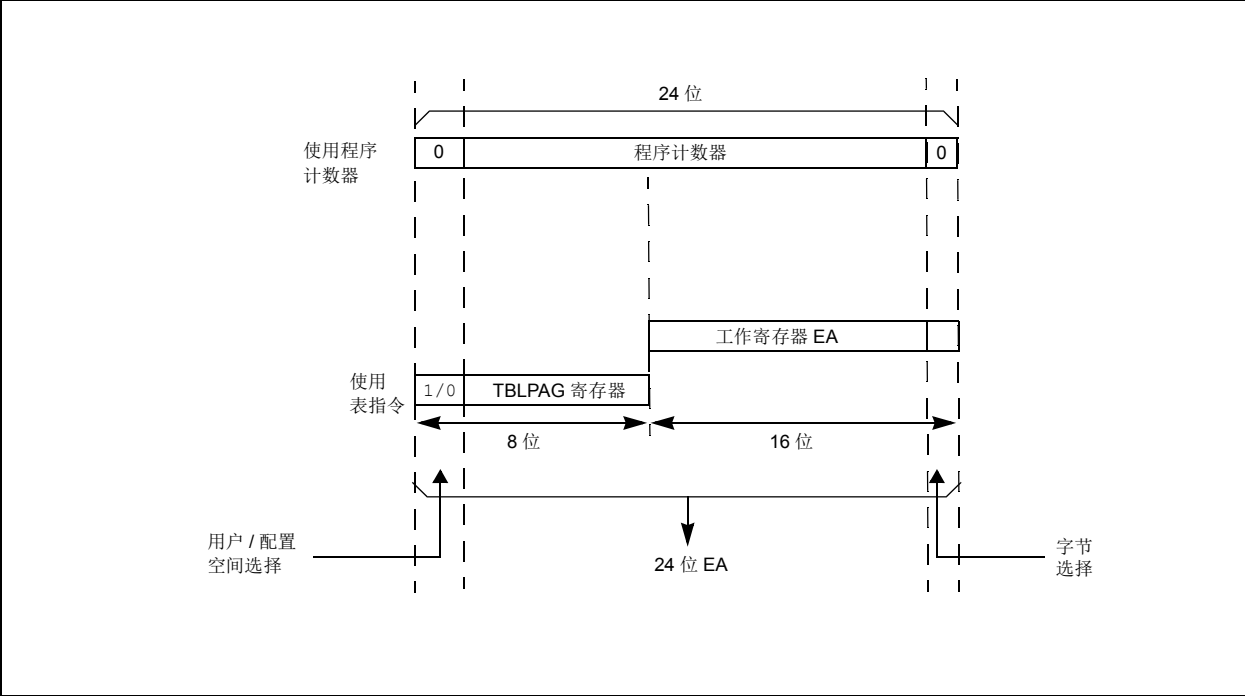
5.1 表指令和闪存编程

闪存编程是通过表读和表写指令完成的，与使用的编程方法无关。这些指令允许在器件正常工作模式下通过数据存储器直接读写程序存储空间。24 位程序存储器目标地址由 TBLPAG<7:0> 位和表指令中指定的 W 寄存器中的有效地址（Effective Address, 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 指令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 指令也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



PIC24FV32KA304 系列

5.2 RTSP 操作

PIC24F 闪存程序存储器阵列以 32 条指令或 96 字节的行构成。RTSP 允许用户一次擦除 1 行、2 行和 4 行（32、64 和 128 条指令）的块并一次编程一行，它还可以编程单字。

1 行（96 字节）、2 行（192 字节）和 4 行（384 字节）擦除块和单行写入块（96 字节）都是从程序存储器开始边沿对齐的。

用 TBLWT 指令将数据写入程序存储器时，数据并未直接写入存储器，而是将使用表写指令写入的数据存储在保持锁存器中，直到执行编程序列。

可以执行任意数量的 TBLWT 指令，且写操作都将成功执行。但是，需要 32 条 TBLWT 指令来写存储器的整行。

RTSP 编程的基本顺序是先建立一个表指针，然后执行一系列 TBLWT 指令以装入缓冲区。通过将 NVMCON 寄存器的控制位置 1 来执行编程。

可按任何顺序装入数据，且在执行写操作之前可以多次写保持寄存器。但后续写操作将覆盖之前的所有写操作写入的内容。

注： 不推荐多次写入一个存储单元而不擦除其内容。

因为只写缓冲区，因此所有的表写操作都是单字写操作（2 个指令周期）。编程每行都需要一个编程周期。

5.3 增强型在线串行编程

增强型在线串行编程使用片上自举程序（称为编程执行程序（Programming Executive, PE））管理编程过程。通过使用 SPI 数据帧格式，编程执行程序可以擦除、编程和校验程序存储器。如需了解更多有关增强型 ICSP 的信息，请参见器件编程规范。

5.4 控制寄存器

有两个用于读写闪存程序存储器的 SFR：NVMCON 和 NVMKEY。

NVMCON 寄存器（[寄存器 5-1](#)）控制要擦除的块、要编程的存储器类型以及编程周期的开始时间。

NVMKEY 是只写寄存器，用于写保护。要启动编程或擦除过程，用户必须把 55h 和 AAh 连续写入 NVMKEY 寄存器。更多详细信息，请参见[第 5.5 节“编程操作”](#)。

5.5 编程操作

在 RTSP 模式下，对内部闪存进行编程或擦除需要完整的编程序列。在编程或擦除操作期间，处理器将停止（等待），直到操作完成。将 WR 位（NVMCON<15>）置 1 启动操作，当操作完成时 WR 位会自动清零。

寄存器 5-1: NVMCON: 闪存控制寄存器

R/SO-0, HC	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	PGMONLY ⁽⁴⁾	—	—	—	—
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
—	ERASE	NVMOP5 ⁽¹⁾	NVMOP4 ⁽¹⁾	NVMOP3 ⁽¹⁾	NVMOP2 ⁽¹⁾	NVMOP1 ⁽¹⁾	NVMOP0 ⁽¹⁾
bit 7				bit 0			

图注:	SO = 只可置 1 位	HC = 可由硬件清零的位	
-n = 上电复位时的值	1 = 置 1	R = 可读位	W = 可写位
0 = 清零	x = 未知	U = 未实现位, 读为 0	

- bit 15 **WR:** 写控制位
1 = 启动闪存编程或擦除操作。操作是自定时的, 一旦操作完成此位即由硬件清零。
0 = 编程或擦除操作完成, 并处于停止状态
- bit 14 **WREN:** 写使能位
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
1 = 尝试执行错误的编程或擦除序列或发生终止 (任何试图将 WR 位置 1 的操作都会自动置 1 此位)
0 = 编程或擦除操作正常完成
- bit 12 **PGMONLY:** 仅编程使能位 ⁽⁴⁾
- bit 11-7 **未实现:** 读为 0
- bit 6 **ERASE:** 擦除 / 编程使能位
1 = 在下一个 WR 命令执行由 NVMOP<5:0> 指定的擦除操作
0 = 在下一个 WR 命令执行由 NVMOP<5:0> 指定的编程操作
- bit 5-0 **NVMOP<5:0>:** 编程操作命令字节位 ⁽¹⁾
擦除操作 (ERASE 位为 1 时):
1010xx = 擦除整个引导块 (包括代码保护的引导块) ⁽²⁾
1001xx = 擦除整个存储器 (包括引导块、配置块和通用块) ⁽²⁾
011010 = 擦除闪存的 4 行 ⁽³⁾
011001 = 擦除闪存的 2 行 ⁽³⁾
011000 = 擦除闪存的 1 行 ⁽³⁾
0101xx = 擦除整个配置块 (代码保护位除外)
0100xx = 擦除整个数据 EEPROM ⁽⁴⁾
0011xx = 擦除整个通用存储器块
编程操作 (当 ERASE 位为 0 时)
0001xx = 写闪存的 1 行 ⁽³⁾

- 注** 1: NVMOP<5:0> 的所有其他组合均无操作。
2: 这些值仅在 ICSP™ 模式下可用。请参见器件编程规范。
3: 表指针中的地址确定将擦除的行。
4: 仅当访问数据 EEPROM 时才使用此位。

PIC24FV32KA304 系列

5.5.1 闪存程序存储器的编程算法

用户可通过擦除可编程行一次编程闪存程序存储器的一行。一般过程如下：

1. 读程序存储器中的 1 行（32 条指令），并将其存储在数据 RAM 中。
2. 用需要的新数据更新 RAM 中对应的程序数据。
3. 擦除行（见例 5-1）：
 - a) 将 NVMOPx 位（NVMCON<5:0>）设置为 011000，以配置为行擦除操作。将 ERASE（NVMCON<6>）和 WREN（NVMCON<14>）位置 1。
 - b) 将要擦除块的起始地址写入 TBLPAG 和 W 寄存器。
 - c) 将 55h 写入 NVMKEY。
 - d) 将 AAh 写入 NVMKEY。
 - e) 将 WR 位（NVMCON<15>）置 1。启动擦除周期，CPU 停止工作等待擦除周期结束。擦除操作完成时，WR 位自动清零。

4. 将数据 RAM 的前 32 条指令写入程序存储器缓冲区（见例 5-1）。
5. 将程序块写入闪存：
 - a) 将 NVMOPx 位设置为 011000，以配置为行编程操作。将 ERASE 位清零，将 WREN 位置 1。
 - b) 将 55h 写入 NVMKEY。
 - c) 将 AAh 写入 NVMKEY。
 - d) 将 WR 位置 1。启动编程周期，CPU 停止工作等待写周期完成。当写闪存的操作完成时，WR 位自动清零。

为了防止意外操作，必须向 NVMKEY 写入启动序列从而允许执行任何擦除或编程操作。执行了编程命令以后，用户必须等待一定的编程时间，直到编程操作完成。紧跟编程启动序列的两条指令必须为 NOP，如例 5-5 所示。

例 5-1: 擦除程序存储器的一行 —— 汇编语言代码

```
; Set up NVMCON for row erase operation
MOV    #0x4058, W0                ;
MOV    W0, NVMCON                 ; Initialize NVMCON
; Init pointer to row to be ERASED
MOV    #tblpage(PROG_ADDR), W0    ;
MOV    W0, TBLPAG                 ; Initialize PM Page Boundary SFR
MOV    #tbloffset(PROG_ADDR), W0  ; Initialize in-page EA[15:0] pointer
TBLWTL W0, [W0]                   ; Set base address of erase block
DISI    #5                         ; Block all interrupts
                                   ; for next 5 instructions

MOV    #0x55, W0
MOV    W0, NVMKEY                 ; Write the 55 key
MOV    #0xAA, W1
MOV    W1, NVMKEY                 ; Write the AA key
BSET   NVMCON, #WR                ; Start the erase sequence
NOP                                         ; Insert two NOPs after the erase
NOP                                         ; command is asserted
```

例 5-2: 擦除程序存储器的一行 —— C 语言代码

```
// C example using MPLAB C30

int __attribute__((space(auto_psv))) progAddr = 0x1234; // Global variable located in Pgm Memory
unsigned int offset;

//Set up pointer to the first memory location to be written

TBLPAG = __builtin_tblpage(&progAddr);           // Initialize PM Page Boundary SFR
offset = __builtin_tbloffset(&progAddr);          // Initialize lower word of address

__builtin_tblwtl(offset, 0x0000);                // Set base address of erase block
                                                // with dummy latch write

NVMCON = 0x4058;                                // Initialize NVMCON

asm("DISI #5");                                  // Block all interrupts for next 5
                                                // instructions
__builtin_write_NVM();                           // C30 function to perform unlock
                                                // sequence and set WR
```

例 5-3: 装载写缓冲区 —— 汇编语言代码

```
; Set up NVMCON for row programming operations
MOV    #0x4004, W0
MOV     W0, NVMCON                               ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
MOV     #0x0000, W0
MOV     W0, TBLPAG                               ; Initialize PM Page Boundary SFR
MOV     #0x6000, W0                               ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
MOV     #LOW_WORD_0, W2
MOV     #HIGH_BYTE_0, W3
TBLWTL W2, [W0]                                  ; Write PM low word into program latch
TBLWTH W3, [W0++]                                ; Write PM high byte into program latch
; 1st_program_word
MOV     #LOW_WORD_1, W2
MOV     #HIGH_BYTE_1, W3
TBLWTL W2, [W0]                                  ; Write PM low word into program latch
TBLWTH W3, [W0++]                                ; Write PM high byte into program latch
; 2nd_program_word
MOV     #LOW_WORD_2, W2
MOV     #HIGH_BYTE_2, W3
TBLWTL W2, [W0]                                  ; Write PM low word into program latch
TBLWTH W3, [W0++]                                ; Write PM high byte into program latch
.
.
; 32nd_program_word
MOV     #LOW_WORD_31, W2
MOV     #HIGH_BYTE_31, W3
TBLWTL W2, [W0]                                  ; Write PM low word into program latch
TBLWTH W3, [W0]                                  ; Write PM high byte into program latch
```

PIC24FV32KA304 系列

例 5-4: 装载写缓冲区 ——C 语言代码

```
// C example using MPLAB C30

#define NUM_INSTRUCTION_PER_ROW 64
int __attribute__((space(auto_psv))) progAddr = 0x1234; // Global variable located in Pgm Memory
unsigned int offset;
unsigned int i;
unsigned int progData[2*NUM_INSTRUCTION_PER_ROW]; // Buffer of data to write

//Set up NVMCON for row programming
NVMCON = 0x4001; // Initialize NVMCON

//Set up pointer to the first memory location to be written
TBLPAG = __builtin_tblpage(&progAddr); // Initialize PM Page Boundary SFR
offset = __builtin_tbloffset(&progAddr); // Initialize lower word of address

//Perform TBLWT instructions to write necessary number of latches
for(i=0; i < 2*NUM_INSTRUCTION_PER_ROW; i++)
{
    __builtin_tblwtl(offset, progData[i++]); // Write to address low word
    __builtin_tblwth(offset, progData[i]); // Write to upper byte
    offset = offset + 2; // Increment address
}
```

例 5-5: 启动编程序列 —— 汇编语言代码

```
DISI    #5                ; Block all interrupts
                        ; for next 5 instructions

MOV     #0x55, W0
MOV     W0, NVMKEY        ; Write the 55 key
MOV     #0xAA, W1
MOV     W1, NVMKEY        ; Write the AA key
BSET    NVMCON, #WR       ; Start the erase sequence
NOP     ; 2 NOPs required after setting WR
NOP     ;
BTSC    NVMCON, #15       ; Wait for the sequence to be completed
BRA     $-2               ;
```

例 5-6: 启动编程序列 ——C 语言代码

```
// C example using MPLAB C30

asm("DISI #5"); // Block all interrupts for next 5 instructions

__builtin_write_NVM(); // Perform unlock sequence and set WR
```

6.0 数据 EEPROM 存储器

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解数据 EEPROM 的更多信息，请参见《PIC24F 系列参考手册》中的第 5 章“数据 EEPROM”（DS39720A_CN）。

与程序存储器和易失性数据 RAM 不同，数据 EEPROM 存储器是一种非易失性存储器（Nonvolatile Memory，NVM）。数据 EEPROM 存储器基于与程序存储器相同的闪存技术，并经过优化，可实现长期数据保存和更高的耐擦写次数。

数据 EEPROM 映射到用户程序存储空间的顶部，最高地址位于程序存储器地址 7FFE00h 至 7FFFFFFh 处。PIC24FV32KA304 系列器件中的数据 EEPROM 大小为 256 字。

数据 EEPROM 按 16 位宽存储器进行组织。在整个 V_{DD} 范围内的正常工作期间，每个字都可直接寻址，并且是可读可写的。

不同于闪存程序存储器，在数据 EEPROM 编程或擦除操作期间，不会停止正常的程序执行。

数据 EEPROM 的编程操作使用 3 个 NVM 控制寄存器进行控制：

- NVMCON：非易失性存储器控制寄存器
- NVMKEY：非易失性存储器密钥寄存器
- NVMADR：非易失性存储器地址寄存器

例 6-1： 数据 EEPROM 解锁序列

```
//Disable Interrupts For 5 instructions
asm volatile ("disi #5");
//Issue Unlock Sequence
asm volatile ("mov #0x55, W0      \n"
              "mov W0, NVMKEY     \n"
              "mov #0xAA, W1      \n"
              "mov W1, NVMKEY     \n");
// Perform Write/Erase operations
asm volatile ("bset NVMCON, #WR   \n"
              "nop                 \n"
              "nop                 \n");
```

6.1 NVMCON 寄存器

NVMCON 寄存器（寄存器 6-1）还是数据 EEPROM 编程 / 擦除操作的主控制寄存器。高字节包含用于启动编程或擦除周期的控制位，以及用于指示操作是否成功执行的标志位。NVMCON 的低字节用于配置将执行的 NVM 操作类型。

6.2 NVMKEY 寄存器

NVMKEY 是一个只写寄存器，用于防止数据 EEPROM 存储单元的误写或误擦除。

要开始任何编程或擦除序列，必须严格按顺序先执行以下两条指令：

1. 将 55h 写入 NVMKEY。
2. 将 AAh 写入 NVMKEY。

在此序列后，就可以在一个指令周期中写入 NVMCON 寄存器。在多数情况下，用户只需要将 NVMCON 寄存器中的 WR 位置 1 就可以开始编程或擦除周期。在解锁序列中应禁止中断。

MPLAB® C30 C 编译器提供了一个已定义的库过程（builtin_write_nvm）用来执行解锁序列。例 6-1 显示了如何使用行内汇编来执行解锁序列。

PIC24FV32KA304 系列

寄存器 6-1: NVMCON: 非易失性存储器控制寄存器

R/S-0, HC	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	PGMONLY	—	—	—	—
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
—	ERASE	NVMOP5	NVMOP4	NVMOP3	NVMOP2	NVMOP1	NVMOP0
bit 7							bit 0

图注:	HC = 可由硬件清零的位	U = 未实现位, 读为 0
R = 可读位	W = 可写位	S = 可置 1 位
-n = 上电复位时的值	1 = 置 1	0 = 清零 x = 未知

- bit 15

WR: 写 (编程或擦除) 控制位
1 = 启动数据 EEPROM 的擦除或写周期 (用软件只能将该位置 1, 但不能清零)
0 = 写周期完成 (由硬件自动清零)
- bit 14

WREN: 写 (编程或擦除) 使能位
1 = 使能擦除或编程操作
0 = 不允许任何操作 (在写 / 擦除操作完成时器件会将该位清零)
- bit 13

WRERR: 写闪存错误标志位
1 = 写操作过早终止 (编程操作期间的任何 $\overline{\text{MCLR}}$ 复位或 WDT 复位)
0 = 写操作成功完成
- bit 12

PGMONLY: 仅编程使能位
1 = 不先擦除目标地址内容即执行写操作
0 = 自动写前擦除。
在执行写操作之前, 自动对目标地址进行擦除操作。
- bit 11-7

未实现: 读为 0
- bit 6

ERASE: 擦除操作选择位
1 = WR 置 1 时执行擦除操作
0 = WR 置 1 时执行写操作
- bit 5-0

NVMOP<5:0>: 编程操作命令字节位
擦除操作 (当 ERASE 位为 1 时):
011010 = 擦除 8 个字
011001 = 擦除 4 个字
011000 = 擦除 1 个字
0100xx = 擦除整个数据 EEPROM
擦除操作 (当 ERASE 位为 0 时):
0010xx = 写 1 个字

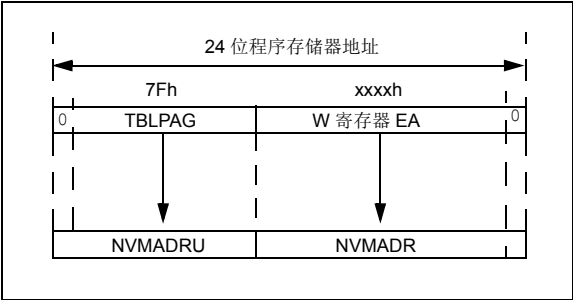
6.3 NVM 地址寄存器

类似于闪存程序存储器，NVM 地址寄存器 NVMADRU 和 NVMADR 构成进行数据 EEPROM 操作的选定行或字的 24 位有效地址（Effective Address, 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。这两个寄存器不会被映射到特殊功能寄存器（SFR）空间。相反，它们直接捕捉已执行的上一条表写指令的 EA<23:0>，并选择要擦除的数据 EEPROM 行。图 6-1 显示了用于编程和擦除操作的程序存储器 EA 是如何构成的。

类似于程序存储器操作，NVMADR 的最低有效位（Least Significant bit, LSb）被限制到偶地址。这是因为数据 EEPROM 空间中的任何给定地址仅包含程序存储器宽度的低位字；高位字（包括最高位的“虚拟字节”）是不可用的。这意味着数据 EEPROM 地址的 LSb 将始终为 0。

类似地，NVMADRU 的最高有效位（Most Significant bit, MSb）始终为 0，因为所有地址都位于用户程序空间中。

图 6-1：使用 TBLPAG 和 NVM 地址寄存器对数据 EEPROM 进行寻址



6.4 数据 EEPROM 操作

EEPROM 块使用类似于用于程序存储器的表读和表写操作进行访问。对于数据 EEPROM 操作，不需要 TBLWTH 和 TBLRDH 指令，因为存储器只有 16 位宽（仅低地址上的数据有效）。对于数据 EEPROM，可以执行以下编程操作：

- 擦除 1、4 或 8 个字
- 批量擦除整个数据 EEPROM
- 写 1 个字
- 读 1 个字

注 1: 如果在编程或擦除操作正在进行时，用户尝试读 EEPROM，则会产生意外的结果。

2: C30 C 编译器包含了一些库过程，用于自动执行表读和表写操作，管理表指针和写缓冲区，以及解锁和启动存储器写序列。这使用户不必用 C 语言为每个应用程序创建汇编器宏或时间要求严格的程序。

在以下几节详细说明了的代码示例中使用了这些库过程。对于未使用 C30 编译器库的用户，以下提供了每个过程的大致说明。

PIC24FV32KA304 系列

6.4.1 擦除数据 EEPROM

数据 EEPROM 可完全擦除或部分擦除，有三种不同大小：1 个字、4 个字或 8 个字。NVMOP<1:0> (NVMCON<1:0>) 位确定要擦除的字数。要部分擦除数据 EEPROM，必须按照以下序列：

1. 将 NVMCON 配置为擦除所需的字数：1、4 或 8。
2. 将要擦除的 EEPROM 地址装入 TBLPAG 和 WREG。
3. 清零 NVMIF 状态位并允许 NVM 中断（可选）。
4. 将密钥序列写入 NVMKEY。
5. 将 WR 位置 1 以开始擦除周期。
6. 查询 WR 位或等待 NVM 中断（置 1 NVMIF）。

例 6-2 给出了典型的擦除序列。该示例显示了如何执行一个字擦除。同样，也可实现擦除 4 个字和 8 个字。该示例使用 C 库过程来管理表指针 (builtin_tblpage 和 builtin_tbloffset) 和 擦除页指针 (builtin_tblwtl)。存储器解锁序列 (builtin_write_NVM) 还会将 WR 位置 1 以启动操作，并在完成时返回控制。

例 6-2: 单字擦除

```
int __attribute__((space(eedata))) eeData = 0x1234;
/*
-----
The variable eeData must be a Global variable declared outside of any method
the code following this comment can be written inside the method that will execute the erase
-----
*/
    unsigned int offset;

    // Set up NVMCON to erase one word of data EEPROM
    NVMCON = 0x4058;

    // Set up a pointer to the EEPROM location to be erased
    TBLPAG = __builtin_tblpage(&eeData);           // Initialize EE Data page pointer
    offset = __builtin_tbloffset(&eeData);          // Initilize lower word of address
    __builtin_tblwtl(offset, 0);                    // Write EEPROM data to write latch

    asm volatile ("disi #5");                       // Disable Interrupts For 5 Instructions
    __builtin_write_NVM();                          // Issue Unlock Sequence & Start Write Cycle
    while (NVMCONbits.WR==1);                      // Optional: Poll WR bit to wait for
                                                    // write sequence to complete
```

6.4.1.1 数据 EEPROM 批量擦除

要擦除整个数据 EEPROM（批量擦除），因为该操作会影响整个数据 EEPROM，所以不需要配置地址寄存器。以下序列帮助执行批量擦除：

1. 配置 NVMCON 为批量擦除模式。
2. 清零 NVMIF 状态位并允许 NVM 中断（可选）。
3. 将密钥序列写入 NVMKEY。
4. 将 WR 位置 1 以开始擦除周期。
5. 查询 WR 位或等待 NVM 中断（置 1 NVMIF）。

例 6-3 给出了典型的批量擦除序列。

6.4.2 单字写

要将单字写入数据 EEPROM，必须遵循以下序列：

1. 将 PGMONLY 位（NVMCON<12>）设置为 1 以擦除一个数据 EEPROM 字（正如上一节中提到的）。
2. 将数据字写入数据 EEPROM 锁存器。
3. 将数据字编程到 EEPROM 中：
 - 配置 NVMCON 寄存器以编程 EEPROM 的一个字（NVMCON<5:0> = 0001xx）。
 - 清零 NVMIF 状态位并允许 NVM 中断（可选）。
 - 将密钥序列写入 NVMKEY。
 - 将 WR 位置 1 以开始擦除周期。
 - 查询 WR 位或等待 NVM 中断（置 1 NVMIF）。
 - 要清零，需要等待直到 NVMIF 置 1。

例 6-4 给出了典型的单字写序列。

例 6-3: 数据 EEPROM 批量擦除

```
// Set up NVMCON to bulk erase the data EEPROM
NVMCON = 0x4050;

// Disable Interrupts For 5 Instructions
asm volatile ("disi #5");

// Issue Unlock Sequence and Start Erase Cycle
__builtin_write_NVM();
```

例 6-4: 将单个字写入数据 EEPROM

```
int __attribute__((space(eedata))) eeData = 0x1234;
int newData; // New data to write to EEPROM
/*-----
The variable eeData must be a Global variable declared outside of any method

the code following this comment can be written inside the method that will execute the write
-----*/
unsigned int offset;

// Set up NVMCON to erase one word of data EEPROM
NVMCON = 0x4004;

// Set up a pointer to the EEPROM location to be erased
TBLPAG = __builtin_tblpage(&eeData); // Initialize EE Data page pointer
offset = __builtin_tbloffset(&eeData); // Initizlize lower word of address
__builtin_tblwtl(offset, newData); // Write EEPROM data to write latch

asm volatile ("disi #5"); // Disable Interrupts For 5 Instructions
__builtin_write_NVM(); // Issue Unlock Sequence & Start Write Cycle
while (NVMCONbits.WR=1); // Optional: Poll WR bit to wait for
// write sequence to complete
```

PIC24FV32KA304 系列

6.4.3 读数据 EEPROM

可使用表读指令从数据 EEPROM 读一个字。因为 EEPROM 阵列只有 16 位宽，所以只需要 TBLRDL 指令。通过将 EEPROM 地址单元的地址装入 TBLPAG 和 WREG，后跟 TBLRDL 指令，来执行读操作。

例 6-5 给出的典型读序列使用来自 C30 编译器库中的表指针管理 (`builtin_tblpage` 和 `builtin_tbloffset`) 和 表 读 过 程 (`builtin_tblrdl`)。

程序空间可视性 (PSV) 也可以用于读取数据 EEPROM 中的地址单元。

例 6-5: 使用 TBLRD 命令读取数据 EEPROM

```
int __attribute__((space(eedata))) eeData = 0x1234;
int data;                                     // Data read from EEPROM
/*-----
The variable eeData must be a Global variable declared outside of any method

the code following this comment can be written inside the method that will execute the read
-----
*/
    unsigned int offset;

    // Set up a pointer to the EEPROM location to be erased
    TBLPAG = __builtin_tblpage(&eeData);           // Initialize EE Data page pointer
    offset = __builtin_tbloffset(&eeData);          // Initizlize lower word of address
    data = __builtin_tblrdl(offset);                 // Write EEPROM data to write latch
```

7.0 复位

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。有关复位的更多信息，请参见《PIC24F 系列参考手册》，第 40 章“包括可编程欠压复位在内的复位机制”（DS39728A_CN）。

复位模块将所有的复位源组合在一起并控制器件主复位信号 **SYSRST**。以下所列是器件的复位源：

- **POR：** 上电复位
- **MCLR：** 引脚复位
- **SWR：** RESET 指令
- **WDTR：** 看门狗定时器复位
- **BOR：** 欠压复位
- 低功耗 BOR/ 深度休眠 BOR
- **TRAPR：** 陷阱冲突复位
- **IOPUWR：** 非法操作码复位
- **UWR：** 未初始化的 W 寄存器复位

复位模块的简化框图如图 7-1 所示。

任何激活的复位源都会激活 **SYSRST** 信号。许多与 CPU 和外设有关的寄存器被强制为一个已知的复位状态。大多数寄存器不受复位的影响；在上电复位（POR）时它们的状态未知，而在所有其他复位时它们的状态不变。

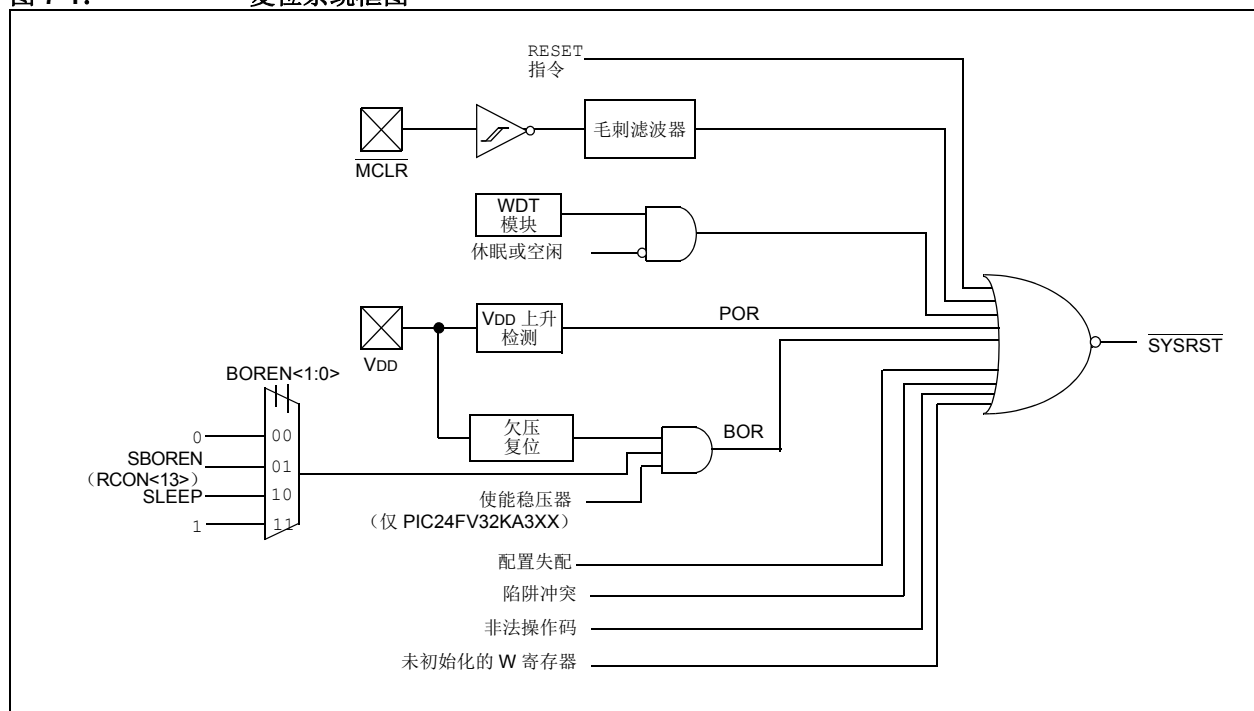
注： 有关寄存器复位状态的信息，请参见本数据手册中特定外设的章节或第 3.0 节“CPU”。

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 以表明复位类型（见寄存器 7-1）。上电复位会清零除 BOR 和 POR 位（**RCON**<1:0>）外的所有位，而 BOR 和 POR 位会被置 1。用户可在代码执行过程中的任何时间置 1 或清零任何位。**RCON** 位仅作为状态位。用软件将特定的复位状态位置 1 不会导致器件发生复位。

RCON 寄存器也有一些与看门狗定时器（WDT）和器件节能状态相关的位。本数据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，这样下一次器件复位后的 **RCON** 寄存器值才会有意义。

图 7-1： 复位系统框图



PIC24FV32KA304 系列

寄存器 7-1: **RCON: 复位控制寄存器⁽¹⁾**

R/W-0, HS	R/W-0, HS	R/W-0	R/W-0	U-0	R/C-0, HS	R/W-0	R/W-0
TRAPR	IOPUWR	SBOREN	RETEN ⁽³⁾	—	DPSLP	CM	PMSLP
bit 15							bit 8

R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-1, HS	R/W-1, HS
EXTR	SWR	SWDTEN ⁽²⁾	WDTO	SLEEP	IDLE	BOR	POR
bit 7							bit 0

图注:		C = 可清零位	HS = 可由硬件置 1 的位
R = 可读位		W = 可写位	U = 未实现位, 读为 0
-n = 上电复位时的值		1 = 置 1	0 = 清零
			x = 未知

- bit 15 **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 寄存器的复位标志位
1 = 检测到非法操作码、非法地址模式或未初始化的 W 寄存器用作地址指针而导致的复位
0 = 未发生由非法操作码或未初始化的 W 寄存器而导致的复位
- bit 13 **SBOREN:** 软件使能 / 禁止 BOR 位
1 = BOR 位由软件开启
0 = BOR 位由软件关闭
- bit 12 **RETEN:** 保持休眠模式控制位⁽³⁾
1 = 休眠模式下仅由保持稳压器 (RETEN) 提供稳压电源
0 = 休眠模式下由主稳压器 (VREG) 提供稳压电源
- bit 11 **未实现:** 读为 0
- bit 10 **DPSLP:** 深度休眠模式标志位
1 = 发生深度休眠
0 = 未发生深度休眠
- bit 9 **CM:** 配置字失配复位标志位
1 = 发生了配置字失配复位
0 = 未发生配置字失配复位
- bit 8 **PMSLP:** 休眠模式下的程序存储器电源位
1 = 休眠模式下程序存储器偏置电压仍然供电
0 = 休眠模式下程序存储器偏置电压掉电, 而稳压器进入待机模式
- bit 7 **EXTR:** 外部复位 (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:** 软件复位 (指令) 标志位
1 = 执行了 RESET 指令
0 = 未执行 RESET 指令

- 注** 1: 所有复位状态位都可以用软件置 1 或清零。用软件将任何一位置 1 都不会导致器件复位。
- 2: 如果 FWDTENx 配置位为 1 (未编程), 则 WDT 总是使能, 而与 SWDTEN 位的设置无关。
- 3: 仅在 PIC24FV32KA3XX 器件上实现, 不能用于 PIC24F32KA3XX 器件。

寄存器 7-1: RCON: 复位控制寄存器⁽¹⁾ (续)

bit 5	SWDTEN: 软件使能 / 禁止 WDT 位 ⁽²⁾ 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
bit 3	SLEEP: 从休眠状态唤醒标志位 1 = 器件处于休眠模式 0 = 器件未处于休眠模式
bit 2	IDLE: 从空闲状态唤醒标志位 1 = 器件处于空闲模式 0 = 器件未处于空闲模式
bit 1	BOR: 欠压复位标志位 1 = 发生欠压复位 (BOR 在 POR 也后置 1) 0 = 未发生欠压复位
bit 0	POR: 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。用软件将任何一位置 1 都不会导致器件复位。
 2: 如果 FWDTENx 配置位为 1 (未编程), 则 WDT 总是使能, 而与 SWDTEN 位的设置无关。
 3: 仅在 PIC24FV32KA3XX 器件上实现, 不能用于 PIC24F32KA3XX 器件。

表 7-1: 复位标志位操作

标志位	置 1 事件	清零事件
TRAPR (RCON<15>)	陷阱冲突事件	POR
IOPUWR (RCON<14>)	非法操作码或访问未初始化的 W 寄存器	POR
CM (RCON<9>)	配置失配复位	POR
EXTR (RCON<7>)	$\overline{\text{MCLR}}$ 复位	POR
SWR (RCON<6>)	RESET 指令	POR
WDTO (RCON<4>)	WDT 超时	PWRSV 指令, POR
SLEEP (RCON<3>)	PWRSV #SLEEP 指令	POR
IDLE (RCON<2>)	PWRSV #IDLE 指令	POR
BOR (RCON<1>)	POR 和 BOR	—
POR (RCON<0>)	POR	—
DPSLP (RCON<10>)	PWRSV #SLEEP 指令, DSEN (DSCON<15>) 置 1	POR

注: 所有复位标志位可由用户软件置 1 或清零。

PIC24FV32KA304 系列

7.1 复位时的时钟源选择

如果使能了时钟切换，器件复位时的系统时钟源可按照表 7-2 进行选择。如果禁止时钟切换，则总是根据振荡器配置位选择系统时钟源。更多信息，请参见第 9.0 节“振荡器配置”。

表 7-2: 振荡器选择与复位类型的关系 (使能时钟切换)

复位类型	时钟源的确定
POR	FNOSC _x 配置位 (FNOSC<10:8>)
BOR	
MCLR	COSC _x 控制位 (OSCCON<14:12>)
WDTO	
SWR	

7.2 器件复位时间

表 7-3 总结了各种类型的器件复位所需的时间。注意，在 POR 和 PWRT 延时结束后，会发出系统复位信号 $\overline{\text{SYSRST}}$ 。

器件实际开始执行代码的时间还取决于系统振荡器延时，它包括振荡器起振定时器 (OST) 延时和 PLL 锁定时间。OST 和 PLL 锁定时间与相应的 $\overline{\text{SYSRST}}$ 延时同时发生。

FSCM 延时决定在 $\overline{\text{SYSRST}}$ 信号发出后 FSCM 开始监视系统时钟源的时间。

表 7-3: 各种器件复位的延时

复位类型	时钟源	$\overline{\text{SYSRST}}$ 延时	系统时钟延时	备注
POR ⁽⁶⁾	EC	TPOR + TPWRT	—	1, 2
	FRC 和 FRCDIV	TPOR + TPWRT	TFRC	1, 2, 3
	LPRC	TPOR + TPWRT	TLPRC	1, 2, 3
	ECPLL	TPOR + TPWRT	TLOCK	1, 2, 4
	FRCPLL	TPOR + TPWRT	TFRC + TLOCK	1, 2, 3, 4
	XT、HS 和 SOSC	TPOR + TPWRT	TOST	1, 2, 5
	XTPLL 和 HSPLL	TPOR + TPWRT	TOST + TLOCK	1, 2, 4, 5
BOR	EC	TPWRT	—	2
	FRC 和 FRCDIV	TPWRT	TFRC	2, 3
	LPRC	TPWRT	TLPRC	2, 3
	ECPLL	TPWRT	TLOCK	2, 4
	FRCPLL	TPWRT	TFRC + TLOCK	2, 3, 4
	XT、HS 和 SOSC	TPWRT	TOST	2, 5
	XTPLL 和 HSPLL	TPWRT	TFRC + TLOCK	2, 3, 4
所有其他	任何时钟	—	—	无

- 注
- 1: TPOR = 上电复位延时。
 - 2: TPWRT = 如果使能上电延时定时器 (PWRT)，标称值为 64 ms；其他情况下为 0。
 - 3: TFRC 和 TLPRC = RC 振荡器起振时间。
 - 4: TLOCK = PLL 锁定时间。
 - 5: TOST = 振荡器起振定时器 (OST) 延迟时间。10 位计数器等待 1024 个振荡器周期后，才将振荡器时钟释放给系统。
 - 6: 如果使能了双速启动，无论选择了哪种主振荡器，器件都将使用 FRC 启动，在这些情况下，需要考虑 FRC 起振时间。

注: 有关工作频率和时序规范的详细信息，请参见第 29.0 节“电气特性”。

7.2.1 POR 和长振荡器起振时间

振荡器起振电路及其相关的延时定时器与上电时发生的器件复位延时没有关系。某些晶振电路（尤其是低频晶振）的起振时间会相对较长。因此，在发出 $\overline{\text{SYSRST}}$ 之后，可能会发生以下一种或多种情况：

- 振荡器电路未起振。
- 振荡器起振定时器（OST）尚未超时（如果使用晶振）。
- PLL 还未锁定（如果使用了 PLL）。

在有效时钟源供系统使用之前，器件不会开始执行代码。因此，如果必须确定复位延时，还须考虑振荡器起振延时和 PLL 锁定时间。

7.2.2 故障保护时钟监视器（FSCM）和器件复位

如果使能了 FSCM，它将在发出 $\overline{\text{SYSRST}}$ 时开始监视系统时钟源。如果此时没有可用的有效时钟源，器件将会自动切换到 FRC 振荡器，且用户可在陷阱服务程序（Trap Service Routine, TSR）中将系统时钟源切换到所需的晶振。

7.3 特殊功能寄存器的复位状态

大部分与 PIC24F CPU 和外设有关的特殊功能寄存器（SFR）会在器件复位时复位为某个特定值。SFR 是按其外设或 CPU 功能分组的，其复位值在本手册的各章节中有所说明。

除了四个寄存器之外，其他 SFR 的复位值都不受复位类型的影响。复位控制寄存器 RCON 的复位值取决于器件复位的类型。振荡器控制寄存器 OSCCON 的复位值取决于复位类型和闪存配置字（FOSCSEL<2:0>）中 FNOSC_x 位的编程值；请参见表 7-2。RCFGCAL 和 NVMCON 寄存器只受 POR 影响。

7.4 深度休眠 BOR（DSBOR）

深度休眠 BOR 是低功耗 BOR 电路，当器件处于深度休眠模式下时使用。由于电流消耗低，精度会因情况而异。

DSBOR 跳变点在 2.0V 左右。通过配置 DSLPBOR（FDS<6>）= 1 使能 DSBOR。DSLPBOR 将重新激活 POR 以确保当 V_{DD} 降至 POR 门限值以下时器件复位。

7.5 欠压复位（BOR）

PIC24FV32KA304 系列器件实现 BOR 电路，它将为用户提供多种配置和节能选择。BOR 由 BORV<1:0> 和 BOREN<1:0> 配置位（FPOR<6:5,1:0>）控制。共计四种 BOR 配置，如表 7-3 所示。

BOR 门限值由 BORV<1:0> 位设置。如果使能 BOR（BOREN<1:0> 除 00 之外的任意值），V_{DD} 降至设置的门限值以下时都将复位器件。芯片仍处于 BOR 直至 V_{DD} 上升超过门限值。

如果使能上电延时定时器，在 V_{DD} 上升超过门限值后，该定时器就会启动；当在上电延时定时器运行时 V_{DD} 降至门限值以下，它将使芯片保持复位状态一个额外的延时 TPWRT。芯片进入 BOR 状态，上电延时定时器会被初始化。一旦 V_{DD} 升到高于门限值，上电延时定时器将执行额外的延时。

BOR 和上电延时定时器（PWRT）是独立配置的。使能欠压复位不会自动使能 PWRT。

7.5.1 软件使能 BOR

当 BOREN<1:0> = 01 时，用户通过软件使能或禁止 BOR。通过控制位 SBOREN（RCON<13>）完成此操作。将 SBOREN 置 1 使能 BOR 以便其像之前描述的那样运行。清零 SBOREN 可完全禁止 BOR。SBOREN 位仅在该模式下工作；否则，它读为 0。

通过软件控制设置 BOR 可使用户能更灵活地定制应用以使其适应环境，而无需通过对器件再编程来更改 BOR 配置。它还使用户可以定制 BOR 消耗的增量电流。当 BOR 电流特别小时，它可能会对低功耗应用造成一些影响。

注：	当 BOR 由软件控制时，BOR 复位电压仍由 BORV<1:0> 配置位设定。它不能由软件控制。
-----------	---

PIC24FV32KA304 系列

7.5.2 检测 BOR

当使能 BOR 时，任何 BOR 或 POR 事件都会使 BOR 位（RCON<1>）复位为 1。因此只通过读 BOR 位的状态很难确定是否发生过 BOR 事件。更可靠的方法是同时检查 POR 和 BOR 的状态。假设 POR 事件后 POR 和 BOR 位将由软件立即复位为 0。如果 BOR 位为 1 而 POR 为 0，那么就可以断定 BOR 事件已发生。

注： 即使器件退出深度休眠模式，POR 和 BOR 位仍会置 1。

7.5.3 在休眠模式下禁止 BOR

当 BOREN<1:0> = 10 时，BOR 仍由硬件控制并如此前描述的一样工作。然而，每当器件进入休眠模式，BOR 就会自动禁止。当器件返回其他工作模式时，BOR 便自动重新使能。

当器件非常需要 BOR 保护时，该模式允许应用从欠压状态恢复，同时有效地执行代码。通过消除休眠模式下小的 BOR 电流，可以在休眠模式下节省更多功耗。

注： BOR 电平根据器件类型而各不相同；PIC24FV32KA3XX 器件具有不同于 PIC24F32KA3XX 器件的 BOR 电平。有关 BOR 电压电平，请见[第29.0节“电气特性”](#)。

8.0 中断控制器

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解更多中断控制器的信息，请参见《PIC24F 系列参考手册》中的第 8 章“中断”（DS39707A_CN）。

PIC24F 中断控制器将诸多外设中断请求信号缩减到一个到 CPU 的中断请求信号。该控制器具有以下特性：

- 多达 8 个处理器异常和软件陷阱
- 7 个用户可选择的优先级
- 具有多达 118 个向量的中断向量表（Interrupt Vector Table, IVT）
- 每个中断或异常源对应一个惟一的向量
- 指定的用户优先级中具有固定优先级
- 用于支持调试的备用中断向量表（Alternate Interrupt Vector Table, AIVT）
- 固定的中断进入和返回延时

8.1 中断向量表（IVT）

IVT 如图 8-1 所示。IVT 位于程序存储器中，起始单元地址为 000004h。IVT 包含 126 个向量，由 8 个不可屏蔽的陷阱向量和多达 118 个中断源组成。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。每个中断向量存储单元中设置的值是其对应中断服务程序（Interrupt Service Routine, ISR）的起始地址。

中断向量有一个自然优先级；也就是说每个中断向量的优先级与其在向量表中的位置有关。如果其他方面都相同，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。

PIC24FV32KA304 系列器件实现了不可屏蔽陷阱和惟一中断；表 8-1 和表 8-2 对此进行了总结。

8.1.1 备用中断向量表（AIVT）

如图 8-1 所示，备用中断向量表（AIVT）位于 IVT 之后。对 AIVT 的访问由 ALTIVT 控制位（INTCON2<15>）控制。如果 ALTIVT 位置 1，所有中断和异常处理将使用备用向量而不是默认向量。备用向量与默认向量的构成方式相同。

AIVT 支持仿真和调试功能，它提供了一种不需要将中断向量再编程就可以在应用程序和支持环境之间切换的方法。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软件算法。如果不需要 AIVT，应该用 IVT 中使用的地址设置 AIVT。

8.2 复位过程

由于复位过程中不涉及到中断控制器，所以器件复位并不是真的异常。作为对复位的响应，PIC24F 器件清零其寄存器，同时强制程序计数器（PC）为零。然后，单片机从单元 000000h 开始执行程序。用户在复位地址中设置一条 GOTO 指令，该指令会使程序执行重新定位到相应的启动程序。

注： 应该使用包含 RESET 指令的默认中断处理程序的地址编程 IVT 和 AIVT 中所有未实现或未使用的向量存储单元。

PIC24FV32KA304 系列

图 8-1: PIC24F 中断向量表

自然优先级降序排列	复位 — GOTO 指令	000000h	中断向量表 (IVT) ⁽¹⁾
	复位 — GOTO 地址	000002h	
	保留	000004h	
	振荡器故障陷阱向量		
	地址错误陷阱向量		
	堆栈错误陷阱向量		
	算术错误陷阱向量		
	保留		
	保留		
	保留		
	中断向量 0	000014h	
	中断向量 1		
	—		
	—		
	—		
	中断向量 52	00007Ch	
	中断向量 53	00007Eh	
	中断向量 54	000080h	
	—		
	—		
	—		
	中断向量 116	0000FCh	
	中断向量 117	0000FEh	
	保留	000100h	
	保留	000102h	
	保留		
	振荡器故障陷阱向量		
	地址错误陷阱向量		
	堆栈错误陷阱向量		
	算术错误陷阱向量		
	保留		
	保留		
	保留		
	中断向量 0	000114h	备用中断向量表 (AIVT) ⁽¹⁾
	中断向量 1		
	—		
	—		
	—		
	中断向量 52	00017Ch	
	中断向量 53	00017Eh	
	中断向量 54	000180h	
	—		
	—		
	—		
	中断向量 116		
	中断向量 117	0001FEh	
	代码起始	000200h	

注 1: 有关中断向量列表的信息, 请参见表 8-2。

PIC24FV32KA304 系列

表 8-1: 陷阱向量详细信息

向量编号	IVT 地址	AIVT 地址	陷阱源
0	000004h	000104h	保留
1	000006h	000106h	振荡器故障
2	000008h	000108h	地址错误
3	00000Ah	00010Ah	堆栈错误
4	00000Ch	00010Ch	算术错误
5	00000Eh	00010Eh	保留
6	000010h	000110h	保留
7	000012h	000112h	保留

表 8-2: 实现的中断向量

中断源	向量编号	IVT 地址	AIVT 地址	中断位位置		
				标志	允许	优先级
ADC1 转换完成	13	00002Eh	00012Eh	IFS0<13>	IEC0<13>	IPC3<6:4>
比较器事件	18	000038h	000138h	IFS1<2>	IEC1<2>	IPC4<10:8>
CRC 发生器	67	00009Ah	00019Ah	IFS4<3>	IEC4<3>	IPC16<14:12>
CTMU	77	0000AEh	0001AEh	IFS4<13>	IEC4<13>	IPC19<6:4>
外部中断 0	0	000014h	000114h	IFS0<0>	IEC0<0>	IPC0<2:0>
外部中断 1	20	00003Ch	00013Ch	IFS1<4>	IEC1<4>	IPC5<2:0>
外部中断 2	29	00004Eh	00014Eh	IFS1<13>	IEC1<13>	IPC7<6:4>
I2C1 主事件	17	000036h	000136h	IFS1<1>	IEC1<1>	IPC4<6:4>
I2C1 从事件	16	000034h	000134h	IFS1<0>	IEC1<0>	IPC4<2:0>
I2C2 主事件	50	000078h	000178h	IFS3<2>	IEC3<2>	IPC12<10:8>
I2C2 从事件	49	000076h	000176h	IFS3<1>	IEC3<1>	IPC12<6:4>
输入捕捉 1	1	000016h	000116h	IFS0<1>	IEC0<1>	IPC0<6:4>
输入捕捉 2	5	00001Eh	00011Eh	IFS0<5>	IEC0<5>	IPC1<6:4>
输入捕捉 3	37	00005Eh	00015Eh	IFS2<5>	IEC2<5>	IPC9<6:4>
输入电平变化通知	19	00003Ah	00013Ah	IFS1<3>	IEC1<3>	IPC4<14:12>
HLVD (高 / 低电压检测)	72	0000A4h	0001A4h	IFS4<8>	IEC4<8>	IPC17<2:0>
NVM——NVM 写完成	15	000032h	000132h	IFS0<15>	IEC0<15>	IPC3<14:12>
输出比较 1	2	000018h	000118h	IFS0<2>	IEC0<2>	IPC0<10:8>
输出比较 2	6	000020h	000120h	IFS0<6>	IEC0<6>	IPC1<10:8>
输出比较 3	25	000046h	000146h	IFS1<9>	IEC1<9>	IPC6<6:4>
实时时钟 / 日历	62	000090h	000190h	IFS3<14>	IEC3<14>	IPC15<10:8>
SPI1 错误	9	000026h	000126h	IFS0<9>	IEC0<9>	IPC2<6:4>
SPI1 事件	10	000028h	000128h	IFS0<10>	IEC0<10>	IPC2<10:8>
SPI2 错误	32	000054h	000154h	IFS2<0>	IEC2<2>	IPC8<2:0>
SPI2 事件	33	000056h	000156h	IFS2<1>	IEC2<1>	IPC8<6:4>
Timer1	3	00001Ah	00011Ah	IFS0<3>	IEC0<3>	IPC0<14:12>
Timer2	7	000022h	000122h	IFS0<7>	IEC0<7>	IPC1<14:12>
Timer3	8	000024h	000124h	IFS0<8>	IEC0<8>	IPC2<2:0>
Timer4	27	00004Ah	00014Ah	IFS1<11>	IEC1<11>	IPC6<14:12>
Timer5	28	00004Ch	00015Ch	IFS1<12>	IEC1<12>	IPC7<2:0>
UART1 错误	65	000096h	000196h	IFS4<1>	IEC4<1>	IPC16<6:4>
UART1 接收器	11	00002Ah	00012Ah	IFS0<11>	IEC0<11>	IPC2<14:12>
UART1 发送器	12	00002Ch	00012Ch	IFS0<12>	IEC0<12>	IPC3<2:0>
UART2 错误	66	000098h	000198h	IFS4<2>	IEC4<2>	IPC16<10:8>
UART2 接收器	30	000050h	000150h	IFS1<14>	IEC1<14>	IPC7<10:8>
UART2 发送器	31	000052h	000152h	IFS1<15>	IEC1<15>	IPC7<14:12>
超低功耗唤醒	80	0000B4h	0001B4h	IFS5<0>	IEC5<0>	IPC20<2:0>

PIC24FV32KA304 系列

8.3 中断控制和状态寄存器

PIC24FV32KA304 系列器件共实现了用于中断控制器的 23 个寄存器：

- INTCON1
- INTCON2
- IFS0、IFS1、IFS3 和 IFS4
- IEC0、IEC1、IEC3 和 IEC4
- IPC0 至 IPC5、IPC7 和 IPC15 至 IPC19
- INTTREG

INTCON1 和 INTCON2 控制全局中断允许（Global Interrupt Enable, GIE）功能。INTCON1 包含中断嵌套禁止（NSTDIS）位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和 AIVT 的使用。

IFSx 寄存器包含所有中断请求标志。每个中断源都有一个状态位，由各自的外设或外部信号置 1，且由软件清零。

IECx 寄存器包含所有中断允许位。这些控制位用于单独允许外设或外部信号中断。

IPCx 寄存器用于设置每个中断源的中断优先级（Interrupt Priority Level, IPL）。可以给每个用户中断源分配 8 个优先级之一。

INTTREG 寄存器包含相关的中断向量编号和新的 CPU 中断优先级，它们被锁存到 INTTREG 寄存器中的向量编号（VECNUM<6:0>）和中断优先级（ILR<3:0>）位域中。新的中断优先级是待处理中断的优先级。

中断源按其在表 8-2 中列出的顺序分配给 IFSx、IECx 和 IPCx 寄存器。例如，INT0（外部中断 0）表示向量编号为 0，且自然优先级为 0。所以 INTOIF 状态位在 IFS0<0> 中，INT0IE 允许位在 IEC0<0> 中，INT0IP<2:0> 优先级位在 IPC0 最前面的位置（IPC0<2:0>）中。

尽管两个 CPU 控制寄存器都不是中断控制硬件的特定组成部分，但它们仍包含控制中断功能的位。ALU 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。用户可以通过写 IPLx 位更改当前 CPU 的中断优先级。

CORCON 寄存器包含 IPL3 位，此位与 IPL<2:0> 位一起指示当前 CPU 的中断优先级。IPL3 是一个只读位，所以用户软件不能屏蔽陷阱事件。

在下面各节中的寄存器 8-1 到寄存器 8-33 说明了所有的中断寄存器。

PIC24FV32KA304 系列

寄存器 8-1: SR: ALU 状态寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0, HSC
—	—	—	—	—	—	—	DC ⁽¹⁾
bit 15							bit 8

R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC
IPL2 ^(2,3)	IPL1 ^(2,3)	IPL0 ^(2,3)	RA ⁽¹⁾	N ⁽¹⁾	OV ⁽¹⁾	Z ⁽¹⁾	C ⁽¹⁾
bit 7							bit 0

图注:	HSC = 可由硬件置 1/ 清零的位						
R = 可读位	W = 可写位		U = 未实现位, 读为 0				
-n = 上电复位时的值	1 = 置 1		0 = 清零		x = 未知		

bit 15-9 未实现: 读为 0

bit 7-5 IPL<2:0>: CPU 中断优先级状态位 ^(2,3)

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: 请参见寄存器 3-1 了解那些不是专门用于中断控制功能的位的描述。
- 2: IPL<2:0> 位与 IPL3 位 (CORCON<3>) 共同决定 CPU 的中断优先级。如果 IPL3 = 1, 则括号中的值表示中断优先级。
- 3: 当 NSTDIS (INTCON1<15>) = 1 时 IPLx 状态位为只读。

注: 第 3.0 节 “CPU” 描述了 bit 8、bit 4 到 bit 0。

PIC24FV32KA304 系列

寄存器 8-2: CORCON: CPU 控制寄存器

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/C-0, HSC	R/W-0	U-0	U-0
—	—	—	—	IPL3 ⁽²⁾	PSV ⁽¹⁾	—	—
bit 7						bit 0	

图注:	C = 可清零位	HSC = 可由硬件置 1/ 清零的位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15-4 未实现: 读为 0

bit 3 **IPL3:** CPU 中断优先级状态位 ⁽²⁾
1 = CPU 中断优先级大于 7
0 = CPU 中断优先级为 7 或更小

bit 1-0 未实现: 读为 0

注 1: 请见 [寄存器 3-2](#) 了解这个不是专门用于中断控制功能的位的描述。

 2: IPL3 位与 IPL<2:0> 位 (SR<7:5>) 共同决定 CPU 中断优先级。

注: [第 3.0 节 “CPU”](#) 描述了 bit 2。

PIC24FV32KA304 系列

寄存器 8-3: INTCON1: 中断控制寄存器 1

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
NSTDIS	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0
—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
bit 7							bit 0

图注:	HS = 可由硬件置 1 的位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	NSTDIS: 中断嵌套禁止位 1 = 禁止中断嵌套 0 = 允许中断嵌套
bit 14-5	未实现: 读为 0
bit 4	MATHERR: 算法错误陷阱状态位 1 = 发生了溢出陷阱 0 = 未发生溢出陷阱
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
bit 2	STKERR: 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	OSCFAIL: 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	未实现: 读为 0

PIC24FV32KA304 系列

寄存器 8-4: INTCON2: 中断控制寄存器 2

R/W-0	R-0, HSC	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	INT2EP	INT1EP	INT0EP
bit 7							bit 0

图注:	HSC = 可由硬件置 1/ 清零的位						
R = 可读位	W = 可写位		U = 未实现位，读为 0				
-n = 上电复位时的值	1 = 置 1		0 = 清零		x = 未知		

- bit 15

ALTIVT: 备用中断向量表使能位
1 = 使用备用中断向量表 (AIVT)
0 = 使用标准 (默认) 中断向量表 (IVT)
- bit 14

DISI: DISI 指令状态位
1 = 执行了 DISI 指令
0 = 没有执行 DISI 指令
- bit 13-3

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

PIC24FV32KA304 系列

寄存器 8-5: IFS0: 中断标志状态寄存器 0

R/W-0, HS	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS
NVMIF	—	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPF1IF	T3IF
bit 15							bit 8

R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS
T2IF	OC2IF	IC2IF	—	T1IF	OC1IF	IC1IF	INT0IF
bit 7							bit 0

图注:	HS = 可由硬件置 1 的位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	NVMIF: NVM 中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 14	未实现: 读为 0
bit 13	AD1IF: A/D 转换完成中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 12	U1TXIF: UART1 发送器中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 11	U1RXIF: UART1 接收器中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 10	SPI1IF: SPI1 事件中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 9	SPF1IF: SPI1 故障中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 8	T3IF: Timer3 中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 7	T2IF: Timer2 中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 6	OC2IF: 输出比较通道 2 中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求
bit 5	IC2IF: 输入捕捉通道 2 中断标志状态位 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	T1IF: Timer1 中断标志状态位 1 = 产生了中断请求 0 = 未产生中断请求

PIC24FV32KA304 系列

寄存器 8-5: IFS0: 中断标志状态寄存器 0 (续)

- | | |
|-------|--|
| bit 2 | OC1IF: 输出比较通道 1 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求 |
| bit 1 | IC1IF: 输入捕捉通道 1 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求 |
| bit 0 | INT0IF: 外部中断 0 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求 |

寄存器 8-6: IFS1: 中断标志状态寄存器 1

R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0	R/W-0, HS	U-0
U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	—	OC3IF	—
bit 15						bit 8	

U-0	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	R/W-0	R/W-0
—	—	—	INT1IF	CNIF	CMIF	MI2C1IF	SI2C1IF
bit 7						bit 0	

图注: HS = 可由硬件置 1 的位
R = 可读位 W = 可写位 U = 未实现位, 读为 0
-n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

- bit 15 **U2TXIF:** UART2 发送器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 14 **U2RXIF:** UART2 接收器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 13 **INT2IF:** 外部中断 2 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 12 **T5IF:** Timer5 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 11 **T4IF:** Timer4 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 10 **未实现:** 读为 0
- bit 9 **OC3IF:** 输出比较通道 3 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 8-5 **未实现:** 读为 0
- bit 4 **INT1IF:** 外部中断 1 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 3 **CNIF:** 输入电平变化通知中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 2 **CMIF:** 比较器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 1 **MI2C1IF:** I2C1 主事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 0 **SI2C1IF:** I2C1 从事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

PIC24FV32KA304 系列

寄存器 8-7: IFS2: 中断标志状态寄存器 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	R/W-0, HS	U-0	U-0	U-0	R/W-0, HS	R/W-0, HS
—	—	IC3IF	—	—	—	SPI2IF	SPF2IF
bit 7							bit 0

图注:	HS = 可由硬件置 1 的位		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-6未实现：读为 0
- bit 5**IC3IF**: 输入捕捉通道 3 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 4-2未实现：读为 0
- bit 1**SPI2IF**: SPI2 事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 0**SPF2IF**: SPI2 故障中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

寄存器 8-8: IFS3: 中断标志状态寄存器 3

U-0	R/W-0, HS	U-0	U-0	U-0	U-0	U-0	U-0
—	RTCIF	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0, HS	R/W-0, HS	U-0
—	—	—	—	—	MI2C2IF	SI2C2IF	—
bit 7							bit 0

图注: HS = 可由硬件置 1 的位
R = 可读位 W = 可写位 U = 未实现位, 读为 0
-n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

bit 15 未实现: 读为 0
bit 14 **RTCIF:** 实时时钟 / 日历中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
bit 13-3 未实现: 读为 0
bit 2 **MI2C2IF:** I2C2 主事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
bit 1 **SI2C2IF:** I2C2 从事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
bit 0 未实现: 读为 0

PIC24FV32KA304 系列

寄存器 8-9: IFS4: 中断标志状态寄存器 4

U-0	U-0	R/W-0, HS	U-0	U-0	U-0	U-0	R/W-0, HS
—	—	CTMUIF	—	—	—	—	HLVDIF
bit 15							bit 8
U-0	U-0	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0
—	—	—	—	CRCIF	U2ERIF	U1ERIF	—
bit 7							bit 0

图注:	HS = 可由硬件置 1 的位		
R = 可读位	W = 可写位		U = 未实现位, 读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-14 未实现: 读为 0
- bit 13 CTMUIF: CTMU 中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 12-9 未实现: 读为 0
- bit 8 HLVDIF: 高 / 低电压检测中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 7-4 未实现: 读为 0
- bit 3 CRCIF: CRC 发生器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 2 U2ERIF: UART2 错误中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 1 U1ERIF: UART1 错误中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 0 未实现: 读为 0

PIC24FV32KA304 系列

寄存器 8-10: IFS5: 中断标志状态寄存器 5

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, HS
—	—	—	—	—	—	—	ULPWUIF
bit 7				bit 0			

图注:

HS = 可由硬件置 1 的位

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-1

未实现: 读为 0

bit 0

ULPWUIF: 超低功耗唤醒中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

PIC24FV32KA304 系列

寄存器 8-11: IEC0: 中断允许控制寄存器 0

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
NVMIE	—	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPF1IE	T3IE
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
T2IE	OC2IE	IC2IE	—	T1IE	OC1IE	IC1IE	INT0IE
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **NVMIE:** NVM 中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 14 **未实现:** 读为 0
- bit 13 **AD1IE:** A/D 转换完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 12 **U1TXIE:** UART1 发送器中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 11 **U1RXIE:** UART1 接收器中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 10 **SPI1IE:** SPI1 传输完成中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 9 **SPF1IE:** SPI1 故障中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 8 **T3IE:** Timer3 中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 7 **T2IE:** Timer2 中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 6 **OC2IE:** 输出比较通道 2 中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 5 **IC2IE:** 输入捕捉通道 2 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 **T1IE:** Timer1 中断允许位
1 = 允许中断请求
0 = 禁止中断请求
- bit 2 **OC1IE:** 输出比较通道 1 中断允许位
1 = 允许中断请求
0 = 禁止中断请求

寄存器 8-11: IEC0: 中断允许控制寄存器 0 (续)

bit 1	IC1IE: 输入捕捉通道 1 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 0	INT0IE: 外部中断 0 允许位 1 = 允许中断请求 0 = 禁止中断请求

PIC24FV32KA304 系列

寄存器 8-12: IEC1: 中断允许控制寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	—	OC3IE	—
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
—	—	—	INT1IE	CNIE	CMIE	MI2C1IE	SI2C1IE
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **U2TXIE:** UART2 发送器中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 14 **U2RXIE:** UART2 接收器中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 13 **INT2IE:** 外部中断 2 允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 12 **T5IE:** Timer5 中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 11 **T4IE:** Timer4 中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 10 **未实现:** 读为 0

bit 9 **OC3IE:** 输出比较通道 3 中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 8-5 **未实现:** 读为 0

bit 4 **INT1IE:** 外部中断 1 允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 3 **CNIE:** 输入电平变化通知中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 2 **CMIE:** 比较器中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 1 **MI2C1IE:** I2C1 主事件中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 0 **SI2C1IE:** I2C1 从事件中断允许位

1 = 允许中断请求

0 = 禁止中断请求

寄存器 8-13: IEC2: 中断允许控制寄存器 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	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	IC3IE	—	—	—	SPI2IE	SPF2IE
bit 7							bit 0

图注:	HS = 可由硬件置 1 的位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-6	未实现: 读为 0
bit 5	IC3IE: 输入捕捉通道 3 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 4-2	未实现: 读为 0
bit 1	SPI2IE: SPI2 事件中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 0	SPF2IE: SPI2 故障中断允许位 1 = 允许中断请求 0 = 禁止中断请求

PIC24FV32KA304 系列

寄存器 8-14: IEC3: 中断允许控制寄存器 3

U-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
—	RTCIE	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	MI2C2IE	SI2C2IE	—
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	未实现: 读为 0
bit 14	RTCIE: 实时时钟 / 日历中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 13-3	未实现: 读为 0
bit 2	MI2C2IE: I2C2 主事件中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 1	SI2C2IE: I2C2 从事件中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 0	未实现: 读为 0

寄存器 8-15: IEC4: 中断允许控制寄存器 4

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
—	—	CTMUIE	—	—	—	—	HLVDIE
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	—	CRCIE	U2ERIE	U1ERIE	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-14 **未实现:** 读为 0
- bit 13 **CTMUIE:** CTMU 中断允许位
 - 1 = 允许中断请求
 - 0 = 禁止中断请求
- bit 12-9 **未实现:** 读为 0
- bit 8 **HLVDIE:** 高 / 低电压检测中断允许位
 - 1 = 允许中断请求
 - 0 = 禁止中断请求
- bit 7-4 **未实现:** 读为 0
- bit 3 **CRCIE:** CRC 发生器中断允许位
 - 1 = 允许中断请求
 - 0 = 禁止中断请求
- bit 2 **U2ERIE:** UART2 错误中断允许位
 - 1 = 允许中断请求
 - 0 = 禁止中断请求
- bit 1 **U1ERIE:** UART1 错误中断允许位
 - 1 = 允许中断请求
 - 0 = 禁止中断请求
- bit 0 **未实现:** 读为 0

PIC24FV32KA304 系列

寄存器 8-16: IEC5: 中断允许控制寄存器 5

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
—	—	—	—	—	—	—	ULPWUIE
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-1 未实现: 读为 0
bit 0 **ULPWUIE:** 超低功耗唤醒中断允许位
 1 = 允许中断请求
 0 = 禁止中断请求

PIC24FV32KA304 系列

寄存器 8-17: **IPC0: 中断优先级控制寄存器 0**

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T1IP2	T1IP1	T1IP0	—	OC1IP2	OC1IP1	OC1IP0
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC1IP2	IC1IP1	IC1IP0	—	INT0IP2	INT0IP1	INT0IP0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **未实现:** 读为 0

bit 14-12 **T1IP<2:0>:** Timer1 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 11 **未实现:** 读为 0

bit 10-8 **OC1IP<2:0>:** 输出比较通道 1 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 7 **未实现:** 读为 0

bit 6-4 **IC1IP<2:0>:** 输入捕捉通道 1 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 3 **未实现:** 读为 0

bit 2-0 **INT0IP<2:0>:** 外部中断 0 优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

PIC24FV32KA304 系列

寄存器 8-18: **IPC1: 中断优先级控制寄存器 1**

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T2IP2	T2IP1	T2IP0	—	OC2IP2	OC2IP1	OC2IP0
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	IC2IP2	IC2IP1	IC2IP0	—	—	—	—
bit 7				bit 0			

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	未实现: 读为 0
bit 14-12	T2IP<2:0> : Timer2 中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 11	未实现: 读为 0
bit 10-8	OC2IP<2:0> : 输出比较通道 2 中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 7	未实现: 读为 0
bit 6-4	IC2IP<2:0> : 输入捕捉通道 2 中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 3-0	未实现: 读为 0

寄存器 8-19: IPC2: 中断优先级控制寄存器 2

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U1RXIP2	U1RXIP1	U1RXIP0	—	SPI1IP2	SPI1IP1	SPI1IP0
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	SPF1IP2	SPF1IP1	SPF1IP0	—	T3IP2	T3IP1	T3IP0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **未实现:** 读为 0

bit 14-12 **U1RXIP<2:0>:** UART1 接收器中断优先级位

111 = 中断优先级为 7 (最高优先级中断)

•

•

•

001 = 中断优先级为 1

000 = 禁止中断源

bit 11 **未实现:** 读为 0

bit 10-8 **SPI1IP<2:0>:** SPI1 事件中断优先级位

111 = 中断优先级为 7 (最高优先级中断)

•

•

•

001 = 中断优先级为 1

000 = 禁止中断源

bit 7 **未实现:** 读为 0

bit 6-4 **SPF1IP<2:0>:** SPI1 故障中断优先级位

111 = 中断优先级为 7 (最高优先级中断)

•

•

•

001 = 中断优先级为 1

000 = 禁止中断源

bit 3 **未实现:** 读为 0

bit 2-0 **T3IP<2:0>:** Timer3 中断优先级位

111 = 中断优先级为 7 (最高优先级中断)

•

•

•

001 = 中断优先级为 1

000 = 禁止中断源

PIC24FV32KA304 系列

寄存器 8-20: **IPC3: 中断优先级控制寄存器 3**

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	NVMIP2	NVMIP1	NVMIP0	—	—	—	—
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	AD1IP2	AD1IP1	AD1IP0	—	U1TXIP2	U1TXIP1	U1TXIP0
bit 7				bit 0			

图注:
R = 可读位 W = 可写位 U = 未实现位，读为 0
-n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

bit 15 **未实现:** 读为 0

bit 14-12 **NVMIP<2:0>:** NVM 中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 ·
 ·
 ·
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 11-7 **未实现:** 读为 0

bit 6-4 **AD1IP<2:0>:** A/D 转换完成中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 ·
 ·
 ·
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 3 **未实现:** 读为 0

bit 2-0 **U1TXIP<2:0>:** UART1 发送器中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 ·
 ·
 ·
 001 = 中断优先级为 1
 000 = 禁止中断源

PIC24FV32KA304 系列

寄存器 8-21: IPC4: 中断优先级控制寄存器 4

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	CNIP2	CNIP1	CNIP0	—	CMIP2	CMIP1	CMIP0
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	MI2C1P2	MI2C1P1	MI2C1P0	—	SI2C1P2	SI2C1P1	SI2C1P0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15	未实现: 读为 0
bit 14-12	CNIP<2:0>: 输入电平变化通知中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 11	未实现: 读为 0
bit 10-8	CMIP<2:0>: 比较器中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 7	未实现: 读为 0
bit 6-4	MI2C1P<2:0>: I2C1 主事件中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 3	未实现: 读为 0
bit 2-0	SI2C1P<2:0>: I2C1 从事件中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源

D = 可读位 W = 可写位 H = 十六进制位 逢 16 进 1

x = 未知

000 = 禁止中断源

PIC24FV32KA304 系列

寄存器 8-23: IPC6: 中断优先级控制寄存器 6

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	T4IP2	T4IP1	T4IP0	—	—	—	—
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	OC3IP2	OC3IP1	OC3IP0	—	—	—	—
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 未实现: 读为 0

bit 14-12 **T4IP<2:0>**: Timer4 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 11-7 未实现: 读为 0

bit 6-4 **OC3IP<2:0>**: 输出比较通道 3 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 3-0 未实现: 读为 0

PIC24FV32KA304 系列

寄存器 8-24: **IPC7: 中断优先级控制寄存器 7**

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U2TXIP2	U2TXIP1	U2TXIP0	—	U2RXIP2	U2RXIP1	U2RXIP0
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	INT2IP2	INT2IP1	INT2IP0	—	T5IP2	T5IP1	T5IP0
bit 7				bit 0			

图注:							
R = 可读位		W = 可写位		U = 未实现位，读为 0			
-n = 上电复位时的值		1 = 置 1		0 = 清零		x = 未知	

bit 15	未实现：读为 0
bit 14-12	U2TXIP<2:0> : UART2 发送器中断优先级位 111 = 中断优先级为 7（最高优先级中断） · · · 001 = 中断优先级为 1 000 = 禁止中断源
bit 11	未实现：读为 0
bit 10-8	U2RXIP<2:0> : UART2 接收器中断优先级位 111 = 中断优先级为 7（最高优先级中断） · · · 001 = 中断优先级为 1 000 = 禁止中断源
bit 7	未实现：读为 0
bit 6-4	INT2IP<2:0> : 外部中断 2 优先级位 111 = 中断优先级为 7（最高优先级中断） · · · 001 = 中断优先级为 1 000 = 禁止中断源
bit 3	未实现：读为 0
bit 2-0	T5IP<2:0> : Timer5 中断优先级位 111 = 中断优先级为 7（最高优先级中断） · · · 001 = 中断优先级为 1 000 = 禁止中断源

PIC24FV32KA304 系列

寄存器 8-25: IPC8: 中断优先级控制寄存器 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	SPI2IP2	SPI2IP1	SPI2IP0	—	SPF2IP2	SPF2IP1	SPF2IP0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-7	未实现：读为 0
bit 6-4	SPI2IP<2:0> : SPI2 事件中断优先级位
	111 = 中断优先级为 7（最高优先级中断）
	•
	•
	001 = 中断优先级为 1
	000 = 禁止中断源
bit 3	未实现：读为 0
bit 2-0	SPF2IP<2:0> : SPI2 故障中断优先级位
	111 = 中断优先级为 7（最高优先级中断）
	•
	•
	001 = 中断优先级为 1
	000 = 禁止中断源

PIC24FV32KA304 系列

寄存器 8-26: **IPC9: 中断优先级控制寄存器 9**

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

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	IC3IP2	IC3IP1	IC3IP0	—	—	—	—
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-7

未实现：读为 0
- bit 6-4

IC3IP<2:0>：输入捕捉通道 3 事件中断优先级位

111 = 中断优先级为 7 （最高优先级中断）

⋮

001 = 中断优先级为 1

000 = 禁止中断源
- bit 3-0

未实现：读为 0

PIC24FV32KA304 系列

寄存器 8-27: IPC12: 中断优先级控制寄存器 12

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	MI2C2IP2	MI2C2IP1	MI2C2IP0
bit 15						bit 8	

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	SI2C2IP2	SI2C2IP1	SI2C2IP0	—	—	—	—
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-11 未实现: 读为 0

bit 10-8 **MI2C2IP<2:0>**: I2C2 主事件中断优先级位

111 = 中断优先级为 7 (最高优先级中断)

•

•

•

001 = 中断优先级为 1

000 = 禁止中断源

bit 7 未实现: 读为 0

bit 6-4 **SI2C2IP<2:0>**: I2C2 从事件中断优先级位

111 = 中断优先级为 7 (最高优先级中断)

•

•

•

001 = 中断优先级为 1

000 = 禁止中断源

bit 3-0 未实现: 读为 0

PIC24FV32KA304 系列

寄存器 8-28: IPC15: 中断优先级控制寄存器 15

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	RTCIP2	RTCIP1	RTCIP0
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 = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-11 **未实现:** 读为 0

bit 10-8 **RTCIP<2:0>:** 实时时钟 / 日历中断优先级位

111 = 中断优先级为 7 （最高优先级中断）

.

.

.

001 = 中断优先级为 1

000 = 禁止中断源

bit 7-0 **未实现:** 读为 0

PIC24FV32KA304 系列

寄存器 8-29: **IPC16: 中断优先级控制寄存器 16**

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	CRCIP2	CRCIP1	CRCIP0	—	U2ERIP2	U2ERIP1	U2ERIP0
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1ERIP2	U1ERIP1	U1ERIP0	—	—	—	—
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **未实现:** 读为 0

bit 14-12 **CRCIP<2:0>:** CRC 发生器错误中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 11 **未实现:** 读为 0

bit 10-8 **U2ERIP<2:0>:** UART2 错误中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 7 **未实现:** 读为 0

bit 6-4 **U1ERIP<2:0>:** UART1 错误中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 3-0 **未实现:** 读为 0

PIC24FV32KA304 系列

寄存器 8-30: IPC18: 中断优先级控制寄存器 18

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-1	R/W-0	R/W-0
—	—	—	—	—	HLVDIP2	HLVDIP1	HLVDIP0
bit 7							bit 0

图注:							
R = 可读位		W = 可写位		U = 未实现位, 读为 0			
-n = 上电复位时的值		1 = 置 1		0 = 清零		x = 未知	

bit 15-3 未实现: 读为 0
bit 2-0 **HLVDIP<2:0>**: 高 / 低电压检测中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

寄存器 8-31: IPC19: 中断优先级控制寄存器 19

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

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	CTMUIP2	CTMUIP1	CTMUIP0	—	—	—	—
bit 7							bit 0

图注:							
R = 可读位		W = 可写位		U = 未实现位, 读为 0			
-n = 上电复位时的值		1 = 置 1		0 = 清零		x = 未知	

bit 15-7 未实现: 读为 0
bit 6-4 **CTMUIP<2:0>**: CTMU 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源
bit 3-0 未实现: 读为 0

PIC24FV32KA304 系列

寄存器 8-32: **IPC20: 中断优先级控制寄存器 20**

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-1	R/W-0	R/W-0
—	—	—	—	—	ULPWUIP2	ULPWUIP1	ULPWUIP0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-3 **未实现:** 读为 0

bit 2-0 **ULPWUIP<2:0>:** 超低功耗唤醒中断优先级位

111 = 中断优先级为 7 (最高优先级中断)

⋮

001 = 中断优先级为 1

000 = 禁止中断源

PIC24FV32KA304 系列

寄存器 8-33: INTTREG: 中断控制和状态寄存器

R-0	U-0	R/W-0	U-0	R-0	R-0	R-0	R-0
CPUIRQ	—	VHOLD	—	ILR3	ILR2	ILR1	ILR0
bit 15				bit 8			
U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	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 = 上电复位时的值		1 = 置 1		0 = 清零		x = 未知	

bit 15	CPUIRQ: 来自中断控制器 CPU 的中断请求位 1 = 产生了中断请求, 但 CPU 尚未应答; 这通常发生在 CPU 的优先级高于所请求的中断优先级的情况下 0 = 没有未应答中断请求
bit 14	未实现: 读为 0
bit 13	VHOLD: 向量保持位 允许捕捉和修改存储在 VECNUM 位中的中断向量编号: 1 = VECNUM 位保存优先级最高的待处理中断的值, 而不是当前中断。 0 = VECNUM 位保存上一次应答的中断值 (即已产生的且优先级高于 CPU 的上一个中断, 即使有其他待处理中断时也是如此)
bit 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	未实现: 读为 0
bit 6-0	VECNUM<6:0>: 待处理中断的向量编号位 0111111 = 待处理的中断向量编号为 135 . . . 0000001 = 待处理的中断向量编号为 9 0000000 = 待处理的中断向量编号为 8

8.4 中断设置过程

8.4.1 初始化

按以下步骤配置中断源：

1. 如果不需要嵌套中断，则将 `NSTDIS` 控制位 (`INTCON1<15>`) 置 1。
2. 通过写相应的 `IPCx` 寄存器中的控制位来为中断源选择用户分配的优先级。优先级将取决于具体的应用和中断源的类型。如果不需要多个优先级，可以将所有允许的中断源的 `IPCx` 寄存器控制位编程为相同的非零值。

注： 在器件复位时，`IPCx` 寄存器被初始化，为所有用户中断源分配优先级 4。

3. 将相应 `IFSx` 寄存器中与外设相关的中断标志状态位清零。
4. 通过将相应 `IECx` 寄存器中与中断源相关的中断允许控制位置 1 来允许中断源。

8.4.2 中断服务程序

用于声明 `ISR`（中断服务程序）以及使用正确的向量地址初始化 `IVT` 的方法取决于编程语言（即 `C` 语言或汇编语言）和用于开发应用程序的语言开发工具套件。一般情况下，用户必须将相应 `IFSx` 寄存器中与 `ISR` 处理的中断源相对应的中断标志清零。否则，在退出中断后会立即再次进入 `ISR`。如果 `ISR` 用汇编语言编写，则必须使用 `RETFIE` 指令结束 `ISR`，以便将保存的 `PC` 值、`SRL` 值和原先的 `CPU` 优先级弹出堆栈。

8.4.3 陷阱服务程序（TSR）

陷阱服务程序（`Trap Service Routine, TSR`）的编码方式类似于 `ISR`，只是必须将 `INTCON1` 寄存器中相应的陷阱状态标志清零，以避免重新进入 `TSR`。

8.4.4 禁止中断

可以通过以下步骤禁止所有用户中断：

1. 使用 `PUSH` 指令将当前 `SR` 值压入软件堆栈。
2. 通过将值 `OEH` 与 `SRL` 进行逻辑或运算来强制把 `CPU` 的优先级设置为 7。

要允许用户中断，可以使用 `POP` 指令恢复先前的 `SR` 值。

只能禁止优先级小于或等于 7 的用户中断。不能禁止陷阱源（优先级为 8-15）。

使用 `DISI` 指令可以方便地将优先级为 1-6 的中断禁止一段固定的时间。`DISI` 指令不能禁止优先级为 7 的中断源。

PIC24FV32KA304 系列

注

9.0 振荡器配置

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。有关使用振荡器配置的更多详细信息，请参见《PIC24F 系列参考手册》中的第 38 章“带 500 kHz 低功耗 FRC 的振荡器”（DS39726A_CN）。

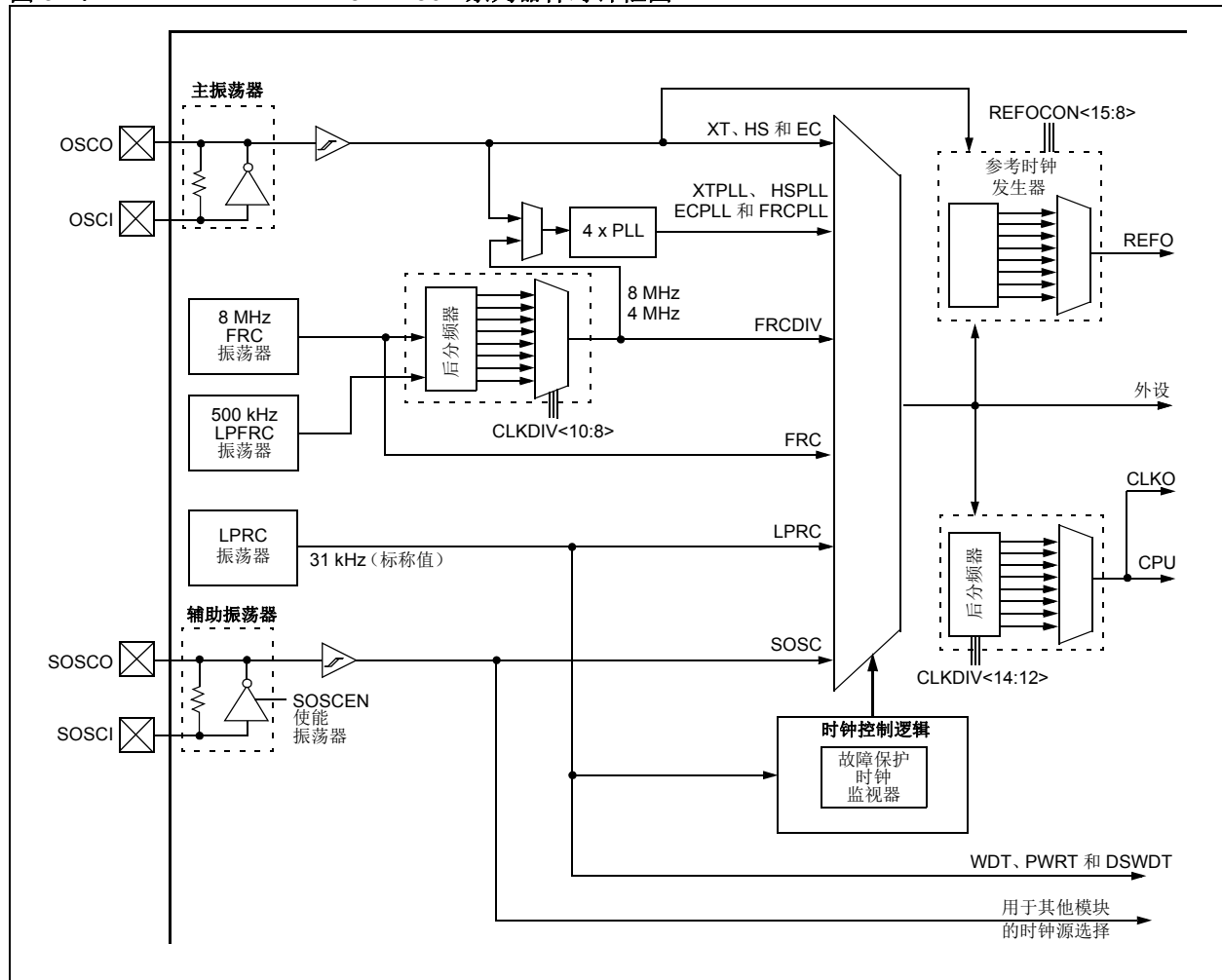
PIC24FV32KA304 系列器件的振荡器系统具有以下特性：

- 共有五个外部和内部振荡器可选作为时钟源，提供 11 种不同的时钟模式
- 片内 4 倍锁相环（PLL），用于在选择内部和外部振荡器源时提高内部工作频率。

- 可通过软件控制在多个时钟源之间切换
- 可通过软件控制后分频器有选择地为 CPU 提供时钟以节省系统功耗
- EC 模式的系统频率范围声明位。当使用外部时钟源时，通过将声明位设为期望的频率范围可降低电流消耗。
- 具有故障保护时钟监视器（FSCM），可检测时钟故障，以使应用可安全地恢复或关闭。

图 9-1 所示为振荡器系统的简化框图。

图 9-1: PIC24FV32KA304 系列器件时钟框图



PIC24FV32KA304 系列

9.1 CPU 时钟机制

系统时钟源可由以下四种时钟源之一提供：

- OSCI 和 OSCO 引脚上的主振荡器（POSC）
- SOSCI 和 SOSCO 引脚上的辅助振荡器（SOSC）

PIC24FV32KA304 系列器件由两种类型的辅助振荡器组成：

- 高功耗辅助振荡器
- 低功耗辅助振荡器

可使用 SOSCSEL（FOSC<5>）位进行选择。

- 内部快速 RC（FRC）振荡器
 - 8 MHz FRC 振荡器
 - 500 kHz 低功耗 FRC 振荡器
- 内部低功耗 RC（LPRC）振荡器，具有两种模式：
 - 高功耗 / 高精度模式
 - 低功耗 / 低精度模式

主振荡器和 8 MHz FRC 源可使用内部 4x PLL 模块。FRC 时钟源的频率可通过可编程时钟分频器降低。选择的时钟源提供处理器和外设的时钟。

将处理器时钟源二分频可以产生内部指令周期时钟 Fcy。在本文档中，指令周期时钟由 Fosc/2 表示。在 OSCO I/O 引脚上可以提供内部指令周期时钟 Fosc/2，用于主振荡器的某些工作模式。

9.2 上电复位时的初始配置

通过使用配置位设置选择器件发生上电复位事件时使用的振荡器源（和工作模式）。振荡器配置位设置将存储在程序存储器的配置寄存器中（更多信息，请参见第26.1节“配置位”）。主振荡器配置位POSCMD<1:0>（FOSC<1:0>）和初始振荡器选择配置位FNOSC<2:0>（FOSCSEL<2:0>）用于选择上电复位时使用的振荡器源。带有后分频器的 FRC 主振荡器（FRCDIV）是默认（未编程）的选择。对这些位单元进行编程可以选择辅助振荡器或其中一个内部振荡器。EC 模式频率范围配置位 POSCFREQ<1:0>（FOSC<4:3>）优化了运行于 EC 模式时的功耗。默认配置是“频率范围大于 8 MHz”。

配置位允许用户在多个时钟模式中进行选择，如表 9-1 所示。

9.2.1 时钟切换模式配置位

FCKSMx 配置位（FOSC<7:6>）用于联合配置器件时钟切换和 FSCM。只有将 FCKSM1 编程为 0 时才可以使能时钟切换功能。只有将 FCKSM<1:0> 编程为 00 时才可以使能 FSCM。

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>	备注
带后分频器的 8 MHz FRC 振荡器（FRCDIV）	内部	11	111	1, 2
带后分频器的 500 kHz FRC 振荡器（LPFRCDIV）	内部	11	110	1
低功耗 RC 振荡器（LPRC）	内部	11	101	1
辅助（Timer1）振荡器（SOSC）	辅助	00	100	1
带 PLL 模块的主振荡器（HSPLL）	主	10	011	
带 PLL 模块的主振荡器（ECPLL）	主	00	011	
主振荡器（HS）	主	10	010	
主振荡器（XT）	主	01	010	
主振荡器（EC）	主	00	010	
带 PLL 模块的 8 MHz FRC 振荡器（FRCPLL）	内部	11	001	1
8 MHz FRC 振荡器（FRC）	内部	11	000	1

- 注
- 1: OSCO 引脚功能由 OSCIOFNC 配置位决定。
 - 2: 对于未编程（已擦除）器件，这是默认的振荡器模式。

9.3 控制寄存器

振荡器的运行由三个特殊功能寄存器（SFR）控制：

- OSCCON
- CLKDIV
- OSCTUN

OSCCON 寄存器（寄存器 9-1）是振荡器的主要控制寄存器。它控制时钟源的切换并允许监视时钟源。

时钟分频器寄存器（寄存器 9-2）控制打盹模式的相关特性，以及 FRC 振荡器的后分频器。

FRC 振荡器调节寄存器（寄存器 9-3）允许用户在大约 $\pm 5.25\%$ 的范围内精确地调整 FRC 振荡器。每位的递增和递减都会将 FRC 振荡器的出厂校准频率改变一个固定的量。

寄存器 9-1: OSCCON: 振荡器控制寄存器

U-0	R-0, HSC	R-0, HSC	R-0, HSC	U-0	R/W-x ⁽¹⁾	R/W-x ⁽¹⁾	R/W-x ⁽¹⁾
—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0
bit 15				bit 8			

R/SO-0, HSC	U-0	R-0, HSC ⁽²⁾	U-0	R/CO-0, HS	R/W-0 ⁽³⁾	R/W-0	R/W-0
CLKLOCK	—	LOCK	—	CF	SOSCDRV	SOSCEN	OSWEN
bit 7				bit 0			

图注:	HSC = 可由硬件置 1/ 清零的位		
HS = 可由硬件置 1 的位	CO = 只可清零位	SO = 只可置 1 位	
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 未实现：读为 0

bit 14-12 **COSC<2:0>**：当前振荡器选择位

- 111 = 带后分频器的 8 MHz 快速 RC 振荡器（FRCDIV）
- 110 = 带后分频器的 500 kHz 低功耗快速 RC 振荡器（FRC）（LPFRCDIV）
- 101 = 低功耗 RC 振荡器（LPRC）
- 100 = 辅助振荡器（SOSC）
- 011 = 带 PLL 模块的主振荡器（XTPLL、HSPLL 和 ECPLL）
- 010 = 主振荡器（XT、HS 和 EC）
- 001 = 带后分频器和 PLL 模块的 8 MHz FRC 振荡器（FRCPLL）
- 000 = 8 MHz FRC 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⁽¹⁾

- 111 = 带后分频器的 8 MHz 快速 RC 振荡器（FRCDIV）
- 110 = 带后分频器的 500 kHz 低功耗快速 RC 振荡器（FRC）（LPFRCDIV）
- 101 = 低功耗 RC 振荡器（LPRC）
- 100 = 辅助振荡器（SOSC）
- 011 = 带 PLL 模块的主振荡器（XTPLL、HSPLL 和 ECPLL）
- 010 = 主振荡器（XT、HS 和 EC）
- 001 = 带后分频器和 PLL 模块的 8 MHz FRC 振荡器（FRCPLL）
- 000 = 8 MHz FRC 振荡器（FRC）

注 1： 这些位的复位值由 FNOSC_x 配置位决定。

2： 在进行有效的时钟切换或选择非 PLL 时钟模式时，该位也复位为 0。

3： 当选择从数字时钟输入运行 SOSC 时而不是从外部晶振（SOSC_{SRC} = 0）运行时，该位不受影响。

PIC24FV32KA304 系列

寄存器 9-1: OSCCON: 振荡器控制寄存器 (续)

bit 7	CLKLOCK: 时钟选择锁定使能位 <u>如果使能 FSCM (FCKSM1 = 1):</u> 1 = 时钟和 PLL 选择被锁定 0 = 时钟和 PLL 选择未锁定, 可通过将 OSWEN 位置 1 来进行修改 <u>如果禁止 FSCM (FCKSM1 = 0):</u> 时钟和 PLL 选择始终未锁定, 可通过将 OSWEN 位置 1 来进行修改。
bit 6	未实现: 读为 0
bit 5	LOCK: PLL 锁定状态位 ⁽²⁾ 1 = PLL 模块处于锁定状态或 PLL 模块的起振定时器延时结束 0 = PLL 模块不处于锁定状态, PLL 起振定时器正在运行或 PLL 被禁止
bit 4	未实现: 读为 0
bit 3	CF: 时钟故障检测位 1 = FSCM 检测到一个时钟故障 0 = 未检测到时钟故障
bit 2	SOSCDRV: 辅助振荡器驱动强度位 ⁽³⁾ 1 = 选择高功耗 SOSC 电路 0 = 通过 SOSCSRC 配置位选择高 / 低功耗
bit 1	SOSCEN: 32 kHz 辅助振荡器 (SOSC) 使能位 1 = 使能辅助振荡器 0 = 禁止辅助振荡器
bit 0	OSWEN: 振荡器切换使能位 1 = 启动振荡器切换, 切换到由 NOSC<2:0> 位指定的时钟源 0 = 完成振荡器切换

- 注 1: 这些位的复位值由 FNOSCx 配置位决定。
- 2: 在进行有效的时钟切换或选择非 PLL 时钟模式时, 该位也复位为 0。
- 3: 当选择从数字时钟输入运行 SOSC 时而不是从外部晶振 (SOSCSRC = 0) 运行时, 该位不受影响。

寄存器 9-2: CLKDIV: 时钟分频器寄存器

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1
ROI	DOZE2	DOZE1	DOZE0	DOZEN ⁽¹⁾	RCDIV2	RCDIV1	RCDIV0
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 = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **ROI:** 中断时恢复位
1 = 中断时清零 DOZEN 位并将 CPU/ 外设的时钟比复位为 1:1
0 = 中断不影响 DOZEN 位

bit 14-12 **DOZE<2:0>:** CPU/ 外设的时钟比选择位
111 = 1:128
110 = 1:64
101 = 1:32
100 = 1:16
011 = 1:8
010 = 1:4
001 = 1:2
000 = 1:1

bit 11 **DOZEN:** 打盹使能位 ⁽¹⁾
1 = DOZE<2:0> 位指定 CPU/ 外设时钟比
0 = 将 CPU 和外设时钟比设置为 1:1

bit 10-8 **RCDIV<2:0>:** FRC 后分频比选择位
当 COSC<2:0> (OSCCON<14:12>) = 111 时:
111 = 31.25 kHz (256 分频)
110 = 125 kHz (64 分频)
101 = 250 kHz (32 分频)
100 = 500 kHz (16 分频)
011 = 1 MHz (8 分频)
010 = 2 MHz (4 分频)
001 = 4 MHz (2 分频) (默认)
000 = 8 MHz (1 分频)
当 COSC<2:0> (OSCCON<14:12>) = 110 时:
111 = 1.95 kHz (256 分频)
110 = 7.81 kHz (64 分频)
101 = 15.62 kHz (32 分频)
100 = 31.25 kHz (16 分频)
011 = 62.5 kHz (8 分频)
010 = 125 kHz (4 分频)
001 = 250 MHz (2 分频) (默认)
000 = 500 MHz (1 分频)

bit 7-0 **未实现:** 读为 0

注 1: 当 ROI 位置 1 并发生中断时, 该位会自动清零。

PIC24FV32KA304 系列

寄存器 9-3: 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
—	—	TUN5 ⁽¹⁾	TUN4 ⁽¹⁾	TUN3 ⁽¹⁾	TUN2 ⁽¹⁾	TUN1 ⁽¹⁾	TUN0 ⁽¹⁾
bit 7							bit 0

图注:

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

bit 15-6 未实现：读为 0

bit 5-0 **TUN<5:0>: FRC 振荡器调节位 ⁽¹⁾**

011111 = 最大频率偏移

011110

⋮

000001

000000 = 中心频率，振荡器以出厂校准频率工作

111111

⋮

100001

100000 = 最小频率偏移

注 1: TUN<5:0> 的增加或减少不能在 FRC 调节范围内同步更改 FRC 频率且可能不具有单调性。

9.4 时钟切换工作原理

应用可在软件控制下随时在四个时钟源（POSC、SOSC、FRC 和 LPRC）间自由切换，几乎没有什么限制。为限制该灵活性可能带来的负面影响，PIC24F 器件在时钟切换过程中采用了安全锁定。

注： 主振荡器模式有三种不同的子模式（XT、HS 和 EC），这三种子模式由 POSCMDx 配置位决定。在应用中可以用软件实现从主振荡器模式切换到其他模式，或从其他模式切换到主振荡器模式，但不能在不对器件进行再编程的情况下在主振荡器模式的不同子模式之间进行切换。

9.4.1 使能时钟切换

要使能时钟切换，必须将 FOSC 配置寄存器中的 FCKSM1 配置位编程为 0。（更多详细信息，请参见第 26.0 节“特殊功能”。）如果没有对 FCKSM1 配置位编程（即为 1），则时钟切换功能和 FSCM 功能都将被禁止。这是默认设置。

在时钟切换被禁止的情况下，NOSC_x（OSCCON<10:8>）控制位不控制时钟选择。但是，COSC_x（OSCCON<14:12>）将会反映 FNOSC_x 配置位选择的时钟源。

禁止了时钟切换功能时，OSWEN（OSCCON<0>）控制位不起作用；它始终保持为 0。

9.4.2 振荡器切换序列

执行时钟切换至少需要下列基本序列：

1. 如果需要，读 COSC_x 位（OSCCON<14:12>）确定当前振荡器源。
2. 执行解锁序列以允许写入 OSCCON 寄存器的高字节。
3. 将新振荡器源的相关值写入 NOSC_x（OSCCON<10:8>）位。
4. 执行解锁序列以允许写入 OSCCON 寄存器的低字节。
5. 将 OSWEN 位置 1 来启动振荡器切换。

一旦基本序列完成，系统时钟硬件将自动进行如下响应：

1. 时钟切换硬件将 NOSC_x 控制位的新值和 COSC_x 位做比较。如果相等，时钟切换为冗余操作。在这种情况下，OSWEN 位被自动清零且时钟切换被中止。
2. 如果启动了有效时钟切换，则 LOCK（OSCCON<5>）和 CF（OSCCON<3>）位被清零。
3. 如果新振荡器现在不运行，硬件会将其开启。如果必须开启晶振，硬件将等待直至 OST 超时。如果新振荡源正在使用 PLL，硬件将等待直到检测到 PLL 锁定（LOCK = 1）。
4. 硬件会等待新时钟源的 10 个时钟周期，然后执行时钟切换。
5. 硬件清零 OSWEN 位以表示时钟切换成功。此外，NOSC_x 位的值被传送到 COSC_x 位。
6. 此时旧时钟源被关闭，但 LPRC（如果使能了以 LPRC 为时钟源的 WDT、FSCM 或 RTCC）或 SOSC（如果 SOSCEEN 位保持使能状态）除外。

注 1： 在整个时钟切换过程中，处理器将继续执行代码。对时序敏感的代码不应在此时执行。

2： 不允许直接在带 PLL 的主振荡器模式和 FRCPLL 模式之间进行时钟切换。这适用于任一方向的时钟切换。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。

PIC24FV32KA304 系列

时钟切换的建议代码序列通常包括：

1. 在 OSCCON 寄存器解锁和写序列过程中禁止中断。
2. 通过两条连续的指令将 78h 和 9Ah 写入 OSCCON<15:8>，以执行 OSCCON 高字节的解锁序列。
3. 执行解锁序列之后，立即使用指令将新振荡器源写入 NOSC_x 位。
4. 通过两条连续的指令将 46h 和 57h 写入 OSCCON<7:0>，以执行 OSCCON 低字节的解锁序列。
5. 执行解锁序列之后，立即使用指令将 OSWEN 位置 1。
6. 继续执行对时钟要求不高的代码（可选）。
7. 调用适当时间的软件延时（周期计数），允许选定的振荡器和 / 或 PLL 启动并稳定下来。
8. 检查 OSWEN 位是否为 0。如果是，则切换成功。如果 OSWEN 位仍为置 1 状态，则检查 LOCK 位以确定故障原因。

解锁 OSCCON 寄存器并启动时钟切换的核心序列如例 9-1 所示。

例 9-1: 时钟切换的基本代码序列

```
;Place the new oscillator selection in W0
;OSCCONH (high byte) Unlock Sequence
MOV      #OSCCONH, w1
MOV      #0x78, w2
MOV      #0x9A, w3
MOV.b    w2, [w1]
MOV.b    w3, [w1]
;Set new oscillator selection
MOV.b    WREG, OSCCONH
;OSCCONL (low byte) unlock sequence
MOV      #OSCCONL, w1
MOV      #0x46, w2
MOV      #0x57, w3
MOV.b    w2, [w1]
MOV.b    w3, [w1]
;Start oscillator switch operation
BSET     OSCCON, #0
```

9.5 参考时钟输出

除了某些振荡器模式中使用的 CLK_O 输出（Fosc/2）外，还可对 PIC24FV32KA304 系列器件中的器件时钟进行配置，以为端口引脚提供参考时钟输出信号。此特性可用于所有的振荡器配置，用户可通过它选择更大范围的时钟子倍频以驱动应用中的外部器件。

REFOCON 寄存器（见寄存器 9-4）控制此参考时钟输出。置 1 ROEN 位（REFOCON<15>）可使 REFO 引脚上的时钟信号可用。RODIV_x 位（REFOCON<11:8>）允许选择 16 种不同的时钟分频比选项。

ROSSLP 和 ROSEL 位（REFOCON<13:12>）控制休眠模式期间参考时钟输出是否可用。ROSEL 位决定是 OSC1 和 OSC2 上的振荡器还是当前系统时钟源提供参考时钟输出。ROSSLP 位决定当器件处于休眠模式时 REFO 上的参考时钟源是否可用。

要在休眠模式下使用参考时钟输出，必须将 ROSSLP 和 ROSEL 位都置 1。器件时钟也必须配置为其中一种主模式（EC、HS 或 XT）；否则，如果 ROSEL 位也没有置 1，那么当器件进入休眠模式时，OSCI 和 OSC2 上的振荡器会掉电。任何时钟切换期间，清零 ROSEL 位都会允许参考时钟输出频率随着系统时钟的变化而变化。

寄存器 9-4: 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
ROEN	—	ROSSLP	ROSEL	RODIV3	RODIV2	RODIV1	RODIV0
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 = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **ROEN:** 参考振荡器输出使能位
1 = 使能 REFO 引脚上的参考振荡器
0 = 禁止参考振荡器
- bit 14 **未实现:** 读为 0
- bit 13 **ROSSLP:** 休眠模式下参考振荡器输出停止位
1 = 休眠模式下参考振荡器继续工作
0 = 休眠模式下禁止参考振荡器
- bit 12 **ROSEL:** 参考振荡器源选择位
1 = 主振荡器用作基本时钟 ⁽¹⁾
0 = 系统时钟用作基本时钟; 基本时钟反映了器件的任何时钟切换
- bit 11-8 **RODIV<3:0>:** 参考振荡器分频比选择位
1111 = 基本时钟值的 32,768 分频
1110 = 基本时钟值的 16,384 分频
1101 = 基本时钟值的 8,192 分频
1100 = 基本时钟值的 4,096 分频
1011 = 基本时钟值的 2,048 分频
1010 = 基本时钟值的 1,024 分频
1001 = 基本时钟值的 512 分频
1000 = 基本时钟值的 256 分频
0111 = 基本时钟值的 128 分频
0110 = 基本时钟值的 64 分频
0101 = 基本时钟值的 32 分频
0100 = 基本时钟值的 16 分频
0011 = 基本时钟值的 8 分频
0010 = 基本时钟值的 4 分频
0001 = 基本时钟值的 2 分频
0000 = 基本时钟值
- bit 7-0 **未实现:** 读为 0

注 1: 必须使用 FOSC<2:0> 位使能晶振; 晶振在休眠模式下继续工作。

PIC24FV32KA304 系列

注:

10.0 节能特性

注： 本数据手册总结了 PIC24F 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手册来使用。如需了解更多信息，请参见《PIC24F 系列参考手册》中的第 39 章“带深度休眠的节能特性”(DS39727B_CN)。

PIC24FV32KA304 系列器件提供了管理功耗的功能，该功能是通过选择性地管理 CPU 和外设的时钟来实现的。一般而言，较低的时钟频率和减少时钟源驱动的电路数目会使功耗降低。所有 PIC24F 器件通过以下四种方法管理功耗：

- 时钟频率
- 基于指令的休眠、空闲和深度休眠模式
- 软件控制的打盹模式
- 通过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应用关键特性（如对于时序要求高的通信）的情况下有选择地调节应用的功耗。

10.1 时钟频率和时钟切换

PIC24F 器件允许在应用控制下，从较宽的时钟频率范围内进行选择。如果未锁定系统时钟配置，用户只需更改 NOSC_x 位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”中进行了较为详细的讨论。

10.2 基于指令的节能模式

PIC24F 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运行并停止所有代码执行；空闲模式下 CPU 停止工作，代码停止执行，但是允许外设模块继续工作。深度休眠模式下时钟停止工作、暂停代码执行、外设（除 RTCC 和 DSWDT 外）停止工作、冻结 I/O 状态并除去 SRAM 和闪存程序存储器的电源。

PWRSAV 指令的汇编语法如例 10-1 所示。

注： SLEEP_MODE 和 IDLE_MODE 为所选器件的汇编程序头文件中定义的常量。

允许的中断、WDT 超时或器件复位会导致器件从休眠和空闲模式退出。器件退出这两种模式的过程称为“唤醒”。

10.2.1 休眠模式

休眠模式包含以下这些特性：

- 系统时钟源关闭。如果使用了片上振荡器，也要关闭它。
- 因为 I/O 引脚不消耗电流，器件电流消耗将降至最低。
- 冻结 I/O 引脚方向和状态。
- 由于禁止了系统时钟源，所以故障保护时钟监视器在休眠模式下不工作。
- 如果使能了以 LPRC 作为时钟源的 WDT 或 RTCC，则 LPRC 时钟将继续在休眠模式下运行。
- 如果使能了 WDT，它将在进入休眠模式之前自动清零。
- 某些器件功能或外设可在休眠模式下继续工作。这包括 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。任何需要系统时钟源工作的外设，在休眠模式下都被禁止。

当发生以下任何事件时，器件将从休眠模式唤醒：

- 产生任何被单独允许的中断源
- 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启动。

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
BSET      DSCON, #DSEN    ; Enable Deep Sleep
PWRSAV    #SLEEP_MODE    ; Put the device into Deep SLEEP mode
```

PIC24FV32KA304 系列

10.2.2 空闲模式

空闲模式具有以下这些特性：

- CPU 将停止执行指令。
- WDT 自动清零。
- 系统时钟源保持工作状态。默认情况下，所有外设模块将使用系统时钟源继续正常工作，但也可被有选择地禁止（见第 10.6 节“选择性外设模块控制”）。
- 若使能了 WDT 或 FSCM，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，为 CPU 重新提供时钟，且立即从 PWRSAV 指令之后的下一条指令或 ISR 中的第一条指令开始执行。

10.2.3 在节能指令执行期间的中断

在 PWRSAV 指令执行期间发生的中断都将延迟到进入休眠或空闲模式后才产生，并导致器件从休眠或空闲模式唤醒。

10.2.4 深度休眠模式

在 PIC24FV32KA304 系列器件中，深度休眠模式旨在提供最低程度的功耗，而无需使用外部开关完全关闭器件的所有电源。进入深度休眠模式完全由软件控制。从深度休眠模式退出可通过以下任一事件触发：

- POR 事件
- MCLR 事件
- RTCC 闹钟（如果有 RTCC）
- 外部中断 0
- 深度休眠看门狗定时器（DSWDT）超时
- 超低功耗唤醒（ULPWU）事件

在深度休眠模式下，可保持器件实时时钟和日历（RTCC）运行，从而不会失去时钟周期。

该器件具有专用的深度休眠欠压复位（DSBOR）和深度休眠看门狗定时器复位（DSWDT），用于监视电压和超时事件。DSBOR 和 DSWDT 独立于用于其他功耗管理模式（休眠、空闲和打盹）的标准 BOR 和 WDT。

10.2.4.1 进入深度休眠模式

通过将 DSCON 寄存器中的 DSEN 位置 1，然后执行 SLEEP 命令（PWRSAV #SLEEP_MODE）进入深度休眠模式。需要一个解锁序列将 DSEN 位置 1。一旦将 DSEN 位置 1，执行 SLEEP 命令前无任何时间限制。退出深度休眠模式时，DSEN 位将自动清零。

注： 要在从深度休眠唤醒之后重新进入深度休眠模式，则在清零 RELEASE 位之后，必须经过至少 3 个 Tcy 的延时。

进入深度休眠模式的序列为：

1. 如果应用需要深度休眠 WDT，使能并配置其时钟源。关于深度休眠 WDT 的更多信息，请参见第 10.2.4.5 节“深度休眠 WDT”。
2. 如果应用需要深度休眠 BOR，通过编程 DSLPBOR 配置位（FDS<6>）使能。
3. 如果应用需要根据 RTCC 闹钟从深度休眠唤醒，使能并配置 RTCC 模块。有关 RTCC 的更多信息，请参见第 19.0 节“实时时钟和日历（RTCC）”。
4. 如果需要，通过写入 DSGPR0 和 DSGPR1 寄存器来保护任何关键应用现场数据（可选）。
5. 通过将 DSEN 位（DSCON<15>）置 1 使能深度休眠模式。

注： 需要一个解锁序列将 DSEN 位置 1。

6. 通过发出 PWRSAV #0 指令进入深度休眠模式。

每次 DSEN 位置 1 时，将自动清零 DSWAKE 寄存器中的所有位。

要将 DSEN 位置 1，需要例 10-2 中的解锁序列。

例 10-2: 解锁序列

```
//Disable Interrupts For 5 instructions
asm volatile("disi #5");
//Issue Unlock Sequence
asm volatile
mov #0x55, W0;
mov W0, NVMKEY;
mov #0xAA, W1;
mov W1, NVMKEY;
bset DSCON, #DSEN
```

10.2.4.2 退出深度休眠模式

在发生以下任一事件时退出深度休眠模式：

- VDD 电源发生 POR 事件。如果没有 DSBOR 电路重新激活 VDD 电源 POR 电路，则外部 VDD 电源必须降到 POR 电路的自然激活电压。
- DSWDT 超时。当 DSWDT 定时器超时，器件将会退出深度休眠模式。
- RTCC 闹钟（如果 RTCEN = 1）。
- MCLR 引脚有效（为 0）。
- INT0 引脚为有效（如果在进入深度休眠模式之前允许了中断）。极性配置用于确定引脚的有效电平（0 或 1），将导致器件退出深度休眠模式。在深度休眠模式下，需要在 INT0 引脚上发生电平变化时才会从深度休眠模式退出。

注： 进入深度休眠模式时，所有待处理中断都会被清除。

退出深度休眠模式相当于器件上电复位（POR），通常不会保持原有状态。这种情况的例外情况包括 RTCC 模块（如果有，它在唤醒后将继续工作）、DSGPRx 寄存器和 DSWDT。

此外，从退出深度休眠到 POR 序列完成之前的所有唤醒事件都会被忽略，不会被捕捉到 DSWAKE 寄存器中。

退出深度休眠模式的序列为：

1. 发生唤醒事件后，器件退出深度休眠模式并执行 POR。自动清零 DSEN 位。在复位向量处恢复代码执行。
2. 要确定器件是否退出深度休眠模式，应读取深度休眠位 DPSLP（RCON<10>）。退出深度休眠模式时，该位将置 1。如果已置 1 该位，请清零。
3. 通过读 DSWAKE 寄存器确定唤醒源。
4. 通过读 DSBOR 位（DSCON<1>）确定处于深度休眠模式期间是否发生 DSBOR 事件。
5. 如果应用程序现场数据已保存，则从 DSGPR0 和 DSGPR1 寄存器中读取这些数据。
6. 将 RELEASE 位（DSCON<0>）清零。

10.2.4.3 使用 DSGPR0/DSGPR1 寄存器保护现场数据

由于退出深度休眠模式导致 POR，大部分特殊功能寄存器会复位为其默认 POR 值。另外，由于在深度休眠期间不提供 V_{CORE} 电源，在退出该模式时数据 RAM 中的信息可能丢失。

需要在退出深度休眠模式前保存关键数据的应用可使用深度休眠通用寄存器 DSGPR0 和 DSGPR1 或数据 EEPROM（如果可用）。不同于其他 SFR，在器件处于深度休眠模式时将保留这些寄存器的内容。退出深度休眠后，软件可通过读取这些寄存器并清零 RELEASE 位（DSCON<0>）恢复数据。

10.2.4.4 深度休眠期间的 I/O 引脚

在深度休眠期间，通用 I/O 引脚将保持它们原先的状态且辅助振荡器（SOSC）将保持运行状态（如果使能）。在进入深度休眠模式之前配置为输入（TRISx 位置 1）的引脚在深度休眠期间将保持高阻态。在进入深度休眠模式之前配置为输出（TRISx 位清零）的引脚在深度休眠期间将保持为输出引脚。处于该模式时，它们驱动的输出电平将由进入深度休眠模式时相应的 LATx 位决定。

当器件唤醒后，所有 I/O 引脚将继续维持它们原先的状态，即使器件已经完成 POR 序列，并再次执行应用代码。配置为输入的引脚在深度休眠期间将保持高阻态，配置为输出的引脚将继续驱动原先的值。唤醒之后，TRIS 和 LAT 寄存器和 SOSCEN 位（OSCCON<1>）复位。如果固件修改了这些位或寄存器中的任一个，I/O 不会立即切换到新配置的状态。在固件清零 RELEASE 位（DSCON<0>）之后，I/O 引脚将被“释放”。这会导致 I/O 引脚切换为由它们相应的 TRISx 和 LATx 位值配置的状态。

这意味着，在器件唤醒后保持 SOSC 运行需要在清零 RELEASE 之前将 SOSCEN 位置 1。

如果使能了深度休眠 BOR（DSBOR）且在深度休眠期间发生 DSBOR 或真正的 POR 事件，I/O 引脚将被立即释放，类似于清零 RELEASE 位。所有原先的状态信息都将丢失，包括通用寄存器 DSGPR0 和 DSGPR1 的内容。

如果在深度休眠期间发生 $\overline{\text{MCLR}}$ 复位事件，DSGPRx、DSCON 和 DSWAKE 寄存器将保持有效且 RELEASE 位将保持置 1。还将保持 SOSC 的状态。然而，I/O 引脚将复位为其 $\overline{\text{MCLR}}$ 复位状态。由于 RELEASE 仍置 1，所以更改 SOSCEN 位（OSCCON<1>）不会生效，直到清零 RELEASE 位。

在所有其他深度休眠唤醒情况下，应用固件必须清零 RELEASE 位才能重新配置 I/O 引脚。

10.2.4.5 深度休眠 WDT

要使能深度休眠模式下的 DSWDT，编程配置位 DSWDTEN (FDS<7>)。不需要使能器件看门狗定时器 (WDT) 即可运行 DSWDT。进入深度休眠模式时自动复位 DSWDT。

通过配置位 DSWDTOSC (FDS<4>) 选择 DSWDT 时钟源。通过配置位 DSWDTPS<3:0> (FDS<3:0>) 设定后分频比选项。可实现的最小超时周期为 2.1 ms，最大为 25.7 天。关于 FDS 配置寄存器和 DSWDT 配置选项的更多详细信息，请参见第 26.0 节“特殊功能”。

10.2.4.6 在深度休眠模式下切换时钟

RTCC 和 DSWDT 都可以使用 SOSC 或 LPRC 时钟源工作。这使 RTCC 和 DSWDT 可以在无需同时使能 LPRC 和 SOSC 的情况下运行，从而降低功耗。

RTCC 使用 LPRC 时钟源运行时，RTCC 的精度会降低大约 5-10%。如果要求 RTCC 保持高精度，则它必须使用 SOSC 时钟源。通过 RTCOSC 配置位 (FDS<5>) 选择 RTCC 时钟源。

在某些情况下，当进入深度休眠模式时，DSWDT 的时钟源可能已关闭。这种情况下，将会自动打开时钟源（如果 DSWDT 已使能），而无需软件干预。但是，这会使 DSWDT 计数器的启动产生延时。在使用 SOSC 作为时钟源时，为了避免这种延时，应用程序可在进入深度休眠模式之前的某一时间先激活 SOSC。

10.2.4.7 检查和清除深度休眠状态

在进入深度休眠模式时，状态位 DPSLP (RCON<10>) 会置 1，且必须用软件清零。

在上电时，软件应读取该状态位，以确定是否由于从深度休眠模式退出而发生复位，如果该位置 1，则清零该位。可以考虑 DPSLP 和 POR 位 4 种可能组合中的 3 种情况：

- DPSLP 和 POR 位均清零。这种情况下，复位是由于退出深度休眠模式之外的某个其他事件而产生的。
- DPSLP 位清零，但 POR 位置 1；这是正常的 POR。
- DPSLP 和 POR 位均置 1。这表示发生了以下情况：进入深度休眠模式、器件掉电，然后从深度休眠模式退出。

10.2.4.8 上电复位 (POR)

器件通过监视 VDD 电压来产生 POR。因为退出深度休眠在功能上类似于 POR，所以应使用第 10.2.4.7 节“[检查和清除深度休眠状态](#)”中描述的技术来区分深度休眠和真正的 POR 事件。

发生真正的 POR 时，包括所有深度休眠逻辑（深度休眠寄存器：RTCC 和 DSWDT 等）在内的整个器件将复位。

10.2.4.9 深度休眠序列汇总

为了方便查看，以下为进入和退出深度休眠模式所涉及的必要步骤：

1. 器件退出复位并开始执行其应用代码。
2. 如果需要 DSWDT 功能，则必须设置相应的配置位。
3. 为 DSWDT 和 RTCC 选择相应的时钟（可选）。
4. 使能并配置 DSWDT（可选）。
5. 使能并配置 RTCC（可选）。
6. 将现场数据写入 DSGPRx 寄存器（可选）。
7. 允许 INTO 中断（可选）。
8. 将 DSCON 寄存器中的 DSEN 位置 1。
9. 通过发出 PWRSV #SLEEP_MODE 命令进入深度休眠模式。
10. 器件由于发生唤醒事件而退出深度休眠。
11. 自动清零 DSEN 位。
12. 读取并清零 RCON 中的 DPSLP 状态位和 DSWAKE 状态位。
13. 读 DSGPRx 寄存器（可选）。
14. 完成所有与状态相关的配置之后，将 RELEASE 位清零。
15. 应用程序恢复正常运行。

PIC24FV32KA304 系列

寄存器 10-1: DSCON: 深度休眠控制寄存器 ⁽¹⁾

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
DSEN	—	—	—	—	—	—	RTCCWDIS
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/C-0, HS
—	—	—	—	—	ULPWUDIS	DSBOR ⁽²⁾	RELEASE
bit 7							bit 0

图注:	C = 可清零位	HS = 可由硬件置 1 的位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15 **DSEN:** 深度休眠使能位
1 = 在执行 PWRSAV #0 时进入深度休眠模式
0 = 在执行 PWRSAV #0 时进入正常休眠模式
- bit 14-9 **未实现:** 读为 0
- bit 8 **RTCCWDIS:** RTCC 唤醒禁止位
1 = 从深度休眠唤醒时禁止 RTCC
0 = 从深度休眠唤醒时使能 RTCC
- bit 7-3 **未实现:** 读为 0
- bit 2 **ULPWUDIS:** ULPWU 唤醒禁止位
1 = 从深度休眠唤醒时禁止 ULPWU
0 = 从深度休眠唤醒时使能 ULPWU
- bit 1 **DSBOR:** 深度休眠 BOR 事件位 ⁽²⁾
1 = DSBOR 先前在工作, 并且在深度休眠期间检测到 BOR 事件
0 = DSBOR 先前不工作, 或者先前在工作, 但在深度休眠期间未检测到 BOR 事件
- bit 0 **RELEASE:** I/O 引脚状态释放位
1 = 从深度休眠模式唤醒时, I/O 引脚维持在进入深度休眠模式之前的状态
0 = 将 I/O 引脚从进入深度休眠模式之前的状态释放, 允许相应的 TRISx 和 LATx 位控制它们的状态

- 注 1:** 只有在从深度休眠模式退出产生 POR 事件时, 所有寄存器位才会复位。
- 2:** 不同于所有其他事件, 深度休眠 BOR 事件不会导致从深度休眠模式唤醒; 它会重新激活 POR。

PIC24FV32KA304 系列

寄存器 10-2: **DSWAKE: 深度休眠唤醒源寄存器** ⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0, HS
—	—	—	—	—	—	—	DSINT0
bit 15							bit 8

R/W-0, HS	U-0	U-0	R/W-0, HS	R/W-0, HS	R/W-0, HS	U-0	R/W-0, HS
DSFLT	—	—	DSWDT	DSRTCC	DSMCLR	—	DSPOR ^(2,3)
bit 7							bit 0

图注:	HS = 可由硬件置 1 的位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-9 **未实现:** 读为 0
- bit 8 **DSINT0:** 深度休眠电平变化中断位
1 = 电平变化中断在深度休眠期间发生
0 = 电平变化中断在深度休眠期间未发生
- bit 7 **DSFLT:** 深度休眠故障检测位
1 = 在深度休眠期间发生了故障, 并且某些深度休眠配置设置可能已被破坏
0 = 在深度休眠期间未检测到故障
- bit 6-5 **未实现:** 读为 0
- bit 4 **DSWDT:** 深度休眠看门狗定时器超时位
1 = 深度休眠看门狗定时器在深度休眠期间超时
0 = 深度休眠看门狗定时器在深度休眠期间未超时
- bit 3 **DSRTCC:** 深度休眠实时时钟和日历 (RTCC) 闹钟位
1 = 实时时钟和日历在深度休眠期间触发了一次闹钟
0 = 实时时钟和日历在深度休眠期间未触发闹钟
- bit 2 **DSMCLR:** 深度休眠 MCLR 事件位
1 = MCLR 引脚先前在工作, 并且在深度休眠期间被置为有效
0 = MCLR 引脚先前不工作, 或者先前在工作, 但在深度休眠期间未被置为有效
- bit 1 **未实现:** 读为 0
- bit 0 **DSPOR:** 深度休眠上电复位事件位 ^(2,3)
1 = VDD 电源 POR 电路先前在工作, 并且检测到 POR 事件
0 = VDD 电源 POR 电路先前不工作, 或者先前在工作, 但未检测到 POR 事件

- 注 1: 所有寄存器位在 DSEN (DSCON<15>) 位置 1 时清零。
- 2: 只有在从深度休眠模式退出产生 POR 事件时, 所有寄存器位才会复位; 但 DSPOR 位除外, 该位在由于退出深度休眠而产生 POR 事件时不会复位。
- 3: 不同于该寄存器中的其他位, 该位可以在不处于深度休眠模式时置 1。

10.3 超低功耗唤醒

引脚 RB0 上的超低功耗唤醒（ULPWU），允许缓慢的下降电压来产生中断，而无需过多的电流消耗。

要使用该特性：

1. 通过将 RB0 引脚配置为输出并将其设置为 1 对 RB0 上的电容充电。
2. 通过将 RB0 配置为输入停止对电容充电。
3. 通过将 ULPWCON 寄存器的 ULPEN 和 ULPSINK 位置 1 使电容放电。
4. 配置休眠模式。
5. 进入休眠模式。

当 RB0 上的电压降至 V_{IL} 以下时，器件唤醒并执行下一条指令。

该功能为定期将器件从休眠模式唤醒提供了低功耗技术。

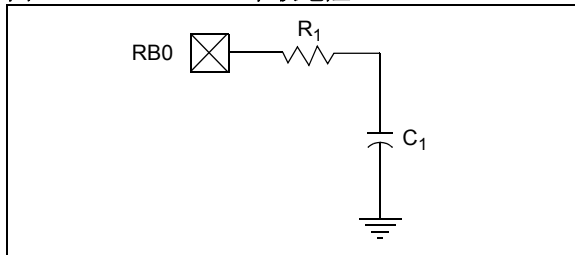
超时取决于 RB0 上 RC 电路的放电时间。

当 ULPWU 模块将器件从休眠模式唤醒时，将 ULPWUIF 位（IFS5<0>）位置 1。唤醒时软件可检查该位以确定唤醒源。

关于初始化 ULPWU 模块，请参见例 10-3。

RB0 和外部电容之间串联的电阻为 RB0/AN0/ULPWU 引脚提供了过流保护并允许通过软件对延迟时间进行校准（见图 10-1）。

图 10-1: 串联电阻



定时器可用于测量电容的充电时间和放电时间。然后可调整充电时间以提供休眠模式所需的延时。该技术弥补了温度、电压和元件精度的影响。也可将该外设配置为一个简单的可编程低电压检测（LVD）或温度传感器。

例 10-3: 初始化超低功耗唤醒

```
//*****
// 1. Charge the capacitor on RB0
//*****

TRISBbits.TRISB0 = 0;
LATBbits.LATB0 = 1;
for(i = 0; i < 10000; i++) Nop();
//*****
//2. Stop Charging the capacitor
//   on RB0
//*****

TRISBbits.TRISB0 = 1;
//*****
//3. Enable ULPWU Interrupt
//*****
IFS5bits.ULPWUIF = 0;
IEC5bits.ULPWUIE = 1;
IPC21bits.ULPWUIP = 0x7;
//*****
//4. Enable the Ultra Low Power
//   Wakeup module and allow
//   capacitor discharge
//*****

ULPWCONbits.ULPEN = 1;
ULPWCONbit.ULPSINK = 1;
//*****
//5. Enter Sleep Mode
//*****

Sleep();
//for sleep, execution will
//resume here
```

PIC24FV32KA304 系列

寄存器 10-3: **ULPWCON: ULPWU 控制寄存器**

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
ULPEN	—	ULPSIDL	—	—	—	—	ULPSINK
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 = 上电复位时的值		1 = 置 1		0 = 清零		x = 未知	

- bit 15

ULPEN: ULPWU 模块使能位
1 = 使能模块
0 = 禁止模块
- bit 14

未实现: 读为 0
- bit 13

ULPSIDL: ULPWU 空闲模式停止选择位
1 = 当器件进入空闲模式时，模块停止工作
0 = 模块在空闲模式下继续工作
- bit 12-9

未实现: 读为 0
- bit 8

ULPSINK: ULPWU 拉电流使能位
1 = 使能拉电流
0 = 禁止拉电流
- bit 7-0

未实现: 读为 0

10.4 基于稳压器的节能特性

PIC24FV32KA304 系列器件具有稳压器，该稳压器可通过改变功能来实现节能。片内稳压器由两个基本模块组成：稳压器（VREG）和保持稳压器（RETREG）。通过组合 VREG 和 RETREG，提供以下功耗模式：

10.4.1 运行模式

在运行模式下，主 VREG 提供一个稳定电压（提供足够大的电流以供器件全速运行），且器件未处于休眠或深度休眠模式下。保持稳压器可能在或不在运行，但未被使用。

10.4.2 休眠（待机）模式

在休眠模式下，器件处于休眠状态且主 VREG 以减少的（待机）供电电流提供稳定电压。由于供电电流减少，该模式提供的功能有限。它需要较长时间从休眠模式唤醒。

10.4.3 保持休眠模式

在保持休眠模式下，器件处于休眠状态且所有稳定电压仅由保持稳压器提供。因此，该模式的功耗比常规休眠模式更低，但具体的功耗与使能了多少功能有关。保持休眠唤醒时间长于休眠模式，是因为需要额外时间来将 V_{CORE} 电源轨提高到正常稳定电压。

注： PIC24F32KA30X 系列器件不具有任何内部稳压器，因此不支持保持休眠模式。

10.4.4 深度休眠模式

在深度休眠模式下，主稳压器和保持稳压器均关闭，从而提供最低器件功耗。然而，该模式不保留任何器件功能且具有最长唤醒时间。

表 10-1: PIC24FV32KA304 系列器件的稳压配置设置

RETCGF 位 (FPOR<2>)	RETEN 位 (RCON<12>)	PMSLP 位 (RCON<8>)	休眠期间的 功耗模式	说明
0	0	1	休眠	VREG 模式（正常）在休眠期间无变化 RETREG 未使用
0	0	0	休眠 (待机)	休眠期间 VREG 进入低电压待机模式 RETREG 未使用
0	1	0	保持 休眠	休眠期间 VREG 关闭 使能 RETREG 并提供休眠稳定电压
1	X	1	休眠	VREG 模式（正常）在休眠期间无变化 始终禁止 RETREG
1	X	0	休眠 (待机)	休眠期间 VREG 进入低电压待机模式 始终禁止 RETREG

10.5 打盹模式

通常，更改时钟速度和进入某种节能模式是降低功耗的首选策略。但在有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，即使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导致通信错误，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。保持两个时钟域同步，以允许外设以较慢的速率执行代码时访问 SFR。

通过置 1 DOZEN 位 (CLKDIV<11>) 使能打盹模式。外设与内核时钟速度之比由 DOZE<2:0> 位 (CLKDIV<14:12>) 决定。有 8 种可能的配置，从 1:1 到 1:128，其中 1:1 是默认设置。

还可使用打盹模式在事件驱动应用中有选择地降低功耗。这样就可以实现不间断地运行对时钟敏感的功能（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，中断事件对打盹工作模式没有影响。

10.6 选择性外设模块控制

空闲和打盹模式允许用户通过降低 CPU 时钟速度或停止 CPU 时钟大幅降低功耗。即使如此，外设模块仍然使用时钟源，因此会有功耗。可能在有些情况下应用需要这些模式无法提供的功能，比如将绝大部分用电资源分配给 CPU 处理，而使外设功耗最低。

PIC24F 器件允许有选择地禁止外设模块，从而降低或消除它们的功耗，以此满足上述需求。可通过两个控制位完成此操作：

- 外设使能位，一般命名为“XXXEN”，位于模块的主控制 SFR 中。
- 外设模块禁止 (PMD) 位，一般命名为“XXXMD”，位于某个 PMD 控制寄存器中。

这两位在使能或禁止相关模块时具有相似的功能。将某个模块的 PMD_x 位置 1 会禁止该模块的所有时钟源，从而将其功耗降至绝对最小值。在此状态下，与此外设相关的控制和状态寄存器也会被禁止，所以写这些寄存器不会有影响且读取值无效。很多外设模块都有对应的 PMD_x 位。

相反，通过清零模块的 XXXEN 位以禁止其功能来禁止该模块，会保留其寄存器的读写功能。这也会降低功耗，但不如将 PMD_x 位置 1 降低得多。大多数外设模块都有一个使能位，捕捉、比较和 RTCC 模块除外。

要实现更多可选择的节能，可在器件进入空闲模式后有选择地禁止外设模块。这可通过通用名称格式为“XXXIDL”的控制位实现。默认情况下，所有能在空闲模式下工作的模块都适用。在空闲模式下使用空闲模式禁止功能禁止模块，使功耗在空闲模式下得到进一步降低，从而增强了对功耗要求异常严格的应用的节能功能。

11.0 I/O 端口

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解有关 I/O 端口的更多信息，请参见《PIC24F 系列参考手册》中的第 12 章“带外设引脚选择 (PPS) 的 I/O 端口” (DS39711B_CN)。请注意 PIC24FV32KA304 系列器件不支持外设引脚选择功能。

所有器件引脚（除了 VDD 和 VSS 以外）均由外设和并行 I/O 端口共用。所有 I/O 输入端口都为施密特触发器输入，以便增强抗噪声能力。

11.1 并行 I/O (PIO) 端口

通常，与某个外设共用一个引脚的并行 I/O 端口总是服从于该外设。外设的输出缓冲器数据和控制信号提供一对多路开关。这对多路开关用于选择 I/O 引脚的输出数据和控制信号是用于外设还是相应的端口。该逻辑同时还会阻止“环回进入” (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输入。图 11-1 中显示端口是如何与其他外设及其连接的关联 I/O 引脚复用的。

当使能某外设并驱动与其相对应的引脚时，将禁止此引脚的通用输出功能。可以读该 I/O 引脚，但并行端口位的输出驱动器将被禁止。若使能某外设但没有驱动引脚时，则该引脚可由一个端口驱动。

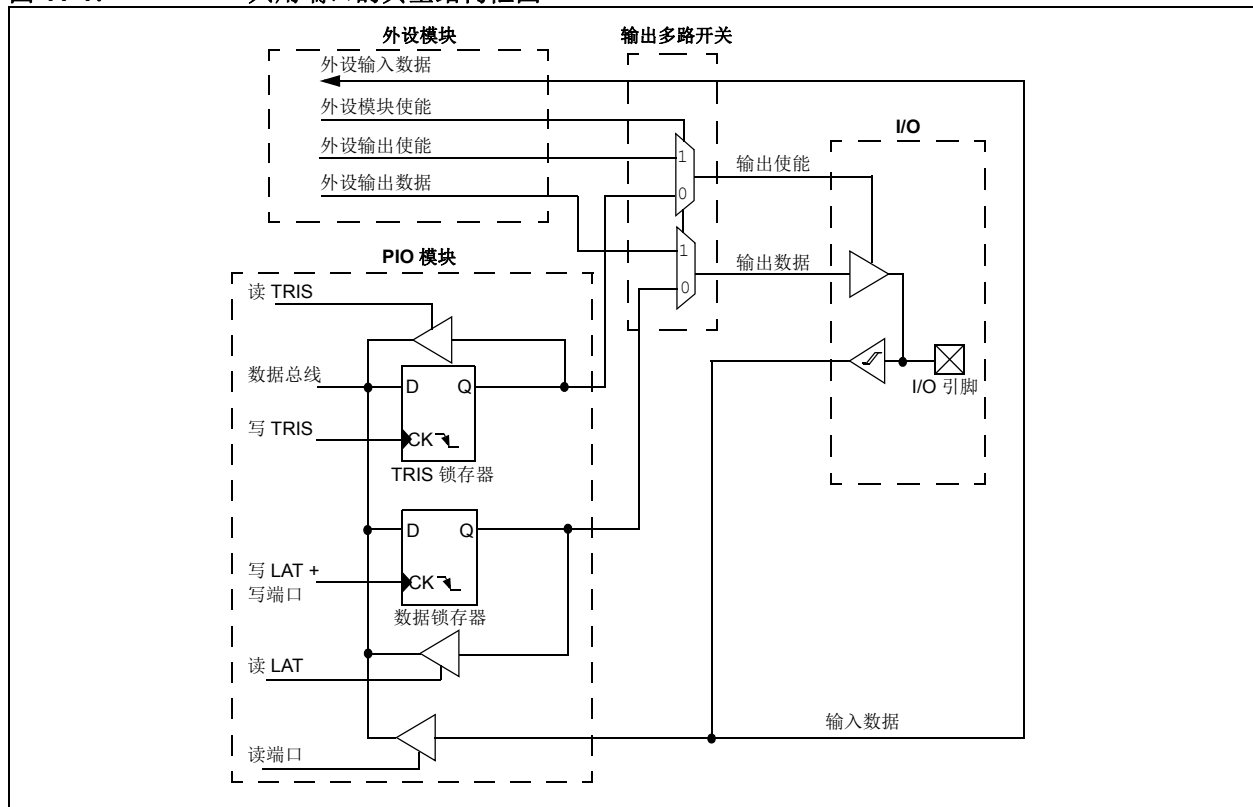
所有端口引脚都有三个与端口引脚作为数字 I/O 时的工作直接相关的寄存器。数据方向寄存器 (TRISx) 决定引脚是输入引脚还是输出引脚。如果数据方向位为 1，则为输入引脚。复位以后，所有端口引脚被定义为输入引脚。读数据锁存寄存器 (LATx) 时，读到的是锁存器中的值；写锁存器时，写入的是锁存器。但读取端口 (PORTx) 时，读到的是端口引脚的值；而写入端口引脚时，写入的是相应的锁存器。

对于某个特定器件，无效的位及其相关的数据和控制寄存器都将被禁止。这意味着相应的 LATx 和 TRISx 寄存器以及该端口引脚将读为 0。

当一个定义为只用作输入的引脚与另一个外设或功能复用，由于没有其他竞争的输入源，它将被视为专用端口。

注： 在深度休眠期间，I/O 引脚保持原有状态。在唤醒时它们将保持该状态，直到软件恢复 (RELEASE) 被清零。

图 11-1: 共用端口的典型结构框图



PIC24FV32KA304 系列

11.1.1 漏极开路配置

除端口、LAT 和 TRIS 寄存器用于数据控制外，每个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对应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。

允许的最大漏极开路电压与最大 V_{IH} 规范相同。

11.2 配置模拟端口引脚

ANS 和 TRIS 寄存器用于控制 A/D 端口引脚的操作。必须将需要作为模拟输入的端口引脚的相应 TRISx 位置 1（输入）。如果 TRISx 位清零（输出），将转换数字输出电平（ V_{OH} 或 V_{OL} ）。

当对端口寄存器进行读操作时，所有配置为模拟输入通道的引脚将被读为零（低电平）。定义为数字输入的任何引脚（包括 ANx 引脚）上的模拟电平都可能导致输入缓冲器消耗的电流超出器件规范。

11.2.1 模拟选择寄存器

具有共享模拟功能的 I/O 引脚（如 A/D 输入和比较器输入）在使用模拟功能时必须将其数字输入功能关闭。请注意，模拟功能包括一个可从外部施加到引脚的模拟电压。

为了进行模拟控制，提供了 ANSx 寄存器。每个端口都有一个 ANS 寄存器（ANSA、ANSB 和 ANSC）。每个 ANSx 寄存器内，每个模拟功能和数字 I/O 功能共用的引脚都有一个对应位。

如果特定引脚没有模拟功能，将不实现该位。要了解实现情况，请参见寄存器 11-1 到寄存器 11-3。

寄存器 11-1: ANSA: 模拟选择寄存器 (PORTA)

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-1	R/W-1	R/W-1	R/W-1
—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0
bit 7				bit 0			

图注:			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-4 未实现：读为 0

bit 3-0 ANSA<3:0>: 模拟选择控制位

 1 = 数字输入缓冲器不工作（用于模拟输入）

 0 = 数字输入缓冲器工作

PIC24FV32KA304 系列

寄存器 11-2: ANSB: 模拟选择寄存器 (PORTB)

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	U-0
ANSB15	ANSB14	ANSB13	ANSB12	—	—	—	—
bit 15				bit 8			
U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	ANSB4	ANSB3 ⁽¹⁾	ANSB2	ANSB1	ANSB0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-12 **ANSB<15:12>**: 模拟选择控制位
 1 = 数字输入缓冲器不工作 (用于模拟输入)
 0 = 数字输入缓冲器工作
- bit 11-5 **未实现**: 读为 0
- bit 4-0 **ANSB<4:0>**: 模拟选择控制位 ⁽¹⁾
 1 = 数字输入缓冲器不工作 (用于模拟输入)
 0 = 数字输入缓冲器工作

注 1: ANSB3 位在 20 引脚器件中不可用。

寄存器 11-3: ANSC: 模拟选择寄存器 (PORTC)

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-1	R/W-1	R/W-1
—	—	—	—	—	ANSC2 ⁽¹⁾	ANSC1 ⁽¹⁾	ANSC0 ⁽¹⁾
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-3 **未实现**: 读为 0
- bit 2-0 **ANSC<2:0>**: 模拟选择控制位 ⁽¹⁾
 1 = 数字输入缓冲器不工作 (用于模拟输入)
 0 = 数字输入缓冲器工作

注 1: 这些位在 20 引脚或 28 引脚器件中不可用。

PIC24FV32KA304 系列

11.2.2 I/O 端口写 / 读时序

端口方向改变或端口写操作与对同一端口的读操作之间需要一个指令周期。这通常通过一条 NOP 指令来实现。

11.3 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 PIC24FV32KA304 系列器件在选定输入引脚的电平状态变化 (Change-Of-State, COS) 时向处理器发出中断请求。即便是在时钟被禁止的休眠模式下, 该特性也可检测到输入电平状态变化。根据器件的引脚数, 最多可以选择 (允许) 23 个外部信号 (CN0 到 CN22) 在电平状态发生变化时产生中断请求。

有六个与 ICN 模块关联的控制寄存器。CNEN1 和 CNEN2 寄存器包含每个 CN 输入引脚的中断允许控制位。将其中任一位置 1 将允许相应引脚的 CN 中断。

每个 CN 引脚也都有一个与之相连的弱上拉 / 下拉电路。上拉电路提供连接到该引脚的拉电流。下拉电路提供灌电流, 以便在连接按钮或键盘设备时不需要使用外部电阻。

在任何引脚上, 都只应使能上拉电阻或下拉电阻, 而不应同时使能两者。如果按钮或键盘连接到 VDD, 则使能下拉电阻; 如果连接到 VSS, 则使能上拉电阻。上拉电路将使用 CNPU1 和 CNPU2 寄存器 (包含每个 CN 引脚的控制位) 分别使能。

将任意控制位置 1 都会使能相应引脚的弱上拉电路。下拉电路将使用 CNPD1 和 CNPD2 寄存器 (包含每个 CN 引脚的控制位) 分别使能。将任意控制位置 1 都会使能相应引脚的弱下拉电路。

选择内部上拉电路时, 引脚将使用 VDD 作为上拉源电压。选择内部下拉电路时, 内部电阻将引脚电平下拉至 VSS。使能内部上拉 / 下拉电路时, 请确保没有外部上拉拉电流和下拉灌电流。

注: 只要端口引脚被配置为数字输出引脚, 电平变化通知引脚上的上拉电路和下拉电路将始终被禁止。

例 11-1: 端口写 / 读示例

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

Equivalent 'C' Code
TRISB = 0xFF00;               //Configure PORTB<15:8> as inputs and PORTB<7:0> as outputs
NOP();                         //Delay 1 cycle
if(PORTBbits.RB13 == 1)      // execute following code if PORTB pin 13 is set.
{
}
```

12.0 TIMER1

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解定时器的更多信息，请参见《PIC24F 系列参考手册》中的第14章“定时器”（DS39704A_CN）。

Timer1 模块是一个 16 位定时器，可作为实时时钟（RTC）的时间计数器，或作为独立运行的间隔定时器 / 计数器。Timer1 可在以下三种模式下工作：

- 16 位定时器
- 16 位同步计数器
- 16 位异步计数器

而且，Timer1 还支持下列特性：

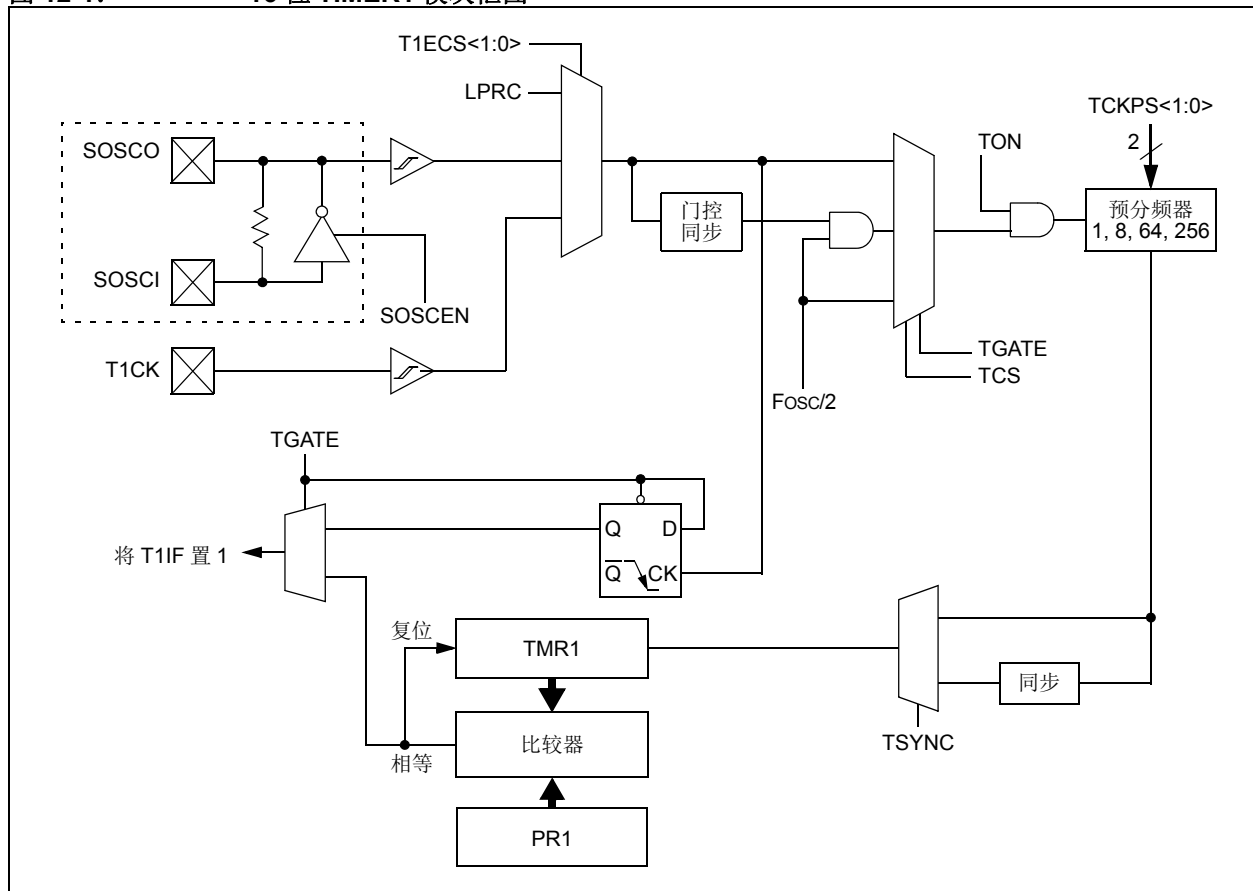
- 定时器门控操作
- 可选的预分频比设置
- CPU 空闲和休眠模式下的定时器操作
- 在 16 位周期寄存器匹配或外部门控信号的下降沿产生中断

图 12-1 给出了 16 位 Timer1 模块的框图。

配置 Timer1 的步骤：

1. 将 TON 位置 1（= 1）。
2. 使用 TCKPS<1:0> 位选择定时器预分频比。
3. 使用 TCS 和 TGATE 位设置时钟和门控模式。
4. 将 TSYNC 位置 1 或清零以配置同步或异步操作。
5. 将定时器周期值装入 PR1 寄存器。
6. 如果需要中断，则将 Timer1 中断允许位 T1IE 置 1。使用 Timer1 中断优先级位 T1IP<2:0> 来设置中断的优先级。

图 12-1: 16 位 TIMER1 模块框图



PIC24FV32KA304 系列

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	R/W-0	R/W-0
TON	—	TSIDL	—	—	—	T1ECS1 ⁽¹⁾	T1ECS0 ⁽¹⁾
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	TCS	—
bit 7						bit 0	

图注:

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

- bit 15 **TON:** Timer1 使能位
 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 = 模块在空闲模式下继续工作
- bit 12-10 **未实现:** 读为 0
- bit 9-8 **T1ECS <1:0>:** Timer1 扩展时钟选择位 ⁽¹⁾
 11 = 保留; 不要使用
 10 = Timer1 使用 LPRC 作为时钟源
 01 = Timer1 使用 T1CK 的外部时钟
 00 = Timer1 使用辅助振荡器 (SOSC) 作为时钟源
- bit 7 **未实现:** 读为 0
- bit 6 **TGATE:** Timer1 门控时间累加使能位
 当 TCS = 1 时:
 此位为无关位。
 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
 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:
 此位为无关位。
- bit 1 **TCS:** Timer1 时钟源选择位
 1 = Timer1 时钟源由 T1ECS<1:0> 选择
 0 = 内部时钟 (Fosc/2)
- bit 0 **未实现:** 读为 0

注 1: 仅当 TCS = 1 时 T1ECSx 位才有效。

13.0 TIMER2/3 和 TIMER4/5

注： 本数据手册总结了PIC24F系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解定时器的更多信息，请参见《PIC24F系列参考手册》中的第14章“定时器”（DS39704A_CN）。

Timer2/3 和 Timer4/5 模块各为一个 32 位定时器，它们也可以被配置为 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 可以在以下三种模式下工作：

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

它们还支持以下功能：

- 定时器门控操作
- 可选的预分频比设置
- 空闲模式下作为定时器工作
- 在 32 位周期寄存器匹配时产生中断
- 触发 A/D 事件

在作为 16 位定时器时，所有 4 个定时器都可以单独用作同步定时器或计数器。它们也可以提供上面列出的功能，但触发 A/D 事件除外（触发 A/D 事件仅能用 Timer3 实现）。工作模式和使能的功能由 T2CON、T3CON、T4CON 和 T5CON 寄存器中的相应位的设置决定。寄存器 13-1 和寄存器 13-2 中分别给出了 T2CON、T3CON、T4CON 和 T5CON 寄存器的一般形式。

当工作在 32 位定时器/计数器模式下时，Timer2/Timer4 作为 32 位定时器的低位字（lsw），而 Timer3/Timer5 则作为其高位字（msw）。

注： 当定时器工作在 32 位模式时，将不使用 T3CON 或 T5CON 控制位。设置和控制只使用 T2CON 和 T4CON 控制位。32 位定时器模块使用 Timer2 和 Timer4 时钟和门控输入，但中断由 Timer3 或 Timer5 中断标志产生。

要将 Timer2/3 或 Timer4/5 配置为 32 位工作模式：

1. 将 T32 位置 1（T2CON<3> 或 T4CON<3> = 1）。
2. 使用 TCKPS<1:0> 位选择 Timer2 或 Timer4 的预分频比。
3. 使用 TCS 和 TGATE 位设置时钟和门控模式。
4. 装载定时器周期值。PR3（或 PR5）将存储该值的高位字，而 PR2（或 PR4）则存储低位字。
5. 如果需要中断，则将 Timerx 中断允许位 TxIE 置 1。使用 Timerx 中断优先级位 TxIP<2:0> 设置中断优先级。
6. 将 TON 位置 1（TxCON<15> = 1）。

任意时刻的定时器值都将存储在寄存器对 TMR3:TMR2（或 TMR5:TMR4）。TMR3（TMR5）始终存储计数值的高位字，而 TMR2（TMR4）存储低位字。

要将任何定时器配置为独立的 16 位工作模式：

1. 将与该定时器对应的位 T32 清零（对于 Timer2 和 Timer3 来说是 T2CON<3>；对于 Timer4 和 Timer5 来说是 T4CON<3>）。
2. 使用 TCKPS<1:0> 位选择定时器预分频比。
3. 使用 TCS 和 TGATE 位设置时钟和门控模式。
4. 将定时器周期值装入 PRx 寄存器。
5. 如果需要中断，将 Timerx 中断允许位 TxIE 置 1，并使用 Timerx 中断优先级位 TxIP<2:0> 设置中断优先级。
6. 将 TON 位置 1（TxCON<15> = 1）。

PIC24FV32KA304 系列

图 13-1: TIMER2/3 和 TIMER4/5 (32 位) 框图

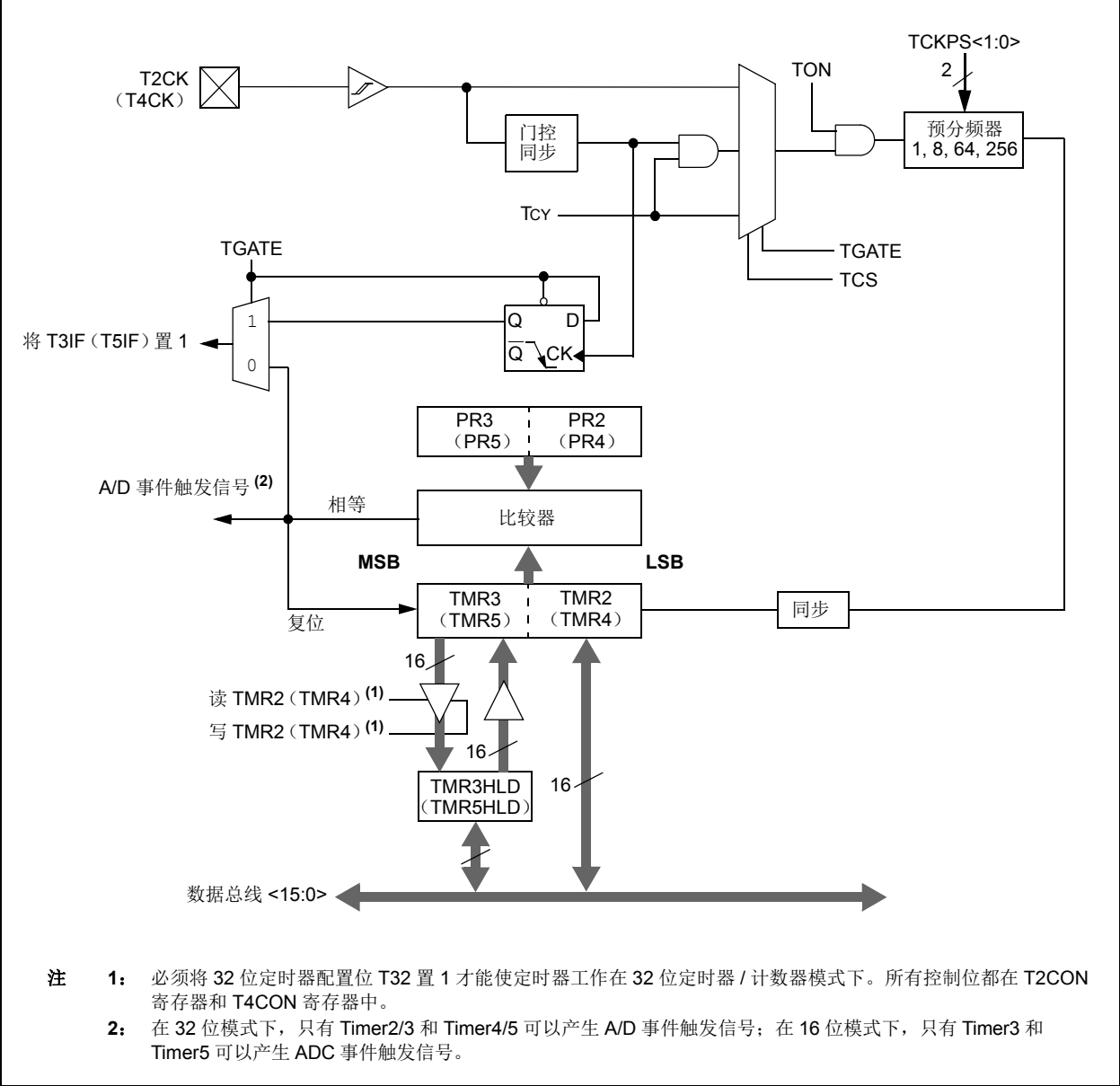


图 13-2: TIMER2 和 TIMER4 (16 位同步) 框图

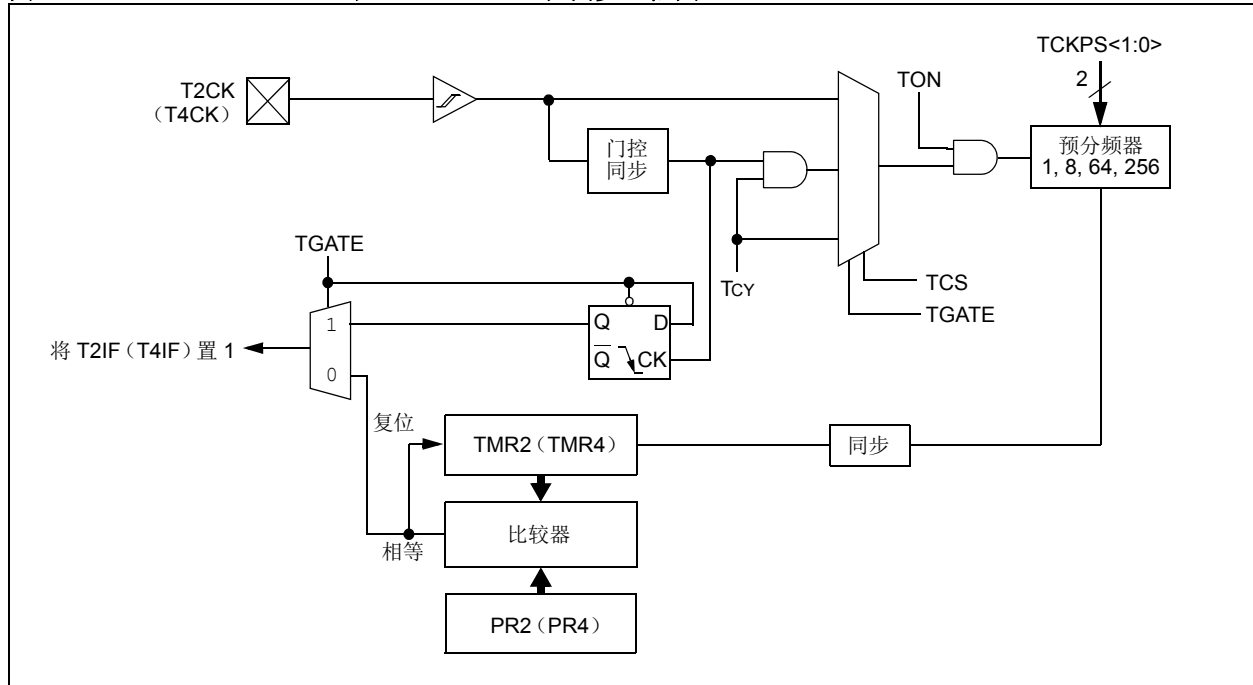
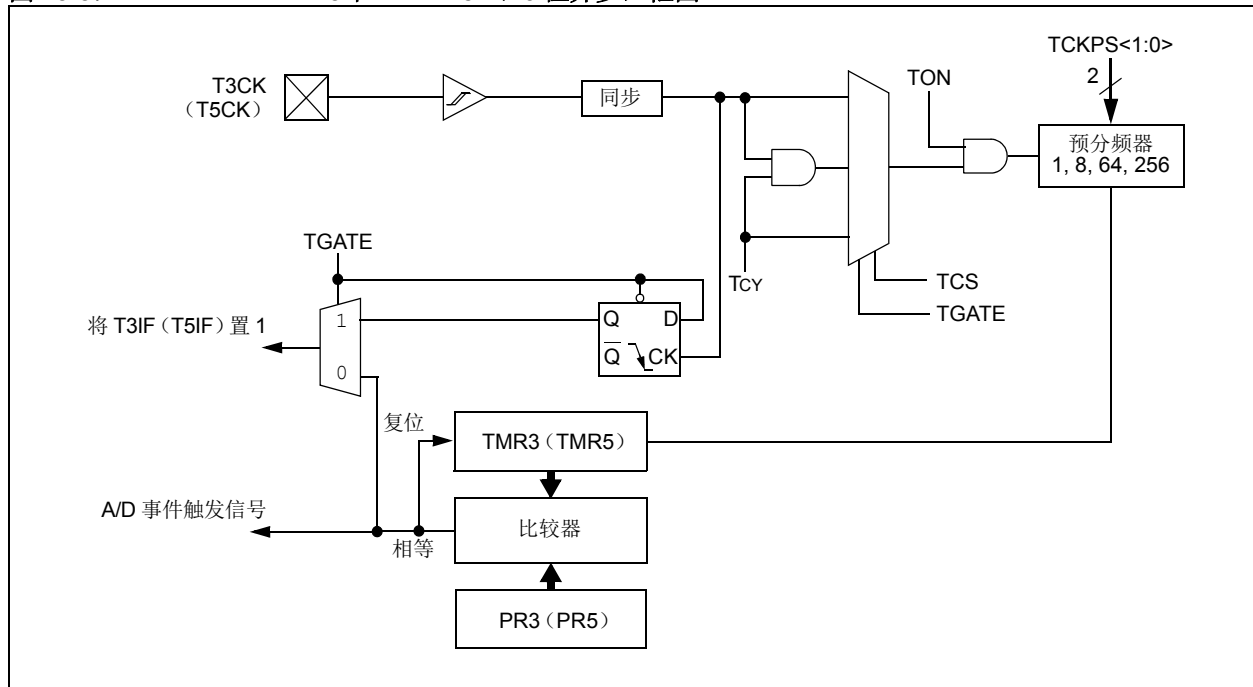


图 13-3: TIMER3 和 TIMER5 (16 位异步) 框图



PIC24FV32KA304 系列

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 ⁽¹⁾	—	TCS	—
bit 7						bit 0	

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	TON: Timerx 使能位 <u>当 TxCON<3> = 1 时:</u> 1 = 启动 32 位 Timerx/y 0 = 停止 32 位 Timerx/y <u>当 TxCON<3> = 0 时:</u> 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 = 模块在空闲模式下继续工作
bit 12-7	未实现: 读为 0
bit 6	TGATE: Timerx 门控时间累加使能位 <u>当 TCS = 1 时:</u> 此位为无关位。 <u>当 TCS = 0 时:</u> 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⁾ 1 = Timer2 和 Timer3 或 Timer4 和 Timer5 构成一个 32 位定时器 0 = Timer2 和 Timer3 或 Timer4 和 Timer5 作为两个 16 位定时器
bit 2	未实现: 读为 0
bit 1	TCS: Timerx 时钟源选择位 1 = TxCK 引脚的外部时钟的上升沿 0 = 内部时钟 (Fosc/2)
bit 0	未实现: 读为 0

注 1: 在 32 位模式下, T3CON 或 T5CON 控制位不影响 32 位定时器的。

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 ⁽¹⁾	—	TSIDL ⁽¹⁾	—	—	—	—	—
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 ⁽¹⁾	TCKPS1 ⁽¹⁾	TCKPS0 ⁽¹⁾	—	—	TCS ⁽¹⁾	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **TON:** Timery 使能位 ⁽¹⁾
 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 ⁽¹⁾
 1 = 当器件进入空闲模式时, 模块停止工作
 0 = 模块在空闲模式下继续工作
- bit 12-7 **未实现:** 读为 0
- bit 6 **TGATE:** Timery 门控时间累加使能位 ⁽¹⁾
 当 TCS = 1 时:
 此位为无关位。
 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 ⁽¹⁾
 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 ⁽¹⁾
 1 = T3CK 引脚的外部时钟的上升沿
 0 = 内部时钟 (Fosc/2)
- bit 0 **未实现:** 读为 0

注 1: 当使能 32 位工作模式时 (TxCON<3> = 1), 这些位不会对 Timery 的工作产生影响。定时器的所有功能都通过 TxCON 寄存器设置。

PIC24FV32KA304 系列

注:

14.0 带专用定时器的输入捕捉

注： 本数据手册总结了PIC24F系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解更多信息，请参见《PIC24F系列参考手册》中的第34章“带专用定时器的输入捕捉”（DS39722A_CN）。

PIC24FV32KA304系列中的所有器件都具有3个独立的输入捕捉模块。每个模块都为捕捉外部脉冲事件和产生中断提供了各种配置和工作选项。

输入捕捉模块的主要特性包括：

- 通过级联两个相邻的模块，采用硬件方式配置为所有的32位工作模式
- 输出比较操作有同步和触发两种模式，最多具有20个用户可选的同步/触发源
- 一个4级深的FIFO缓冲区，用于捕捉并保存多个事件的定时器值
- 可配置中断产生
- 为每个模块提供了最多6个时钟源，驱动独立的内部16位计数器

此模块可通过两个寄存器进行控制：ICxCON1（寄存器14-1）和ICxCON2（寄存器14-2）。模块的一般框图如图14-1所示。

14.1 一般工作模式

14.1.1 同步和触发模式

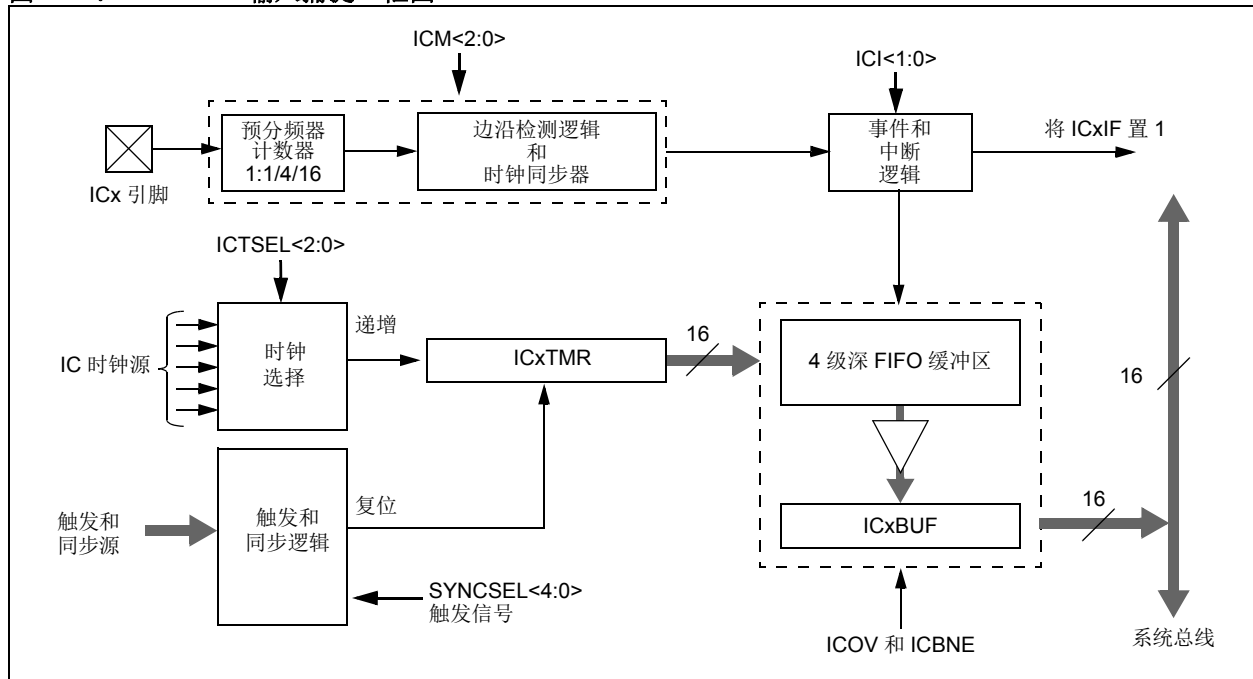
默认情况下，输入捕捉模块工作在自由运行模式下。内部16位计数器ICxTMR可连续向上计数，并在每次溢出时从FFFFh计满返回0000h，其周期与所选外部时钟源同步。发生捕捉事件时，将内部计数器的当前16位值写入FIFO缓冲区。

在同步模式下，一旦使能了所选的时钟源，模块就开始捕捉ICx引脚上的事件。只要所选的同步源发生事件，内部计数器就会复位。在触发模式下，模块等待来自其他内部模块的同步事件发生后，才允许内部计数器运行。

通过把SYNCSELx位设置为00000并清零ICTRIG位（ICxCON2<7>）选择标准的自由运行工作模式。当SYNCSELx位设置为除00000以外的其他值时，将选择同步和触发模式。ICTRIG位选择同步或触发模式；置1该位将选择触发工作模式。在这两种模式下，SYNCSELx位用于决定同步/触发源。

当SYNCSELx位设置为00000且ICTRIG置1时，模块工作在软件触发模式下。在此情况下，可通过手动将TRIGSTAT位（ICxCON2<6>）置1来启动捕捉操作。

图 14-1: 输入捕捉 x 框图



PIC24FV32KA304 系列

14.1.2 级联（32 位）模式

默认情况下，每个模块都可以使用自己的 16 位定时器工作。要提高分辨率，可将相邻的偶数编号的模块和奇数编号的模块配置为一个 32 位模块。（例如，模块 1 和 2 配对，模块 3 和 4 配对等）。奇数编号的模块（ICx）提供 32 位寄存器对中的低 16 位，而偶数编号的模块（ICy）提供高 16 位。ICx 寄存器的计满返回操作会导致其相应的 ICy 寄存器递增。

通过将两个模块的 IC32 位（ICxCON2<8>）置 1 使用硬件来配置级联操作。

14.2 捕捉操作

输入捕捉模块可配置为在 ICx 信号的上升沿或所有跳变沿处捕捉定时器值并产生中断。捕捉可配置为在所有上升沿上发生，或仅在某些上升沿（例如每 4 个或 16 个上升沿）发生。可单独配置中断以在每个事件或一组事件发生时产生中断。

将模块设置为捕捉操作：

1. 如果要使用同步模式，应在继续前禁止同步源。
2. 确保 ICBNE 位（ICxCON1<3>）清零之前已通过读 ICxBUF 从 FIFO 中除去了所有以前的数据。
3. 设置 SYNCSELx 位（ICxCON2<4:0>）以得到所需的同步 / 触发源。
4. 设置 ICTSELx 位（ICxCON1<12:10>）以得到所需的时钟源。如果所需的时钟源正在运行，请在使能输入捕捉模块前设置 ICTSELx 位以与所需的时钟源正确同步。
5. 设置 ICIX 位（ICxCON1<6:5>）以得到所需的中断频率。
6. 选择同步或触发工作模式：
 - a) 确认 SYNCSELx 位未设置为 00000。
 - b) 对于同步模式，清零 ICTRIG 位（ICxCON2<7>）。
 - c) 对于触发模式，将 ICTRIG 位置 1 并清零 TRIGSTAT 位（ICxCON2<6>）。
7. 设置 ICMx 位（ICxCON1<2:0>）以得到所需的工作模式。
8. 使能所选的同步 / 触发源。

对于 32 位级联操作，设置过程稍有不同：

1. 将两个模块的 IC32 位（ICyCON2<8> 和 ICxCON2<8>）置 1，首先使能偶数编号的模块。这可以确保两个模块可以一起启动。
2. 分别设置两个模块的 ICTSELx 和 SYNCSELx 位以选择相同的同步 / 触发模式和时基源。首先设置偶数编号的模块，然后设置奇数编号的模块。两个模块必须使用相同的 ICTSELx 和 SYNCSELx 位设置。
3. 清零偶数编号的模块的 ICTRIG 位（ICyCON2<7>）。这可以强制该模块与奇数编号的模块同步运行（不管其触发设置如何）。
4. 使用奇数编号的模块的 ICIX 位（ICxCON1<6:5>）设置所需的中断频率。
5. 使用奇数编号的模块的 ICTRIG 位（ICxCON2<7>）配置触发或同步工作模式。

注： 对于同步工作模式，最后一个步骤是使能同步源。使能同步源之前，两个输入捕捉模块都保持复位状态。

6. 使用奇数编号的模块的 ICMx 位（ICxCON1<2:0>）设置所需的捕捉模式。

使能了时基和同步 / 触发源之后，模块准备就绪，可以捕捉事件了。ICBNE 位（ICxCON1<3>）置 1 时，FIFO 中至少有一个捕捉值。从 FIFO 读取输入捕捉值，直到 ICBNE 清零。

对于 32 位操作，读取 ICxBUF 和 ICyBUF 以得到整个 32 位定时器值（ICxBUF 用于低 16 位，ICyBUF 用于高 16 位）。当奇数编号的模块的 ICBNE 位（ICxCON1<3>）置 1 时，FIFO 缓冲区中至少有一个捕捉值。继续读缓冲寄存器直到 ICBNE 清零（由硬件自动执行）。

PIC24FV32KA304 系列

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, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0
—	ICI1	ICI0	ICOV	ICBNE	ICM2	ICM1	ICM0
bit 7						bit 0	

图注:	HSC = 可由硬件置 1/ 清零的位						
R = 可读位	W = 可写位		U = 未实现位, 读为 0				
-n = 上电复位时的值	1 = 置 1		0 = 清零		x = 未知		

bit 15-14 **未实现:** 读为 0

bit 13 **ICSIDL:** 空闲时输入捕捉 x 模块停止控制位
 1 = 输入捕捉模块将在 CPU 空闲模式下停止工作
 0 = 输入捕捉模块将在 CPU 空闲模式下继续工作

bit 12-10 **ICTSEL<2:0>:** 输入捕捉 x 定时器选择位
 111 = 系统时钟 (Fosc/2)
 110 = 保留
 101 = 保留
 100 = Timer1
 011 = Timer5
 010 = Timer4
 001 = Timer2
 000 = Timer3

bit 9-7 **未实现:** 读为 0

bit 6-5 **ICI<1:0>:** 选择每次中断发生的捕捉次数的位
 11 = 每四次捕捉事件中断一次
 10 = 每三次捕捉事件中断一次
 01 = 每两次捕捉事件中断一次
 00 = 每次捕捉事件中断一次

bit 4 **ICOV:** 输入捕捉 x 溢出状态标志位 (只读)
 1 = 已发生输入捕捉溢出
 0 = 未发生输入捕捉溢出

bit 3 **ICBNE:** 输入捕捉 x 缓冲区空状态位 (只读)
 1 = 输入捕捉缓冲区不为空, 至少可以读取一个以上的捕捉值
 0 = 输入捕捉缓冲区为空

bit 2-0 **ICM<2:0>:** 输入捕捉模式选择位
 111 = 中断模式: 只有当器件处于休眠或空闲模式时输入捕捉才可作为中断引脚 (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 = 预分频器捕捉模式: 每 16 个上升沿捕捉一次
 100 = 预分频器捕捉模式: 每 4 个上升沿捕捉一次
 011 = 简单捕捉模式: 每个上升沿捕捉一次
 010 = 简单捕捉模式: 每个下降沿捕捉一次
 001 = 边沿检测捕捉模式: 每个沿 (上升和下降) 捕捉一次, ICI<1:0> 位不控制此模式下中断的产生
 000 = 输入捕捉模块关闭

PIC24FV32KA304 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	TRIGSTAT	—	SYNCSEL4	SYNCSEL3	SYNCSEL2	SYNCSEL1	SYNCSEL0
bit 7							bit 0

图注: HS = 可由硬件置 1 的位
R = 可读位 W = 可写位 U = 未实现位, 读为 0
-n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

- bit 15-9 未实现: 读为 0
- bit 8 **IC32:** 级联两个输入捕捉模块使能位 (32 位操作)
1 = ICx 和 ICy 以级联的形式作为 32 位模块使用 (两个模块中的此位都必须置 1)
0 = ICx 作为 16 位模块独立工作
- bit 7 **ICTRIG:** 输入捕捉 x 同步 / 触发选择位
1 = 用 SYNCSELx 位指定的源触发 ICx
0 = 使 ICx 与 SYNCSELx 位指定的源同步
- bit 6 **TRIGSTAT:** 定时器触发状态位
1 = 定时器源被触发且正在运行 (该位可用硬件置 1, 也可用软件置 1)
0 = 定时器源未被触发, 保持清零状态
- bit 5 未实现: 读为 0
- bit 4-0 **SYNCSEL<4:0>:** 触发 / 同步源选择位
11111 = 保留
11110 = 保留
11101 = 保留
11100 = CTMU⁽¹⁾
11011 = A/D⁽¹⁾
11010 = 比较器 3⁽¹⁾
11001 = 比较器 2⁽¹⁾
11000 = 比较器 1⁽¹⁾
10111 = 输入捕捉 4
10110 = 输入捕捉 3
10101 = 输入捕捉 2
10100 = 输入捕捉 1
10011 = 保留
10010 = 保留
1000x = 保留
01111 = Timer5
01110 = Timer4
01101 = Timer3
01100 = Timer2
01011 = Timer1
01010 = 输入捕捉 5
01001 = 保留
01000 = 保留
00111 = 保留
00110 = 保留
00101 = 输出比较 5
00100 = 输出比较 4
00011 = 输出比较 3
00010 = 输出比较 2
00001 = 输出比较 1
00000 = 不与任何其他模块同步

注 1: 这些输入仅用作触发源, 不能用作同步源。

15.0 带专用定时器的输出比较

注： 本数据手册总结了本系列 PIC24F 器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解更多信息，请参见《PIC24F 系列参考手册》中的第 35 章“带专用定时器的输出比较”（DS39723A_CN）。

PIC24FV32KA304 系列器件都具有 3 个独立的输出比较模块。每个模块都为发生器件内部事件时产生连续脉冲信号提供了各种配置和工作选项。另外，这些模块还可产生脉宽调制（PWM）波形以驱动功率应用。

输出比较模块的主要特性包括：

- 通过级联两个相邻的模块，可采用硬件方式配置为所有的 32 位工作模式
- 输出比较操作有同步和触发两种模式，可使用最多 21 个用户可选的同步 / 触发源
- 两个单独的周期寄存器（一个主寄存器 OCxR 和一个辅助寄存器 OCxRS），可更加灵活地产生不同宽度的脉冲
- 可配置为在发生输出事件时产生单脉冲信号或连续脉冲信号，或产生连续的 PWM 波形
- 每个模块最多可提供 6 种时钟源来驱动独立的内部 16 位计数器

15.1 一般工作模式

15.1.1 同步和触发模式

默认情况下，输出比较模块在自由运行模式下工作。内部 16 位计数器 OCxTMR 可连续向上计数，并在每次溢出时从 FFFF 计满返回 0000，其周期与所选的外部时钟源同步。当内部计数器与其中一个周期寄存器发生匹配时，产生一个比较或 PWM 事件。

在同步模式下，一旦使能了所选的时钟源，模块就开始执行比较或 PWM 操作。只要所选的同步源发生事件，模块的内部计数器都会复位。在触发模式下，模块等待来自其他内部模块的同步事件发生后，才允许该计数器运行。

默认情况下或 SYNCSELx 位（OCxCON2<4:0>）设置为 00000 时，选择自由运行模式。当 SYNCSELx 位设置为 00000 之外的其他值时，将选择同步或触发模式。OCTRIG 位（OCxCON2<7>）用于选择同步或触发模式。置 1 该位将选择触发工作模式。在这两种模式下，SYNCSELx 位决定同步 / 触发源。

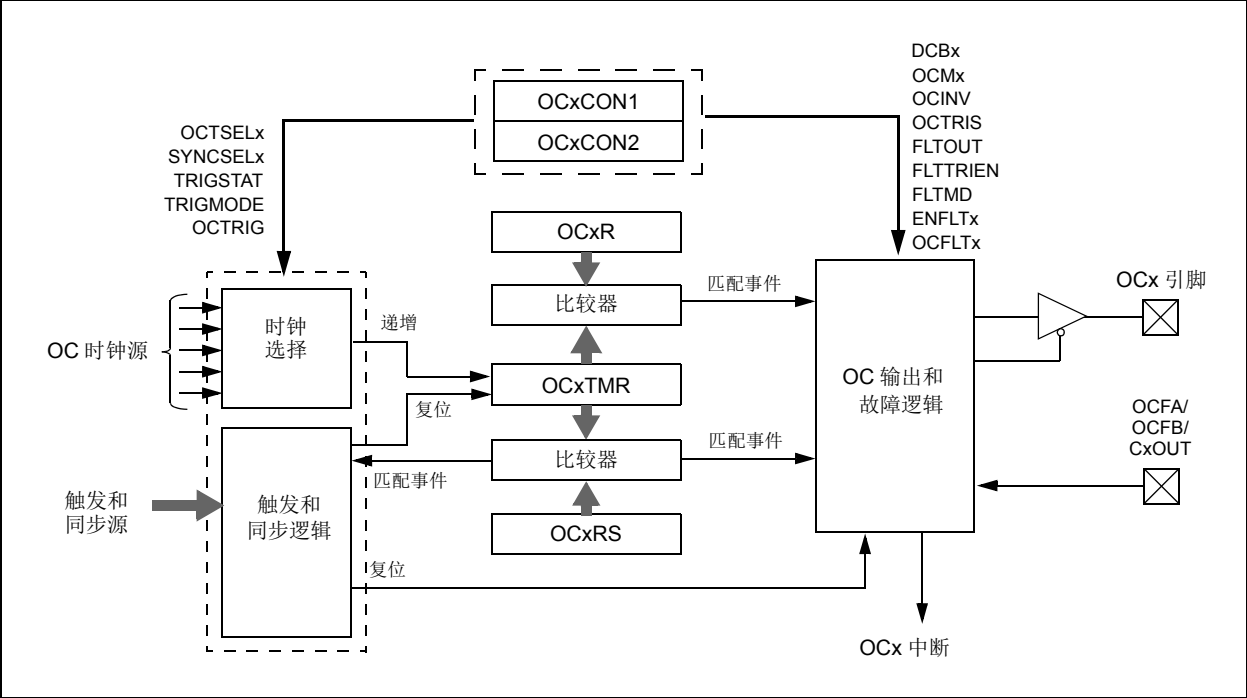
15.1.2 级联（32 位）模式

默认情况下，每个模块都可以使用自己的 16 位定时器和占空比寄存器独立运行。要提高分辨率，可将相邻的偶数编号的模块和奇数编号的模块配置为一个 32 位模块。（例如，模块 1 和 2 配对，模块 3 和 4 配对等）。奇数编号的模块（OCx）提供 32 位寄存器对中的低 16 位，而偶数编号的模块（OCy）提供高 16 位。OCx 寄存器的进位操作会导致其相应的 OCy 寄存器递增。

通过将两个模块的 OC32 位（OCxCON2<8>）置 1 使用硬件来配置级联操作。

PIC24FV32KA304 系列

图 15-1: 输出比较 x 框图（16 位模式）



15.2 比较操作

在比较模式下（图 15-1），输出比较模块可配置为产生单脉冲信号或产生连续脉冲信号。它还可在发生每个定时器事件时重复翻转输出引脚。

要将模块设置为比较操作：

1. 计算 OCxR 和（用于双比较模式）OCxRS 占空比寄存器所需的值：
 - a) 确定指令时钟周期。考虑定时器源的外部时钟频率（如果使用）和定时器预分频比设置。
 - b) 计算从定时器起始值（0000h）到输出脉冲的上升沿所需的时间。
 - c) 根据所需的脉冲宽度，计算出现脉冲下降沿的时间和出现脉冲上升沿的时间。
2. 将上升沿的值写入 OCxR，下降沿的值写入 OCxRS。
3. 如需触发工作模式，置 1 OCTRIG 以启动触发模式。置 1 或清零 TRIGMODE 可配置触发工作模式，置 1 或清零 TRIGSTAT 可选择硬件触发或软件触发。如需同步模式，清零 OCTRIG。
4. 设置 SYNCSEL<4:0> 位可配置触发或同步源。如需定时器在自由运行模式下工作，请将 SYNCSELx 位设置为 00000（无同步/触发源）。
5. 使用 OCTSEL<2:0> 位选择时基源。当运行所需的时钟源时，在使能输出比较模块之前，设置 OCTSEL<2:0> 位以与该时钟源同步。如有必要，将使能比较时基进行计数的选定定时器的 TON 位置 1。使能同步源后，将立即启动同步工作模式；发生触发源事件后，启动触发工作模式。
6. 设置 OCM<2:0> 位（0xx）以选择合适的比较操作。

对于 32 位级联操作，还包含以下必需步骤：

1. 将两个寄存器的 OC32 位（即 OCyCON2<8> 和 OCxCON2<8>）置 1。首先使能偶数编号的模块，以确保两个模块一起启动。
2. 清零偶数编号的模块的 OCTRIG 位（OCyCON2），使其在同步模式下运行。
3. 为 OCy 配置所需的输出和故障设置。
4. 通过清零 OCTRIS 位强制 OCx 的输出引脚为输出状态。
5. 如需触发工作模式，请使用 OCTRIG（OCxCON2<7>）、TRIGSTAT（OCxCON2<6>）和 SYNCSELx（OCxCON1<4:0>）位配置触发 OCx 的选项。
6. 首先为 OCy 配置所需的比较或 PWM 工作模式（OCM<2:0>），然后再配置 OCx。

根据选择的输出模式，模块保持 OCx 引脚处于默认状态并在 OCxR 中的值与定时器中的值匹配时强制翻转该引脚的状态。在双比较模式下，当定时器中的值与 OCxRS 中的值匹配时，OCx 被强制返回默认状态。在单比较模式下定时器与 OCxR 发生匹配、以及在双比较模式下每次定时器与 OCxRS 发生匹配后，OCxIF 中断标志都会置 1。

单脉冲事件仅发生一次，但是可通过简单地重写 OCxCON1 寄存器的值使其重复发生。连续脉冲事件在终止之前可无限次地发生。

PIC24FV32KA304 系列

15.3 脉宽调制 (PWM) 模式

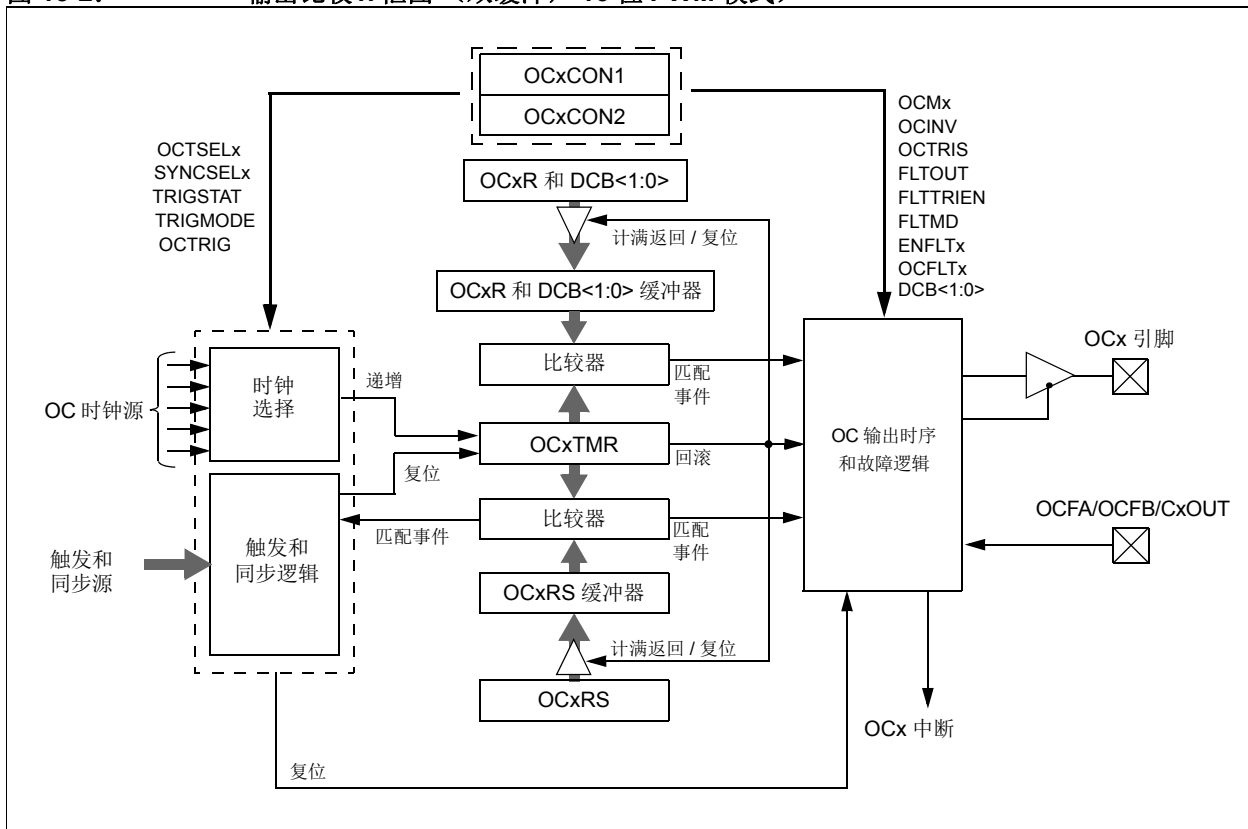
在 PWM 模式下，输出比较模块可配置为产生边沿对齐或中心对齐的脉冲波形。所有 PWM 操作都是双缓冲的（缓冲寄存器是模块内部的寄存器，没有映射到 SFR 空间）。

将输出比较模块配置为边沿对齐的 PWM 工作模式：

1. 计算所需的占空比并将计算结果装入 OCxR 寄存器。
2. 计算所需的周期并将计算结果装入 OCxRS 寄存器。
3. 选择当前的 OCx 作为同步源，方法是把 0x1F 写入 SYNCSEL<4:0> 位 (OCxCON2<4:0>)，把 0 写入 OCTRIG 位 (OCxCON2<7>)。

4. 通过写 OCTSEL2<2:0> 位 (OCxCON<12:10>) 选择时钟源。
5. 如果需要的话，允许定时器和输出比较模块中断。使用 PWM 故障引脚时需要输出比较中断。
6. 通过 OCM<2:0> (OCxCON1<2:0>) 位选择所需的 PWM 模式。
7. 如果定时器被选为时钟源，请设置 TMRy 预分频值，并通过将 TON (TxCON<15>) 位置 1 使能时基。

图 15-2: 输出比较 x 框图 (双缓冲, 16 位 PWM 模式)



15.3.1 PWM 周期

在边沿对齐的 PWM 模式下，通过 OCxRS 寄存器的值指定周期。在中心对齐的 PWM 模式下，同步源的周期（如定时器 PRy）指定周期。这两种模式下的周期可使用公式 15-1 计算。

公式 15-1: 计算 PWM 周期⁽¹⁾

$$\text{PWM 周期} = [\text{Value} + 1] \times \text{Tcy} \times (\text{预分频值})$$

其中:

Value = 边沿对齐的 PWM 模式下的 OCxRS，也可以是中心对齐的 PWM 模式下的 PRy（如果 TMRy 为同步源）。

注 1: 基于 $\text{Tcy} = \text{Tosc} \times 2$ ；禁止打盹模式和 PLL。

15.3.2 PWM 占空比

PWM 占空比是通过写 OCxRS 和 OCxR 寄存器指定的。在任何时间都可以写入 OCxRS 和 OCxR 寄存器，但是在周期完成前不会锁存占空比值。这可以为 PWM 占空比提供双重缓冲，对于 PWM 的无毛刺操作是极其重要的。

以下是 PWM 占空比的部分重要边界参数：

- 边沿对齐的 PWM：
 - 如果 OCxR 和 OCxRS 装入 0000h，那么 OCx 引脚将保持低电平（占空比为 0%）。
 - 如果 OCxRS 大于 OCxR，则引脚将保持高电平（占空比为 100%）。
- 中心对齐 PWM（当 TMRy 作为同步源）：
 - 如果 OCxR、OCxRS 和 PRy 全部装入 0000h，那么 OCx 引脚将保持低电平（占空比为 0%）。
 - 如果 OCxRS 大于 PRy，则引脚将进入高电平（占空比为 100%）。

请参见例 15-3 了解 PWM 模式时序的详细信息。表 15-1 和表 15-2 给出了当器件分别工作在 4 MIPS 和 10 MIPS 时，所对应的 PWM 频率和分辨率的示例。

公式 15-2: 计算最大 PWM 分辨率⁽¹⁾

$$\text{最大 PWM 分辨率 (位)} = \frac{\log_{10}\left(\frac{\text{Fcy}}{\text{FPWM} \cdot (\text{预分频值})}\right)}{\log_{10}(2)} \text{ 位}$$

注 1: 基于 $\text{Fcy} = \text{Fosc}/2$ ；禁止打盹模式和 PLL。

公式 15-3: 计算 PWM 周期和占空比⁽¹⁾

1. 在所需的 PWM 频率为 52.08 kHz、Fosc = 8 MHz 并带 PLL（32 MHz 器件时钟速率）且使用边沿对齐 PWM 模式下时预分频比设为 1:1 的条件下计算 OCxRS 寄存器的值。

$$\text{Tcy} = 2 \cdot \text{Tosc} = 62.5 \text{ ns}$$

$$\text{PWM 周期} = 1/\text{PWM 频率} = 1/52.08 \text{ kHz} = 19.2 \mu\text{s}$$

$$\text{PWM 周期} = (\text{OCxRS} + 1) \cdot \text{Tcy} \cdot (\text{OCx 预分频值})$$

$$19.2 \mu\text{s} = (\text{OCxRS} + 1) \cdot 62.5 \text{ ns} \cdot 1$$

$$\text{OCxRS} = 306$$

2. 在 PWM 频率为 52.08 kHz 且器件的时钟速率为 32 MHz 时，计算占空比的最大分辨率：

$$\text{PWM 分辨率} = \log_{10}(\text{Fcy}/\text{FPWM})/\log_{10}(2) \text{ 位}$$

$$= (\log_{10}(16 \text{ MHz}/52.08 \text{ kHz})/\log_{10}(2)) \text{ 位}$$

$$= 8.3 \text{ 位}$$

注 1: 基于 $\text{Tcy} = 2 \cdot \text{Tosc}$ ；禁止打盹模式和 PLL。

PIC24FV32KA304 系列

15.4 小于一个指令周期的分辨率

DCBx 位（OCxCON2<10:9>）提供比一个指令周期更好的分辨率。当使用 DCB 位时，将导致因匹配事件生成的下降沿延时小于一个指令周期。

例如，设置 DCB<1:0> = 10 可能导致在匹配事件发生的指令周期内中途产生下降沿，而不是在开始时。当 OCM<2:0> = 001 时，不能使用这些位。当在 PWM 模式（OCM<2:0> = 110 或 111）下运行该模块时，DCBx 位是双缓冲位。

DCBx 位用于与系统时钟相同的时钟源。当使用带有使能的预分频器的 OCx 模块时，由 DCBx 位引起的下降沿延时将以系统时钟周期而不是 OCx 模块的周期作为参考。

表 15-1: 器件工作在 4 MIPS（Fcy = 4 MHz）时的 PWM 频率和分辨率示例⁽¹⁾

PWM 频率	7.6 Hz	61 Hz	122 Hz	977 Hz	3.9 kHz	31.3 kHz	125 kHz
预分频比	8	1	1	1	1	1	1
周期值	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
分辨率（位）	16	16	15	12	10	7	5

注 1: 基于 Fcy = Fosc/2；禁止打盹模式和 PLL。

表 15-2: 器件工作在 16 MIPS（Fcy = 16 MHz）时的 PWM 频率和分辨率示例⁽¹⁾

PWM 频率	30.5 Hz	244 Hz	488 Hz	3.9 kHz	15.6 kHz	125 kHz	500 kHz
预分频比	8	1	1	1	1	1	1
周期值	FFFFh	FFFFh	7FFFh	0FFFh	03FFh	007Fh	001Fh
分辨率（位）	16	16	15	12	10	7	5

注 1: 基于 Fcy = Fosc/2；禁止打盹模式和 PLL。

PIC24FV32KA304 系列

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	R/W-0	R/W-0
—	—	OCSIDL	OCTSEL2	OCTSEL1	OCTSEL0	ENFLT2	ENFLT1
bit 15						bit 8	

R/W-0	R/W-0, HSC	R/W-0, HSC	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R/W-0
ENFLT0	OCFLT2	OCFLT1	OCFLT0	TRIGMODE	OCM2 ⁽¹⁾	OCM1 ⁽¹⁾	OCM0 ⁽¹⁾
bit 7						bit 0	

图注:	HSC = 可由硬件置 1/ 清零的位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14 **未实现:** 读为 0

bit 13 **OCSIDL:** 输出比较 x 在空闲模式下停止控制位
 1 = 输出比较 x 在 CPU 空闲模式下停止
 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 定时器选择位
 111 = 系统时钟
 110 = 保留
 101 = 保留
 100 = Timer1
 011 = Timer5
 010 = Timer4
 001 = Timer3
 000 = Timer2

bit 9 **ENFLT2:** 比较器故障输入使能位
 1 = 使能比较器故障输入
 0 = 禁止比较器故障输入

bit 8 **ENFLT1:** OCFB 故障输入使能位
 1 = 使能 OCFB 故障输入
 0 = 禁止 OCFB 故障输入

bit 7 **ENFLT0:** OCFA 故障输入使能位
 1 = 使能 OCFA 故障输入
 0 = 禁止 OCFA 故障输入

bit 6 **OCFLT2:** PWM 比较器故障条件状态位
 1 = PWM 比较器故障条件已发生 (只能由硬件清零)
 0 = PWM 比较器故障条件未发生 (仅当 OCM<2:0> = 111 时该位可用)

bit 5 **OCFLT1:** PWM OCFB 故障输入使能位
 1 = PWM OCFB 故障条件已发生 (只能由硬件清零)
 0 = PWM OCFB 故障条件未发生 (仅当 OCM<2:0> = 111 时该位可用)

bit 4 **OCFLT0:** PWM OCFA 故障条件状态位
 1 = PWM OCFA 故障条件已发生 (只能由硬件清零)
 0 = PWM OCFA 故障条件未发生 (仅当 OCM<2:0> = 111 时该位可用)

bit 3 **TRIGMODE:** 触发状态模式选择位
 1 = 当 OCxRS = OCxTMR 时清零 TRIGSTAT (OCxCON2<6>), 或由软件清零
 0 = TRIGSTAT 只能由软件清零

注 1: 用于故障输入的比较器模块由 OCx 模块决定。OC1 和 OC2 使用比较器 1；OC3 和 OC4 使用比较器 2；OC5 使用比较器 3。

PIC24FV32KA304 系列

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 ⁽¹⁾

111 = OCx 处于中心对齐 PWM 模式

110 = OCx 处于边沿对齐 PWM 模式

101 = 双比较连续脉冲模式: 初始化 OCx 引脚为低电平, 与 OCxR 和 OCxRS 交替匹配时连续翻转 OCx 的状态

100 = 双比较单脉冲模式: 初始化 OCx 引脚为低电平, 在一个周期内 OCxR 与 OCxRS 匹配时翻转 OCx 的状态

011 = 单比较连续脉冲模式: 比较事件连续翻转 OCx 引脚

010 = 单比较单脉冲模式: 初始化 OCx 引脚为高电平, 比较事件强制 OCx 引脚为低电平

001 = 单比较单脉冲模式: 初始化 OCx 引脚为低电平, 比较事件强制 OCx 引脚为高电平

000 = 禁止输出比较通道

注 1: 用于故障输入的比较器模块由 OCx 模块决定。OC1 和 OC2 使用比较器 1; OC3 和 OC4 使用比较器 2; OC5 使用比较器 3。

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	R/W-0	R/W-0	R/W-0
FLTMD	FLTOUT	FLTTRIEN	OCINV	—	DCB1 ⁽³⁾	DCB0 ⁽³⁾	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 = 可由硬件置 1 的位						
R = 可读位	W = 可写位			U = 未实现位, 读为 0			
-n = 上电复位时的值	1 = 置 1			0 = 清零		x = 未知	

- bit 15 **FLTMD:** 故障模式选择位
 1 = 故障模式保持, 直到消除故障源并且相应的 OCFLT_x 位由软件清零
 0 = 故障模式保持, 直到消除故障源并且启动了新的 PWM 周期
- bit 14 **FLTOUT:** 故障输出位
 1 = 发生故障时 PWM 输出被驱动为高电平
 0 = 发生故障时 PWM 输出被驱动为低电平
- bit 13 **FLTTRIEN:** 故障输出状态选择位
 1 = 发生故障条件时引脚被强制为输出状态
 0 = 引脚 I/O 状态不受故障影响
- bit 12 **OCINV:** 输出比较 x 反相位
 1 = OC_x 输出反相
 0 = OC_x 输出不反相
- bit 11 **未实现:** 读为 0
- bit 10-9 **DCB<1:0>:** 输出比较 x 脉宽最低有效位 ⁽³⁾
 11 = OC_x 下降沿延迟 3/4 指令周期
 10 = OC_x 下降沿延迟 1/2 指令周期
 01 = OC_x 下降沿延迟 1/4 指令周期
 00 = OC_x 下降沿在指令周期一开始就出现
- bit 8 **OC32:** 级联两个输出比较模块使能位 (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 同步 / 触发选择位
 1 = 用 SYNCSEL_x 位指定的源触发 OC_x
 0 = OC_x 与 SYNCSEL_x 位指定的源同步
- bit 6 **TRIGSTAT:** 定时器触发状态位
 1 = 定时器源已被触发并运行
 0 = 定时器源未被触发, 保持清零状态
- bit 5 **OCTRIS:** 输出比较 x 输出引脚方向选择位
 1 = OC_x 引脚是三态引脚
 0 = OC_x 引脚上连接了输出比较 x 外设

- 注** 1: 不要通过以下方式将输出比较模块作为自己的触发源: 选择这种模式或其他同等 SYNCSEL_x 设置。
 2: 这些输入仅用作触发源, 不能用作同步源。
 3: 当 OCMx = 1 时, 这些位影响上升沿。当 OCMx 位 (OCxCON1<2:0>) = 001 时, 该位不起作用。

PIC24FV32KA304 系列

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 = 此输出比较模块 ⁽¹⁾

11110 = 保留

11101 = 保留

11100 = CTMU ⁽²⁾

11011 = A/D ⁽²⁾

11010 = 比较器 3 ⁽²⁾

11001 = 比较器 2 ⁽²⁾

11000 = 比较器 1 ⁽²⁾

10111 = 输入捕捉 4 ⁽²⁾

10110 = 输入捕捉 3 ⁽²⁾

10101 = 输入捕捉 2 ⁽²⁾

10100 = 输入捕捉 1 ⁽²⁾

100xx = 保留

01111 = Timer5

01110 = Timer4

01101 = Timer3

01100 = Timer2

01011 = Timer1

01010 = 输入捕捉 5 ⁽²⁾

01001 = 保留

01000 = 保留

00111 = 保留

00110 = 保留

00101 = 输出比较 5 ⁽¹⁾

00100 = 输出比较 4 ⁽¹⁾

00011 = 输出比较 3 ⁽¹⁾

00010 = 输出比较 2 ⁽¹⁾

00001 = 输出比较 1 ⁽¹⁾

00000 = 不与任何其他模块同步

- 注
- 1: 不要通过以下方式将输出比较模块作为自己的触发源: 选择这种模式或其他同等 SYNCSELx 设置。
 - 2: 这些输入仅用作触发源, 不能用作同步源。
 - 3: 当 OCINV = 1 时, 这些位影响上升沿。当 OCMx 位 (OCxCON1<2:0>) = 001 时, 该位不起作用。

16.0 串行外设接口 (SPI)

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解串行外设接口的更多信息，请参见《PIC24F 系列参考手册》中的第 23 章“串行外设接口 (SPI)” (DS39699A_CN)。

串行外设接口 (Serial Peripheral Interface, SPI) 模块是一个同步串行接口，可用于与其他外设或者单片机进行通信。这些外设器件可以是串行数据 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。

该模块可在两种缓冲模式下工作。在标准模式下，通过单个串行缓冲器移动数据。在增强型缓冲模式下，通过 8 级深 FIFO 缓冲器移动数据。

注： 无论是在标准还是增强型缓冲模式下，都不要对 SPI1BUF 寄存器执行读 - 修改 - 写操作 (如位操作指令)。

工作在主模式或从模式下时，该模块还支持基本的帧 SPI 协议。共支持四种帧 SPI 配置。

SPI 串行接口由以下四个引脚组成：

- SDI1: 串行数据输入
- SDO1: 串行数据输出
- SCK1: 移位时钟输入或输出
- SS1: 低电平有效从选择或帧同步 I/O 脉冲

SPI 模块可以被配置为使用 2 个、3 个或 4 个引脚工作。在 3 引脚模式下，不使用 SS1。在 2 引脚模式下，不使用 SDO1 和 SS1。

图 16-1 和图 16-2 所示为标准型和增强型缓冲模式下该模块的框图。

PIC24FV32KA304 系列器件上提供两个 SPI 模块：

注： 在本章中，SPI 模块统称为 SPIx。特殊功能寄存器 (SFR) 也使用类似的符号表示。例如，SPI1CON1 或 SPI1CON2 表示 SPI1 模块的控制寄存器。

要将 SPI1 模块设置为工作在标准主模式下，请遵循以下步骤：

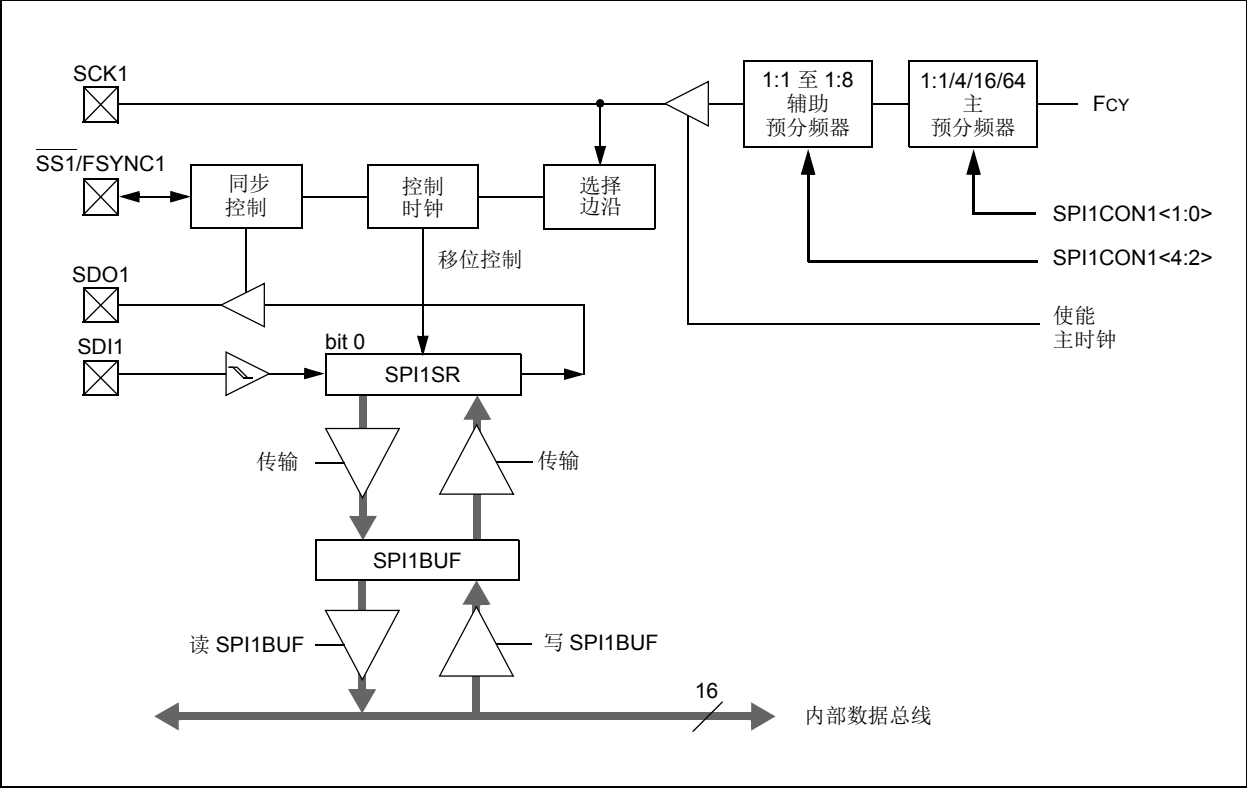
1. 如果使用中断：
 - a) 将 IFS0 寄存器中的 SPI1IF 位清零。
 - b) 将 IEC0 寄存器中的 SPI1IE 位置 1。
 - c) 通过写 IPC2 寄存器中的相应 SPI1IPx 位来设置中断优先级。
2. 将所需设置写入 SPI1CON1 和 SPI1CON2 寄存器，同时 MSTEN (SPI1CON1<5>) = 1。
3. 将 SPIROV 位 (SPI1STAT<6>) 清零。
4. 通过将 SPIEN 位 (SPI1STAT<15>) 置 1 使能 SPI 操作。
5. 将待发送数据写入 SPI1BUF 寄存器。数据一写入 SPI1BUF 寄存器发送 (和接收) 就会立即开始。

要将 SPI1 模块设置为工作在标准从模式下，请遵循以下步骤：

1. 将 SPI1BUF 寄存器清零。
2. 如果使用中断：
 - a) 将 IFS0 寄存器中的 SPI1IF 位清零。
 - b) 将 IEC0 寄存器中的 SPI1IE 位置 1。
 - c) 通过写 IPC2 寄存器中的 SPI1IPx 位来设置中断优先级。
3. 将所需设置写入 SPI1CON1 和 SPI1CON2 寄存器，同时 MSTEN (SPI1CON1<5>) = 0。
4. 将 SMP 位清零。
5. 如果将 CKE 位置 1，则 SSEN 位 (SPI1CON1<7>) 也必须置 1 以使能 SS1 引脚。
6. 将 SPIROV 位 (SPI1STAT<6>) 清零。
7. 通过将 SPIEN 位 (SPI1STAT<15>) 置 1 使能 SPI 操作。

PIC24FV32KA304 系列

图 16-1: SPI1 模块框图 (标准缓冲模式)



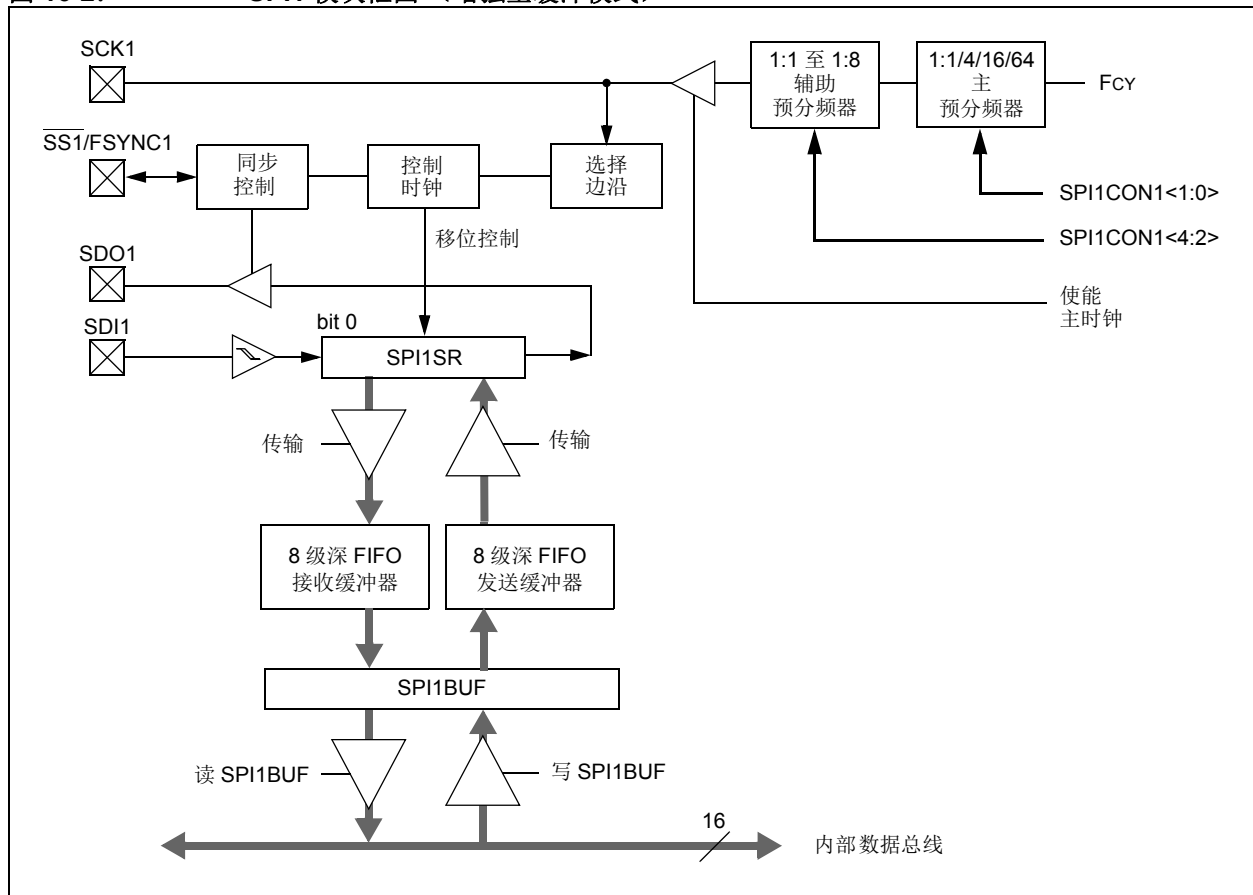
要将 SPI1 模块设置为工作在增强型缓冲主模式 (Enhanced Buffer Master, EBM) 下, 请遵循以下步骤:

1. 如果使用中断:
 - a) 将 IFS0 寄存器中的 SPI1IF 位清零。
 - b) 将 IEC0 寄存器中的 SPI1IE 位置 1。
 - c) 写 IPC2 寄存器中的相应 SPI1IPx 位。
2. 将所需设置写入 SPI1CON1 和 SPI1CON2 寄存器, 同时 MSTEN (SPI1CON1<5>) = 1。
3. 将 SPIROV 位 (SPI1STAT<6>) 清零。
4. 通过将 SPIBEN 位 (SPI1CON2<0>) 置 1 选择增强型缓冲模式。
5. 通过将 SPIEN 位 (SPI1STAT<15>) 置 1 使能 SPI 操作。
6. 将待发送数据写入 SPI1BUF 寄存器。数据一写入 SPI1BUF 寄存器发送 (和接收) 就会立即开始。

要将 SPI1 模块设置为工作在增强型缓冲从模式下, 请遵循以下步骤:

1. 将 SPI1BUF 寄存器清零。
2. 如果使用中断:
 - a) 将 IFS0 寄存器中的 SPI1IF 位清零。
 - b) 将 IEC0 寄存器中的 SPI1IE 位置 1。
 - c) 通过写 IPC2 寄存器中的相应 SPI1IPx 位来设置中断优先级。
3. 将所需设置写入 SPI1CON1 和 SPI1CON2 寄存器, 同时 MSTEN (SPI1CON1<5>) = 0。
4. 将 SMP 位清零。
5. 如果 CKE 位置 1, 则 SS1EN 位也必须置 1, 从而使能 SS1 引脚。
6. 将 SPIROV 位 (SPI1STAT<6>) 清零。
7. 通过将 SPIBEN 位 (SPI1CON2<0>) 置 1 选择增强型缓冲模式。
8. 通过将 SPIEN 位 (SPI1STAT<15>) 置 1 使能 SPI 操作。

图 16-2: SPI1 模块框图 (增强型缓冲模式)



PIC24FV32KA304 系列

寄存器 16-1: SPIxSTAT: SPIx 状态和控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
SPIEN	—	SPISIDL	—	—	SPIBEC2	SPIBEC1	SPIBEC0
bit 15					bit 8		

R-0, HSC	R/C-0, HS	R/W-0, HSC	R/W-0	R/W-0	R/W-0	R-0, HSC	R-0, HSC
SRMPT	SPIROV	SRXMPT	SISEL2	SISEL1	SISEL0	SPITBF	SPIRBF
bit 7							bit 0

图注:	C = 可清零位	HS = 可由硬件置 1 的位	HSC = 可由硬件置 1/ 清零的位
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15 **SPIEN:** SPIx 使能位
1 = 使能模块并将 SCKx、SDOx、SDIx 和 $\overline{SSx}$ 配置为串行端口引脚
0 = 禁止模块
- bit 14 未实现: 读为 0
- bit 13 **SPISIDL:** SPIx 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 = 模块在空闲模式下继续工作
- bit 12-11 未实现: 读为 0
- bit 10-8 **SPIBEC<2:0>:** SPIx 缓冲器元素计数位 (在增强型缓冲模式下有效)
主模式:
等待中的 SPI 传输数。
从模式:
未读取的 SPI 传输数。
- bit 7 **SRMPT:** SPIx 移位寄存器 (SPIxSR) 空位 (在增强型缓冲模式下有效)
1 = SPIx 移位寄存器为空并已做好发送或接收准备
0 = SPIx 移位寄存器不为空
- bit 6 **SPIROV:** SPIx 接收溢出标志位
1 = 一个新字节 / 字已被完全接收并丢弃。(用户软件尚未读取 SPIxBUF 寄存器中先前接收到的数据。)
0 = 未发生溢出
- bit 5 **SRXMPT:** SPIx 接收 FIFO 空位 (在增强型缓冲模式下有效)
1 = 接收 FIFO 为空
0 = 接收 FIFO 不为空
- bit 4-2 **SISEL<2:0>:** SPIx 缓冲器中断模式位 (在增强型缓冲模式下有效)
111 = 当 SPIx 发送缓冲器满时产生中断 (SPITBF 位置 1)
110 = 最后一位移入 SPIxSR 导致发送 FIFO 为空时产生中断
101 = 最后一位被移出 SPIxSR 时产生中断, 此时发送完成
100 = 有一个数据字节移入 SPIxSR 导致发送 FIFO 有待发送数据时产生中断
011 = SPIx 接收缓冲器满时产生中断 (SPIRBF 位置 1)
010 = SPIx 接收缓冲器被填满 3/4 或更多时产生中断
001 = 接收缓冲器中有数据时产生中断 (SRMPT 位置 1)
000 = 读取接收缓冲器中的最后一个数据, 导致缓冲器为空时产生中断 (SRXMPT 位置 1)

寄存器 16-1: SPIxSTAT: SPIx 状态和控制寄存器 (续)

bit 1 **SPITBF:** SPIx 发送缓冲器满状态位

1 = 未开始发送, SPIxTXB 已满

0 = 已开始发送, SPIxTXB 为空

在标准缓冲模式下:

当 CPU 通过写 SPIxBUF 地址单元装载 SPIxTXB 时, 该位由硬件自动置 1。

当 SPIx 模块将数据从 SPIxTXB 传输到 SPIxSR 时, 该位由硬件自动清零。

在增强型缓冲模式下:

当 CPU 通过写 SPIxBUF 地址单元装载最后一个可用缓冲单元时, 该位由硬件自动置 1。

当有空的缓冲单元可由 CPU 写入时, 该位由硬件自动清零。

bit 0 **SPIRBF:** SPIx 接收缓冲器满状态位

1 = 接收完成, SPIxRXB 为满

0 = 接收未完成, SPIxRXB 为空

在标准缓冲模式下:

当 SPIx 将数据从 SPIxSR 传输到 SPIxRXB 时, 该位由硬件自动置 1。

当内核通过读 SPIxBUF 地址单元读 SPIxRXB 时, 该位由硬件自动清零。

在增强型缓冲模式下:

当 SPIx 将数据从 SPIxSR 传输到缓冲器填充了最后一个未读的缓冲单元时, 该位由硬件自动置 1。

当有空的缓冲单元可接收来自 SPIxSR 的传输数据时, 该位由硬件自动清零。

PIC24FV32KA304 系列

寄存器 16-2: SPIxCON1: SPIx 控制寄存器 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE ⁽¹⁾
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	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0
bit 7							
							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-13 未实现: 读为 0

bit 12 **DISSCK**: 禁止 SCKx 引脚位 (仅限 SPIx 主模式)

1 = 禁止内部 SPIx 时钟, 引脚用作 I/O

0 = 使能内部 SPIx 时钟

bit 11 **DISSDO**: 禁止 SDOx 引脚位

1 = 模块不使用 SDOx 引脚, 引脚用作 I/O

0 = SDOx 引脚受模块控制

bit 10 **MODE16**: 字 / 字节通信选择位

1 = 通信为字宽 (16 位)

0 = 通信为字节宽 (8 位)

bit 9 **SMP**: SPIx 数据输入采样阶段位

主模式:

1 = 在数据输出时间的结尾采样输入数据

0 = 在数据输出时间的中间采样输入数据

从模式:

当在从模式下使用 SPIx 时, 必须将 SMP 清零。

bit 8 **CKE**: 时钟边沿选择位 ⁽¹⁾

1 = 串行输出数据在时钟由有效状态变为空闲状态 (见 bit 6) 时改变

0 = 串行输出数据在时钟由空闲状态变为有效状态 (见 bit 6) 时改变

bit 7 **SSEN**: 从选择使能位 (从模式)

1 = SSx 引脚用于从模式

0 = 模块不使用 SSx 引脚; 引脚由端口功能控制

bit 6 **CKP**: 时钟极性选择位

1 = 时钟信号空闲状态为高电平; 有效状态为低电平

0 = 时钟信号空闲状态为低电平; 有效状态为高电平

bit 5 **MSTEN**: 主模式使能位

1 = 主模式

0 = 从模式

bit 4-2 **SPRE<2:0>**: 辅助预分频比位 (主模式)

111 = 辅助预分频比为 1:1

110 = 辅助预分频比为 2:1

.

.

.

000 = 辅助预分频比为 8:1

注 1: 在帧 SPI 模式下不使用 CKE 位。在帧 SPI 模式 (FRMEN = 1) 下, 用户应将该位编程为 0。

寄存器 16-2: SPIxCON1: SPIx 控制寄存器 1 (续)

bit 1-0 **PPRE<1:0>**: 主预分频比位 (主模式)
 11 = 主预分频比为 1:1
 10 = 主预分频比为 4:1
 01 = 主预分频比为 16:1
 00 = 主预分频比为 64:1

注 1: 在帧 SPI 模式下不使用 CKE 位。在帧 SPI 模式 (FRMEN = 1) 下, 用户应将该位编程为 0。

寄存器 16-3: SPIxCON2: SPIx 控制寄存器 2

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	SPIFPOL	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SPIFE	SPIBEN
bit 7							bit 0

图注:

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

bit 15 **FRMEN**: 帧 SPIx 支持位
 1 = 使能帧 SPIx 支持
 0 = 禁止帧 SPIx 支持

bit 14 **SPIFSD**: SPIx $\overline{SSx}$ 引脚上的帧同步脉冲方向控制位
 1 = 帧同步脉冲输入 (从模式)
 0 = 帧同步脉冲输出 (主模式)

bit 13 **SPIFPOL**: SPIx 帧同步脉冲极性位 (仅用于帧模式)
 1 = 帧同步脉冲高电平有效
 0 = 帧同步脉冲低电平有效

bit 12-2 **未实现**: 读为 0

bit 1 **SPIFE**: SPIx 帧同步脉冲边沿选择位
 1 = 帧同步脉冲与第一个位时钟同步
 0 = 帧同步脉冲比第一个位时钟超前

bit 0 **SPIBEN**: SPIx 增强型缓冲器使能位
 1 = 使能增强型缓冲器
 0 = 禁止增强型缓冲器 (传统模式)

PIC24FV32KA304 系列

公式 16-1: 器件速率和 SPIx 时钟速率之间的关系 (1)

$$F_{SCK} = \frac{F_{CY}}{\text{主预分频值} * \text{辅助预分频值}}$$

注 1: 基于 $F_{CY} = F_{OSC}/2$; 禁止打盹模式和 PLL。

表 16-1: 采样 SCKx 频率 (1,2)

Fcy = 16 MHz		辅助预分频比设置				
		1:1	2:1	4:1	6:1	8:1
主预分频比设置	1:1	无效	8000	4000	2667	2000
	4:1	4000	2000	1000	667	500
	16:1	1000	500	250	167	125
	64:1	250	125	63	42	31
Fcy = 5 MHz						
主预分频比设置	1:1	5000	2500	1250	833	625
	4:1	1250	625	313	208	156
	16:1	313	156	78	52	39
	64:1	78	39	20	13	10

注 1: 基于 $F_{CY} = F_{OSC}/2$; 禁止打盹模式和 PLL。
2: SCKx 频率的单位为 kHz。

17.0 I²C™

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解 I²C 的更多信息，请参见《PIC24F 系列参考手册》中的第 24 章 “I²C™” (DS39702A_CN)。

I²C™ 模块是用来与其他外设或单片机通信的串行接口。这些外设器件可以是串行数据 EEPROM、显示驱动器和 A/D 转换器等。

I²C 模块支持以下特性：

- 独立的主 / 从逻辑
- 7 位和 10 位器件地址
- I²C 协议中定义的广播呼叫地址
- 时钟延长，为处理器响应从器件数据请求提供延时
- 支持 100 kHz 和 400 kHz 两种总线规范
- 可配置的地址掩码
- 多主器件模式以防止仲裁时报文丢失
- 总线重发器模式，允许作为从器件接收来自所有地址的所有报文
- 自动 SCL

模块的框图如图 17-1 所示。

17.1 引脚重新映射选项

I²C 模块将连接到固定引脚。要灵活进行外设复用，可以在器件配置期间，将 28 引脚器件中的 I2C1 模块重新指定给备用引脚，这些备用引脚为 SCL1 和 SDA1。

引脚分配由 I2CxSEL 配置位控制。编程此位 (= 0) 会将该模块复用到 SCL1 和 SDA1 引脚。

注： 在本节中，若要引用与特定 I²C 模块相关的寄存器和位，通常在其名称中使用 x 来代替特定的模块编号。因此，“I2CxSTAT”可能指的是 I2C1 或 I2C2 的接收状态寄存器。

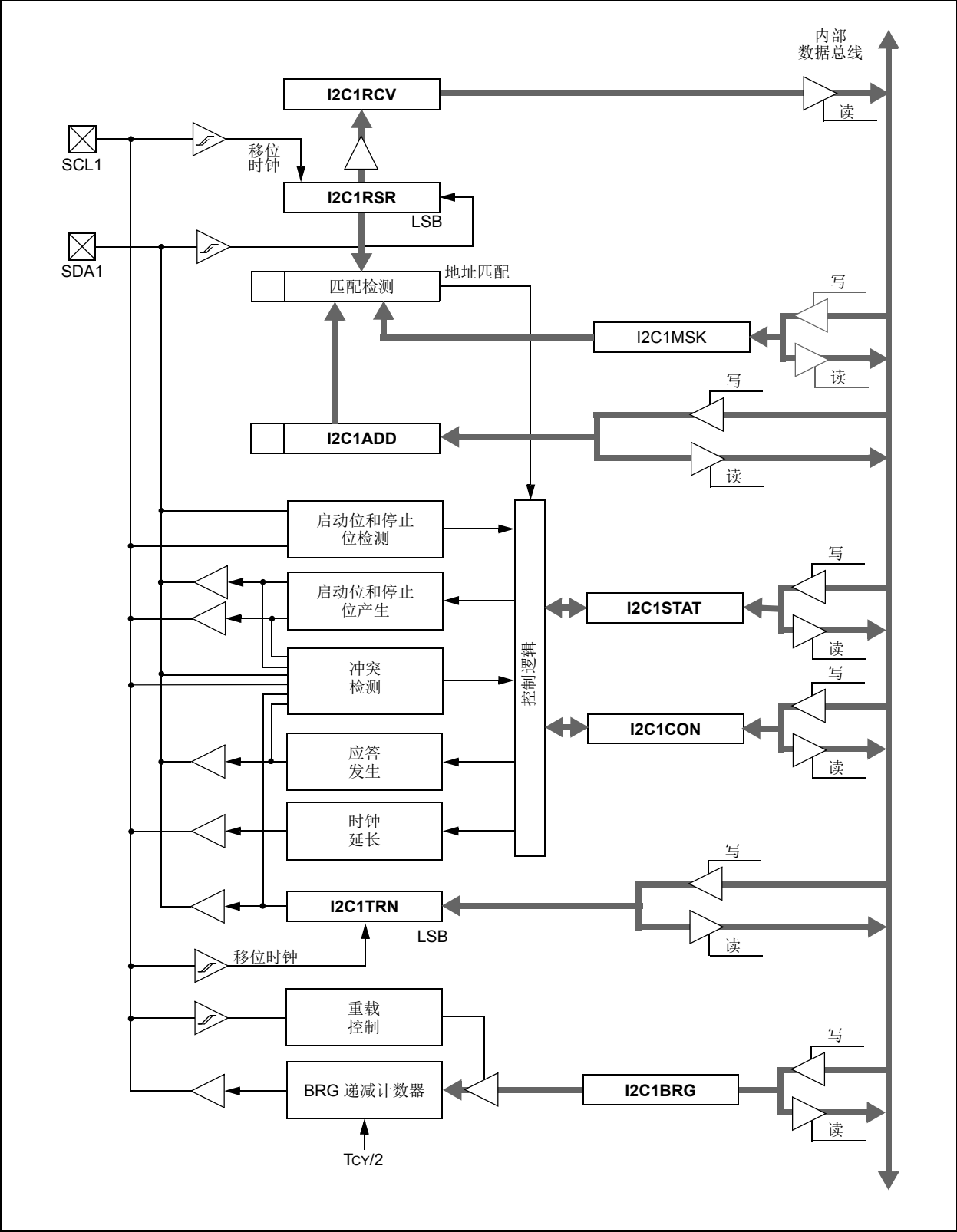
17.2 作为主器件在单主器件环境中通信

在主模式下发送报文的细节取决于用于与器件通信的通信协议。通常，事件序列如下：

1. 在 SDAx 和 SCLx 上发送一个启动条件。
2. 发送一个 I²C 器件地址字节到从器件，表明要执行写操作。
3. 等待并验证从器件的应答。
4. 发送第一个数据字节（有时是命令）到从器件。
5. 等待并验证从器件的应答。
6. 发送串行存储器地址低字节到从器件。
7. 重复步骤 4 和 5，直至发送完所有数据字节。
8. 在 SDAx 和 SCLx 上发送一个重复启动条件。
9. 发送一个器件地址字节到从器件，表明要执行读操作。
10. 等待并验证从器件的应答。
11. 使能主器件接收以接收串行存储器数据。
12. 在数据字节接收完毕时，产生 ACK 或 NACK 条件。
13. 在 SDAx 和 SCLx 上产生一个停止条件。

PIC24FV32KA304 系列

图 17-1: I²C™ 框图 (此处显示的是 I2C1 模块)



17.3 设置用作总线主器件时的波特率

使用公式 17-1 计算波特率发生器 (Baud Rate Generator, BRG) 的重载值。

公式 17-1: 计算波特率重载值⁽¹⁾

$$F_{SCL} = \frac{F_{CY}}{I2CxBRG + 1 + \frac{F_{CY}}{10,000,000}}$$

或

$$I2CxBRG = \left(\frac{F_{CY}}{F_{SCL}} - \frac{F_{CY}}{10,000,000} \right) - 1$$

注 1: 基于 $F_{CY} = F_{OSC}/2$; 禁止打盹模式和 PLL。

17.4 从地址掩码

I2CxMSK 寄存器 (见寄存器 17-3) 指定了 7 位和 10 位寻址模式下某些“无关”地址位的位置。将 I2CxMSK 寄存器中某个特定位置 1 (= 1)，不论相应的地址位值是 0 或 1，从模块都会做出响应。例如，当将 I2CxMSK 设置为 00100000 时，从模块将检测两个地址：00000000 和 00100000。

为了使能地址掩码，必须通过将 IPMIEN 位 (I2CxCON<11>) 清零来禁止智能外设管理接口 (Intelligent Peripheral Management Interface, IPMI)。

注: 由于修改了 I²C 协议，表 17-2 中的地址将被保留，并且在从模式下不会被应答。这同样适用于包含任意这些地址在内的所有地址掩码设置。

表 17-1: I²C™ 时钟速率⁽¹⁾

所需的系统 F _{SCL}	F _{CY}	I2CxBRG 值		实际 F _{SCL}
		(十进制)	(十六进制)	
100 kHz	16 MHz	157	9D	100 kHz
100 kHz	8 MHz	78	4E	100 kHz
100 kHz	4 MHz	39	27	99 kHz
400 kHz	16 MHz	37	25	404 kHz
400 kHz	8 MHz	18	12	404 kHz
400 kHz	4 MHz	9	9	385 kHz
400 kHz	2 MHz	4	4	385 kHz
1 MHz	16 MHz	13	D	1.026 MHz
1 MHz	8 MHz	6	6	1.026 MHz
1 MHz	4 MHz	3	3	0.909 MHz

注 1: 基于 $F_{CY} = F_{OSC}/2$; 禁止打盹模式和 PLL。

表 17-2: I²C™ 保留地址⁽¹⁾

从地址	R/W 位	说明
0000 000	0	广播呼叫地址 ⁽²⁾
0000 000	1	启动字节
0000 001	x	CBus 地址
0000 010	x	保留
0000 011	x	保留
0000 1xx	x	HS 模式主代码
1111 1xx	x	保留
1111 0xx	x	10 位从地址高字节 ⁽³⁾

注 1: 上述地址位永远不会导致地址匹配，与地址掩码的设置无关。

2: 仅当 GCEN = 1 时，才会应答地址。

3: 仅在 10 位寻址模式下才会与地址的高字节匹配。

PIC24FV32KA304 系列

寄存器 17-1: I2CxCON: I2Cx 控制寄存器

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	IPMIEN	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	

图注:	HC = 可由硬件清零的位
R = 可读位	W = 可写位
-n = 上电复位时的值	1 = 置 1
	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

bit 15	I2CEN: I2Cx 使能位 1 = 使能 I2Cx 模块并将 SDAx 和 SCLx 引脚配置为串行端口引脚 0 = 禁止 I2Cx 模块; 所有 I ² C™ 引脚由端口功能控制
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 = 模块在空闲模式下继续工作
bit 12	SCLREL: SCLx 释放控制位 (作为 I ² 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) <u>如果 STREN = 1:</u> 该位为 R/W (即软件可写入 0 来启动时钟延长或写入 1 来释放时钟)。在从器件发送开始时由硬件清零。 在从器件接收结束时由硬件清零。 <u>如果 STREN = 0:</u> 该位为 R/S (即软件只能写入 1 来释放时钟)。在从器件发送开始时由硬件清零。
bit 11	IPMIEN: 智能外设管理接口 (IPMI) 使能位 1 = 使能 IPMI 支持模式; 应答所有地址 0 = 禁止 IPMI 支持模式
bit 10	A10M: 10 位从器件寻址位 1 = I2CxADD 是一个 10 位从地址 0 = I2CxADD 是一个 7 位从地址
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 引脚门限值 0 = 禁止 SMBus 输入门限值
bit 7	GCEN: 广播呼叫使能位 (作为 I ² C 从器件工作时) 1 = 允许在 I2CxRSR 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
bit 6	STREN: SCLx 时钟延长使能位 (作为 I ² C 从器件工作时) 与 SCLREL 位配合使用。 1 = 使能软件或接收时钟延长 0 = 禁止软件或接收时钟延长

寄存器 17-1: I2CxCON: I2Cx 控制寄存器 (续)

- bit 5 **ACKDT:** 应答数据位 (作为 I²C 主器件工作时; 适用于主器件接收操作)
当软件启动应答序列时将发送的值。
1 = 在应答期间发送 NACK
0 = 在应答期间发送 ACK
- bit 4 **ACKEN:** 应答序列使能位 (作为 I²C 主器件工作时; 适用于主器件接收操作)
1 = 在 SDAx 和 SCLx 引脚上启动应答序列, 并发送 ACKDT 数据位; 在主器件应答序列结束时由硬件清零
0 = 没有启动应答序列
- bit 3 **RCEN:** 接收使能位 (作为 I²C 主器件工作时)
1 = 使能 I²C 接收模式。在主器件接收完数据字节的 8 位后由硬件清零
0 = 没有启动接收序列
- bit 2 **PEN:** 停止条件使能位 (作为 I²C 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 = 没有发出停止条件
- bit 1 **RSEN:** 重复启动条件使能位 (作为 I²C 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 = 没有发出重复启动条件
- bit 0 **SEN:** 启动条件使能位 (作为 I²C 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 = 没有发出启动条件

PIC24FV32KA304 系列

寄存器 17-2: I2CxSTAT: I2Cx 状态寄存器

R-0, HSC	R-0, HSC	U-0	U-0	U-0	R/C-0, HS	R-0, HSC	R-0, HSC
ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10
bit 15							bit 8

R/C-0, HS	R/C-0, HS	R-0, HSC	R/C-0, HSC	R/C-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

图注:	C = 可清零位	HS = 可由硬件置 1 的位	HSC = 可由硬件置 1/ 清零的位
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **ACKSTAT:** 应答状态位

1 = 上次检测到 NACK
0 = 上次检测到 ACK
在应答结束时由硬件置 1 或清零。

bit 14 **TRSTAT:** 发送状态位

(作为 I²C 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 = 主器件未在进行发送
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 13-11 **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 = 未发生冲突
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。当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。当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 I²C 模块忙, 尝试写 I2CxTRN 寄存器的操作失败
0 = 未发生冲突
当总线忙时写 I2CxTRN 会使硬件将该位置 1 (由软件清零)。

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 = 未溢出
尝试将 I2CxRSR 的内容传输到 I2CxRCV 时由硬件置 1 (由软件清零)。

bit 5 **D/A:** 数据 / 地址位 (作为 I²C 从器件工作时)

1 = 表示上次接收的字节为数据
0 = 表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; 在写入 I2CxTRN 或接收从器件字节时由硬件置 1。

寄存器 17-2: I2CxSTAT: I2Cx 状态寄存器 (续)

- bit 4 **P:** 停止位
1 = 表示上次检测到停止位
0 = 表示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 = 表示上次检测到启动位 (或重复启动位)
0 = 表示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 (作为 I²C 从器件工作时)
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 I²C 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 为满
0 = 接收未完成, I2CxRCV 为空
用接收到的字节写 I2CxRCV 时由硬件置 1; 当用软件读 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, I2CxTRN 为满
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。

PIC24FV32KA304 系列

寄存器 17-3: 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
—	—	—	—	—	—	AMSK9	AMSK8
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
AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-10 未实现: 读为 0

bit 9-0 **AMSK<9:0>**: 地址位 bit x 的掩码选择位

1 = 使能输入报文地址 bit x 的掩码; 在此位置上不需要位匹配

0 = 禁止 bit x 的掩码; 在此位置上需要位匹配

寄存器 17-4: PADCFG1: 焊盘配置控制寄存器

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	U-0	U-0	U-0	U-0
—	—	SMBUSDEL2	SMBUSDEL1	—	—	—	—
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-6 未实现: 读为 0

bit 5 **SMBUSDEL2**: SMBus SDA2 输入延时选择位

1 = 为 I2C2 模块配置了较长的 SMBus 输入延时 (标称延时为 300 ns)

0 = 为 I2C2 模块配置了传统的输入延时 (标称延时为 150 ns)

bit 4 **SMBUSDEL1**: SMBus SDA1 输入延时选择位

1 = 为 I2C1 模块配置了较长的 SMBus 输入延时 (标称延时为 300 ns)

0 = 为 I2C1 模块配置了传统的输入延时 (标称延时为 150 ns)

bit 3-0 未实现: 读为 0

18.0 通用异步收发器 (UART)

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解通用异步收发器的更多信息，请参见《PIC24F 系列参考手册》中的第 21 章 “UART” (DS39708A_CN)。

通用异步收发器 (Universal Asynchronous Receiver Transmitter, UART) 模块是 PIC24F 系列器件提供的串行 I/O 模块之一。UART 是可以和外设 (例如个人电脑、LIN/J2602、RS-232 和 RS-485 接口) 通信的全双工异步系统。UART 模块还通过 UxCTS 和 UxRTS 引脚支持硬件流控制，该模块中还包含 IrDA® 编码器和解码器。

UART 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 引脚实现硬件流控制
- 完全集成的波特率发生器 (Integrated Baud Rate Generator, IBRG)，具有 16 位预分频器

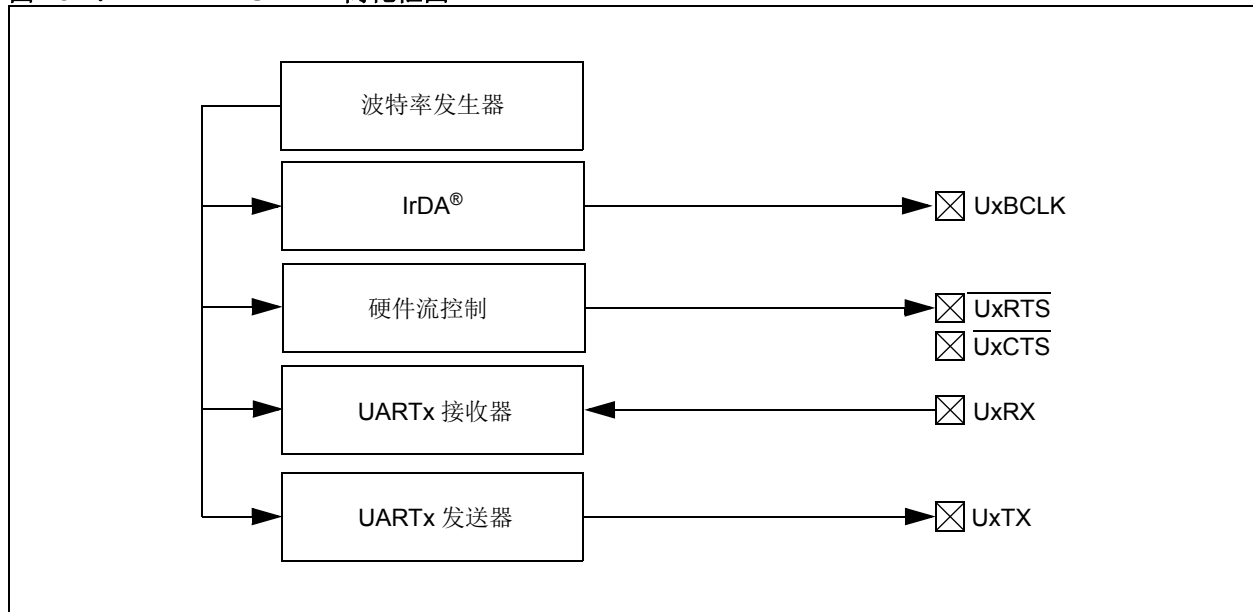
- 当器件工作在 16 MIPS 时，波特率范围从 15 bps 到 1 Mbps
- 4 级深先进先出 (First-In-First-Out, FIFO) 发送数据缓冲器
- 4 级深 FIFO 接收数据缓冲器
- 奇偶校验、帧和缓冲器溢出错误检测
- 支持带地址检测的 9 位模式 (第 9 位 = 1)
- 发送和接收中断
- 支持诊断的环回模式
- 支持同步和间隔字符
- 支持自动波特率检测
- IrDA® 编码器和解码器逻辑
- 支持 IrDA 的 16 倍频波特率时钟输出

图 18-1 显示了 UARTx 的简化框图。UARTx 模块由以下重要硬件组成：

- 波特率发生器
- 异步发送器
- 异步接收器

注： 在本节中，若要引用与特定 UART 模块相关的寄存器和位，通常在其名称中使用 x 来代替特定的模块编号。因此，“UxSTA”可能指的是 UART1 或 UART2 的 UART 状态寄存器。

图 18-1: UARTx 简化框图



PIC24FV32KA304 系列

18.1 UARTx 波特率发生器 (BRG)

UARTx 模块包含一个专用的 16 位波特率发生器 (BRG)。UxBRG 寄存器控制一个独立运行的 16 位定时器的周期。公式 18-1 给出了 BRGH = 0 时计算波特率的公式。

公式 18-1: **BRGH = 0 时的 UARTx 波特率⁽¹⁾**

$$\text{波特率} = \frac{F_{CY}}{16 \cdot (UxBRG + 1)}$$
$$UxBRG = \frac{F_{CY}}{16 \cdot \text{波特率}} - 1$$

注 1: 基于 $F_{CY} = F_{OSC}/2$; 禁止打盹模式和 PLL。

例 18-1 给出了下列条件下的波特率误差计算:

- $F_{CY} = 4 \text{ MHz}$
- 目标波特率 = 9600

例 18-1: **波特率误差计算 (BRGH = 0)⁽¹⁾**

目标波特率

=

$F_{CY}/(16 (UxBRG + 1))$

求解 UxBRG 值:

$UxBRG$

=

$((F_{CY}/\text{目标波特率})/16) - 1$

$UxBRG$

=

$((4000000/9600)/16) - 1$

$UxBRG$

=

25

计算的波特率

=

$4000000/(16 (25 + 1))$

=

9615

误差

=

$\frac{(\text{计算的波特率} - \text{目标波特率})}{\text{目标波特率}}$

=

$(9615 - 9600)/9600$

=

0.16%

注 1: 基于 $F_{CY} = F_{OSC}/2$; 禁止打盹模式和 PLL。

可能的最大波特率 (BRGH = 0) 是 $F_{CY}/16$ (当 $UxBRG = 0$ 时), 可能的最小波特率是 $F_{CY}/(16 \cdot 65536)$ 。

公式 18-2 给出了 BRGH = 1 时计算波特率的公式。

公式 18-2: **BRGH = 1 时的 UARTx 波特率⁽¹⁾**

$$\text{波特率} = \frac{F_{CY}}{4 \cdot (UxBRG + 1)}$$
$$UxBRG = \frac{F_{CY}}{4 \cdot \text{波特率}} - 1$$

注 1: 基于 $F_{CY} = F_{OSC}/2$; 禁止打盹模式和 PLL。

可能的最大波特率 (BRGH = 1) 是 $F_{CY}/4$ (当 $UxBRG = 0$ 时), 可能的最小波特率是 $F_{CY}/(4 \cdot 65536)$ 。

向 UxBRG 寄存器中写入新值会导致 BRG 定时器复位 (清零)。这保证了 BRG 在产生新的波特率之前不需要等待定时器溢出。

18.2 8 位数据模式下的发送

1. 设置 UARTx:
 - a) 将适当的值写入数据位、奇偶校验位和停止位。
 - b) 将适当的波特率值写入 UxBRG 寄存器。
 - c) 设置发送和接收中断允许位和优先级位。
2. 使能 UARTx。
3. 将 UTXEN 位置 1（置 1 后两个周期内产生发送中断）。
4. 将数据字节写入 UxTXREG 字的低字节。该数据字节将被立即传送给发送移位寄存器（TSR）且在波特率时钟的下一个上升沿开始移出串行比特流。
5. 另外，数据字节也可在 UTXEN = 0 时被传送，且随后用户可将 UTXEN 置 1。由于波特率时钟是从清零状态启动的，所以该行为能立即开始发送串行比特流。
6. 将根据中断控制位 UTXISELx 的设置产生发送中断。

18.3 9 位数据模式下的发送

1. 设置 UARTx（如第 18.2 节“8 位数据模式下的发送”所述）。
2. 使能 UARTx。
3. 将 UTXEN 位置 1（置 1 后两个周期内产生发送中断）。
4. 只将 16 位值写入 UxTXREG。
5. 向 UxTXREG 写入一个字可触发 9 位数据向 TSR 传送。串行比特流将会在波特率时钟的第一个上升沿开始移出。
6. 将根据中断控制位 UTXISELx 的设置产生发送中断。

18.4 间隔和同步发送序列

下面的序列将发送一个由间隔字符和其后的自动波特率同步字节组成的报文帧头。

1. 将 UARTx 配置为所需的模式。
2. 将 UTXEN 和 UTXBRK 置 1——设置间隔字符。
3. 将一个“虚拟”字符装入 UxTXREG 寄存器中以启动发送（该值会被忽略）。
4. 向 UxTXREG 写入 55h——将同步字符装入发送 FIFO 中。
5. 当间隔字符发送完成后，由硬件将 UTXBRK 位复位。然后开始发送同步字符。

18.5 8 位或 9 位数据模式下的接收

1. 设置 UARTx（如第 18.2 节“8 位数据模式下的发送”所述）。
2. 使能 UARTx。
3. 当接收到一个或多个数据字符时，将会根据中断控制位 URXISELx 的设置产生接收中断。
4. 读 OERR 位以确定是否发生了溢出错误。必须用软件将 OERR 位复位。
5. 读 UxRXREG。

读取 UxRXREG 字符的行为会将下一个字符传送到接收 FIFO 的顶部，其中包含一组新的 PERR 和 FERR 值。

18.6 UxCTS 和 UxRTS 控制引脚的操作

UARTx 允许发送（ $\overline{\text{UxCTS}}$ ）和 UARTx 请求发送（ $\overline{\text{UxRTS}}$ ）是两个与 UARTx 模块有关由硬件控制的引脚。这两个引脚允许 UARTx 运行在单工模式和流控制模式下。这两个引脚控制数据终端设备（Data Terminal Equipment, DTE）之间的数据发送和接收。通过 UxMODE 寄存器的 UEN<1:0> 位来配置这两个引脚。

18.7 红外支持

UARTx 模块提供两种类型的红外 UARTx 支持：一种是 IrDA 时钟输出，支持外部 IrDA 编码器和解码器（支持传统模块）；另一种是完全实现的 IrDA 编码器和解码器。

因为 IrDA 模式需要 16 倍频的波特率时钟，所以它们仅在 BRGH 位（UxMODE<3>）为 0 时才能工作。

18.7.1 支持外部 IrDA——IrDA 时钟输出

为了支持外部 IrDA 编码器和解码器，可将 UxBCLK 引脚（与 UxRTS 引脚相同）配置为产生 16 倍频的波特率时钟。当使能了 UARTx 模块且 UEN<1:0> = 11 时，UxBCLK 引脚将输出 16 倍频的波特率时钟，用于支持 IrDA 编解码器芯片。

18.7.2 内置 IrDA 编码器和解码器

UARTx 模块在其内部完全实现了 IrDA 编码器和解码器。内置 IrDA 编码器和解码器的功能可通过 IREN 位（UxMODE<12>）来使能。当使能（IREN = 1）时，接收引脚（UxRX）可作为红外接收器的输入引脚。发送引脚（UxTX）可作为红外发送器的输出引脚。

PIC24FV32KA304 系列

寄存器 18-1: 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 ⁽²⁾	R/W-0 ⁽²⁾
UARTEN	—	USIDL	IREN ⁽¹⁾	RTSMD	—	UEN1	UEN0
bit 15						bit 8	

R/C-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	RXINV	BRGH	PDSEL1	PDSEL0	STSEL
bit 7						bit 0	

图注:	C = 可清零位	HC = 可由硬件清零的位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零
		x = 未知

bit 15	UARTEN: UARTx 使能位 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 = 模块在空闲模式下继续工作
bit 12	IREN: IrDA [®] 编码器和解码器使能位 ⁽¹⁾ 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: <u>UxRTS</u> 引脚模式选择位 1 = <u>UxRTS</u> 引脚处于单工模式 0 = <u>UxRTS</u> 引脚处于流控制模式
bit 10	未实现: 读为 0
bit 9-8	UEN<1:0>: UARTx 使能位 ⁽²⁾ 11 = 使能并使用 UxTX、UxRX 和 UxBCLK 引脚; <u>UxCTS</u> 引脚由端口锁存器控制 10 = 使能并使用 UxTX、UxRX、 <u>UxCTS</u> 和 <u>UxRTS</u> 引脚 01 = 使能并使用 UxTX、UxRX 和 <u>UxRTS</u> 引脚; <u>UxCTS</u> 引脚由端口锁存器控制 00 = 使能并使用 UxTX 和 UxRX 引脚; <u>UxCTS</u> 和 <u>UxRTS/UxBCLK</u> 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时唤醒的使能位 1 = UARTx 将继续采样 UxRX 引脚; 在出现下降沿时产生中断; 在随后的上升沿该位由硬件清零 0 = 未使能唤醒
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
bit 5	ABAUD: 自动波特率使能位 1 = 使能在下一个字符传输过程中对波特率进行测量 —— 需要接收到同步字段 (55h); 完成时由硬件清零 0 = 禁止或已完成波特率测量
bit 4	RXINV: 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

注 1: 此功能仅适用于 16 倍 BRG 模式 (BRGH = 0)。

2: 位的可用性取决于引脚的可用性。

寄存器 18-1: UxMODE: UARTx 模式寄存器 (续)

bit 3	BRGH: 高波特率使能位 1 = BRG 每个位周期内产生 4 个时钟 (4 倍频的波特率时钟, 高速模式) 0 = BRG 每个位周期内产生 16 个时钟 (16 倍频的波特率时钟, 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: 此功能仅适用于 16 倍 BRG 模式 (BRGH = 0)。
2: 位的可用性取决于引脚的可用性。

PIC24FV32KA304 系列

寄存器 18-2: 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, HSC	R-1, HSC
UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-1, HSC	R-0, HSC	R-0, HSC	R/C-0, HS	R-0, HSC
URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7							bit 0

图注:	HC = 可由硬件清零的位		
HS = 可由硬件置 1 的位	C = 可清零位	HSC = 可由硬件置 1/ 清零的位	
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15,13	UTXISEL<1:0>: UARTx 发送中断模式选择位 11 = 保留; 不要使用 10 = 当一个字符被传送到发送移位寄存器 (TSR) 导致发送缓冲器为空时, 产生中断 01 = 当最后一个字符被移出发送移位寄存器时产生中断; 发送操作全部完成 00 = 当一个字符被传送到发送移位寄存器 (前提是发送缓冲器中至少还有一个字符可供发送) 时产生中断
bit 14	UTXINV: IrDA® 编码器发送极性翻转位 如果 IREN = 0: 1 = UxTX 空闲状态为 0 0 = UxTX 空闲状态为 1 如果 IREN = 1: 1 = UxTX 空闲状态为 1 0 = 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 = 在下次发送时发出同步间隔字符 —— 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 1 = 使能发送, UARTx 控制 UxTX 引脚 0 = 禁止发送; 终止所有等待的发送, 缓冲器复位。端口寄存器控制 UxTX 引脚。
bit 9	UTXBF: UARTx 发送缓冲器满状态位 (只读) 1 = 发送缓冲器满 0 = 发送缓冲器未满, 至少还可写入一个字符
bit 8	TRMT: UARTx 发送移位寄存器空位 (只读) 1 = 发送移位寄存器为空, 同时发送缓冲器为空 (上一次发送已完成) 0 = 发送移位寄存器非空, 发送正在进行或等待发送
bit 7-6	URXISEL<1:0>: UARTx 接收中断模式选择位 11 = 当 RSR 传输使接收缓冲器为满时 (即, 有 4 个数据字符), 中断标志位置 1 10 = 当 RSR 传输使接收缓冲器为 3/4 满时 (即, 有 3 个数据字符), 中断标志位置 1 0x = 当接收缓冲器接收到来自 RSR 的任何字符时, 中断标志位置 1。接收缓冲器有一个或多个字符

寄存器 18-2: UxSTA: UARTx 状态和控制寄存器 (续)

bit 5	ADDEN: 地址字符检测位 (接收数据的第 8 位 = 1) 1 = 使能地址检测模式; 如果没有选择 9 位模式, 这个控制位将无效 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 (在接收 FIFO 顶部的字符) 的奇偶校验错误 0 = 没有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 (在接收 FIFO 顶部的字符) 的帧错误 0 = 没有检测到帧错误
bit 1	OERR: 接收缓冲器溢出错误状态位 (只读 / 清零) 1 = 接收缓冲器已经溢出 0 = 接收缓冲器没有溢出。清零先前置 1 的 OERR 位 (1 → 0 的跳变) 会使接收缓冲器和 RSR 复位为空状态
bit 0	URXDA: UARTx 接收缓冲器中数据可用位 (只读) 1 = 接收缓冲器中有数据, 有至少一个字符可被读取 0 = 接收缓冲器为空

PIC24FV32KA304 系列

寄存器 18-3: UxTXREG: UARTx 发送寄存器

U-x	U-x	U-x	U-x	U-x	U-x	U-x	W-x
—	—	—	—	—	—	—	UTX8
bit 15							bit 8
W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
UTX7	UTX6	UTX5	UTX4	UTX3	UTX2	UTX1	UTX0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-9 未实现: 读为 0
- bit 8 UTX8: UARTx 发送的字符数据位 (在 9 位模式下)
- bit 7-0 UTX<7:0>: UARTx 发送的字符数据位

寄存器 18-4: UxRXREG: UARTx 接收寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0, HSC
—	—	—	—	—	—	—	URX8
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
URX7	URX6	URX5	URX4	URX3	URX2	URX1	URX0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-9 未实现: 读为 0
- bit 8 URX8: UARTx 接收的字符数据位 (在 9 位模式下)
- bit 7-0 URX<7:0>: UARTx 接收的字符数据位

19.0 实时时钟和日历 (RTCC)

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。关于实时时钟和日历的更多信息，请参见《PIC24F 系列参考手册》中的第 29 章“实时时钟和日历 (RTCC)” (DS39696A_CN)。

RTCC 为用户提供了可校准的实时时钟和日历 (RTCC) 功能。

RTCC 模块的主要特性包括：

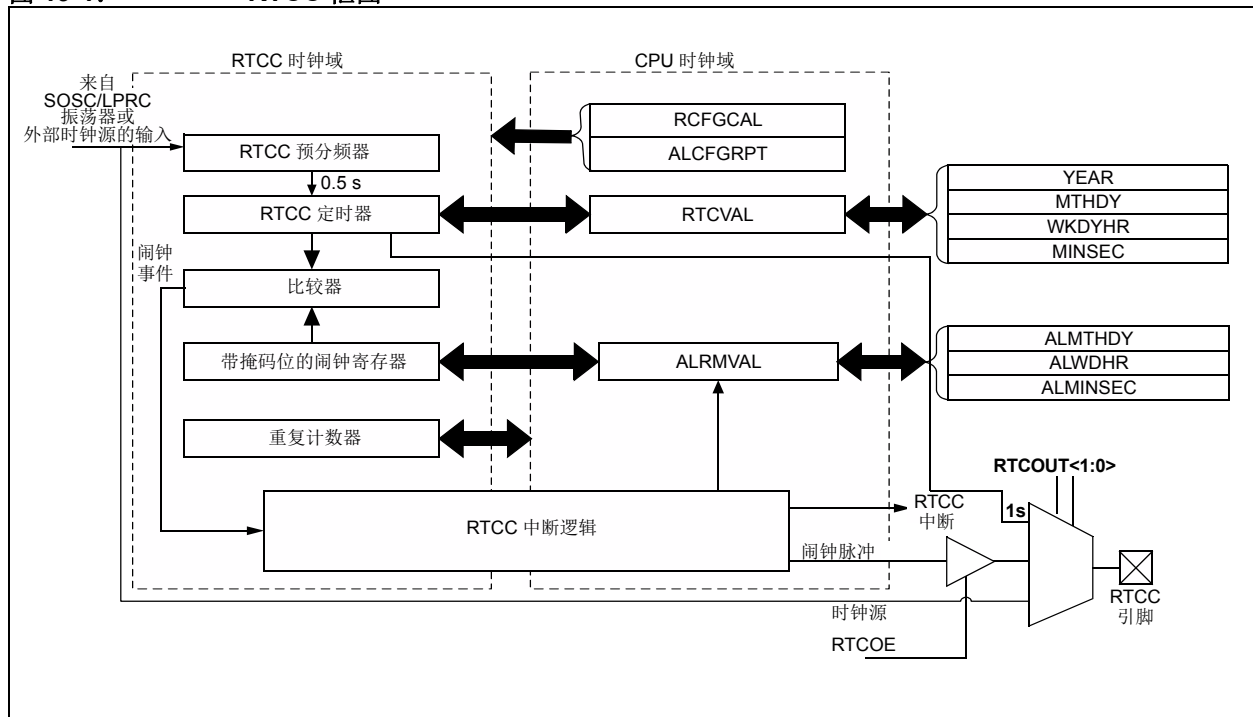
- 可在深度休眠模式下工作
- 可选时钟源
- 采用 24 小时格式提供小时、分和秒
- 可分辨半秒的时长
- 提供日历：星期、日期、月和年
- 闹钟可配置半秒、1 秒、10 秒、1 分钟、10 分钟、1 小时、1 天、1 周、1 个月或 1 年
- 闹钟使用递减计数器进行重复
- 闹钟具有无限重复响铃
- 年份为 2000 至 2099，带闰年修正

- 采用 BCD 格式，以减小软件开销
- 为长期电池工作进行了优化
- 32.768 kHz 时钟晶振的用户校准 / 带周期性自动调整的 32 kHz INTRC 频率
- 小数秒同步
- 校准每月误差在 ± 2.64 秒内
- 校准最高 260 ppm 的晶振误差
- 可定期唤醒外部器件，而无需 CPU 干预（外部功率控制）
- 功率控制输出用于控制外部电路
- 校准每 15 秒生效
- 以下列任一时钟运行：
 - 32.768 kHz 的外部实时时钟
 - 内部 31.25 kHz LPRC 时钟
 - 50 Hz 或 60 Hz 外部输入

19.1 RTCC 时钟源

用户可选择 SOSC 晶振、LPRC 内部振荡器或外部 50 Hz/60 Hz 电力线输入作为 RTCC 模块的参考时钟。这使用户可根据整体系统需要在系统成本、精度和功耗间进行权衡选择。

图 19-1: RTCC 框图



PIC24FV32KA304 系列

19.2 RTCC 模块寄存器

RTCC 模块寄存器分为三类：

- RTCC 控制寄存器
- RTCC 值寄存器
- 闹钟值寄存器

19.2.1 寄存器映射

要限制寄存器接口，则需通过相应的寄存器指针访问 RTCC 定时器和闹钟时间寄存器。RTCC 值寄存器窗口（RTCVALH 和 RTCVALL）使用 RTCPTR 位（RCFGCAL<9:8>）来选择需要的定时器寄存器对（见表 19-1）。

通过写 RTCVALH 字节，使 RTCC 指针值（RTCPTR<1:0> 位）逐次递减 1 直至 00。一旦其达到 00，在手动更改指针值前，可通过 RTCVALH 和 RTCVALL 来访问分钟和秒值。

表 19-1: RTCC 值寄存器映射

RTCPTR<1:0>	RTCC 值寄存器窗口	
	RTCVAL<15:8>	RTCVAL<7:0>
00	分钟数	秒数
01	星期	小时数
10	月	日
11	—	年

闹钟值寄存器窗口（ALRMVALH 和 ALRMVALL）使用 ALRMPTRx 位（ALCFGRPT<9:8>）来选择所需的闹钟寄存器对（见表 19-2）。

通过写 ALRMVALH 字节，使闹钟指针值（ALRMPTR<1:0> 位）逐次递减 1 直至 00。一旦其达到 00，在手动更改指针值前，可通过 ALRMVALH 和 ALRMVALL 来访问 ALRMMIN 和 ALRMSEC 的值。

例 19-1: 将 RTCWREN 位置 1

asm	volatile	("push w7");	
asm	volatile	("push w8");	
asm	volatile	("disi #5");	
asm	volatile	("mov #0x55, w7");	
asm	volatile	("mov w7, _NVMKEY");	
asm	volatile	("mov #0xAA, w8");	
asm	volatile	("mov w8, _NVMKEY");	
asm	volatile	("bset _RCFGCAL, #13");	//set the RTCWREN bit
asm	volatile	("pop w8");	
asm	volatile	("pop w7");	

表 19-2: ALRMVAL 寄存器映射

ALRMPTR<1:0>	闹钟值寄存器窗口	
	ALRMVAL<15:8>	ALRMVAL<7:0>
00	ALRMMIN	ALRMSEC
01	ALRMWD	ALRMHR
10	ALRMMNTH	ALRMDAY
11	PWCSTAB	PWCSAMP

考虑到 16 位内核不能区分 8 位和 16 位读操作，用户必须注意读 ALRMVALH 或 ALRMVALL 字节会递减 ALRMPTR<1:0> 的值。同样，读 RTCVALH 或 RTCVALL 字节会使 RTCPTR<1:0> 的值递减。

注： 这仅适用于读操作，而不适用于写操作。

19.2.2 写锁定

为了对任何 RTCC 定时器寄存器执行写操作，必须将 RTCWREN 位（RTCPWC<13>）置 1（见例 19-1）。

注： 为避免意外写入定时器，建议在除要特意执行写操作之外的任何时候都保持 RTCWREN 位（RCFGCAL<13>）清零。对于要使 RTCWREN 位置 1 的操作，由于在 55h/AAh 序列和 RTCWREN 置 1 之间仅允许有一个指令周期的时间；因此，建议按照例 19-1 中的过程执行代码。

19.2.3 选择 RTCC 时钟源

RTCC 模块的时钟源可通过 RTCCSEL<1:0> 位进行选择：00 = 辅助振荡器，01 = LPRC，10 = 50 Hz 外部时钟，且 11 = 60 Hz 外部时钟。

19.2.4 RTCC 控制寄存器

寄存器 19-1: RCFGAL: RTCC 校准和配置寄存器⁽¹⁾

R/W-0	U-0	R/W-0	R-0, HSC	R-0, HSC	R/W-0	R/W-0	R/W-0
RTCEN ⁽²⁾	—	RTCWREN	RTCSYNC	HALFSEC ⁽³⁾	RTCOE	RTCPTR1	RTCPTR0
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
CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0
bit 7							bit 0

图注:	HSC = 可由硬件置 1/ 清零的位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **RTCEN:** RTCC 使能位⁽²⁾

1 = 使能 RTCC 模块
0 = 禁止 RTCC 模块

bit 14 **未实现:** 读为 0

bit 13 **RTCWREN:** RTCC 值寄存器写使能位

1 = RTCVALH 和 RTCVALL 寄存器可被用户写入
0 = RTCVALH 和 RTCVALL 寄存器被锁定不允许用户写入

bit 12 **RTCSYNC:** RTCC 值寄存器读同步位

1 = 由于计满返回, RTCVALH、RTCVALL 和 ALCFGRPT 寄存器可能会在读取期间变化, 从而导致读到的数据无效。如果两次读取寄存器得到的数据相同, 则认为数据有效。
0 = RTCVALH、RTCVALL 或 ALCFGRPT 寄存器可被读取, 而不用担心计满返回

bit 11 **HALFSEC:** 半秒状态位⁽³⁾

1 = 一秒的后半周期
0 = 一秒的前半周期

bit 10 **RTCOE:** RTCC 输出使能位

1 = 使能 RTCC 输出
0 = 禁止 RTCC 输出

bit 9-8 **RTCPTR<1:0>:** RTCC 值寄存器窗口指针位

当读 RTCVALH 和 RTCVALL 寄存器时, 指向相应的 RTCC 值寄存器。每次读或写 RTCVALH 时, RTCPTR<1:0> 的值都会减 1, 直至 00。

RTCVAL<15:8>:

00 = 分钟数
01 = 周期
10 = 月
11 = 保留

RTCVAL<7:0>:

00 = 秒数
01 = 小时数
10 = 日
11 = 年

注 1: RCFGAL 寄存器仅受到 POR 影响。

2: 仅当 RTCWREN = 1 时才允许写入 RTCEN 位。

3: 该位是只读位。当写入 MINSEC 寄存器的低半部分时, 该位清 0。

PIC24FV32KA304 系列

寄存器 19-1: **RCFGCAL: RTCC 校准和配置寄存器 ⁽¹⁾** (续)

bit 7-0 **CAL<7:0>**: RTC 漂移校准位

01111111 = 最大正向调整; 每分钟增加 508 个 RTC 时钟脉冲

·

·

·

00000001 = 最小正向调整; 每分钟增加 4 个 RTC 时钟脉冲

00000000 = 无调整

11111111 = 最小负向调整; 每分钟减少 4 个 RTC 时钟脉冲

·

·

·

10000000 = 最大负向调整; 每分钟减少 512 个 RTC 时钟脉冲

- 注 **1:** RCFGCAL 寄存器仅受到 POR 影响。
- 2:** 仅当 RTCWREN = 1 时才允许写入 RTCEN 位。
- 3:** 该位是只读位。当写入 MINSEC 寄存器的低半部分时, 该位清 0。

PIC24FV32KA304 系列

寄存器 19-2: RTCPWC: RTCC 配置寄存器 2⁽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
PWCEN	PWCPOL	PWCCPRE	PWCSPRE	RTCCLK1 ⁽²⁾	RTCCLK0 ⁽²⁾	RTCOUT1	RTCOUT0
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 = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **PWCEN:** 功率控制使能位
1 = 使能功率控制
0 = 禁止功率控制
- bit 14 **PWCPOL:** 功率控制极性位
1 = 功率控制输出为高电平有效
0 = 功率控制输出为低电平有效
- bit 13 **PWCCPRE:** 功率控制 / 稳定预分频比位
1 = PWC 稳定窗口时钟为 RTCC 源时钟的 2 分频
0 = PWC 稳定窗口时钟为 RTCC 源时钟的 1 分频
- bit 12 **PWCSPRE:** 功率控制采样预分频比位
1 = PWC 采样窗口时钟为 RTCC 源时钟的 2 分频
0 = PWC 采样窗口时钟为 RTCC 源时钟的 1 分频
- bit 11-10 **RTCCLK<1:0>:** RTCC 时钟选择位 ⁽²⁾
确定内部 RTCC 时钟源, 该源用于所有 RTCC 定时器操作。
00 = 外部辅助振荡器 (SOSC)
01 = 内部 LPRC 振荡器
10 = 外部电力线电源 – 50 Hz
11 = 外部电力线电源 – 60 Hz
- bit 9-8 **RTCOUT<1:0>:** RTCC 输出选择位
确定 RTCC 引脚输出源。
00 = RTCC 闹钟脉冲
01 = RTCC 秒时钟
10 = RTCC 时钟
11 = 功率控制
- bit 7-0 **未实现:** 读为 0

注 1: RTCPWC 寄存器仅受到 POR 影响。

2: 当新值被写入这些寄存器位时, 也应被写入秒值寄存器以正确重新设置 RTCC 中的时钟预分频比。

PIC24FV32KA304 系列

寄存器 19-3: **ALCFGRPT: 闹钟配置寄存器**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ALRMEN	CHIME	AMASK3	AMASK2	AMASK1	AMASK0	ALRMPTR1	ALRMPTR0
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
ARPT7	ARPT6	ARPT5	ARPT4	ARPT3	ARPT2	ARPT1	ARPT0
bit 7						bit 0	

图注:			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	ALRMEN: 闹钟使能位 1 = 使能闹钟（只要 ARPT<7:0> = 00h 且 CHIME = 0，就会在闹钟事件后自动清零） 0 = 禁止闹钟
bit 14	CHIME: 响铃使能位 1 = 使能响铃；允许 ARPT<7:0> 位从 00h 返回到 FFh 0 = 禁止响铃；ARPT<7:0> 位到达 00h 时停止
bit 13-10	AMASK<3:0>: 闹钟掩码配置位 0000 = 每半秒 0001 = 每秒 0010 = 每 10 秒 0011 = 每分钟 0100 = 每 10 分钟 0101 = 每小时 0110 = 一天一次 0111 = 一周一次 1000 = 一月一次 1001 = 一年一次（除了配置为 2 月 29 日，亦即每 4 年一次的情况外） 101x = 保留；不要使用 11xx = 保留；不要使用
bit 9-8	ALRMPTR<1:0>: 闹钟值寄存器窗口指针位 当读 ALRMVALH 和 ALRMVALL 寄存器时指向相应的闹钟值寄存器。每次读或写 ALRMVALH 时，ALRMPTR<1:0> 的值都会减 1，直至 00。 <u>ALRMVAL<15:8>:</u> 00 = ALRMMIN 01 = ALRMWD 10 = ALRMMNTH 11 = 未实现 <u>ALRMVAL<7:0>:</u> 00 = ALRMSEC 01 = ALRMHR 10 = ALRMDAY 11 = 未实现
bit 7-0	ARPT<7:0>: 闹钟重复计数器值位 11111111 = 闹钟将重复 255 次 . . . 00000000 = 闹钟不重复 每发生一次闹钟事件，计数器就减 1；除非 CHIME = 1，否则计数器不能从 00h 返回到 FFh。

19.2.5 RTCVAL 寄存器映射

寄存器 19-4: YEAR: 年值寄存器⁽¹⁾

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

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
YRTEN3	YRTEN2	YRTEN1	YRTEN0	YRONE3	YRONE2	YRONE1	YRONE0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 未实现: 读为 0

bit 7-4 **YRTEN<3:0>**: 年的十位数字的二 – 十进制码 (Binary Coded Decimal, BCD) 值位
包含从 0 到 9 的值。

bit 3-0 **YRONE<3:0>**: 年的个位数字的 BCD 值位
包含从 0 到 9 的值。

注 1: 仅当 RTCWREN = 1 时才允许写入 YEAR 寄存器。

寄存器 19-5: MTHDY: 月和日值寄存器⁽¹⁾

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MHTEN0	MTHONE3	MTHONE2	MTHONE1	MTHONE0
bit 15				bit 8			

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-13 未实现: 读为 0

bit 12 **MHTEN0**: 月的十位数字的 BCD 值位
值为 0 或 1。

bit 11-8 **MTHONE<3:0>**: 月的个位数字的 BCD 值位
包含从 0 到 9 的值。

bit 7-6 未实现: 读为 0

bit 5-4 **DAYTEN<1:0>**: 日的十位数字的 BCD 值位
包含从 0 到 3 的值。

bit 3-0 **DAYONE<3:0>**: 日的个位数字的 BCD 值位
包含从 0 到 9 的值。

注 1: 仅当 RTCWREN = 1 时才允许写入该寄存器。

PIC24FV32KA304 系列

寄存器 19-6: WKDYHR: 星期和小时值寄存器⁽¹⁾

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY2	WDAY1	WDAY0
bit 15						bit 8	

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
bit 7						bit 0	

图注:

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

- bit 15-11 **未实现:** 读为 0
 bit 10-8 **WDAY<2:0>:** 星期数字的 BCD 值位
 包含从 0 到 6 的值。
 bit 7-6 **未实现:** 读为 0
 bit 5-4 **HRTEN<1:0>:** 小时的十位数字的 BCD 值位
 包含从 0 到 2 的值。
 bit 3-0 **HRONE<3:0>:** 小时的个位数字的 BCD 值位
 包含从 0 到 9 的值。

注 1: 仅当 RTCWREN = 1 时才允许写入该寄存器。

寄存器 19-7: MINSEC: 分钟和秒值寄存器

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	MINTEN2	MINTEN1	MINTEN0	MINONE3	MINONE2	MINONE1	MINONE0
bit 15						bit 8	

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN2	SECTEN1	SECTEN0	SECONE3	SECONE2	SECONE1	SECONE0
bit 7						bit 0	

图注:

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

- bit 15 **未实现:** 读为 0
 bit 14-12 **MINTEN<2:0>:** 分钟的十位数字的 BCD 值位
 包含从 0 到 5 的值。
 bit 11-8 **MINONE<3:0>:** 分钟的个位数字的 BCD 值位
 包含从 0 到 9 的值。
 bit 7 **未实现:** 读为 0
 bit 6-4 **SECTEN<2:0>:** 秒的十位数字的 BCD 值位
 包含从 0 到 5 的值。
 bit 3-0 **SECONE<3:0>:** 秒的个位数字的 BCD 值位
 包含从 0 到 9 的值。

19.2.6 ALRMVAL 寄存器映射

寄存器 19-8: ALMTHDY: 闹钟月和日值寄存器⁽¹⁾

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MTHTEN0	MTHONE3	MTHONE2	MTHONE1	MTHONE0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
bit 7							bit 0

图注:

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

- bit 15-13 **未实现:** 读为 0
bit 12 **MTHTEN0:** 月的十位数字的 BCD 值位
 值为 0 或 1。
bit 11-8 **MTHONE<3:0>:** 月的个位数字的 BCD 值位
 包含从 0 到 9 的值。
bit 7-6 **未实现:** 读为 0
bit 5-4 **DAYTEN<1:0>:** 日的十位数字的 BCD 值位
 包含从 0 到 3 的值。
bit 3-0 **DAYONE<3:0>:** 日的个位数字的 BCD 值位
 包含从 0 到 9 的值。

注 1: 仅当 RTCWREN = 1 时才允许写入该寄存器。

寄存器 19-9: ALWDHR: 闹钟星期和小时值寄存器⁽¹⁾

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY2	WDAY1	WDAY0
bit 15							bit 8

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	HRTEN1	HRTEN0	HRONE3	HRONE2	HRONE1	HRONE0
bit 7							bit 0

图注:

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

- bit 15-11 **未实现:** 读为 0
bit 10-8 **WDAY<2:0>:** 星期数字的 BCD 值位
 包含从 0 到 6 的值。
bit 7-6 **未实现:** 读为 0
bit 5-4 **HRTEN<1:0>:** 小时的十位数字的 BCD 值位
 包含从 0 到 2 的值。
bit 3-0 **HRONE<3:0>:** 小时的个位数字的 BCD 值位
 包含从 0 到 9 的值。

注 1: 仅当 RTCWREN = 1 时才允许写入该寄存器。

PIC24FV32KA304 系列

寄存器 19-10: ALMINSEC: 闹钟分钟和秒值寄存器

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	MINTEN2	MINTEN1	MINTEN0	MINONE3	MINONE2	MINONE1	MINONE0
bit 15							bit 8

U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SECTEN2	SECTEN1	SECTEN0	SECONE3	SECONE2	SECONE1	SECONE0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15未实现: 读为 0
- bit 14-12MINTEN<2:0>: 分钟的十位数字的 BCD 值位
包含从 0 到 5 的值。
- bit 11-8MINONE<3:0>: 分钟个位数字的 BCD 值位
包含从 0 到 9 的值。
- bit 7未实现: 读为 0
- bit 6-4SECTEN<2:0>: 秒的十位数字的 BCD 值位
包含从 0 到 5 的值。
- bit 3-0SECONE<3:0>: 秒的个位数字的 BCD 值位
包含从 0 到 9 的值。

寄存器 19-11: **RTCCSWT: 稳定 / 采样窗口定时器寄存器** ⁽¹⁾

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PWCSTAB7	PWCSTAB6	PWCSTAB5	PWCSTAB4	PWCSTAB3	PWCSTAB2	PWCSTAB1	PWCSTAB0
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
PWCSAMP7	PWCSAMP6	PWCSAMP5	PWCSAMP4	PWCSAMP3	PWCSAMP2	PWCSAMP1	PWCSAMP0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 **PWCSTAB<7:0>: PWM 稳定窗口定时器位**

11111111 = 稳定窗口为 255 个 TPWCCLK 时钟周期

.

.

.

00000000 = 稳定窗口为 0 个 TPWCCLK 时钟周期

当触发闹钟事件时, 采样窗口启动。当 PWCEN = 1 时, 稳定窗口定时器在发生每个闹钟事件时开始计数。

bit 7-0 **PWCSAMP<7:0>: PWM 采样窗口定时器位**

11111111 = 始终使能采样窗口, 即使 PWCEN = 0

11111110 = 采样窗口为 254 个 TPWCCLK 时钟周期

.

.

.

00000000 = 采样窗口为 0 个 TPWCCLK 时钟周期

当 PWCEN = 1 时, 采样窗口定时器在稳定窗口结束时开始计数。如果 PWCSTAB<7:0> = 00000000 且 PWCEN = 1 时, 采样窗口定时器在发生每个闹钟事件时开始计数。

注 1: 仅当 RTCWREN = 1 时才允许写入该寄存器。

PIC24FV32KA304 系列

19.3 校准

实时晶振输入可使用周期性自动调整功能来校准。正确校准后，RTCC 可实现每月小于 3 秒的误差。这可通过确定误差时钟脉冲数并将其存储到 RCFGAL 寄存器的低半部分来完成。装入到 RCFGAL 低半部分的 8 位有符号值与 4 相乘，然后每分钟与 RTCC 定时器的值相加或相减一次。请参见以下步骤校准 RTCC：

1. 使用器件上的其他定时器资源，用户必须确定 32.768 kHz 晶振的误差。
2. 知道误差后，必须将其转换为每分钟的误差时钟脉冲数。
3. a) 如果振荡器频率高于理想值（步骤 2 中的计算结果为负），那么 RCFGAL 寄存器的值就必须为负。这样每分钟会在定时器计数值上减去指定的时钟脉冲数。
b) 如果振荡器频率低于理想值（步骤 2 中的计算结果为正），那么 RCFGAL 寄存器的值就必须为正。这样每分钟会在定时器计数值上加上指定的时钟脉冲数。

公式 19-1:

$$(\text{理想频率} \uparrow - \text{测量频率}) * 60 = \text{每分钟时钟数}$$
$$\uparrow \text{理想频率} = 32,768 \text{ Hz}$$

仅当定时器关闭或者紧跟在秒脉冲的上升沿 (SECONDS = 00、15、30 或 45 时除外) 后，才可写入 RCFGAL 寄存器的低半部分。这是因为 RTCC 每 15 秒会自动调整一次。

注：由用户决定误差值是否包含晶振初始误差、由温度引起的漂移以及晶体老化引起的漂移。

19.4 闹钟

- 闹钟的时间间隔可配置为从半秒到一年
- 使用 ALRMEN 位 (ALCFGRPT<15>) 使能
- 可选择一次性闹钟和重复闹钟

19.4.1 配置闹钟

使用 ALRMEN 位使能闹钟功能。闹钟事件发生后该位清零。只有在 ALRMEN = 0 时才允许写入 ALRMVAL。

如图 19-2 所示，可通过 AMASKx 位 (ALCFGRPT<13:10>) 配置闹钟时间间隔。这些位确定要触发闹钟，闹钟的哪些位、多少位必须和时钟值匹配。

也可配置闹钟根据预配置的时间间隔重复。一旦使能了闹钟，发生的总次数就会被存储到 ARPT<7:0> 位 (ALCFGRPT<7:0>)。当 ARPTx 位的值等于 00h 且 CHIME 位 (ALCFGRPT<14>) 清零时，禁止重复功能，仅发生单次闹钟。通过将 FFh 装入 ARPT<7:0>，可使闹钟重复最多 255 次。

每发生一次闹钟事件，ARPTx 位的值都会减 1。值达到 00h，将最后一次发出闹钟，然后 ALRMEN 位自动清零，关闭闹钟。

如果 CHIME 位 = 1，闹钟可无限次重复发生。在这种情况下，当 ARPTx 位的值达到 00h 时，不会禁止闹钟，而是返回 FFh 并继续无限次地计数。

19.4.2 闹钟中断

每一次闹钟事件都会产生一个中断。此外会提供闹钟脉冲输出，其频率是闹钟频率的一半。该输出与 RTCC 时钟完全同步，且可用作其他外设的触发时钟。

注：当使能闹钟 (ALRMEN = 1) 时，更改除 RCFGAL 和 ALCFGRPT 寄存器外的任何寄存器以及 CHIME 位都将导致错误的闹钟事件，从而引起错误的闹钟中断。为避免错误闹钟事件，只能在禁止闹钟 (ALRMEN = 0) 时更改定时器和闹钟值。建议在 RTCSYNC = 0 时更改 ALCFGRPT 寄存器和 CHIME 位。

图 19-2: 闹钟掩码设置

闹钟掩码设置 (AMASK<3:0>)	星期值	月	日	小时	分钟	秒
0000 - 每半秒	☐	☐ ☐	☐ ☐	☐ ☐	☐ ☐	☐ ☐
0001 - 每秒	☐	☐ ☐	☐ ☐	☐ ☐	☐ ☐	☐ ☐
0010 - 每 10 秒	☐	☐ ☐	☐ ☐	☐ ☐	☐ ☐	☐ s
0011 - 每分钟	☐	☐ ☐	☐ ☐	☐ ☐	☐ ☐	☐ s ☐ s
0100 - 每 10 分钟	☐	☐ ☐	☐ ☐	☐ ☐	☐ m	☐ s ☐ s
0101 - 每小时	☐	☐ ☐	☐ ☐	☐ h ☐ h	☐ m ☐ m	☐ s ☐ s
0110 - 每天	☐	☐ ☐	☐ ☐	☐ h ☐ h	☐ m ☐ m	☐ s ☐ s
0111 - 每周	☐ d	☐ ☐	☐ ☐	☐ h ☐ h	☐ m ☐ m	☐ s ☐ s
1000 - 每月	☐	☐ ☐	☐ d ☐ d	☐ h ☐ h	☐ m ☐ m	☐ s ☐ s
1001 - 每年 ⁽¹⁾	☐	☐ m ☐ m	☐ d ☐ d	☐ h ☐ h	☐ m ☐ m	☐ s ☐ s

注 1: 每年，除非配置为 2 月 29 日。

19.5 功率控制

RTCC 包含功率控制功能，使器件可定期唤醒外部器件，在从该器件采样唤醒事件前等待器件稳定，然后关闭外部器件。这可完全由 RTCC 自动完成，而无需从当前低功耗模式（休眠模式和深度休眠模式等）唤醒。

要使能该功能，必须使能 RTCC（RTCEN = 1），必须将 PWCEN 寄存器位置 1 且 RTCC 引脚必须正在驱动 PWC 控制信号（RTCOE = 1 和 RTCOUT<1:0> = 11）。

可使用 PWCP 寄存器位选择 PWCPOL 控制信号的极性。可将低电平有效或高电平有效用于相应的外部开关来打开或关闭一个或多个外部器件的电源。低电平有效设置也可与 RTCC 引脚的漏极开路设置结合使用。该设置可以直接驱动外部器件的 GND 引脚（相应的外部 VDD 上拉器件），而无需外部开关。最后，应将 CHIME 位置 1 以定期使能 PWC。

PIC24FV32KA304 系列

注:

20.0 32 位可编程循环冗余校验 (CRC) 发生器

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解更多信息，请参见《PIC24F 系列参考手册》中的第 41 章“32 位可编程循环冗余校验 (CRC)” (DS39729A_CN)。

可编程 CRC 发生器为各种网络和安全应用提供快速产生校验和的硬件实现方法。它提供以下特性：

- 用户可编程的 CRC 多项式方程（最多 32 位）
- 可编程移位方向（小尾数或大尾数格式）
- 独立的数据和多项式长度
- 可配置中断输出
- 数据 FIFO

CRC 发生器的简化框图如图 20-1 所示。图 20-2 中给出了 CRC 移位引擎的简化形式。

图 20-1: CRC 框图

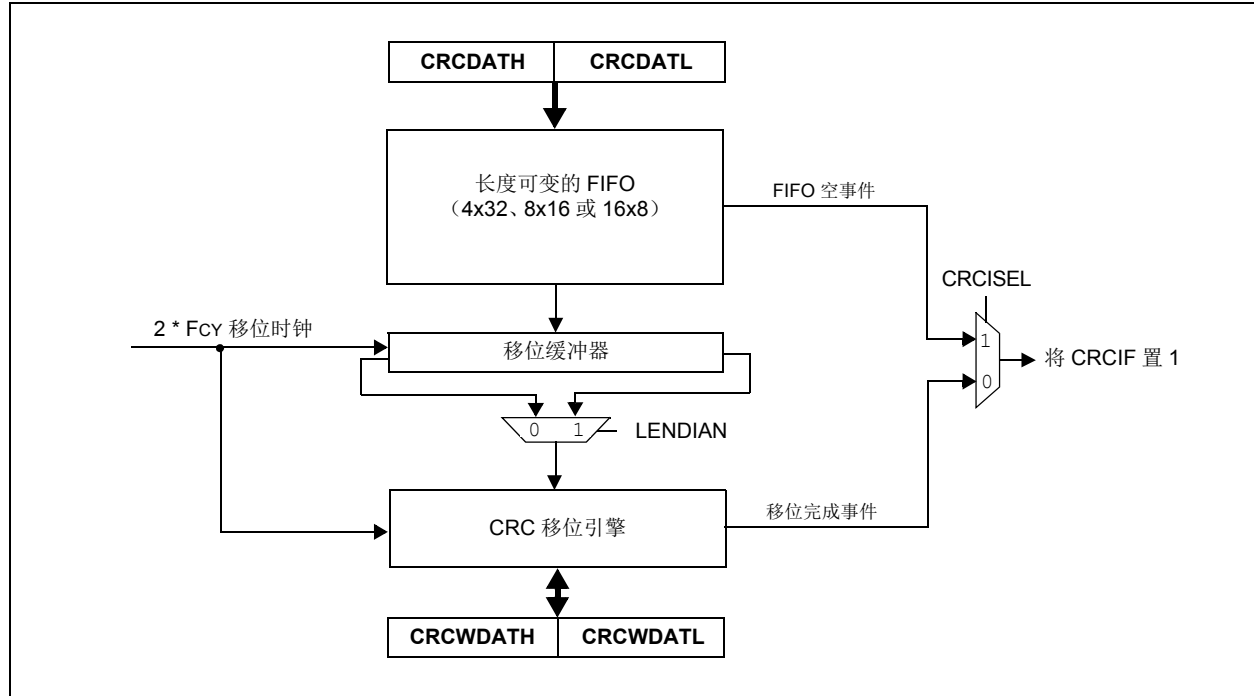
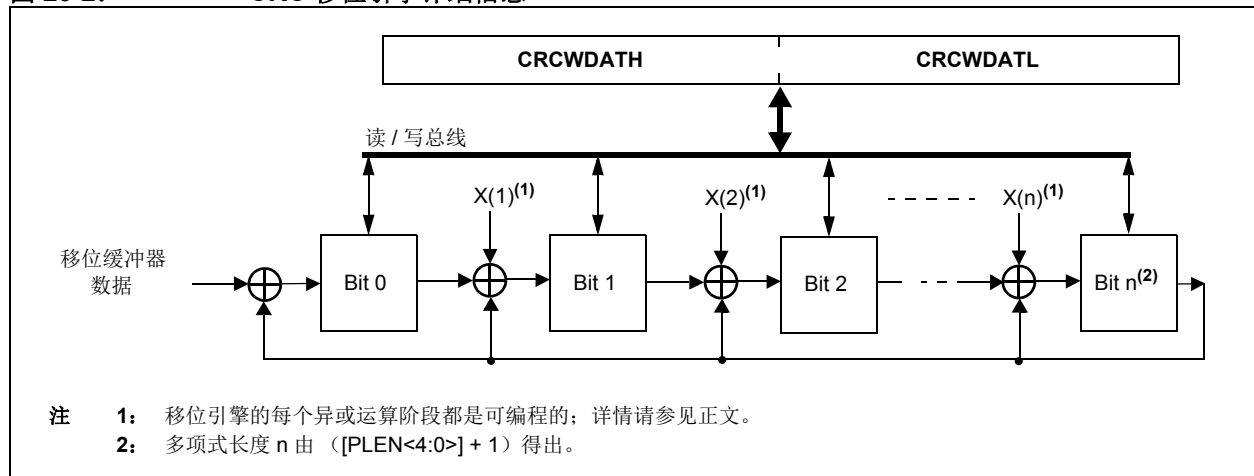


图 20-2: CRC 移位引擎详细信息



PIC24FV32KA304 系列

20.1 用户接口

20.1.1 多项式接口

可使用最多 32 位，将 CRC 模块编程为生成最多 32 阶的 CRC 多项式。通过 PLEN<4:0> (CRCCON2<4:0>) 选择反映方程中最高次幂的多项式长度。

CRCXORL 和 CRCXORH 寄存器控制方程中包含的幂项。将寄存器中的某个位置 1 会在方程中包含相应的幂项，在功能上相当于包含在 CRC 引擎的相应位上执行的异或操作。清零该位会禁止此异或操作。

例如，考虑两个 CRC 多项式，一个是 16 位方程，另一个是 32 位方程：

$$x^{16} + x^{12} + x^5 + 1$$

和

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

要将这些多项式编程到 CRC 发生器中，请设置表 20-1 中的寄存器位。

注意，请将适当的位置 1 以指示在方程中使用了它们（例如 X26 和 X23）。方程所需的 0 位始终为其他位的异或值，因此 X0 的取值对方程式没有影响。对于长度为 N 的多项式，假设无论 CRC 发生器中第 N 位设置如何都将始终使用该位。因此，对于长度为 32 的多项式，CRCXOR 寄存器中没有第 32 位。

20.1.2 数据接口

CRC 模块具有可缓存各种数据宽度的 FIFO。可使用 DWIDTHx<4:0> 位 (CRCCON2<12:8>) 将输入数据宽度配置为 1 到 32 位之间的任何数值。当数据宽度大于 15 时，FIFO 为 4 字深。当 DWIDTHx 值在 8 至 15 之间时，FIFO 为 8 字深。当 DWIDTHx 值小于 8 时，FIFO 为 16 字深。

首先将要进行 CRC 计算所需的数据写入 FIFO。即使数据宽度小于 8，可被写入 FIFO 的最小数据元素也是 1 个字节。例如，如果 DWIDTHx 值为 5，那么数据大小为 DWIDTHx + 1，即 6。数据作为整个字节写入，CRC 模块忽略了两个未使用的高位。

数据一旦被写入 CRCDAT 寄存器的最高位（由数据宽度定义），VWORD<4:0> 位 (CRCCON1<12:8>) 的值将递增 1。例如，如果 DWIDTHx 值为 24，则写入 CRCDATH 的 bit 7 时，VWORDx 位递增 1。因此，CRCDATL 必须始终先于 CRCDATH 写入。

当 CRCGO 位置 1 且 VWORDx 的值大于零时，CRC 引擎开始移入数据。每个字都被从 FIFO 复制到缓冲寄存器（这会导致 VWORDx 减 1）。然后从缓冲器中移出数据。CRC 引擎会以每指令周期两个位的速率继续移动数据，直到 VWORDx 值达到 0。因此对于给定的数据宽度，每个字完成计算所需的指令周期数为数据宽度的一半。例如，完成 32 位数据单字的 CRC 计算需要 16 个周期。

当 VWORDx 值达到所配置 DWIDTHx 值（4、8 或 16）的最大值时，CRCFUL 位置 1。当 VWORDx 值达到零时，CRCMPT 位置 1。当 CRCEN 为 0 时，FIFO 为空且 VWORD<4:0> 设置为 00000。

在写入 CRCDAT 之后，必须至少经过一个指令周期才可以读 VWORDx 位。

表 20-1: CRC 设置示例（16 和 32 位多项式）

CRC 控制位	位值	
	16 位多项式	32 位多项式
PLEN<4:0>	01111	11111
X<31:16>	0000 0000 0000 000x	0000 0100 1100 0001
X<15:0>	0001 0000 0010 000x	0001 1101 1011 011x

20.1.3 数据移位方向

LENDIAN 位 (CRCCON1<3>) 用于控制移位方向。默认情况下, CRC 通过引擎从最高有效位开始移入数据。将 LENDIAN 置 1 (= 1) 可导致 CRC 从最低有效位开始移入数据。此设置可使 CRC 与各种通信机制更好地集成, 并省去了用软件使位反序的开销。注意, 这只是改变了将数据移入引擎的方向。CRC 计算的结果仍然是正常的 CRC 结果, 不是反序的 CRC 结果。

20.1.4 中断操作

用户可以配置该模块在以下两种条件下产生中断。如果 CRCISEL 为 0, 当 VWORD<4:0> 位的值从 1 跳变到 0 时产生中断。如果 CRCISEL 为 1, 在 CRC 操作完成且模块将 CRCGO 位设置为 0 时产生中断。手动将 CRCGO 设置为 0 不会产生中断。

20.1.5 典型操作

要使用模块进行典型 CRC 计算, 请执行以下步骤:

1. 置 1 CRCEN 位以使能模块。
2. 配置模块以实现所需的操作:
 - a) 使用 CRCXORL 和 CRCXORH 寄存器以及 PLEN<4:0> 位对所需的多项式编程
 - b) 使用 DWIDTHxx 和 LENDIAN 位配置数据宽度和移位方向
 - c) 使用 CRCISEL 位选择所需的中断模式
3. 在 CRCFUL 位置 1 之前或 FIFO 中无数据时, 通过写入 CRCDATL 和 CRCDATH 寄存器来预装载 FIFO。
4. 通过写入 00h 到 CRCWDATL 和 CRCWDATH 来清除旧数据。CRCWDAT 寄存器也可以保持不变以继续先前暂停的计算。
5. 置 1 CRCGO 位以启动计算。
6. 在 FIFO 有可用空间时将剩余数据写入 FIFO 中。
7. 计算完成时, CRCGO 自动清零。如果 CRCISEL = 1, 则将产生中断。
8. 读取 CRCWDATL 和 CRCWDATH 以获得计算结果。

20.2 寄存器

有 8 个寄存器与该模块关联:

- CRCCON1
- CRCCON2
- CRCXORL
- CRCXORH
- CRCDATL
- CRCDATH
- CRCWDATL
- CRCWDATH

CRCCON1 和 CRCCON2 寄存器 (寄存器 20-1 和寄存器 20-2) 用于控制模块的操作和配置各种设置。CRCXOR 寄存器 (寄存器 20-3 和寄存器 20-4) 用于选择 CRC 方程中要使用的多项式项。CRCDAT 和 CRCWDAT 寄存器是两对分别用作双字输入数据缓冲器和 CRC 处理结果输出缓冲器的寄存器。

PIC24FV32KA304 系列

寄存器 20-1: **CRCCON1: CRC 控制寄存器 1**

R/W-0	U-0	R/W-0	R-0	R-0	R-0	R-0	R-0
CRCEN	—	CSIDL	VWORD4	VWORD3	VWORD2	VWORD1	VWORD0
bit 15							bit 8

R-0, HSC	R-1, HSC	R/W-0	R/W-0, HC	R/W-0	U-0	U-0	U-0
CRCFUL	CRCMPT	CRCISEL	CRCGO	LENDIAN	—	—	—
bit 7							bit 0

图注:	HC = 可由硬件清零的位	HSC = 可由硬件置 1/ 清零的位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零 x = 未知

bit 15	CRCEN: CRC 使能位 1 = 使能模块 0 = 禁止模块。 所有状态机、指针和 CRCWDAT/CRCDAT 寄存器都复位; 其他 SFR 不复位。
bit 14	未实现: 读为 0
bit 13	CSIDL: 空闲模式 CRC 停止位 1 = 当器件进入空闲模式时, 模块停止工作 0 = 模块在空闲模式下继续工作
bit 12-8	VWORD<4:0>: 指针值位 指出 FIFO 中有效字的个数。当 PLEN<4:0> > 7 时, 最大值为 8, 当 PLEN<4:0> ≤ 7 时, 最大值为 16。
bit 7	CRCFUL: CRC FIFO 满位 1 = FIFO 为满 0 = FIFO 未满
bit 6	CRCMPT: CRC FIFO 空位 1 = FIFO 为空 0 = FIFO 非空
bit 5	CRCISEL: CRC 中断选择位 1 = FIFO 为空时中断; CRC 计算未完成 0 = 移位完成时中断且 CRCWDAT 结果已计算好
bit 4	CRCGO: 启动 CRC 位 1 = 启动 CRC 串行移位器 0 = 关闭 CRC 串行移位器
bit 3	LENDIAN: 数据移位方向选择位 1 = 数据字从最低有效位开始移入 CRC (小尾数格式) 0 = 数据字从最高有效位开始移入 CRC (大尾数格式)
bit 2-0	未实现: 读为 0

PIC24FV32KA304 系列

寄存器 20-2: CRCCON2: CRC 控制寄存器 2

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DWIDTHx4	DWIDTHx3	DWIDTHx2	DWIDTHx1	DWIDTHx0
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
—	—	—	PLEN4	PLEN3	PLEN2	PLEN1	PLEN0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-13 未实现: 读为 0

bit 12-8 **DWIDTHx<4:0>**: 数据宽度选择位
定义数据字的宽度 (数据字宽度 = (DWIDTHx<4:0>) + 1)。

bit 7-5 未实现: 读为 0

bit 4-0 **PLEN<4:0>**: 多项式长度选择位
定义 CRC 多项式的长度 (多项式长度 = (PLEN<4:0>) + 1)。

PIC24FV32KA304 系列

寄存器 20-3: CRCXORL: CRC 多项式异或运算寄存器 (低字节)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
X15	X14	X13	X12	X11	X10	X9	X8
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	U-0
X7	X6	X5	X4	X3	X2	X1	—
bit 7							bit 0

图注:

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

bit 15-1 **X<15:1>**: 多项式的项 X^n 的异或操作使能位
bit 0 未实现: 读为 0

寄存器 20-4: CRCXORH: CRC 异或运算多项式寄存器 (高字节)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
X31	X30	X29	X28	X27	X26	X25	X24
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
X23	X22	X21	X20	X19	X18	X17	X16
bit 7							bit 0

图注:

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

bit 15-0 **X<31:16>**: 多项式的项 X^n 的异或运算使能位

21.0 高 / 低电压检测 (HLVD)

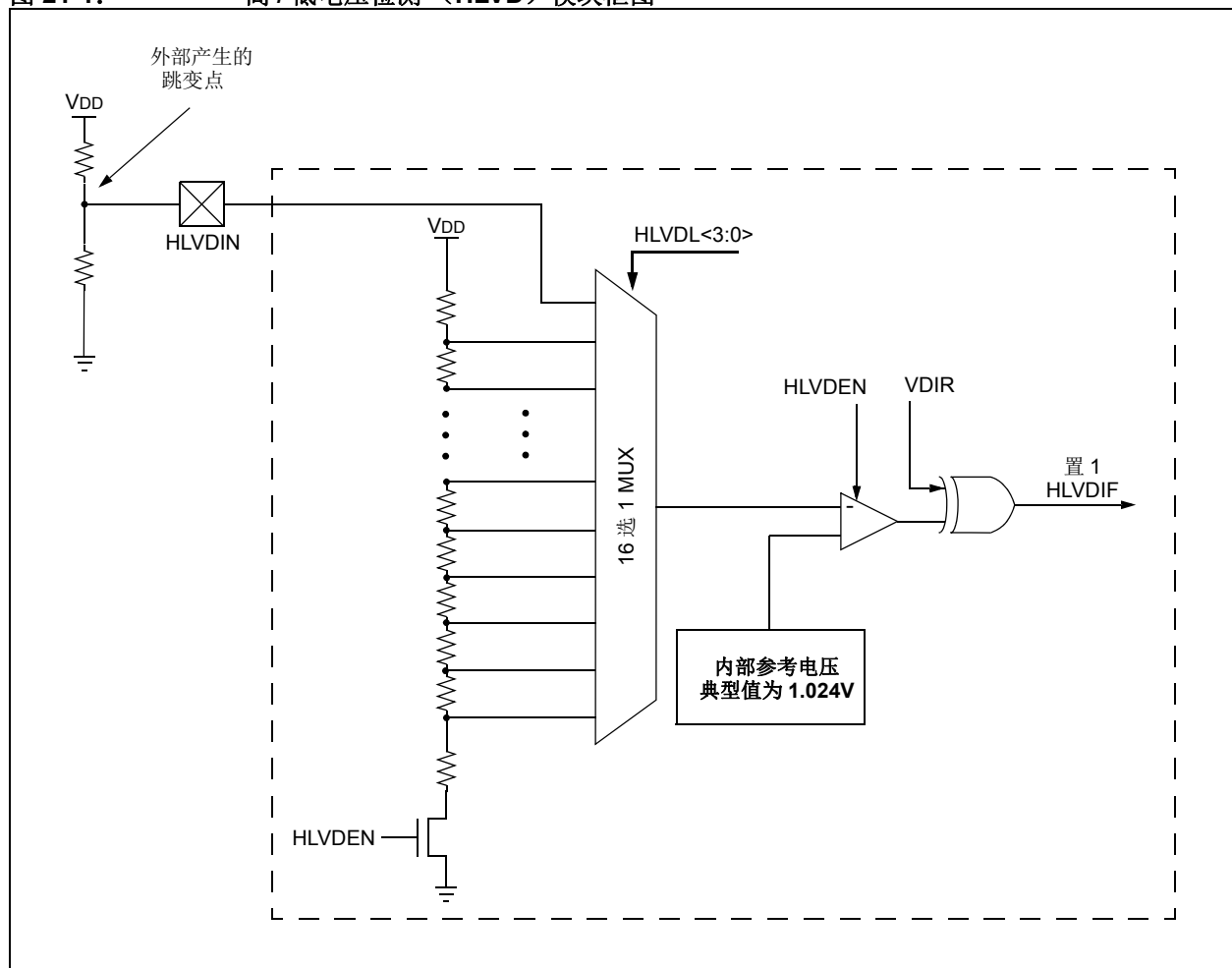
注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解高 / 低电压检测的更多信息，请参见《PIC24F 系列参考手册》中的第 36 章“高度集成的可编程高 / 低电压检测 (HLVD) 模块” (DS39725A_CN)。

高 / 低电压检测模块 (HLVD) 是一个可编程电路，允许用户指定器件电压跳变点和变化方向。

如果器件电压按照特定的变化方向相对于跳变点发生了偏离，就会将中断标志位置 1。如果允许中断，程序执行将跳转到中断向量地址处执行，随后软件可以对中断做出响应。

HLVD 控制寄存器（见寄存器 21-1）完全控制 HLVD 模块的操作。用户可通过软件控制将电路“关闭”，从而使器件的电流消耗降至最低。

图 21-1: 高 / 低电压检测 (HLVD) 模块框图



PIC24FV32KA304 系列

寄存器 21-1: HLVDCON: 高 / 低电压检测控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
HLVDEN	—	HLSIDL	—	—	—	—	—
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
VDIR	BGVST	IRVST	—	HLVDL3	HLVDL2	HLVDL1	HLVDL0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	HLVDEN: 高 / 低电压检测电源使能位 1 = 使能 HLVD 0 = 禁止 HLVD
bit 14	未实现: 读为 0
bit 13	HLSIDL: 空闲模式 HLVD 停止位 1 = 当器件进入空闲模式时, 模块停止工作 0 = 模块在空闲模式下继续工作
bit 12-8	未实现: 读为 0
bit 7	VDIR: 电压变化方向选择位 1 = 电压等于或大于跳变点 (HLVDL<3:0>) 时发生事件 0 = 电压等于或小于跳变点 (HLVDL<3:0>) 时发生事件
bit 6	BGVST: 带隙电压稳定标志位 1 = 指示带隙电压稳定 0 = 指示带隙电压不稳定
bit 5	IRVST: 内部参考电压稳定标志位 1 = 指示内部参考电压稳定, 高电压检测逻辑在指定的电压范围内生成中断标志 0 = 指示内部参考电压不稳定, 高电压检测逻辑在指定的电压范围内不生成中断标志, 且不应该允许 HLVD 中断
bit 4	未实现: 读为 0
bit 3-0	HLVDL<3:0>: 高 / 低电压检测限制位 1111 = 将使用外部模拟输入 (输入来自 HLVDIN 引脚) 1110 = 跳变点 1 ⁽¹⁾ 1101 = 跳变点 2 ⁽¹⁾ 1100 = 跳变点 3 ⁽¹⁾ . . . 0000 = 跳变点 15 ⁽¹⁾

注 1: 关于实际跳变点, 请参见第 29.0 节“电气特性”。

22.0 带阈值检测的 12 位 A/D 转换器

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解带阈值检测的 12 位 A/D 转换器的更多信息，请参见《PIC24F 系列参考手册》中的第 5 章“带阈值检测功能的 12 位 A/D 转换器 (DS39739A_CN)”。

12 位 A/D 转换器模块是一些 PIC24F 器件中提供的 10 位模块的增强版本。两个模块的核心都是逐次逼近寄存器 (SAR) 转换器，此外还有一些可灵活配置的硬件功能。本版本的模块通过提供 12 位分辨率、较多的自动采样选项和更紧密集成其他模拟模块（如 CTMU 和一个可配置结果缓冲区）来扩展功能。该模块还包含一个独特的阈值检测功能，使模块本身可根据转换结果作出简单的决定。

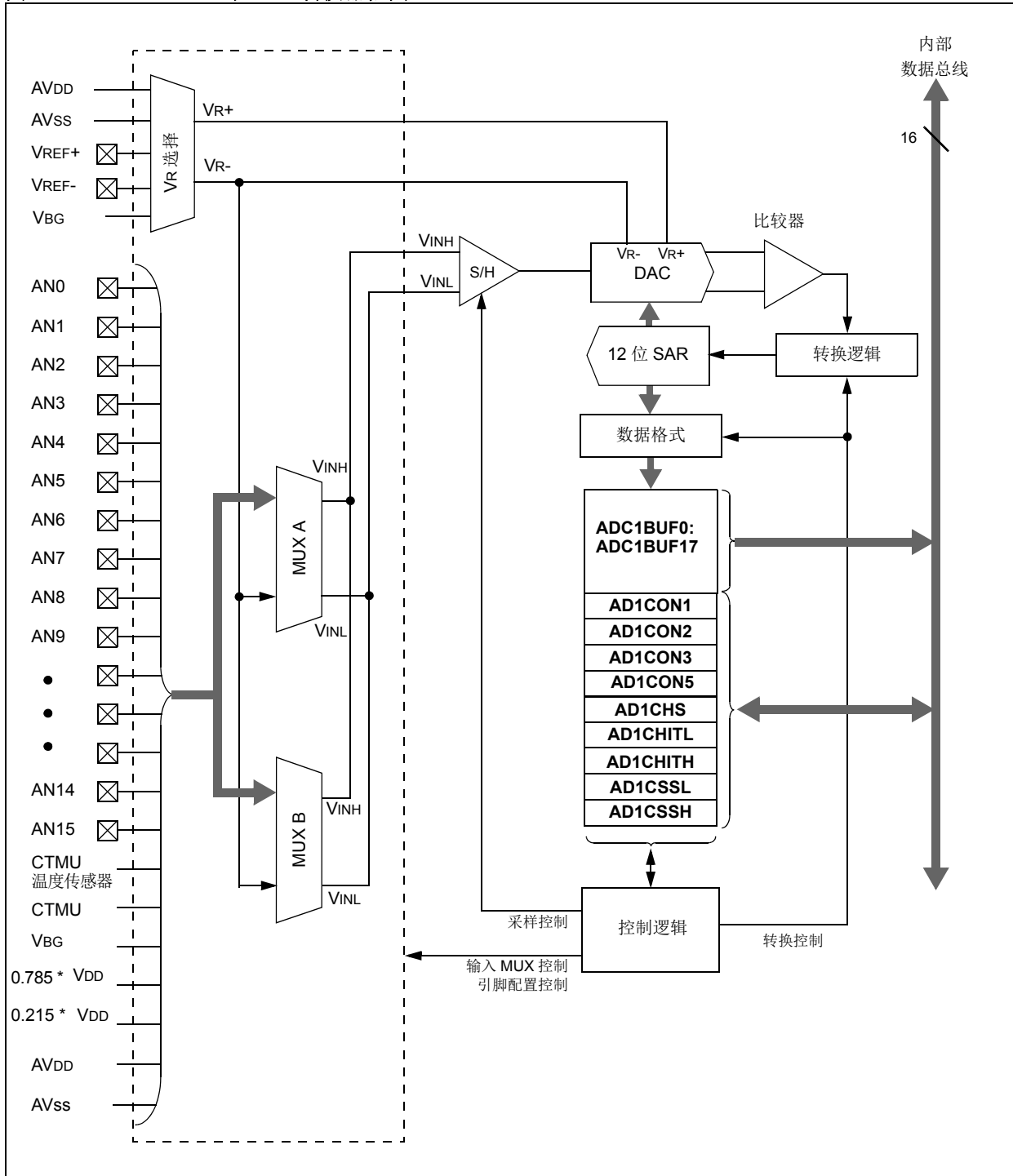
模块的简化框图如图 22-1 所示。

PIC24F 12 位 A/D 转换器具有以下主要特性：

- 逐次逼近寄存器 (Successive Approximation Register, SAR) 转换
- 转换速度最高可达 100 ksps
- 最多 32 路模拟输入通道（内部和外部）
- 多个内部参考输入通道
- 外部参考电压输入引脚
- 单极型差分采样 / 保持 (Sample-and-Hold, S/H) 放大器
- 自动阈值扫描和比较操作，可预先估计转换结果
- 可选转换触发源
- 固定长度（每路通道一个字）的可配置转换结果缓冲区
- 4 种结果对齐选项
- 可配置中断产生
- 可在 CPU 休眠和空闲模式下继续工作

PIC24FV32KA304 系列

图 22-1: 12 位 A/D 转换器框图



要执行一个 A/D 转换：

1. 配置 A/D 模块：
 - a) 将端口引脚配置为模拟输入和 / 或选择带隙参考输入 (ANS<12:10> 和 ANS<5:0>)。
 - b) 选择参考电压源以匹配模拟输入的预期范围 (AD1CON2<15:13>)。
 - c) 选择模拟转换时钟以便使期望的数据速率与处理器时钟匹配 (AD1CON3<7:0>)。
 - d) 选择适当的采样/转换序列 (AD1CON1<7:4> 和 AD1CON3<12:8>)。
 - e) 选择转换结果在缓冲区中的存储方式 (AD1CON1<9:8>)。
 - f) 选择中断发生的频率 (AD1CON2<6:2>)。
 - g) 启动 A/D 模块 (AD1CON1<15>)。
2. 配置 A/D 中断 (如果需要)：
 - a) 清零 AD1IF 位。
 - b) 选择 A/D 中断优先级。

要使用阈值检测扫描执行 A/D 采样和转换：

1. 配置 A/D 模块：
 - a) 将端口引脚配置为模拟输入 (ANS<12:10> 和 ANS<5:0>)。
 - b) 选择参考电压源以匹配模拟输入的预期范围 (AD1CON2<15:13>)。
 - c) 选择模拟转换时钟以便使期望的数据速率与处理器时钟匹配 (AD1CON3<7:0>)。
 - d) 选择适当的采样 / 转换序列 (AD1CON1<7:4> 和 AD1CON3<12:8>)。
 - e) 选择转换结果在缓冲区中的存储方式 (AD1CON1<9:8>)。
 - f) 选择中断发生的频率 (AD1CON2<6:2>)。
2. 配置阈值比较通道：
 - a) 使能自动扫描 — ASEN 位 (AD1CON5<15>)。
 - b) 选择比较模式“大于、小于或窗口式” — CMx 位 (AD1CON5<1:0>)。
 - c) 选择要扫描的阈值比较通道 (ADCSSH 和 ADCSSL)。
 - d) 如果需要 CTMU 作为阈值比较通道的电流源，使能相应的 CTMU 通道 (ADCCTMUENH 和 ADCCTMUENL)。
 - e) 将阈值写入相应的 ADC1BUF_n 寄存器。
 - f) 启动 A/D 模块 (AD1CON1<15>)。

注：	如果在休眠模式下使用阈值检测执行 A/D 采样和转换，在进入休眠模式前必须选择 RC A/D 时钟源。
-----------	---

3. 配置 A/D 中断 (可选)：
 - a) 清零 AD1IF 位。
 - b) 选择 A/D 中断优先级。

PIC24FV32KA304 系列

22.1 A/D 控制寄存器

12 位 A/D 转换器模块使用多达 43 个寄存器来执行操作。所有寄存器均映射到数据存储空间。

22.1.1 控制寄存器

根据具体的器件，模块最多具有 11 个控制和状态寄存器：

- AD1CON1: A/D 控制寄存器 1
- AD1CON2: A/D 控制寄存器 2
- AD1CON3: A/D 控制寄存器 3
- AD1CON5: A/D 控制寄存器 5
- AD1CHS: A/D 采样选择寄存器
- AD1CHITH 和 AD1CHITL: A/D 扫描比较命中寄存器
- AD1CSSL 和 AD1CSSH: A/D 输入扫描选择寄存器
- AD1CTMUENH 和 AD1CTMUENL: CTMU 使能寄存器

AD1CON1、AD1CON2 和 AD1CON3 寄存器（[寄存器 22-1](#)、[寄存器 22-2](#) 和 [寄存器 22-3](#)）控制 A/D 模块的整个操作。这包括使能模块、配置转换时钟和参考电压源、选择采样和转换触发器和手动控制采样/转换序列。AD1CON5 寄存器（[寄存器 22-4](#)）专门控制阈值检测操作的功能，包括其在节能模式下的操作。

AD1CHS 寄存器（[寄存器 22-5](#)）选择输入通道以连接至 S/H 放大器。它还允许选择输入多路开关和参考电压源以进行差分采样。

AD1CHITH 和 AD1CHITL 寄存器（[寄存器 22-6](#) 和 [寄存器 22-7](#)）为信号量寄存器，用于阈值检测操作。在某些

情况下，各个位或位对的状态指示匹配条件是否发生。AD1CHITL 始终实现，而 AD1CHITH 可能在具有 16 路或更少通道的器件中未实现。

AD1CSSH/L 寄存器（[寄存器 22-8](#) 和 [寄存器 22-9](#)）用于选择顺序扫描的通道。

AD1CTMUENH/L 寄存器（[寄存器 22-10](#) 和 [寄存器 22-11](#)）用于选择转换期间 CTMU 使用的通道。选择一个特定的通道使 A/D 转换器通过该通道控制 CTMU（特别是它的电流源）并读取它的数据。AD1CTMUENL 始终实现，而 AD1CTMUENH 可能在具有 16 路或更少通道的器件中未实现。

22.1.2 A/D 结果缓冲区

该模块集成了一个多字双端口 RAM，称为 ADC1BUF。对于特定的器件，缓冲区中应至少有与外部模拟通道相同数量的字地址单元，最多为 32 个。缓冲区地址数始终为偶数。每个地址单元都映射到数据存储空间并可单独寻址。缓冲区地址单元记为 ADC1BUF0 至 ADC1BUF_n（最高 31）。

A/D 结果缓冲区是可读写的。当模块活动（AD1CON<15> = 1）时，缓冲区为只读且存储 A/D 转换的结果。当模块未活动（AD1CON<15> = 0）时，缓冲区是可读写的。在该状态下，写入缓冲区地址单元即可设定阈值检测操作的转换阈值。

当通过 ADON 位（AD1CON1<15>）禁止该模块时，缓冲区的内容不会被清零。转换结果和任何设定的阈值在 ADON 置 1 或清零时保持不变。

PIC24FV32KA304 系列

寄存器 22-1: AD1CON1: A/D 控制寄存器 1

R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
ADON	—	ADSIDL	—	—	MODE12	FORM1	FORM0
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0,HSC	R/C-0,HSC
SSRC3	SSRC2	SSRC1	SSRC0	—	ASAM	SAMP	DONE
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	HSC = 可由硬件置 1/ 清零的位
-n = 上电复位时的值	1 = 置 1	0 = 清零 x = 未知

bit 15	ADON: A/D 工作模式位 1 = A/D 转换器模块正在工作 0 = A/D 转换器关闭
bit 14	未实现: 读为 0
bit 13	ADSIDL: A/D 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 = 模块在空闲模式下继续工作
bit 12-11	未实现: 读为 0
bit 10	MODE12: 12 位工作模式位 1 = 作为 12 位 A/D 工作 0 = 作为 10 位 A/D 工作
bit 9-8	FORM<1:0>: 数据输出格式位 (见以下格式) 11 = 有符号小数结果, 左对齐 10 = 无符号绝对小数结果, 左对齐 01 = 有符号十进制结果, 右对齐 00 = 无符号绝对十进制结果, 右对齐
bit 7-4	SSRC<3:0>: 采样时钟源选择位 1111 = 不可用; 不要使用 . . . 1000 = 不可用; 不要使用 0111 = 内部计数器结束采样并启动转换 (自动转换) 0110 = 不可用; 不要使用 0101 = Timer1 事件结束采样并启动转换 0100 = CTMU 事件结束采样并启动转换 0011 = Timer5 事件结束采样并启动转换 0010 = Timer3 事件结束采样并启动转换 0001 = INTO 事件结束采样并启动转换 0000 = 由软件清零 SAMP 位结束采样并开始转换
bit 3	未实现: 读为 0
bit 2	ASAM: A/D 采样自动启动位 1 = 采样在上次转换完成后立即开始; SAMP 位自动置 1 0 = SAMP 位手动置 1 时开始采样
bit 1	SAMP: A/D 采样使能位 1 = A/D 采样 / 保持放大器正在采样输入 0 = A/D 采样 / 保持放大器正在保持采样结果
bit 0	DONE: A/D 转换状态位 1 = A/D 转换周期完成 0 = A/D 转换周期未启动或正在进行中

PIC24FV32KA304 系列

寄存器 22-2: AD1CON2: A/D 控制寄存器 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
PVCFG1	PVCFG0	NVCFG0	OFFCAL	BUFREGEN	CSCNA	—	—
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
BUFS ⁽¹⁾	SMPI4	SMPI3	SMPI2	SMPI1	SMPI0	BUFM ⁽¹⁾	ALTS
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-14 **PVCFG<1:0>**: 转换器正参考电压配置位

11 = 4 * 内部 V_{BG}⁽²⁾

10 = 2 * 内部 V_{BG}⁽³⁾

01 = 外部 V_{REF+}

00 = AV_{DD}

bit 13 **NVCFG0**: 转换器负参考电压配置位

1 = 外部 V_{REF-}

0 = AV_{SS}

bit 12 **OFFCAL**: 失调校准模式选择位

1 = 采样 / 保持通道的反相和同相输入连接到 AV_{SS}

0 = 采样 / 保持通道的反相和同相输入连接到正常输入

bit 11 **BUFREGEN**: A/D 缓冲寄存器使能位

1 = 转换结果装入到转换通道确定的缓冲地址单元

0 = 将 A/D 结果缓冲区视为 FIFO

bit 10 **CSCNA**: 多路开关 A 的 CH0+ S/H 输入的扫描输入选择设置位

1 = 扫描输入

0 = 不扫描输入

bit 9-8 **未实现**: 读为 0

bit 7 **BUFS**: 缓冲区填充状态位 ⁽¹⁾

1 = A/D 正在填充缓冲区的上半部分; 用户应该访问下半部分中的数据

0 = A/D 正在填充缓冲区的下半部分; 用户应该访问上半部分中的数据

bit 6-2 **SMPI<4:0>**: 中断采样速率选择位

11111 = 每完成 32 次采样 / 转换产生一次中断

11110 = 每完成 31 次采样 / 转换产生一次中断

⋮

⋮

00001 = 每完成 2 次采样 / 转换产生一次中断

00000 = 每完成一次采样 / 转换产生一次中断

bit 1 **BUFM**: 缓冲区填充模式选择位 ⁽¹⁾

1 = 在第一次产生中断时从 AD1BUF0 开始填充缓冲区, 产生下一次中断时从 AD1BUF(n/2) 开始填充 (分割缓冲区模式)

0 = 从地址 ADCBUF0 开始填充缓冲区, 且在产生连续中断时按顺序填充每个地址 (FIFO 模式)

注 1: 仅在缓冲区用于 FIFO 模式 (BUFREGEN = 0) 时适用。另外, 仅当 BUFM = 1 时使用 BUFS。

2: 当 V_{DD} 低于 4.5V 时, 参考电压设置值可能不在规定范围内。

3: 当 V_{DD} 低于 2.3V 时, 参考电压设置值可能不在规定范围内。

寄存器 22-2: AD1CON2: A/D 控制寄存器 2 (续)

bit 0 **ALTS:** 交替输入采样模式选择位
 1 = 第一次采样时使用采样 A 选择的通道输入，下一次采样使用采样 B 选择的通道输入
 0 = 始终使用采样 A 选择的通道输入

- 注 1: 仅在缓冲器用于 FIFO 模式 (BUFREGEN = 0) 时适用。另外，仅当 BUFM = 1 时使用 BUFS。
 2: 当 VDD 低于 4.5V 时，参考电压设置值可能不在规定范围内。
 3: 当 VDD 低于 2.3V 时，参考电压设置值可能不在规定范围内。

寄存器 22-3: AD1CON3: A/D 控制寄存器 3

R/W-0	R-0	r-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	EXTSAM	r	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0
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
ADCS7	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
bit 7							bit 0

图注:	r = 保留位		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 **ADRC:** A/D 转换时钟源位
 1 = RC 时钟
 0 = 时钟由系统时钟产生

bit 14 **EXTSAM:** 扩展采样时间位
 1 = 在 SAMP = 0 后 A/D 仍正在采样
 0 = A/D 完成采样

bit 13 **保留:** 保持为 0

bit 12-8 **SAMC<4:0>:** 自动采样时间选择位
 11111 = 31 TAD
 .
 .
 .
 00001 = 1 TAD
 00000 = 0 TAD

bit 7-0 **ADCS<7:0>:** A/D 转换时钟选择位
 11111111-01000000 = 保留
 00111111 = 64 · TCY = TAD
 .
 .
 .
 00000001 = 2 · TCY = TAD
 00000000 = TCY = TAD

PIC24FV32KA304 系列

寄存器 22-4: AD1CON5: A/D 控制寄存器 5

R/W-0	R/W-0	R/W-0	R/W-0	r-0	U-0	R/W-0	R/W-0
ASEN ⁽¹⁾	LPEN	CTMREQ	BGREQ	r	—	ASINT1	ASINT0
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	WM1	WM0	CM1	CM0
bit 7							bit 0

图注:	r = 保留位
R = 可读位	W = 可写位
-n = 上电复位时的值	1 = 置 1
	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

- bit 15 **ASEN:** 自动扫描使能位 ⁽¹⁾
1 = 使能自动扫描
0 = 禁止自动扫描
- bit 14 **LPEN:** 低功耗使能位
1 = 扫描后返回低功耗模式
0 = 扫描后保持在全功耗模式
- bit 13 **CTMREQ:** CTMU 请求位
1 = 当 A/D 使能且活动时使能 CTMU
0 = A/D 未使能 CTMU
- bit 12 **BGREQ:** 带隙请求位
1 = 当 A/D 使能且活动时使能带隙
0 = A/D 未使能带隙
- bit 11 **保留:** 保持为 0
- bit 10 **未实现:** 读为 0
- bit 9-8 **ASINT<1:0>:** 自动扫描 (阈值检测) 中断模式位
11 = 阈值检测序列完成且发生有效比较后产生中断
10 = 发生有效比较后产生中断
01 = 阈值检测序列完成后产生中断
00 = 无中断
- bit 7-4 **未实现:** 读为 0
- bit 3-2 **WM<1:0>:** 写模式位
11 = 保留
10 = 仅自动比较 (未保存转换结果, 但当发生有效匹配 (由 CMx 和 ASINTx 位定义) 时产生中断)
01 = 转换并保存 (发生匹配 (由 CMx 位定义) 时转换结果保存到寄存器位确定的地址单元)
00 = 传统操作 (转换数据保存到缓冲寄存器确定的地址单元)
- bit 1-0 **CM<1:0>:** 比较模式位
11 = 窗口外部模式 (如果转换结果在对应缓冲区间定义的窗口外, 将发生有效匹配)
10 = 窗口内部模式 (如果转换结果在对应缓冲区间定义的窗口内, 将发生有效匹配)
01 = 大于模式 (如果结果大于对应缓冲寄存器中的值, 将发生有效匹配)
00 = 小于模式 (如果结果小于对应缓冲寄存器中的值, 将发生有效匹配)

注 1: 当使用带阈值检测的自动扫描 (ASEN = 1) 时, 不要将采样时钟源配置为自动转换模式 (SSRCx = 7)。任何其他可用的 SSRCx 选择均有效。要使用采样时钟源的自动转换模式 (SSRCx = 7), 应确保 ASEN 清零。

PIC24FV32KA304 系列

寄存器 22-5: AD1CHS: A/D 采样选择寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB2	CH0NB1	CH0NB0	CH0SB4	CH0SB3	CH0SB2	CH0SB1	CH0SB0
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
CH0NA2	CH0NA1	CH0NA0	CH0SA4	CH0SA3	CH0SA2	CH0SA1	CH0SA0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-13 **CH0NB<2:0>:** 采样 B 通道 0 负输入选择位

111 = AN6⁽¹⁾

110 = AN5⁽²⁾

101 = AN4

100 = AN3

011 = AN2

010 = AN1

001 = AN0

000 = AVss

bit 12-8 **CH0SB<4:0>:** MUX B 多路开关设置的采样 / 保持放大器正输入选择位

11111 = 未实现, 不要使用

11101 = AVDD

11101 = AVss

11100 = 保护电压上限 (0.785 * VDD)

11011 = 保护电压下限 (0.215 * VDD)

11010 = 内部带隙参考电压 (VBG) ⁽³⁾

11001-10010 = 未实现, 不要使用

10001 = 未连接任何通道, 所有输入均处于悬空状态 (用于 CTMU)

10000 = 未连接任何通道, 所有输入均处于悬空状态 (用于 CTMU 温度传感器输入)

01111 = AN15

01110 = AN14

01101 = AN13

01100 = AN12

01011 = AN11

01010 = AN10

01001 = AN9

01000 = AN8⁽¹⁾

00111 = AN7⁽¹⁾

00110 = AN6⁽¹⁾

00101 = AN5⁽²⁾

00100 = AN4

00011 = AN3

00010 = AN2

00001 = AN1

00000 = AN0

bit 7-5 **CH0NA<2:0>:** 采样 A 通道 0 负输入选择位
与 CH0NB<2:0> 的定义相同。

bit 4-0 **CH0SA<4:0>:** 采样 A 通道 0 正输入选择位
与 CH0NA<4:0> 的定义相同。

注 1: 仅在 44 引脚器件上实现。

2: 仅在 28 和 44 引脚器件上实现。

3: 该输入使用的带隙值是内部 VBG 的 2 倍或 4 倍, 可在 PVCFG<1:0> = 1x 时进行选择。

PIC24FV32KA304 系列

寄存器 22-6: AD1CHITH: A/D 扫描比较命中寄存器 (高字) ⁽¹⁾

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	R/W-0	R/W-0
—	—	—	—	—	—	CHH17	CHH16
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-2

未实现: 读为 0。

bit 1-0

CHH<17:16>: A/D 比较命中位

如果 CM<1:0> = 11:

1 = A/D 结果缓冲区 x 已写入数据或发生匹配

0 = A/D 结果缓冲区 x 尚未写入数据

对于 CM<1:0> 的所有其他值:

1 = A/D 结果通道 x 上发生匹配

0 = A/D 结果通道 x 上未发生任何匹配

注 1: 未实现的通道读为 0。

寄存器 22-7: AD1CHITL: A/D 扫描比较命中寄存器 (低字) ⁽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
CHH15	CHH14	CHH13	CHH12	CHH11	CHH10	CHH9	CHH8
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
CHH7	CHH6	CHH5	CHH4	CHH3	CHH2	CHH1	CHH0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-0

CHH<15:0>: A/D 比较命中位

如果 CM<1:0> = 11:

1 = A/D 结果缓冲区 x 已写入数据或发生匹配

0 = A/D 结果缓冲区 x 已写入数据

对于 CM<1:0> 的所有其他值:

1 = A/D 结果通道 n 上发生匹配

0 = A/D 结果通道 n 上未发生任何匹配

注 1: 未实现的通道读为 0。

PIC24FV32KA304 系列

寄存器 22-8: **AD1CSSH: A/D 输入扫描选择寄存器 (高字)** ⁽¹⁾

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
—	CSS30	CSS29	CSS28	CSS27	CSS26	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	CSS17	CSS16
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **未实现:** 读为 0

bit 14-10 **CSS<30:26>: A/D 输入扫描选择位**

1 = 对相应通道进行输入扫描

0 = 输入扫描时跳过相应通道

bit 9-2 **未实现:** 读为 0

bit 1-0 **CSS<17:16>: A/D 输入扫描选择位**

1 = 对相应通道进行输入扫描

0 = 输入扫描时跳过相应通道

注 1: 未实现的通道读为 0。不要选择未实现的通道进行采样, 因为可能会产生不确定结果。

寄存器 22-9: **AD1CSSL: A/D 输入扫描选择寄存器 (低字)** ⁽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
CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8
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
CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-0 **CSS<15:0>: A/D 输入扫描选择位**

1 = 对相应 ANx 输入进行扫描

0 = 输入扫描时跳过相应通道

注 1: 未实现的通道读为 0。不要选择未实现的通道进行采样, 因为可能会产生不确定结果。

PIC24FV32KA304 系列

寄存器 22-10: AD1CTMUENH: A/D CTMU 使能寄存器 (高字) ⁽¹⁾

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	R/W-0	R/W-0
—	—	—	—	—	—	CTMEN17	CTMEN16
bit 7						bit 0	

图注:

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

bit 15-2 未实现: 读为 0。

bit 1-0 **CTMEN<17:16>**: 转换期间使能 CTMU 位

 1 = 转换期间使能 CTMU 并连接到选择的通道

 0 = CTMU 未连接到该通道

注 1: 未实现的通道读为 0。

寄存器 22-11: AD1CTMUENL: A/D CTMU 使能寄存器 (低字) ⁽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
CTMEN15	CTMEN14	CTMEN13	CTMEN12	CTMUEN11	CTMEN10	CTMEN9	CTMEN8
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
CTMEN7	CTMEN6	CTMEN5	CTMEN4	CTMEN3	CTMEN2	CTMEN1	CTMEN0
bit 7						bit 0	

图注:

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

bit 15-0 **CTMEN<15:0>**: 转换期间使能 CTMU 位

 1 = 转换期间使能 CTMU 并连接到选择的通道

 0 = CTMU 未连接到该通道

注 1: 未实现的通道读为 0。

22.2 A/D 采样要求

图 22-2 给出了 12 位 A/D 转换器的模拟输入模型。A/D 的总采样时间是保持电容充电时间的函数。

为了使 A/D 转换器达到规定精度，必须让充电保持电容（CHOLD）充满至模拟输入引脚的电平。信号源阻抗（Rs）、走线阻抗（Ric）和内部采样开关阻抗（Rss）共同直接影响对 CHOLD 充电所需的时间。因此，模拟信号源的总阻抗必须足够小才能在所选的采样时间内对保持电容充分充电。为了将引脚泄漏电流对 A/D 转换器精度的影响降到最低，建议的最大信号源阻抗 Rs 为 2.5 kΩ。选择（改变）模拟输入通道后，必须先完成该

采样功能才能开始转换。在每次采样操作之前，内部保持电容将处于放电状态。

两次转换之间应该至少留出 1 个 TAD 时间周期作为采样时间。更多信息，请参见第 29.0 节“电气特性”。

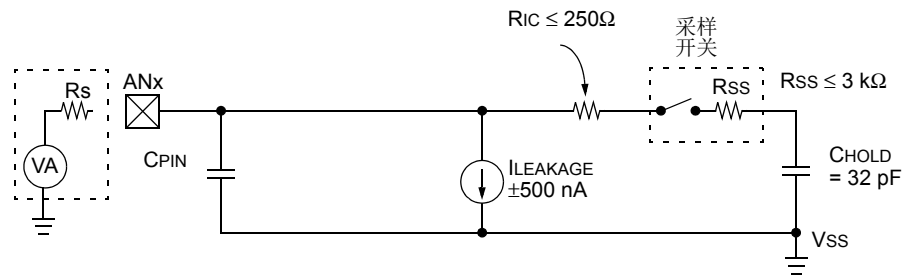
公式 22-1: A/D 转换时钟周期

$$T_{AD} = T_{CY}(ADCS + 1)$$

$$ADCS = \frac{T_{AD}}{T_{CY}} - 1$$

注：基于 $T_{CY} = 2/F_{OSC}$ ；禁止打盹模式和 PLL。

图 22-2: 12 位 A/D 转换器模拟输入模型



图注	CPIN	= 输入电容
	VT	= 阈值电压
	ILEAKAGE	= 各个连接点在引脚上产生的泄漏电流
	RIC	= 内部走线等效阻抗
	RSS	= 采样开关阻抗
	CHOLD	= 采样 / 保持电容 (来自 DAC)

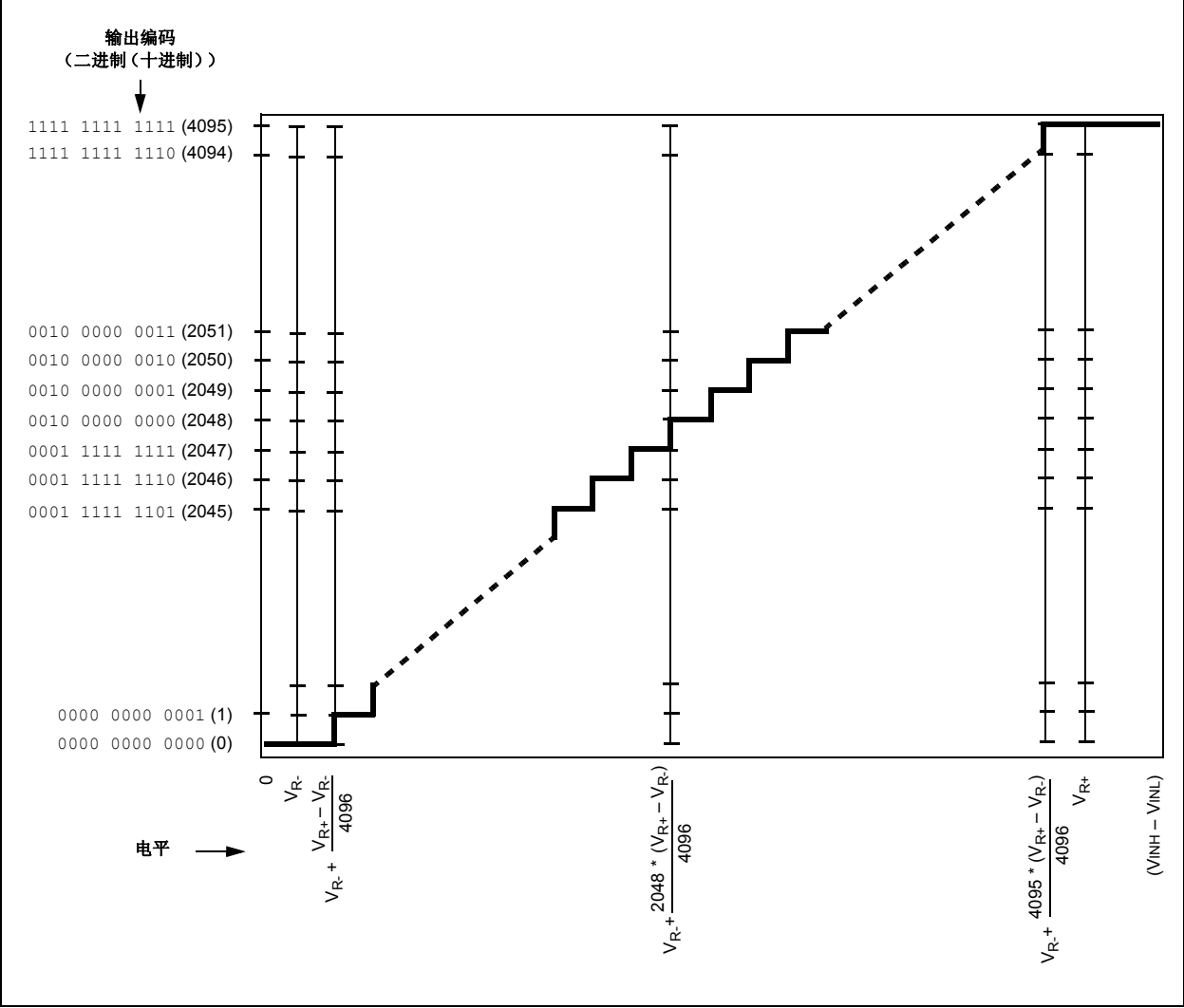
注：CPIN 值取决于器件封装，未经测试。如果 $R_s \leq 5 \text{ k}\Omega$ ，忽略 CPIN 的影响。

22.3 传递函数

图 22-3 给出了 12 位分辨率的 A/D 转换器的传递函数。将输入电压差值（VINH – VINL）与参考电压差值（VR+ – VR-）作比较。

- 当输入电压是 $((VR+)-(VR-))/4096$ 或 1.0 LSB 时，发生第一个编码转换。
- 0000 0000 0001 编码位于 $VR- + (1.5 * ((VR+)-(VR-))/4096)$ 中央。
- 0010 0000 0000 编码位于 $VREFL + (2048.5 * ((VR+)-(VR-))/4096)$ 。
- 低于 $VR- + ((VR-)-(VR-))/4096$ 的输入电压转换为 0000 0000 0000。
- 高于 $VR- + (4095 * ((VR+)-(VR-))/4096)$ 的输入电压转换为 1111 1111 1111。

图 22-3: 12 位 A/D 传递函数



22.4 缓冲区数据格式

当 $MODE12 = 1$ （通过 $AD1CON1<10>$ 设置）时，A/D 转换为全差分 12 位值，当 $MODE12 = 0$ 时，为 10 位值。当使用绝对小数或绝对整数格式时，结果分别为 12 位或 10 位宽。当采用有符号的十进制格式时，转换还包含一个符号位，使得 12 位转换变为 13 位宽，10 位转换

变为 11 位宽。有符号十进制格式分别产生 12 位和 10 位值。符号位（bit12 或 bit10）是填充置缓冲区的扩展位。FORM<1:0> 位（AD1CON1<9:8>）选择格式。图 22-4 和图 22-5 给出了可以选择的数据输出格式。表 22-1 至表 22-4 给出了各种转换结果编码所代表的等效数值。

图 22-4: A/D 输出数据格式（12 位）

RAM 内容	<table><tr><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>															d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																
读到总线																											
整数	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	<table><tr><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00									
0	0	0	0																								
d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																
有符号整数	<table><tr><td>s0</td><td>s0</td><td>s0</td><td>s0</td></tr></table>	s0	s0	s0	s0	<table><tr><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00									
s0	s0	s0	s0																								
d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																
小数（1.15）	<table><tr><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0									
d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																
0	0	0	0																								
有符号小数（1.15）	<table><tr><td>s0</td><td>d11</td><td>d10</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>	s0	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	<table><tr><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0									
s0	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00															
0	0	0																									

表 22-1: 各种结果编码的等效数值：12 位整数格式

V _{IN} /V _{REF}	12 位差分输出编码 (13 位结果)	16 位整数格式 / 等效 10 进制数值	16 位有符号整数格式 / 等效 10 进制数值
+4095/4096	0 1111 1111 1111	0000 1111 1111 1111	+4095
+4094/4096	0 1111 1111 1110	0000 1111 1111 1110	+4094
...			
+1/4096	0 1000 0000 0001	0000 0000 0000 0001	+1
0/4096	0 0000 0000 0000	0000 0000 0000 0000	0
-1/4096	1 0111 1111 1111	0000 0000 0000 0000	0
...			
-4095/4096	1 0000 0000 0001	0000 0000 0000 0000	0
-4096/4096	1 0000 0000 0000	0000 0000 0000 0000	0

PIC24FV32KA304 系列

表 22-2: 各种结果编码的等效数值：12 位小数格式

V _{IN} /V _{REF}	12 位差分输出编码 (13 位结果)	16 位小数格式 / 等效 10 进制数值		16 位有符号小数格式 / 等效 10 进制数值	
+4095/4096	0 1111 1111 1111	1111 1111 1111 0000	0.999	0111 1111 1111 1000	0.999
+4094/4096	0 1111 1111 1110	1111 1111 1110 0000	0.998	0111 1111 1110 1000	0.998
...					
+1/4096	0 0000 0000 0001	0000 0000 0001 0000	0.001	0000 0000 0000 1000	0.001
0/4096	0 0000 0000 0000	0000 0000 0000 0000	0.000	0000 0000 0000 0000	0.000
-1/4096	1 0111 1111 1111	0000 0000 0000 0000	0.000	1111 1111 1111 1000	-0.001
...					
-4095/4096	1 0000 0000 0001	0000 0000 0000 0000	0.000	1000 0000 0000 1000	-0.999
-4096/4096	1 0000 0000 0000	0000 0000 0000 0000	0.000	1000 0000 0000 0000	-1.000

图 22-5: A/D 输出数据格式 (10 位)

RAM 内容:						<table><tr><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>										d09	d08	d07	d06	d05	d04	d03	d02	d01	d00						
d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																						
读到总线:																															
整数	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>															0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
0	0	0	0	0	0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																
有符号整数	<table><tr><td>s0</td><td>s0</td><td>s0</td><td>s0</td><td>s0</td><td>s0</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td></tr></table>															s0	s0	s0	s0	s0	s0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
s0	s0	s0	s0	s0	s0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00																
小数 (1.15)	<table><tr><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>															d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0
d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0	0																
有符号小数 (1.15)	<table><tr><td>s0</td><td>d09</td><td>d08</td><td>d07</td><td>d06</td><td>d05</td><td>d04</td><td>d03</td><td>d02</td><td>d01</td><td>d00</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>															s0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0
s0	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0	0																

表 22-3: 各种结果编码的等效数值：10 位整数格式

V _{IN} /V _{REF}	10 位差分输出编码 (11 位结果)	16 位整数格式 / 等效 10 进制数值		16 位有符号整数格式 / 等效 10 进制数值	
+1023/1024	011 1111 1111	0000 0011 1111 1111	1023	0000 0001 1111 1111	1023
+1022/1024	011 1111 1110	0000 0011 1111 1110	1022	0000 0001 1111 1110	1022
...					
+1/1024	000 0000 0001	0000 0000 0000 0001	1	0000 0000 0000 0001	1
0/1024	000 0000 0000	0000 0000 0000 0000	0	0000 0000 0000 0000	0
-1/1024	101 1111 1111	0000 0000 0000 0000	0	1111 1111 1111 1111	-1
...					
-1023/1024	100 0000 0001	0000 0000 0000 0000	0	1111 1110 0000 0001	-1023
-1024/1024	100 0000 0000	0000 0000 0000 0000	0	1111 1110 0000 0000	-1024

表 22-4: 各种结果编码的等效数值: 10 位小数格式

V _{IN} /V _{REF}	10 位差分输出编码 (11 位结果)	16 位小数格式 / 等效 10 进制数值		16 位有符号小数格式 / 等效 10 进制数值	
+1023/1024	011 1111 1111	1111 1111 1100 0000	0.999	0111 1111 1110 0000	0.999
+1022/1024	011 1111 1110	1111 1111 1000 0000	0.998	0111 1111 1000 0000	0.998
...					
+1/1024	000 0000 0001	0000 0000 0100 0000	0.001	0000 0000 0010 0000	0.001
0/1024	000 0000 0000	0000 0000 0000 0000	0.000	0000 0000 0000 0000	0.000
-1/1024	101 1111 1111	0000 0000 0000 0000	0.000	1111 1111 1110 0000	-0.001
...					
-1023/1024	100 0000 0001	0000 0000 0000 0000	0.000	1000 0000 0010 0000	-0.999
-1024/1024	100 0000 0000	0000 0000 0000 0000	0.000	1000 0000 0000 0000	-1.000

PIC24FV32KA304 系列

注:

23.0 比较器模块

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本参考手册当作无所不包的参考手册来使用。如需了解比较器模块的更多信息，请参见《PIC24F 系列参考手册》中的第 46 章“可扩展的比较器模块”（DS39734A_CN）。

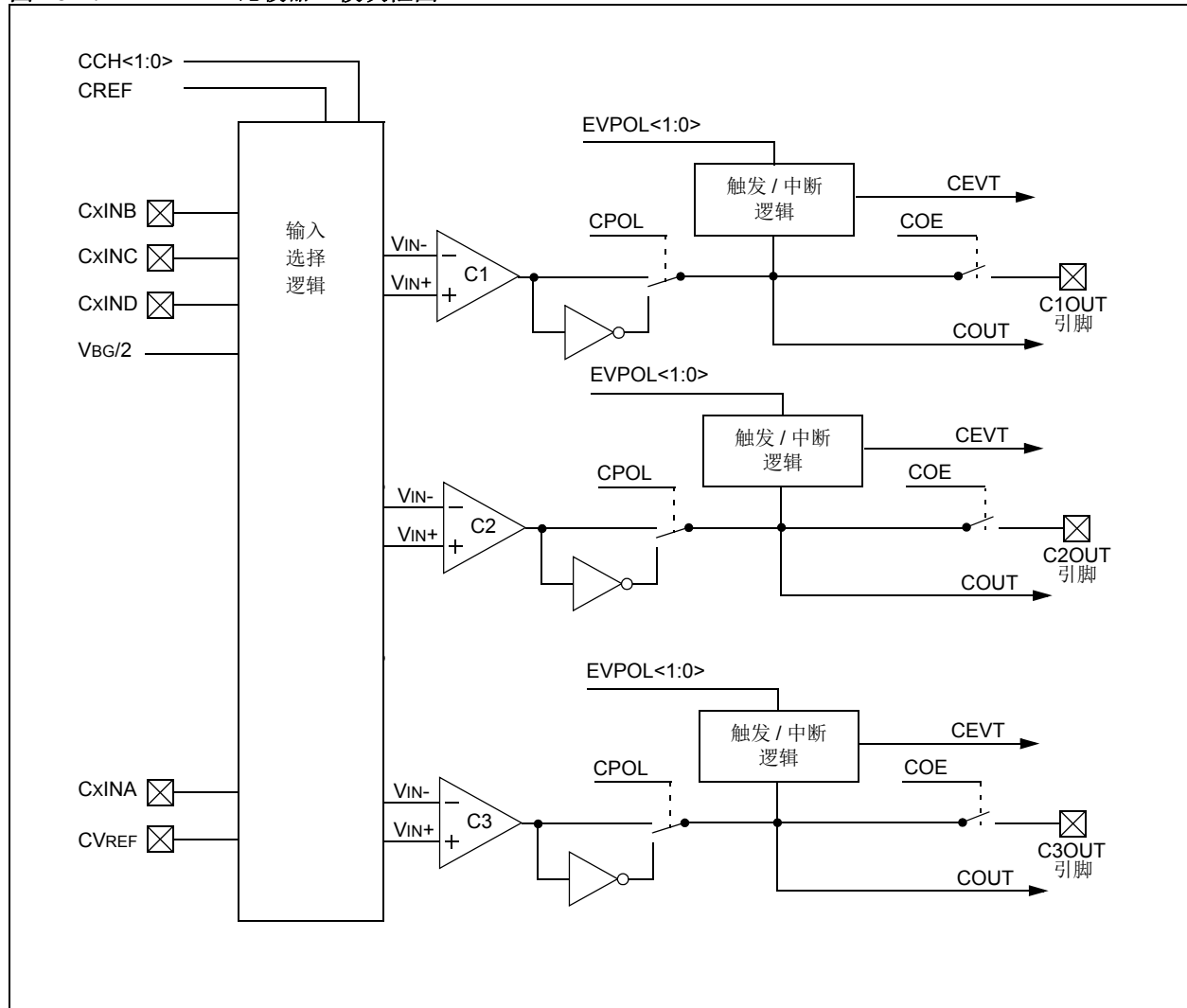
比较器模块提供三个双输入比较器。至比较器的输入可配置为使用四个外部模拟输入以及参考电压输入，后者由内部带隙参考除以 2 ($V_{BG}/2$) 得出或来自比较器参考电压发生器。

比较器输出可直接连接到 CxOUT 引脚。当对应的 COE 等于 1 时，I/O 引脚逻辑使比较器的未同步输出在引脚上可用。

模块的简化框图如图 23-1 所示。图 23-2 中给出了各种可能的比较器配置图。

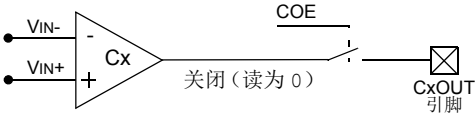
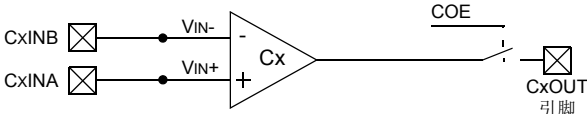
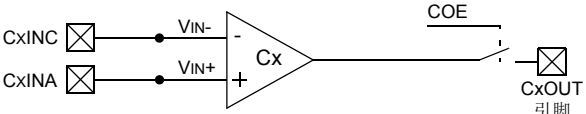
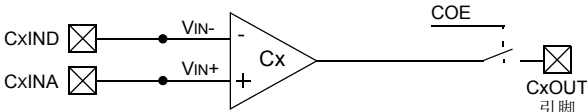
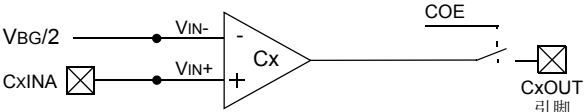
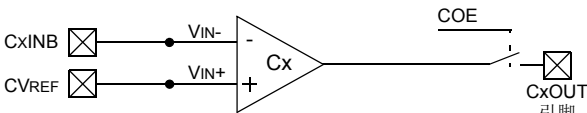
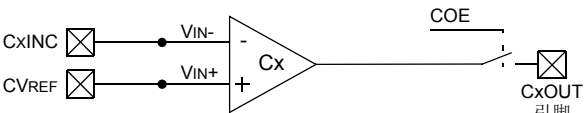
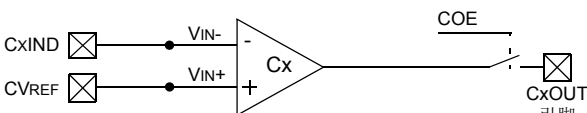
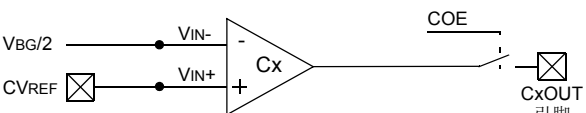
每个比较器都有自己的控制寄存器 CMxCON（寄存器 23-1），用于使能和配置其操作。所有三个比较器的输出和事件状态都在 CMSTAT 寄存器（寄存器 23-2）中给出。

图 23-1: 比较器 x 模块框图



PIC24FV32KA304 系列

图 23-2: 各个比较器配置

比较器关闭 $CON = 0, CREF = x, CCH<1:0> = xx$ 	
比较器 $CxINB > CxINA$ $CON = 1, CREF = 0, CCH<1:0> = 00$ 	比较器 $CxINC > CxINA$ $CON = 1, CREF = 0, CCH<1:0> = 01$ 
比较器 $CxIND > CxINA$ $CON = 1, CREF = 0, CCH<1:0> = 10$ 	比较器 $V_{BG} > CxINA$ $CON = 1, CREF = 0, CCH<1:0> = 11$ 
比较器 $CxINB > CVREF$ $CON = 1, CREF = 1, CCH<1:0> = 00$ 	比较器 $CxINC > CVREF$ $CON = 1, CREF = 1, CCH<1:0> = 01$ 
比较器 $CxIND > CVREF$ $CON = 1, CREF = 1, CCH<1:0> = 10$ 	比较器 $V_{BG} > CVREF$ $CON = 1, CREF = 1, CCH<1:0> = 11$ 

PIC24FV32KA304 系列

寄存器 23-1: CMxCON: 比较器 x 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R-0
CON	COE	CPOL	CLPWR	—	—	CEVT	COUT
bit 15				bit 8			
R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **CON:** 比较器 x 使能位
 1 = 使能比较器 x
 0 = 禁止比较器 x
- bit 14 **COE:** 比较器 x 输出使能位
 1 = 在 CxOUT 引脚上提供比较器输出
 0 = 比较器输出仅供内部使用
- bit 13 **CPOL:** 比较器 x 输出极性选择位
 1 = 比较器输出反相
 0 = 比较器输出不反相
- bit 12 **CLPWR:** 比较器 x 低功耗模式选择位
 1 = 比较器在低功耗模式下工作
 0 = 比较器不在低功耗模式下工作
- bit 11-10 **未实现:** 读为 0
- bit 9 **CEVT:** 比较器 x 事件位
 1 = 发生由 EVPOL<1:0> 定义的比较器事件; 该位清零之前禁止后续触发和中断
 0 = 未产生比较器事件
- bit 8 **COUT:** 比较器 x 输出位
 当 CPOL = 0 时:
 1 = $V_{IN+} > V_{IN-}$
 0 = $V_{IN+} < V_{IN-}$
 当 CPOL = 1 时:
 1 = $V_{IN+} < V_{IN-}$
 0 = $V_{IN+} > V_{IN-}$
- bit 7-6 **EVPOL<1:0>:** 触发 / 事件 / 中断极性选择位
 11 = 比较器输出的任何变化导致发生触发 / 事件 / 中断 (当 CEVT = 0 时)
 10 = 比较器输出的以下跳变导致发生触发 / 事件 / 中断:
 如果 CPOL = 0 (同相极性):
 仅从高水平到低电平的跳变。
 如果 CPOL = 1 (反相极性):
 仅从低电平到高水平的跳变。
 01 = 比较器输出的以下跳变导致发生触发 / 事件 / 中断:
 如果 CPOL = 0 (同相极性):
 仅从低电平到高水平的跳变。
 如果 CPOL = 1 (反相极性):
 仅从高水平到低电平的跳变。
 00 = 禁止发生触发 / 事件 / 中断
- bit 5 **未实现:** 读为 0

PIC24FV32KA304 系列

寄存器 23-1: CMxCON: 比较器 x 控制寄存器 (续)

- bit 4 **CREF:** 比较器 x 参考电压选择位 (同相输入)
1 = 同相输入连接到内部 CVREF 电压
0 = 同相输入连接到 CxINA 引脚
- bit 3-2 **未实现:** 读为 0
- bit 1-0 **CCH<1:0>:** 比较器 x 通道选择位
11 = 比较器反相输入连接到 VBG
10 = 比较器反相输入连接到 CxIND 引脚
01 = 比较器反相输入连接到 CxINC 引脚
00 = 比较器反相输入连接到 CxINB 引脚

寄存器 23-2: CMSTAT: 比较器 x 模块状态寄存器

R/W-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
CMIDL	—	—	—	—	C3EVT	C2EVT	C1EVT
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC	R-0, HSC
—	—	—	—	—	C3OUT	C2OUT	C1OUT
bit 7				bit 0			

图注: HSC = 可由硬件置 1/ 清零的位
R = 可读位 W = 可写位 U = 未实现位, 读为 0
-n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

- bit 15 **CMIDL:** 比较器 x 空闲模式停止位
1 = 空闲模式下禁止比较器中断, 已使能的比较器保持工作
0 = 所有使能的比较器在空闲模式下继续工作
- bit 14-11 **未实现:** 读为 0
- bit 10 **C3EVT:** 比较器 3 事件状态位 (只读)
显示比较器 3 的当前事件状态 (CM3CON<9>)。
- bit 9 **C2EVT:** 比较器 2 事件状态位 (只读)
显示比较器 2 的当前事件状态 (CM2CON<9>)。
- bit 8 **C1EVT:** 比较器 1 事件状态位 (只读)
显示比较器 1 的当前事件状态 (CM1CON<9>)。
- bit 7-3 **未实现:** 读为 0
- bit 2 **C3OUT:** 比较器 3 输出状态位 (只读)
显示比较器 3 的当前输出 (CM3CON<8>)。
- bit 1 **C2OUT:** 比较器 2 输出状态位 (只读)
显示比较器 2 的当前输出 (CM2CON<8>)。
- bit 0 **C1OUT:** 比较器 1 输出状态位 (只读)
显示比较器 1 的当前输出 (CM1CON<8>)。

24.0 比较器参考电压

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解比较器参考电压的更多信息，请参见《PIC24F 系列参考手册》中的第 20 章“比较器参考电压模块”（DS39709A_CN）。

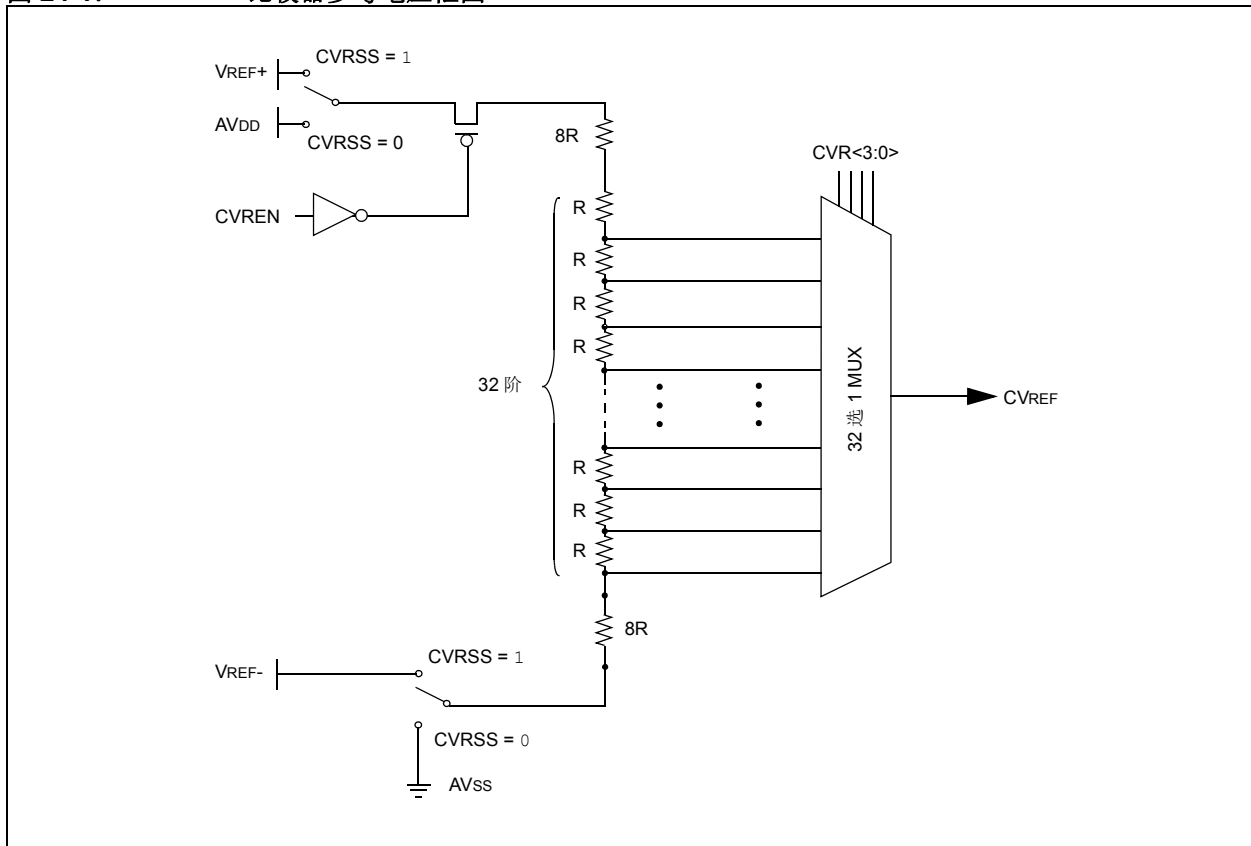
24.1 配置比较器参考电压

比较器参考电压模块由 CVRCON 寄存器（寄存器 24-1）控制。该比较器参考电压提供一个输出电压范围，该范围具有 32 种不同的电平。

比较器参考电压模块的电压可以来自 VDD 和 VSS 或外部 VREF+ 和 VREF-。由 CVRSS 位（CVRCON<5>）选择电压源。

在改变 CVREF 输出值时，必须考虑到比较器参考电压的稳定时间。

图 24-1: 比较器参考电压框图



PIC24FV32KA304 系列

寄存器 24-1: **CVRCON**: 比较器参考电压控制寄存器

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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-8 未实现: 读为 0
- bit 7 **CVREN**: 比较器参考电压使能位
 1 = CVREF 电路上电
 0 = CVREF 电路掉电
- bit 6 **CVROE**: 比较器 VREF 输出使能位
 1 = CVREF 电压从 CVREF 引脚输出
 0 = CVREF 电压与 CVREF 引脚断开
- bit 5 **CVRSS**: 比较器 VREF 源选择位
 1 = 比较器参考源 CVRSRC = VREF+ – VREF-
 0 = 比较器参考源 CVRSRC = AVDD – AVSS
- bit 4-0 **CVR<4:0>**: 比较器 VREF 值选择位 (0 ≤ CVR<4:0> ≤ 31)
 当 CVRSS = 1 时:
 $CVREF = (VREF-) + (CVR<4:0>/32) \cdot (VREF+ - VREF-)$
 当 CVRSS = 0 时:
 $CVREF = (AVSS) + (CVR<4:0>/32) \cdot (AVDD - AVSS)$

25.0 充电时间测量单元 (CTMU)

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本参考手册当作无所不包的参考手册来使用。如需了解充电时间测量单元的更多信息，请参见《PIC24F 系列参考手册》中的第 53 章“带阈值检测功能的充电时间测量单元 (CTMU)” (DS39743A_CN)。

充电时间测量单元 (CTMU) 是一个灵活的模拟模块，它提供充电时间测量、脉冲源之间的精确时间差测量，以及异步脉冲生成。它的主要功能包括：

- 13 个外部边沿输入触发源
- 每个边沿源的极性控制
- 边沿序列控制
- 控制对边沿电平或边沿跳变的响应
- 1 ns 的时间测量分辨率
- 适合测量电容的精确电流源

CTMU 可与其他片内模拟模块一起使用，以精确地测量时间、电容以及电容的相对变化或生成不依赖于系统时钟的输出脉冲。CTMU 模块是与电容式触摸传感器接口的理想选择。

CTMU 可通过三个寄存器进行控制：CTMUCON1、CTMUCON2 和 CTMUICON。CTMUCON1 用于使能模块和控制 CTMU 的工作模式，以及控制边沿序列。CTMUCON2 用于控制边沿源选择和边沿源极性的选择。CTMUICON 寄存器用于选择电流源范围并微调电流。

25.1 测量电容

CTMU 模块通过生成脉冲宽度与两个独立输入通道上的边沿事件间的时间宽度相等的输出脉冲，来测量电容。这两个输入通道上的脉冲边沿事件源可从几款内部外设模块 (OC1、Timer1、任何输入捕捉或输出比较模块) 和最多 13 个外部引脚 (CTED1 到 CTED13) 进行选择。该脉冲和该模块的精确电流源一起使用，可根据以下关系计算电容：

公式 25-1：

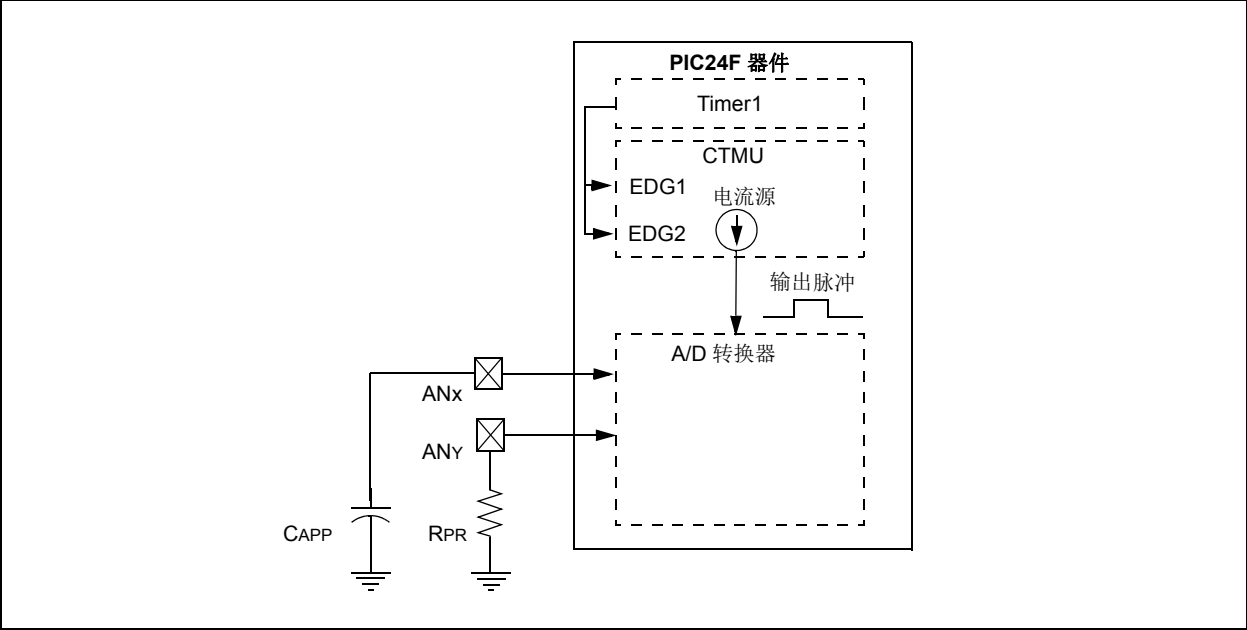
$$I = C \cdot \frac{dV}{dT}$$

如需测量电容，A/D 转换器在 CTMU 输出的脉冲信号之后对它的其中一个输入通道采样外部电容 (CAPP)。由第二个 A/D 通道上的高精度电阻 (RPR) 对电流源进行校准。脉冲信号结束后，转换器确定电容上的电压。电容的实际计算在软件中由应用程序执行。

图 25-1 给出了电容测量使用的外部连接以及此应用中 CTMU 和 A/D 模块的关系。此示例还展示了来自 Timer1 的边沿事件，但使用外部边沿源的其他配置也是可能的。关于使用 CTMU 模块测量电容和时间的详细说明在《PIC24F 系列参考手册》中提供。

PIC24FV32KA304 系列

图 25-1: 电容测量的典型连接和内部配置

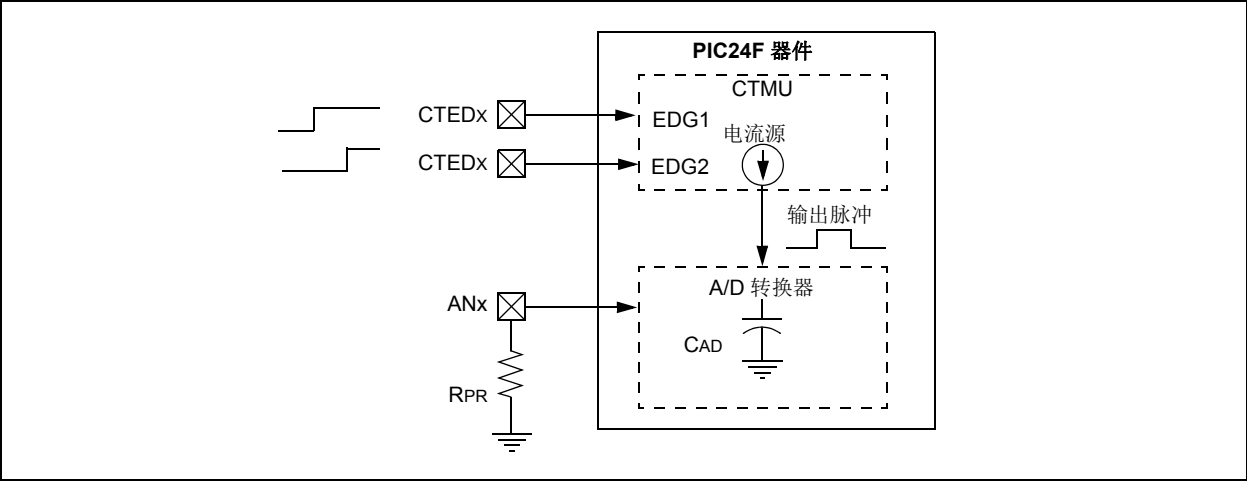


25.2 测量时间

对脉冲宽度的时间测量也可类似地执行，需要使用 A/D 模块的内部电容（CAD）和用于校准电流的高精度电阻。图 25-3 给出了时间测量使用的外部连接以及此应用

中 CTMU 和 A/D 模块的关系。此示例还显示了来自外部 CTEDx 引脚的边沿事件，但使用内部边沿源的其他配置也是可能的。

图 25-2: 测量时间的典型连接和内部配置



25.3 脉冲生成和延时

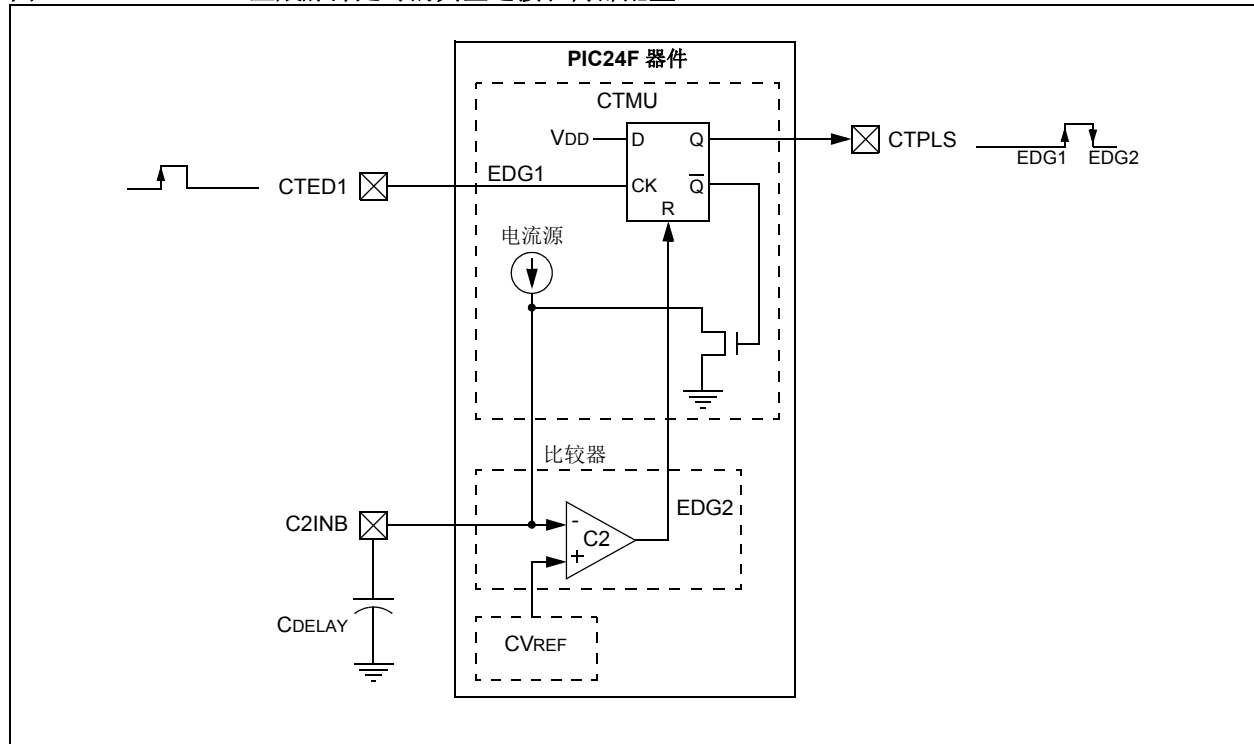
CTMU 模块也可生成边沿和器件的系统时钟异步的输出脉冲。更明确地说，它可以生成边沿事件输入到该模块的延时可编程的脉冲。

当模块通过将 TGEN 位 (CTMUCON<12>) 置 1 配置为脉冲生成延时，内部电流源连接到比较器 2 的 B 输入。将电容 (CDELAY) 连接到比较器 2 的引脚 C2INB，且比较器参考电压 CVREF 连接到 C2INA。CVREF 随后被配置为特定跳变点。当检测到边沿事件时，模块开始对 CDELAY 充电。当 CVREF 高于 CDELAY 上的电压时，CTPLS 为高电平。

当 CDELAY 上的电压等于 CVREF 时，CTPLS 变为低电平。当比较器 2 输出第 2 个边沿时，停止对 CTMU 充电。在此状态事件中，CTMU 自动接地。在下一个 CTPLS 周期之前，无需将 IDISSEN 位置 1 或清零。

图 25-3 给出了脉冲生成的外部连接，以及所需的不同模拟模块之间的关系。CTED1 显示为输入脉冲源时，其他选项可用。关于使用 CTMU 模块生成脉冲的详细说明在《PIC24F 系列参考手册》中提供。

图 25-3: 生成脉冲延时的典型连接和内部配置



PIC24FV32KA304 系列

寄存器 25-1: CTMUCON1: CTMU 控制寄存器 1

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CTMUEN	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG
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 = 上电复位时的值		1 = 置 1		0 = 清零		x = 未知	

- bit 15

CTMUEN: CTMU 使能位
1 = 使能模块
0 = 禁止模块
- bit 14

未实现: 读为 0
- bit 13

CTMUSIDL: CTMU 在空闲模式停止位
1 = 当器件进入空闲模式时, 模块停止工作
0 = 模块在空闲模式下继续工作
- bit 12

TGEN: 时间生成使能位
1 = 使能生成边沿延时
0 = 禁止生成边沿延时
- bit 11

EDGEN: 边沿使能位
1 = 不阻止边沿
0 = 阻止边沿
- bit 10

EDGSEQEN: 边沿发生顺序使能位
1 = 边沿 1 事件必须在边沿 2 事件之前发生
0 = 边沿无需按顺序发生
- bit 9

IDISSEN: 模拟电流源控制位
1 = 模拟电流源输出接地
0 = 模拟电流源输出不接地
- bit 8

CTTRIG: CTMU 触发器控制位
1 = 使能触发器输出
0 = 禁止触发器输出
- bit 7-0

未实现: 读为 0

寄存器 25-2: CTMUCON2: CTMU 控制寄存器 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
EDG1MOD	EDG1POL	EDG1SEL3	EDG1SEL2	EDG1SEL1	EDG1SEL0	EDG2STAT	EDG1STAT
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
EDG2MOD	EDG2POL	EDG2SEL3	EDG2SEL2	EDG2SEL1	EDG2SEL0	—	—
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **EDG1MOD:** 边沿 1 边沿敏感选择位

1 = 输入边沿敏感

0 = 输入电平敏感

bit 14 **EDG1POL:** 边沿 1 极性选择位

1 = 边沿 1 编程为正边沿响应

0 = 边沿 1 编程为负边沿响应

bit 13-10 **EDG1SEL<3:0>:** 边沿 1 源选择位

1111 = 边沿 1 源为比较器 3 输出

1110 = 边沿 1 源为比较器 2 输出

1101 = 边沿 1 源为比较器 1 输出

1100 = 边沿 1 源为 IC3

1011 = 边沿 1 源为 IC2

1010 = 边沿 1 源为 IC1

1001 = 边沿 1 源为 CTED8

1000 = 边沿 1 源为 CTED7

0111 = 边沿 1 源为 CTED6

0110 = 边沿 1 源为 CTED5

0101 = 边沿 1 源为 CTED4

0100 = 边沿 1 源为 CTED3⁽²⁾

0011 = 边沿 1 源为 CTED1

0010 = 边沿 1 源为 CTED2

0001 = 边沿 1 源为 OC1

0000 = 边沿 1 源为 Timer1

bit 9 **EDG2STAT:** 边沿 2 状态位

指示边沿 2 的状态, 可写入以控制电流源。

1 = 已发生边沿 2 事件

0 = 未发生边沿 2 事件

bit 8 **EDG1STAT:** 边沿 1 状态位

指示边沿 1 的状态, 可写入以控制电流源。

1 = 已发生边沿 1 事件

0 = 未发生边沿 1 事件

bit 7 **EDG2MOD:** 边沿 2 边沿敏感选择位

1 = 输入边沿敏感

0 = 输入电平敏感

注 1: 边沿源、CTED11 和 CTED12 在 PIC24FV32KA302 器件上不可用。

2: 边沿源、CTED3、CTED11、CTED12 和 CTED13 在 PIC24FV32KA301 器件上不可用。

PIC24FV32KA304 系列

寄存器 25-2: CTMUCON2: CTMU 控制寄存器 2 (续)

bit 6	EDG2POL: 边沿 2 极性选择位 1 = 边沿 2 编程为正边沿 0 = 边沿 2 编程为负边沿
bit 5-2	EDG2SEL<3:0>: 边沿 2 源选择位 1111 = 边沿 2 源为比较器 3 输出 1110 = 边沿 2 源为比较器 2 输出 1101 = 边沿 2 源为比较器 1 输出 1100 = 未实现; 不要使用 1011 = 边沿 2 源为 IC3 1010 = 边沿 2 源为 IC2 1001 = 边沿 2 源为 IC1 1000 = 边沿 2 源为 CTED13 ⁽²⁾ 0111 = 边沿 2 源为 CTED12 ^(1,2) 0110 = 边沿 2 源为 CTED11 ^(1,2) 0101 = 边沿 2 源为 CTED10 0100 = 边沿 2 源为 CTED9 0011 = 边沿 2 源为 CTED1 0010 = 边沿 2 源为 CTED2 0001 = 边沿 2 源为 OC1 0000 = 边沿 2 源为 Timer1
bit 1-0	未实现: 读为 0

- 注 **1:** 边沿源、CTED11 和 CTED12 在 PIC24FV32KA302 器件上不可用。
 2: 边沿源、CTED3、CTED11、CTED12 和 CTED13 在 PIC24FV32KA301 器件上不可用。

PIC24FV32KA304 系列

寄存器 25-3: CTMUICON: CTMU 电流控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ITRIM5	ITRIM4	ITRIM3	ITRIM2	ITRIM1	ITRIM0	IRNG1	IRNG0
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 = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-10 ITRIM<5:0>: 电流源调节位

011111 = 标称电流的最大正向调整

011110

.

.

.

000001 = 标称电流的最小正向调整

000000 = IRNG<1:0> 指定的标称电流输出

111111 = 标称电流的最小负向调整

.

.

.

100010

100001 = 标称电流的最大负向调整

bit 9-8 IRNG<1:0>: 电流源范围选择位

11 = 100 倍基本电流

10 = 10 倍基本电流

01 = 基本电流级别 (标称值为 0.55 μ A)

00 = 1000 倍基本电流

bit 7-0 未实现: 读为 0

PIC24FV32KA304 系列

注:

26.0 特殊功能

注： 本数据手册总结了 PIC24F 系列器件的功能。但是不应把本数据手册当作无所不包的参考手册来使用。如需了解看门狗定时器、高级器件集成和编程诊断的更多信息，请参见《PIC24F 系列参考手册》的以下章节：

- 第 9 章“看门狗定时器 (WDT)” (DS39697B_CN)
- 第 36 章“高度集成的可编程高/低电压检测 (HLVD) 模块” (DS39725A_CN)
- 第 33 章“编程和诊断” (DS39716A_CN)

PIC24FV32KA304 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的使用将成本降至最低。这些特殊功能包括：

- 灵活的配置
- 看门狗定时器 (WDT)
- 代码保护
- 在线串行编程 (ICSP™)
- 在线仿真

26.1 配置位

可以通过对配置位编程（读为 0）或不编程（读为 1）来选择不同的器件配置。这些配置位被映射到程序存储器中从 F80000h 开始的单元。表 26-1 给出了配置寄存器地址的完整列表。寄存器 26-1 到寄存器 26-8 详细说明了各配置位的不同功能。

注意，地址 F80000h 超出了用户程序存储空间的范围。事实上，它属于配置存储空间（800000h-FFFFFFh），这一空间仅能通过表读和表写进行访问。

表 26-1: 配置寄存器位置

配置寄存器	地址
FBS	F80000
FGS	F80004
FOSCSEL	F80006
FOSC	F80008
FWDT	F8000A
FPOR	F8000C
FICD	F8000E
FDS	F80010

寄存器 26-1: FBS: 引导段配置寄存器

U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	—	BSS2	BSS1	BSS0	BWRP
bit 7				bit 0			

图注：

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

bit 7-4 **未实现：** 读为 0

bit 3-1 **BSS<2:0>:** 引导段程序闪存代码保护位

111 = 无引导程序闪存段
 011 = 保留
 110 = 标准安全性，引导程序闪存段起始于 200h，结束于 000AFEh
 010 = 高安全性，引导程序闪存段起始于 200h，结束于 000AFEh
 101 = 标准安全性，引导程序闪存段起始于 200h，结束于 0015FEh⁽¹⁾
 001 = 高安全性，引导程序闪存段起始于 200h，结束于 0015FEh⁽¹⁾
 100 = 标准安全性，引导程序闪存段起始于 200h，结束于 002BFEh⁽¹⁾
 000 = 高安全性，引导程序闪存段起始于 200h，结束于 002BFEh⁽¹⁾

bit 0 **BWRP:** 引导段程序闪存写保护位

1 = 可写引导段
 0 = 引导段被写保护

注 1: 该选择不应用于 PIC24FV16KA3XX 器件。

PIC24FV32KA304 系列

寄存器 26-2: **FGS: 通用段配置寄存器**

U-0	U-0	U-0	U-0	U-0	U-0	R/C-1	R/C-1
—	—	—	—	—	—	GSS0	GWRP
bit 7						bit 0	

图注:

R = 可读位 C = 可清零位 U = 未实现位, 读为 0
-n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

- bit 7-2 **未实现:** 读为 0
- bit 1 **GSS0:** 通用段代码闪存代码保护位
1 = 无保护
0 = 使能标准安全性
- bit 0 **GWRP:** 通用段代码闪存写保护位
1 = 可写通用段
0 = 通用段被写保护

寄存器 26-3: **FOSCSEL: 振荡器选择配置寄存器**

R/P-1	R/P-1	R/P-1	U-0	U-0	R/P-1	R/P-1	R/P-1
IESO	LPRCSEL	SOSCSRC	—	—	FNOSC2	FNOSC1	FNOSC0
bit 7						bit 0	

图注:

R = 可读位 P = 可编程位 U = 未实现位, 读为 0
-n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

- bit 7 **IESO:** 内部 / 外部切换位
1 = 使能内部 / 外部切换模式 (使能双速启动)
0 = 禁止内部 / 外部切换模式 (禁止双速启动)
- bit 6 **LPRCSEL:** 内部 LPRC 振荡器功率选择位
1 = 高功率 / 高精度模式
0 = 低功率 / 低精度模式
- bit 5 **SOSCSRC:** 辅助振荡器时钟源配置位
1 = SOSC 模拟晶振功能在 SOSC/SOSCO 引脚上可用
0 = 禁止 SOSC 晶振; 在 SOSCO 引脚上选择数字 SCLKI 功能
- bit 4-3 **未实现:** 读为 0
- bit 2-0 **FNOSC<2:0>:** 振荡器选择位
000 = 快速 RC 振荡器 (FRC)
001 = 带 PLL 模块的 N 分频快速 RC 振荡器 (FRCDIV+PLL)
010 = 主振荡器 (XT、HS 和 EC)
011 = 带 PLL 模块的主振荡器 (HS+PLL 和 EC+PLL)
100 = 辅助振荡器 (SOSC)
101 = 低功耗 RC 振荡器 (LPRC)
110 = 可 N 分频的 500 kHz 低功耗 FRC 振荡器 (LPFRCDIV)
111 = 可 N 分频的 8 MHz FRC 振荡器 (FRCDIV)

PIC24FV32KA304 系列

寄存器 26-4: FOSC: 振荡器配置寄存器

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
FCKSM1	FCKSM0	SOSCSEL	POSCFREQ1	POSCFREQ0	OSCIOFNC	POSCMD1	POSCMD0
bit 7							bit 0

图注:

R = 可读位

P = 可编程位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 7-6 **FCKSM<1:0>**: 时钟切换和故障保护时钟监视器选择配置位

1x = 禁止时钟切换和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 = 使能时钟切换和故障保护时钟监视器

bit 5 **SOSCSEL**: 辅助振荡器功耗选择配置位

1 = 辅助振荡器配置为高功耗运行

0 = 辅助振荡器配置为低功耗运行

bit 4-3 **POSCFREQ<1:0>**: 主振荡器频率范围配置位

11 = 主振荡器 / 外部时钟输入频率大于 8 MHz

10 = 主振荡器 / 外部时钟输入频率介于 100 kHz 和 8 MHz 之间

01 = 主振荡器 / 外部时钟输入频率小于 100 kHz

00 = 保留, 不要使用

bit 2 **OSCIOFNC**: CLKO 使能配置位

1 = CLKO 输出信号在 OSCO 引脚上有效; 必须禁止主振荡器或将其配置为外部时钟模式 (EC) 以使 CLKO 有效 (POSCMD<1:0> = 11 或 00)

0 = 禁止 CLKO 输出

bit 1-0 **POSCMD<1:0>**: 主振荡器配置位

11 = 禁止主振荡器模式

10 = 选择 HS 振荡器模式

01 = 选择 XT 振荡器模式

00 = 选择外部时钟模式

PIC24FV32KA304 系列

寄存器 26-5: FWDT: 看门狗定时器配置寄存器

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
FWDTEN1	WINDIS	FWDTEN0	FWPSA	WDTPS3	WDTPS2	WDTPS1	WDTPS0
bit 7							bit 0

图注:			
R = 可读位	P = 可编程位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 7,5

FWDTEN<1:0>: 看门狗定时器使能位

11 = 由硬件使能 WDT

10 = 通过设置 SWDTEN 位控制 WDT

01 = 仅当器件工作时使能 WDT ; 在器件处于休眠模式禁止; 禁止 SWDTEN 位

00 = 由硬件禁止 WDT ; 禁止 SWDTEN 位
- bit 6

WINDIS: 窗口看门狗定时器禁止位

1 = 选择标准 WDT ; 禁止窗口 WDT

0 = 使能窗口 WDT ; 请注意, 当由硬件和软件禁止 WDT (FWDTEN<1:0> = 00 且 SWDTEN (RCON<5>) = 0) 时, 执行 CLRWDTPS 指令不会导致器件复位
- bit 4

FWPSA: WDT 预分频比位

1 = WDT 预分频比为 1:128

0 = WDT 预分频比为 1:32
- bit 3-0

WDTPS<3:0>: 看门狗定时器后分频比选择位

1111 = 1:32,768

1110 = 1:16,384

1101 = 1:8,192

1100 = 1:4,096

1011 = 1:2,048

1010 = 1:1,024

1001 = 1:512

1000 = 1:256

0111 = 1:128

0110 = 1:64

0101 = 1:32

0100 = 1:16

0011 = 1:8

0010 = 1:4

0001 = 1:2

0000 = 1:1

PIC24FV32KA304 系列

寄存器 26-6: FPOR: 复位配置寄存器

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
MCLRE ⁽²⁾	BORV1 ⁽³⁾	BORV0 ⁽³⁾	I2C1SEL ⁽¹⁾	PWRTEN	RETCFG ⁽¹⁾	BOREN1	BOREN0
bit 7							bit 0

图注:

R = 可读位

P = 可编程位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 7 **MCLRE:** MCLR 引脚使能位 ⁽²⁾

1 = 使能 MCLR 引脚; 禁止 RA5 输入引脚

0 = 使能 RA5 输入引脚; 禁止 MCLR

bit 6-5 **BORV<1:0>:** 欠压复位使能位 ⁽³⁾

11 = 欠压复位设置为最低电压

10 = 欠压复位

01 = 欠压复位设置为最高电压

00 = 使能 POR 时电压下降保护 —— 选择“零功耗”

bit 4 **I2C1SEL:** 备用 I2C1 引脚映射位 ⁽¹⁾

1 = SCL1/SDA1 引脚的默认位置

0 = SCL1/SDA1 引脚的备用位置

bit 3 **PWRTEN:** 上电延时定时器使能位

1 = 使能 PWRT

0 = 禁止 PWRT

bit 2 **RETCFG:** 保持稳压器配置位 ⁽¹⁾

1 = 保持稳压器不可用

0 = 休眠期间保持稳压器可用且由 RETEN 位 (RCON<12>) 控制

bit 1-0 **BOREN<1:0>:** 欠压复位使能位

11 = 由硬件使能欠压复位; 禁止 SBOREN 位

10 = 仅当器件工作时使能欠压复位, 在器件处于休眠模式时禁止; 禁止 SBOREN 位

01 = 通过设置 SBOREN 位控制欠压复位

00 = 由硬件禁止欠压复位; 禁止 SBOREN 位

注 1: 该设置仅适用于 FV 器件。在 F 器件上, 该位保留且应保持为 1。

2: 仅当使用基于 VPP 的方式进入 ICSP™ 模式时可更改 MCLRE 熔丝。这可以防止用户意外地锁定器件, 无法进入低压测试。

3: 有关 BOR 电压, 请参见第 29.0 节“电气特性”。

PIC24FV32KA304 系列

寄存器 26-7: FICD: 在线调试器配置寄存器

R/P-1	U-0	U-0	U-0	U-0	U-0	R/P-1	R/P-1
DEBUG	—	—	—	—	—	FICD1	FICD0
bit 7							bit 0

图注:			
R = 可读位	P = 可编程位	U = 未实现位, 读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

- bit 7 **DEBUG:** 后台调试器使能位
- 1 = 禁止后台调试器
- 0 = 使能后台调试器功能
- bit 6-2 **未实现:** 读为 0
- bit 1-0 **FICD<1:0>:** ICD 引脚选择位
- 11 = PGEC1/PGED1 用于编程和调试器件
- 10 = PGEC2/PGED2 用于编程和调试器件
- 01 = PGEC3/PGED3 用于编程和调试器件
- 00 = 保留; 不要使用

寄存器 26-8: FDS: 深度休眠配置寄存器

R/P-1	R/P-1	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
DSWDTEN	DSBOREN	—	DSWDTOSC	DSWDTPS3	DSWDTPS2	DSWDTPS1	DSWDTPS0
bit 7							bit 0

图注:

R = 可读位

P = 可编程位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

- bit 7 **DSWDTEN:** 深度休眠看门狗定时器使能位
 1 = 使能 DSWDT
 0 = 禁止 DSWDT
- bit 6 **DSBOREN:** 深度休眠 / 低功耗 BOR 使能位
 (不会影响非深度休眠模式下的工作)
 1 = 在深度休眠模式下, 使能深度休眠 BOR
 0 = 在深度休眠模式下, 禁止深度休眠 BOR
- bit 5 **未实现:** 读为 0
- bit 4 **DSWDTOSC:** DSWDT 参考时钟选择位
 1 = DSWDT 使用 LPRC 作为参考时钟
 0 = DSWDT 使用 SOSC 作为参考时钟
- bit 3-0 **DSWDTPS<3:0>:** 深度休眠看门狗定时器后分频比选择位
 DSWDT 分频比为 32; 这创建一个大约 1 ms 的时基单位。
 1111 = 1:2,147,483,648 (25.7 天) 标称值
 1110 = 1:536,870,912 (6.4 天) 标称值
 1101 = 1:134,217,728 (38.5 小时) 标称值
 1100 = 1:33,554,432 (9.6 小时) 标称值
 1011 = 1:8,388,608 (2.4 小时) 标称值
 1010 = 1:2,097,152 (36 分钟) 标称值
 1001 = 1:524,288 (9 分钟) 标称值
 1000 = 1:131,072 (135 秒) 标称值
 0111 = 1:32,768 (34 秒) 标称值
 0110 = 1:8,192 (8.5 秒) 标称值
 0101 = 1:2,048 (2.1 秒) 标称值
 0100 = 1:512 (528 ms) 标称值
 0011 = 1:128 (132 ms) 标称值
 0010 = 1:32 (33 ms) 标称值
 0001 = 1:8 (8.3 ms) 标称值
 0000 = 1:2 (2.1 ms) 标称值

PIC24FV32KA304 系列

寄存器 26-9: DEVID: 器件 ID 寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 23				bit 16			
R	R	R	R	R	R	R	R
FAMID7	FAMID6	FAMID5	FAMID4	FAMID3	FAMID2	FAMID1	FAMID0
bit 15				bit 8			
R	R	R	R	R	R	R	R
DEV7	DEV6	DEV5	DEV4	DEV3	DEV2	DEV1	DEV0
bit 7				bit 0			

图注:							
R = 可读位		W = 可写位		U = 未实现位, 读为 0			
-n = 上电复位时的值		1 = 置 1		0 = 清零		x = 未知	

bit 23-16	未实现: 读为 0
bit 15-8	FAMID<7:0> : 器件系列标识符位 01000101 = PIC24FV32KA304 系列
bit 7-0	DEV<7:0> : 单个器件标识符位 00010111 = PIC24FV32KA304 00000111 = PIC24FV16KA304 00010011 = PIC24FV32KA302 00000011 = PIC24FV16KA302 00011001 = PIC24FV32KA301 00001001 = PIC24FV16KA301 00010110 = PIC24F32KA304 00000110 = PIC24F16KA304 00010010 = PIC24F32KA302 00000010 = PIC24F16KA302 00011000 = PIC24F32KA301 00001000 = PIC24F16KA301

PIC24FV32KA304 系列

寄存器 26-10: DEVREV: 器件版本寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 23				bit 16			
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	R	R	R
—	—	—	—	REV3	REV2	REV1	REV0
bit 7				bit 0			

图注:			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = 上电复位时的值	1 = 置 1	0 = 清零	x = 未知

bit 23-4 未实现：读为 0
bit 3-0 **REV<3:0>**: 次要版本标识符位

PIC24FV32KA304 系列

26.2 片内稳压器

所有 PIC24FV32KA304 系列器件使用标称值为 3.0V 的电压为其内核数字逻辑供电。对于需要工作在一个更高的典型电压值（如 5.0V）的设计来讲，这可能会带来问题。为简化系统设计，FV 系列中的所有器件均包含一个片内稳压器，可使器件内核逻辑工作在 VDD 下。

始终使能稳压器并通过其他 VDD 引脚向内核供电。必须在 VCAP 引脚连接低 ESR 电容（如陶瓷电容）（图 26-1）。这有利于保持稳压器的稳定性。第 2.4 节“稳压器引脚（VCAP）”和第 29.1 节“直流特性”提供了该滤波电容的推荐值。

对于“F”器件，禁止稳压器。内核逻辑由 VDD 直接供电。这使得器件可在整个允许的低电压范围（1.8V-3.6V）内工作。

26.2.1 稳压器跟踪模式和低电压检测

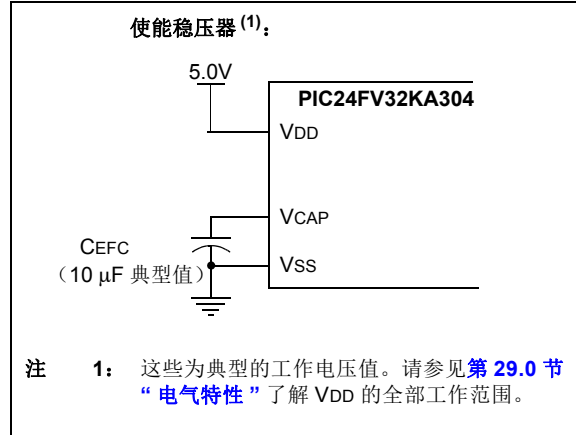
对于所有 PIC24FV32KA304 器件，片内稳压器为数字内核逻辑提供标称值为 3.2V 的恒定电压。稳压器可提供从约为 3.2V 的 VDD 电压直至器件的 VDDMAX 的电压范围。该稳压器无法将低于 3.2V 的 VDD 电压提高。为防止用于稳压器的电压降得过低时产生“欠压”条件，稳压器进入跟踪模式。在跟踪模式下，稳压器输出跟随 VDD，通常比 VDD 小 150 mV。

当器件进入跟踪模式时，不再能以全速工作。为了提供器件何时进入跟踪模式的信息，片内稳压器包含了一个简单的高 / 低电压检测（HLVD）电路。当 VDD 下降至低于全速工作电压时，该电路置 1 高 / 低电压检测中断标志位 HLVDIF（IFS4<8>）。这可用于产生中断并将应用于低功耗工作模式或触发正常关闭。第 29.1 节“直流特性”的图 29-1 和图 29-2 给出了最大器件速度与 VDD 的函数关系。

26.2.2 片内稳压器和 POR

对于 PIC24FV32KA304 器件，稳压器需稍等一段时间（称为 TPM）才能产生稳定输出。在这段时间内，禁止代码执行。每次器件在掉电（包括休眠模式）后恢复工作时都需要 TPM（直流参数 SY71）。

图 26-1: 片内稳压器的连接



26.3 看门狗定时器（WDT）

PIC24FV32KA304 系列器件的 WDT 由 LPRC 振荡器驱动。当使能 WDT 时，也将同时使能该时钟源。

由 LPRC 提供的 WDT 时钟源的频率标称值为 31 kHz。将此时钟源提供给可配置为 5 位（32 分频）或 7 位（128 分频）工作模式的预分频器。分频比由 FWPSA 配置位设置。31 kHz 的输入使预分频器生成标称的 WDT 超时周期（TWDT）——5 位模式下为 1 ms，7 位模式下为 4 ms。

分频比可变的后分频器对 WDT 预分频器的输出进行分频，从而获得更大范围的超时周期。后分频比由 WDTPS<3:0> 配置位（FWDT<3:0>）控制，该配置位共允许选择 16 种设置，从 1:1 到 1:32,768。使用预分频器和后分频器后，可获得 1 ms 到 131 秒的超时周期。

WDT、预分频器和后分频器在以下条件下复位：

- 任何器件复位
- 在时钟切换完成时，无论时钟切换是由软件（即改变 NOSC_x 位后将 OSWEN 位置 1）或是硬件（即故障保护时钟监视器）引起
- 执行 PWRSAV 指令时（即进入休眠模式或空闲模式）
- 当器件退出休眠模式或空闲模式恢复正常工作时
- 在正常执行过程中，执行 CLRWD_T 指令

如果由硬件使能 WDT (FWDTEN<1:0> = 11)，它将在休眠或空闲模式下继续运行。当发生 WDT 超时，将唤醒器件并且代码将从 PWRSAV 指令处继续执行。器件被唤醒后，需要用软件将相应的 SLEEP 或 IDLE 位 (RCON<3:2>) 清零。

WDT 标志位 WDTO (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清零。

注： 当执行 CLRWDT 和 PWRSAV 指令时，预分频器和后分频器的计数值将被清零。

26.3.1 窗口操作

看门狗定时器具有可选的固定窗口工作模式。在此窗口模式下，CLRWDT 指令只能在编程的 WDT 周期的后 1/4 周期复位 WDT。在该窗口前执行 CLRWDT 指令会导致 WDT 复位，这与 WDT 超时类似。

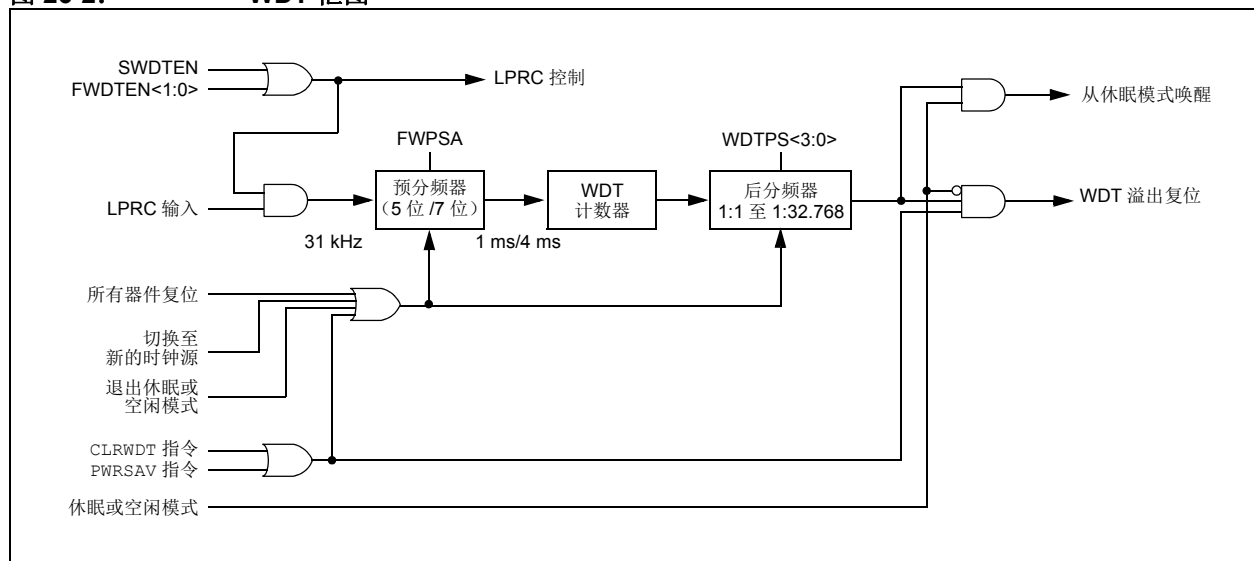
将 WINDIS 配置位 (FWDTE<6>) 编程为 0 使能 WDT 窗口模式。

26.3.2 控制寄存器

通过 FWDTEN<1:0> 配置位使能或禁止 WDT。当 FWDTEN<1:0> 配置位均置 1 时，始终使能 WDT。

FWDTEN<1:0> 配置位被编程为 10 后，也可以使用软件来控制 WDT。通过在软件中将 SWDTEN 控制位 (RCON<5>) 置 1 使能 WDT。任何器件复位都会使 SWDTEN 控制位清零。软件 WDT 选项允许用户在关键代码段使能 WDT，并在非关键代码段禁止 WDT，以最大限度地降低功耗。当 FWDTEN<1:0> 位被设置为 01 时，仅在运行和空闲模式下使能 WDT，而在休眠模式下禁止。通过此设置禁止软件控制 SWDTEN 位 (RCON<5>)。

图 26-2: WDT 框图



PIC24FV32KA304 系列

26.4 深度休眠看门狗定时器 (DSWDT)

在 PIC24FV32KA304 系列器件中，除了 WDT 模块，还具有 DSWDT 模块，如果使能该模块，它可在器件处于深度休眠模式下运行。它由 SOSC 或 LPRC 振荡器驱动。通过配置位 DSWDTOSC (FDS<4>) 选择时钟源。

可通过选择对应的后分频比将 DSWDT 配置为在 2.1 ms 或 25.7 天产生超时。可通过配置位 DSWDTPS<3:0> (FDS<3:0>) 选择后分频比。当使能 DSWDT 时，也将同时使能该时钟源。

DSWDT 为可将器件从深度休眠模式唤醒的源之一。

26.5 程序校验和代码保护

对于 PIC24FV32KA304 系列中的所有器件，引导段的代码保护由配置位 BSS0 控制，而通用段的代码保护由配置位 GSS0 控制。这些位禁止外部读 / 写程序存储空间。这对正常执行模式无任何直接影响。

在配置字中，引导段的代码保护由 BWRP 位控制，而通用段由 GWRP 位控制。当这两个位被编程为 0 时，还禁止在内部对程序存储器执行写和擦除操作。

26.6 在线串行编程

可以在最终应用电路中对 PIC24FV32KA304 系列单片机进行串行编程。只需要 5 根线即可完成这一操作，其中时钟线 (PGECx) 和数据线 (PGEDx) 各一根，其余 3 根分别是电源线、接地线和编程电压线。这允许用户使用未编程器件制造电路板，仅在产品交付前才对单片机进行编程，从而可以将最新版本的固件或定制固件烧写到单片机中。

26.7 在线调试器

当选择 MPLAB® ICD 3、MPLAB REAL ICE™ 或 PICkit™ 3 作为调试器时，使能在线调试功能。这一功能允许结合 MPLAB IDE 进行一些简单的调试。通过 PGECx 和 PGEDx 引脚控制调试功能。

要使用器件的在线调试功能，在设计中必须实现至 MCLR、VDD、VSS，及 PGECx/PGEDx 引脚对的 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 个字节和两个 I/O 引脚。

27.0 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

27.1 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

27.2 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

27.3 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

27.4 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

27.5 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

27.6 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。

27.7 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（RJ-11）或新型抗噪声、高速低压差分信号（LVDS）互连电缆（CAT5）与目标板相连。

可通过 MPLAB 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。

27.8 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（RJ-11）与目标板相连。MPLAB ICD 3 支持所有 MPLAB ICD 2 连接器。

27.9 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（In-Circuit Serial Programming™，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）。

27.10 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。

PIC24FV32KA304 系列

27.11 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、 Σ - Δ ADC、流速传感器，等等。

同时还提供入门工具包，其中包含体验指定器件功能所需的所有软硬件。通常提供单个应用以及调试功能，都包含在一块电路板上。

有关演示、开发和评估工具包的完整列表，请访问 Microchip 网站 (www.microchip.com)。

27.12 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

28.0 指令集汇总

注： 本章是PIC24F指令集架构的简要汇总，并不能作为详尽的参考资料使用。

PIC24F指令集与以前的PIC®MCU指令集相比，添加了许多增强功能，并保持了易于从以前的PICMCU指令集移植的特点。大部分指令只占用一个程序存储字。只有3条指令需要两个程序存储单元。

每条单字指令都是一个24位字，由一个8位的操作码（指明指令类型）和一个或多个操作数（指定指令具体操作）组成。整个指令集具有高度的正交性，分为以下4种基本类型：

- 面向字或字节的操作类指令
- 面向位的操作类指令
- 立即数操作类指令
- 控制操作类指令

表28-1给出了在说明指令时要用到的通用符号。表28-2中的PIC24F指令集汇总列出了所有指令及每条指令影响的状态标志。

大部分面向字或字节的W寄存器指令（包含桶形移位寄存器指令）含有三个操作数：

- 第一个源操作数通常是不带地址修改符的Wb寄存器
- 第二个源操作数通常是带或不带地址修改符的Ws寄存器
- 结果的目标地址通常是带或不带地址修改符的Wd寄存器

而面向字或字节的文件寄存器指令具有两个操作数：

- 文件寄存器（由f值指定）
- 目标寄存器（可以是文件寄存器f也可以是记作WREG的W0寄存器）

大部分面向位的操作类指令（包括简单循环移位/单次移位指令）含有两个操作数：

- W寄存器（带或不带地址修改符）或文件寄存器（由Ws或f的值指定）
- W寄存器或文件寄存器中的位（由立即数直接指定或由Wb寄存器中的内容间接指定）

涉及数据传送的立即数指令可能使用以下操作数：

- 将被装载到W寄存器或文件寄存器的立即数（由k的值指定）
- 将要装载立即数的W寄存器或文件寄存器（由Wb或f指定）

而涉及算术或逻辑运算的立即数指令使用以下操作数：

- 第一个源操作数是不带地址修改符的Wb寄存器
- 第二个源操作数是立即数
- 结果的目标寄存器（只有在与第一个源操作数不同的情况下）通常为带或不带地址修改符的Wd寄存器

控制操作类指令可使用以下操作数：

- 程序存储器地址
- 表读和表写指令的模式

除某些双字指令外所有指令都是单字指令。双字指令中所有必需的信息都在这48位中。第二个字的高8位全为0。如果第二个字作为一条指令（由自身）执行，它将会被当作NOP指令执行。

除非条件测试结果为true或者指令执行后改变了程序计数器（PC）的值，否则执行大多数单字指令都只需要一个指令周期。对于上述两种特殊情况，指令执行需要两个指令周期，第二个指令周期执行一条NOP指令。值得注意的特殊指令有：BRA（无条件/计算转移指令）、间接CALL/GOTO指令、所有表读和表写指令以及RETURN/RETFIE指令，这些指令都是单字指令，但执行起来需要2或3个指令周期。

某些涉及跳过下一条指令的指令，在执行跳过时需要两或三个指令周期，具体周期数取决于被跳过的指令是单字指令还是双字指令。此外需要传送两个字的指令需要两个周期。执行双字指令需要两个指令周期。

PIC24FV32KA304 系列

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“地址为 text 的单元”
{ }	可选字段或操作
<n:m>	寄存器位域
.b	字节模式选择
.d	双字模式选择
.S	影子寄存器选择
.w	字模式选择（默认情况）
bit4	4 位位选择字段（用于字寻址指令） $\in \{0...15\}$
C、DC、N、OV 和 Z	MCU 状态位：进位、半进位、负标志、溢出标志和全零标志
Expr	绝对地址、标号或表达式（由链接器解析）
f	文件寄存器地址 $\in \{0000h...1FFFh\}$
lit1	1 位无符号立即数 $\in \{0,1\}$
lit4	4 位无符号立即数 $\in \{0...15\}$
lit5	5 位无符号立即数 $\in \{0...31\}$
lit8	8 位无符号立即数 $\in \{0...255\}$
lit10	10 位无符号立即数，字节模式下， $\in \{0...255\}$ ，字模式下 $\in \{0...1023\}$
lit14	14 位无符号立即数 $\in \{0...16384\}$
lit16	16 位无符号立即数 $\in \{0...65535\}$
lit23	23 位无符号立即数 $\in \{0...8388608\}$ ；LSB 必须为 0
None	该字段不必有输入项，可以为空白
PC	程序计数器
Slit10	10 位有符号立即数 $\in \{-512...511\}$
Slit16	16 位有符号立即数 $\in \{-32768...32767\}$
Slit6	6 位有符号立即数 $\in \{-16...16\}$
Wb	基本 W 寄存器 $\in \{W0..W15\}$
Wd	目标 W 寄存器 $\in \{Wd, [Wd], [Wd++] , [Wd--], [++Wd], [--Wd] \}$
Wdo	目标 W 寄存器 $\in \{Wnd, [Wnd], [Wnd++] , [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb] \}$
Wm,Wn	被除数和除数工作寄存器对（直接寻址）
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 $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb] \}$

PIC24FV32KA304 系列

表 28-2: 指令集汇总

汇编指令 助记符	汇编语法	说明	字数	周期数	受影响的状态标志
ADD	ADD f	$f = f + WREG$	1	1	C、DC、N、OV 和 Z
	ADD f, WREG	$WREG = f + WREG$	1	1	C、DC、N、OV 和 Z
	ADD #lit10, Wn	$Wd = lit10 + Wd$	1	1	C、DC、N、OV 和 Z
	ADD Wb, Ws, Wd	$Wd = Wb + Ws$	1	1	C、DC、N、OV 和 Z
	ADD Wb, #lit5, Wd	$Wd = Wb + lit5$	1	1	C、DC、N、OV 和 Z
ADDC	ADDC f	$f = f + WREG + (C)$	1	1	C、DC、N、OV 和 Z
	ADDC f, WREG	$WREG = f + WREG + (C)$	1	1	C、DC、N、OV 和 Z
	ADDC #lit10, Wn	$Wd = lit10 + Wd + (C)$	1	1	C、DC、N、OV 和 Z
	ADDC Wb, Ws, Wd	$Wd = Wb + Ws + (C)$	1	1	C、DC、N、OV 和 Z
	ADDC Wb, #lit5, Wd	$Wd = Wb + lit5 + (C)$	1	1	C、DC、N、OV 和 Z
AND	AND f	$f = f .AND. WREG$	1	1	N 和 Z
	AND f, WREG	$WREG = f .AND. WREG$	1	1	N 和 Z
	AND #lit10, Wn	$Wd = lit10 .AND. Wd$	1	1	N 和 Z
	AND Wb, Ws, Wd	$Wd = Wb .AND. Ws$	1	1	N 和 Z
	AND Wb, #lit5, Wd	$Wd = Wb .AND. lit5$	1	1	N 和 Z
ASR	ASR f	f = 算术右移 f	1	1	C、N、OV 和 Z
	ASR f, WREG	WREG = 算术右移 f	1	1	C、N、OV 和 Z
	ASR Ws, Wd	Wd = 算术右移 Ws	1	1	C、N、OV 和 Z
	ASR Wb, Wns, Wnd	Wnd = 将 Wb 算术右移 Wns 位	1	1	N 和 Z
	ASR Wb, #lit5, Wnd	Wnd = 将 Wb 算术右移 lit5 位	1	1	N 和 Z
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
BRA	BRA C, Expr	如果进位位为 1 则转移	1	1 (2)	无
	BRA GE, Expr	如果大于或等于则转移	1	1 (2)	无
	BRA GEU, Expr	如果无符号大于或等于则转移	1	1 (2)	无
	BRA GT, Expr	如果大于则转移	1	1 (2)	无
	BRA GTU, Expr	如果无符号大于则转移	1	1 (2)	无
	BRA LE, Expr	如果小于或等于则转移	1	1 (2)	无
	BRA LEU, Expr	如果无符号小于或等于则转移	1	1 (2)	无
	BRA LT, Expr	如果小于则转移	1	1 (2)	无
	BRA LTU, Expr	如果无符号小于则转移	1	1 (2)	无
	BRA N, Expr	如果为负则转移	1	1 (2)	无
	BRA NC, Expr	如果进位位为 0 则转移	1	1 (2)	无
	BRA NN, Expr	如果非负则转移	1	1 (2)	无
	BRA NOV, Expr	如果未溢出则转移	1	1 (2)	无
	BRA NZ, Expr	如果非零则转移	1	1 (2)	无
	BRA OV, Expr	如果溢出则转移	1	1 (2)	无
	BRA Expr	无条件转移	1	2	无
	BRA Z, Expr	如果为零则转移	1	1 (2)	无
	BRA Wn	计算转移	1	2	无
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
BSW	BSW.C Ws, Wb	将 C 位内容写入 Ws<Wb>	1	1	无
	BSW.Z Ws, Wb	将 Z 位内容写入 Ws<Wb>	1	1	无
BTG	BTG f, #bit4	将 f 中的某位取反	1	1	无
	BTG Ws, #bit4	将 Ws 中的某位取反	1	1	无
BTSC	BTSC f, #bit4	对 f 中的指定位进行检测, 如果为零则跳过	1	1 (2 或 3)	无
	BTSC Ws, #bit4	对 Ws 中的指定位进行检测, 如果为零则跳过	1	1 (2 或 3)	无

PIC24FV32KA304 系列

表 28-2: 指令集汇总 (续)

汇编指令 助记符	汇编语法	说明	字数	周期数	受影响的状态标志
BTSS	BTSS $f, \#bit4$	对 f 中的指定位进行检测, 如果为 1 则跳过	1	1 (2 或 3)	无
	BTSS $Ws, \#bit4$	对 Ws 中的指定位进行检测, 如果为 1 则跳过	1	1 (2 或 3)	无
BTST	BTST $f, \#bit4$	对 f 中的指定位进行检测	1	1	Z
	BTST.C $Ws, \#bit4$	对 Ws 中的指定位进行检测, 并将结果存储到进位标志位 C 中	1	1	C
	BTST.Z $Ws, \#bit4$	对 Ws 中的指定位进行检测, 并将结果存储到全零标志位 Z 中	1	1	Z
	BTST.C Ws, Wb	对 $Ws<Wb>$ 位进行检测, 并将结果存储到进位标志位 C 中	1	1	C
	BTST.Z Ws, Wb	对 $Ws<Wb>$ 位进行检测, 并将结果存储到全零标志位 Z 中	1	1	Z
BTSTS	BTSTS $f, \#bit4$	对 f 中的指定位进行检测, 并将该位置 1	1	1	Z
	BTSTS.C $Ws, \#bit4$	对 Ws 中的指定位进行检测, 并将结果存储到进位标志位 C 中, 然后将 Ws 中的该位置 1	1	1	C
	BTSTS.Z $Ws, \#bit4$	对 Ws 中的指定位进行检测, 并将结果存储到全零标志位 Z 中, 然后将 Ws 中的该位置 1	1	1	Z
CALL	CALL $lit23$	调用子程序	2	2	无
	CALL Wn	间接调用子程序	1	2	无
CLR	CLR f	$f = 0x0000$	1	1	无
	CLR WREG	WREG = 0x0000	1	1	无
	CLR Ws	$Ws = 0x0000$	1	1	无
CLRWDT	CLRWDT	将看门狗定时器清零	1	1	WDTO 和 SLEEP
COM	COM f	$f = \bar{f}$	1	1	N 和 Z
	COM $f, WREG$	WREG = $\bar{f}$	1	1	N 和 Z
	COM Ws, Wd	$Wd = \overline{Ws}$	1	1	N 和 Z
CP	CP f	比较 f 和 WREG	1	1	C、DC、N、OV 和 Z
	CP $Wb, \#lit5$	比较 Wb 和 $lit5$	1	1	C、DC、N、OV 和 Z
	CP Wb, Ws	比较 Wb 和 Ws ($Wb - Ws$)	1	1	C、DC、N、OV 和 Z
CP0	CP0 f	比较 f 和 0x0000	1	1	C、DC、N、OV 和 Z
	CP0 Ws	比较 Ws 和 0x0000	1	1	C、DC、N、OV 和 Z
CPB	CPB f	比较 f 和 WREG (通过带借位减法实现)	1	1	C、DC、N、OV 和 Z
	CPB $Wb, \#lit5$	比较 Wb 和 $lit5$ (通过带借位减法实现)	1	1	C、DC、N、OV 和 Z
	CPB Wb, Ws	比较 Wb 和 Ws (通过带借位减法实现) ($Wb - Ws - C$)	1	1	C、DC、N、OV 和 Z
CPSEQ	CPSEQ Wb, Wn	比较 Wb 和 Wn , 如果相等则跳过	1	1 (2 或 3)	无
CPSGT	CPSGT Wb, Wn	比较 Wb 和 Wn , 如果大于则跳过	1	1 (2 或 3)	无
CPSLT	CPSLT Wb, Wn	比较 Wb 和 Wn , 如果小于则跳过	1	1 (2 或 3)	无
CPSNE	CPSNE Wb, Wn	比较 Wb 和 Wn , 如果不相等则跳过	1	1 (2 或 3)	无
DAW	DAW Wn	$Wn =$ 对 Wn 进行十进制调整	1	1	C
DEC	DEC f	$f = f - 1$	1	1	C、DC、N、OV 和 Z
	DEC $f, WREG$	WREG = $f - 1$	1	1	C、DC、N、OV 和 Z
	DEC Ws, Wd	$Wd = Ws - 1$	1	1	C、DC、N、OV 和 Z
DEC2	DEC2 f	$f = f - 2$	1	1	C、DC、N、OV 和 Z
	DEC2 $f, WREG$	WREG = $f - 2$	1	1	C、DC、N、OV 和 Z
	DEC2 Ws, Wd	$Wd = Ws - 2$	1	1	C、DC、N、OV 和 Z
DISI	DISI $\#lit14$	在 k 个指令周期内禁止中断	1	1	无
DIV	DIV.SW Wm, Wn	有符号 16/16 位整数除法	1	18	N、Z、C 和 OV
	DIV.SD Wm, Wn	有符号 32/16 位整数除法	1	18	N、Z、C 和 OV
	DIV.UW Wm, Wn	无符号 16/16 位整数除法	1	18	N、Z、C 和 OV
	DIV.UD Wm, Wn	无符号 32/16 位整数除法	1	18	N、Z、C 和 OV
EXCH	EXCH Wns, Wnd	将 Wns 和 Wnd 交换	1	1	无
FF1L	FF1L Ws, Wnd	从左边 (MSb) 查找第一个 1	1	1	C
FF1R	FF1R Ws, Wnd	从右边 (LSb) 查找第一个 1	1	1	C

PIC24FV32KA304 系列

表 28-2: 指令集汇总 (续)

汇编指令 助记符	汇编语法	说明	字数	周期数	受影响的状态标志
GOTO	GOTO Expr	转移到地址	2	2	无
	GOTO Wn	间接转移到地址	1	2	无
INC	INC f	$f = f + 1$	1	1	C、DC、N、OV 和 Z
	INC f, WREG	$WREG = f + 1$	1	1	C、DC、N、OV 和 Z
	INC Ws, Wd	$Wd = Ws + 1$	1	1	C、DC、N、OV 和 Z
INC2	INC2 f	$f = f + 2$	1	1	C、DC、N、OV 和 Z
	INC2 f, WREG	$WREG = f + 2$	1	1	C、DC、N、OV 和 Z
	INC2 Ws, Wd	$Wd = Ws + 2$	1	1	C、DC、N、OV 和 Z
IOR	IOR f	$f = f .IOR. WREG$	1	1	N 和 Z
	IOR f, WREG	$WREG = f .IOR. WREG$	1	1	N 和 Z
	IOR #lit10, Wn	$Wd = lit10 .IOR. Wd$	1	1	N 和 Z
	IOR Wb, Ws, Wd	$Wd = Wb .IOR. Ws$	1	1	N 和 Z
	IOR Wb, #lit5, Wd	$Wd = Wb .IOR. lit5$	1	1	N 和 Z
LNK	LNK #lit14	链接帧指针	1	1	无
LSR	LSR f	$f = \text{逻辑右移 } f$	1	1	C、N、OV 和 Z
	LSR f, WREG	$WREG = \text{逻辑右移 } f$	1	1	C、N、OV 和 Z
	LSR Ws, Wd	$Wd = \text{逻辑右移 } Ws$	1	1	C、N、OV 和 Z
	LSR Wb, Wns, Wnd	$Wnd = \text{将 } Wb \text{ 逻辑右移 } Wns \text{ 位}$	1	1	N 和 Z
	LSR Wb, #lit5, Wnd	$Wnd = \text{将 } Wb \text{ 逻辑右移 } lit5 \text{ 位}$	1	1	N 和 Z
MOV	MOV f, Wn	将 f 中的内容送入 Wn	1	1	无
	MOV [Wns+Slit10], Wnd	将 [Wns+Slit10] 中的内容送入 Wnd	1	1	无
	MOV f	将 f 中的内容送入目标寄存器	1	1	N 和 Z
	MOV f, WREG	将 f 中的内容送入 WREG	1	1	N 和 Z
	MOV #lit16, Wn	将 16 位立即数送入 Wn	1	1	无
	MOV.b #lit8, Wn	将 8 位立即数送入 Wn	1	1	无
	MOV Wn, f	将 Wn 中的内容送入 f	1	1	无
	MOV Wns, [Wns+Slit10]	将 Wns 中的内容送入 [Wns+Slit10]	1	1	无
	MOV Wso, Wdo	将 Ws 中的内容送入 Wd	1	1	无
	MOV WREG, f	将 WREG 中的内容送入 f	1	1	N 和 Z
	MOV.D Wns, Wd	将 W(ns):W(ns + 1) 中的双字内容送入 Wd	1	2	无
	MOV.D Ws, Wnd	将 Ws 中的双字内容送入 W(nd + 1):W(nd)	1	2	无
MUL	MUL.SS Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{Signed}(Wb) * \text{Signed}(Ws)$	1	1	无
	MUL.SU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{Signed}(Wb) * \text{Unsigned}(Ws)$	1	1	无
	MUL.US Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{Unsigned}(Wb) * \text{Signed}(Ws)$	1	1	无
	MUL.UU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{Unsigned}(Wb) * \text{Unsigned}(Ws)$	1	1	无
	MUL.SU Wb, #lit5, Wnd	$\{Wnd+1, Wnd\} = \text{Signed}(Wb) * \text{Unsigned}(lit5)$	1	1	无
	MUL.UU Wb, #lit5, Wnd	$\{Wnd+1, Wnd\} = \text{Unsigned}(Wb) * \text{Unsigned}(lit5)$	1	1	无
	MUL f	$W3:W2 = f * WREG$	1	1	无
NEG	NEG f	$f = \bar{f} + 1$	1	1	C、DC、N、OV 和 Z
	NEG f, WREG	$WREG = \bar{f} + 1$	1	1	C、DC、N、OV 和 Z
	NEG Ws, Wd	$Wd = \overline{Ws} + 1$	1	1	C、DC、N、OV 和 Z
NOP	NOP	空操作	1	1	无
	NOPR	空操作	1	1	无
POP	POP f	从栈顶 (TOS) 弹出 f 寄存器的内容	1	1	无
	POP Wdo	将栈顶 (TOS) 的内容弹出到 Wdo 中	1	1	无
	POP.D Wnd	将栈顶 (TOS) 的内容弹出到 W(nd):W(nd+1) 中	1	2	无
	POP.S	将影子寄存器的内容弹出到主寄存器	1	1	全部
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) 中的内容压入栈顶 (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

PIC24FV32KA304 系列

表 28-2: 指令集汇总 (续)

汇编指令 助记符	汇编语法	说明	字数	周期数	受影响的状态标志
PWRSV	PWRSV #lit1	进入休眠或空闲模式	1	1	WDTO 和 SLEEP
RCALL	RCALL Expr	相对调用	1	2	无
	RCALL Wn	计算调用	1	2	无
REPEAT	REPEAT #lit14	将下一条指令重复执行 lit14 + 1 次	1	1	无
	REPEAT Wn	将下一条指令重复执行 (Wn) + 1 次	1	1	无
RESET	RESET	用软件使器件复位	1	1	无
RETFIE	RETFIE	从中断返回	1	3 (2)	无
RETLW	RETLW #lit10, Wn	返回并将立即数存入 Wn	1	3 (2)	无
RETURN	RETURN	从子程序返回	1	3 (2)	无
RLC	RLC f	f = 对 f 执行带进位的循环左移	1	1	C、N 和 Z
	RLC f, WREG	WREG = 对 f 执行带进位的循环左移	1	1	C、N 和 Z
	RLC Ws, Wd	Wd = 对 Ws 执行带进位的循环左移	1	1	C、N 和 Z
RLNC	RLNC f	f = 循环左移 f (不带进位)	1	1	N 和 Z
	RLNC f, WREG	WREG = 循环左移 f (不带进位)	1	1	N 和 Z
	RLNC Ws, Wd	Wd = 循环左移 Ws (不带进位)	1	1	N 和 Z
RRC	RRC f	f = 对 f 执行带进位的循环右移	1	1	C、N 和 Z
	RRC f, WREG	WREG = 对 f 执行带进位的循环右移	1	1	C、N 和 Z
	RRC Ws, Wd	Wd = 对 Ws 执行带进位的循环右移	1	1	C、N 和 Z
RRNC	RRNC f	f = 循环右移 f (不带进位)	1	1	N 和 Z
	RRNC f, WREG	WREG = 循环右移 f (不带进位)	1	1	N 和 Z
	RRNC Ws, Wd	Wd = 循环右移 Ws (不带进位)	1	1	N 和 Z
SE	SE Ws, Wnd	Wnd = 对 Ws 进行符号扩展	1	1	C、N 和 Z
SETM	SETM f	f = FFFFh	1	1	无
	SETM WREG	WREG = FFFFh	1	1	无
	SETM Ws	Ws = FFFFh	1	1	无
SL	SL f	f = 左移 f	1	1	C、N、OV 和 Z
	SL f, WREG	WREG = 左移 f	1	1	C、N、OV 和 Z
	SL Ws, Wd	Wd = 左移 Ws	1	1	C、N、OV 和 Z
	SL Wb, Wns, Wnd	Wnd = 将 Wb 左移 Wns 位	1	1	N 和 Z
	SL Wb, #lit5, Wnd	Wnd = 将 Wb 左移 lit5 位	1	1	N 和 Z
SUB	SUB f	f = f - WREG	1	1	C、DC、N、OV 和 Z
	SUB f, WREG	WREG = f - WREG	1	1	C、DC、N、OV 和 Z
	SUB #lit10, Wn	Wn = Wn - lit10	1	1	C、DC、N、OV 和 Z
	SUB Wb, Ws, Wd	Wd = Wb - Ws	1	1	C、DC、N、OV 和 Z
	SUB Wb, #lit5, Wd	Wd = Wb - lit5	1	1	C、DC、N、OV 和 Z
SUBB	SUBB f	f = f - WREG - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
	SUBB f, WREG	WREG = f - WREG - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
	SUBB #lit10, Wn	Wn = Wn - lit10 - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
	SUBB Wb, Ws, Wd	Wd = Wb - Ws - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
	SUBB Wb, #lit5, Wd	Wd = Wb - lit5 - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
SUBR	SUBR f	f = WREG - f	1	1	C、DC、N、OV 和 Z
	SUBR f, WREG	WREG = WREG - f	1	1	C、DC、N、OV 和 Z
	SUBR Wb, Ws, Wd	Wd = Ws - Wb	1	1	C、DC、N、OV 和 Z
	SUBR Wb, #lit5, Wd	Wd = lit5 - Wb	1	1	C、DC、N、OV 和 Z
SUBBR	SUBBR f	f = WREG - f - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
	SUBBR f, WREG	WREG = WREG - f - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
	SUBBR Wb, Ws, Wd	Wd = Ws - Wb - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
	SUBBR Wb, #lit5, Wd	Wd = lit5 - Wb - ($\overline{C}$)	1	1	C、DC、N、OV 和 Z
SWAP	SWAP.b Wn	Wn = 将 Wn 的两个半字节相交换	1	1	无
	SWAP Wn	Wn = 将 Wn 的两个字节相交换	1	1	无
TBLRDH	TBLRDH Ws, Wd	将程序计数器的 <23:16> 读入 Wd<7:0>	1	2	无

表 28-2: 指令集汇总 (续)

汇编指令 助记符	汇编语法	说明	字数	周期数	受影响的状态标志
TBLRDL	TBLRDL Ws, Wd	将程序计数器的 <15:0> 读入 Wd	1	2	无
TBLWTH	TBLWTH Ws, Wd	将 Ws<7:0> 写入程序计数器的 <23:16>	1	2	无
TBLWTL	TBLWTL Ws, Wd	将 Ws 写入程序计数器的 <15:0>	1	2	无
ULNK	ULNK	释放堆栈帧	1	1	无
XOR	XOR f	f = f.XOR. WREG	1	1	N 和 Z
	XOR f, WREG	WREG = f.XOR. WREG	1	1	N 和 Z
	XOR #lit10, Wn	Wd = lit10.XOR. Wd	1	1	N 和 Z
	XOR Wb, Ws, Wd	Wd = Wb.XOR. Ws	1	1	N 和 Z
	XOR Wb, #lit5, Wd	Wd = Wb.XOR. lit5	1	1	N 和 Z
ZE	ZE Ws, Wnd	Wnd = 对 Ws 进行零扩展	1	1	C、N 和 Z

PIC24FV32KA304 系列

注:

29.0 电气特性

本章提供 PIC24FV32KA304 系列器件电气特性的概述。在文档后续版本中会添加其他信息。

下面列出了 PIC24FV32KA304 系列器件的绝对最大值。器件长时间工作在最大值条件下，其稳定性会受到影响。我们建议不要使器件在该规范规定的参数范围外工作。

绝对最大值 ^(†)

环境温度.....	-40°C 至 +135°C
存储温度.....	-65°C 至 +150°C
V _{DD} 引脚相对于 V _{SS} 的电压 (PIC24FVXXKA30X)	-0.3V 至 +6.5V
V _{DD} 相对于 V _{SS} 的电压 (PIC24FXXKA30X)	-0.3V 至 +4.5V
模拟数字组合引脚相对于 V _{SS} 的电压	-0.3V 至 (V _{DD} + 0.3V)
只能用作数字功能的引脚相对于 V _{SS} 的电压	-0.3V 至 (V _{DD} + 0.3V)
MCLR/V _{PP} 引脚相对于 V _{SS} 的电压	-0.3V 至 +9.0V
V _{SS} 引脚的最大输出电流	300 mA
V _{DD} 引脚的最大输入电流 ⁽¹⁾	250 mA
任一 I/O 引脚的最大输出灌电流	25 mA
任一 I/O 引脚的最大输出拉电流	25 mA
所有端口的最大灌电流	200 mA
所有端口的最大拉电流 ⁽¹⁾	200 mA

注 1: 允许的最大电流由器件的最大功耗决定 (见表 29-1)。

注：如果器件的工作条件超过上述“绝对最大值”，可能会对器件造成永久性损坏。上述值仅为运行条件极大值，我们建议不要使器件在该规范规定的范围以外运行。器件长时间工作在最大值条件下，其稳定性会受到影响。

PIC24FV32KA304 系列

29.1 直流特性

图 29-1: PIC24FV32KA304 系列器件电压 —— 频率关系图（工业级和扩展级）

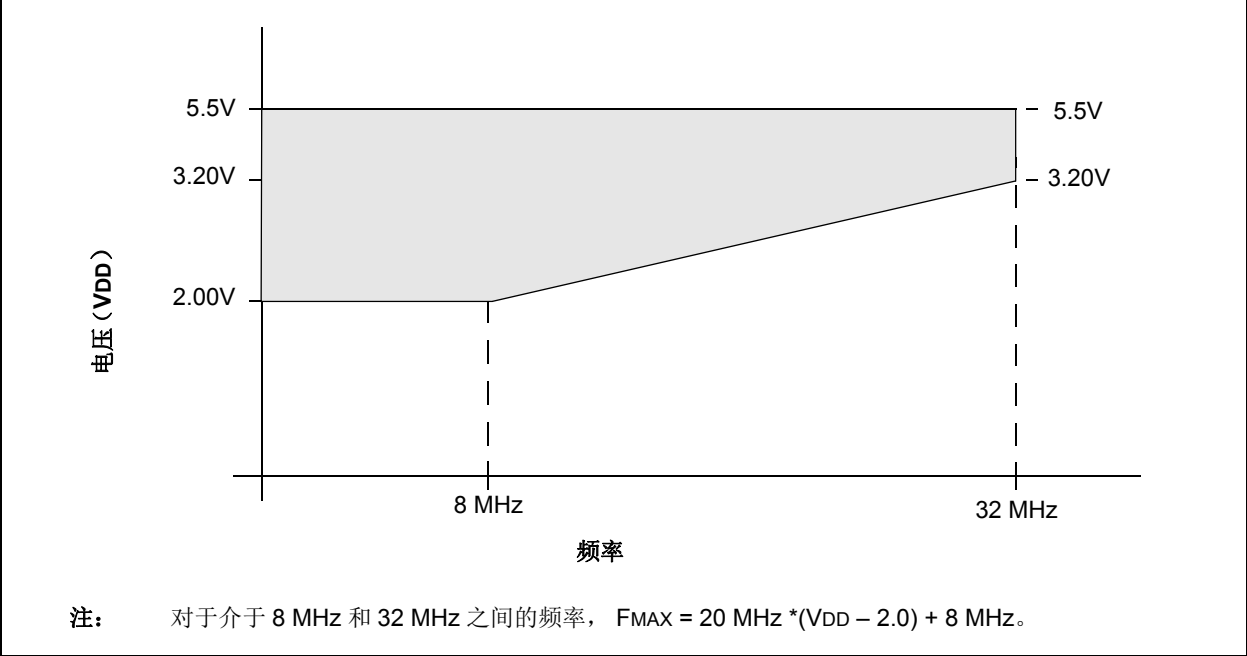
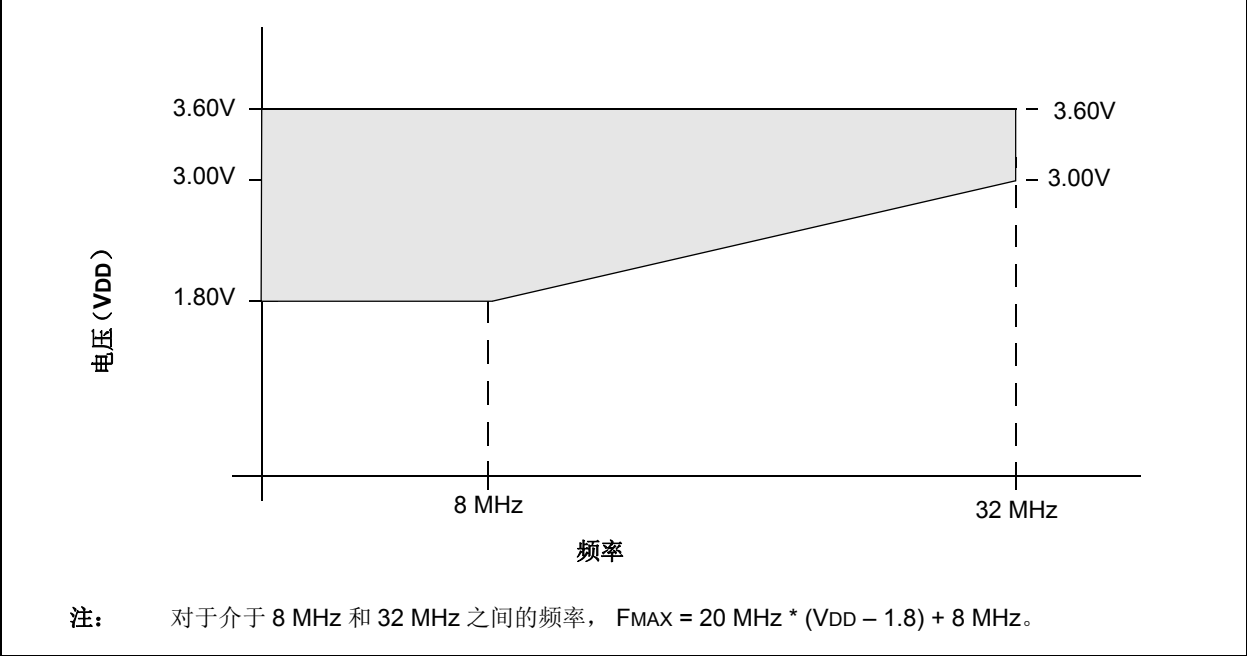


图 29-2: PIC24F32KA304 系列器件电压 —— 频率关系图（工业级和扩展级）



PIC24FV32KA304 系列

表 29-1: 热工作条件

额定值	符号	最小值	典型值	最大值	单位
工作结温范围	T _J	-40	—	+140	°C
工作环境温度范围	T _A	-40	—	+125	°C
功耗 内部芯片功耗: $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$ I/O 引脚功耗: $P_{I/O} = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	P _D	P _{INT} + P _{I/O}			W
允许的最大功耗	P _D MAX	(T _J - T _A)/θ _{JA}			W

表 29-2: 封装热阻特性

特性	符号	典型值	最大值	单位	注
封装热阻, 20 引脚 SPDIP	θ _{JA}	62.4	—	°C/W	1
封装热阻, 28 引脚 SPDIP	θ _{JA}	60	—	°C/W	1
封装热阻, 20 引脚 SSOP	θ _{JA}	108	—	°C/W	1
封装热阻, 28 引脚 SSOP	θ _{JA}	71	—	°C/W	1
封装热阻, 20 引脚 SOIC	θ _{JA}	75	—	°C/W	1
封装热阻, 28 引脚 SOIC	θ _{JA}	80.2	—	°C/W	1
封装热阻, 28 引脚 QFN	θ _{JA}	32	—	°C/W	1
封装热阻, 44 引脚 QFN	θ _{JA}	29	—	°C/W	1
封装热阻, 48 引脚 UQFN	θ _{JA}	—	—	°C/W	1

注 1: 通过封装模拟获得结点与环境的热阻值 (θ_{JA})。

表 29-3: 直流特性: 温度和电压规范

直流特性			标准工作条件: 1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX) 工作温度 -40°C ≤ T _A ≤ +85°C (工业级) -40°C ≤ T _A ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
DC10	V _{DD}	电源电压	1.8	—	3.6	V	对于 F 器件
			2.0	—	5.5	V	对于 FV 器件
DC12	V _{DR}	RAM 数据保持电压 ⁽²⁾	1.5	—	—	V	对于 F 器件
			1.7	—	—	V	对于 FV 器件
DC16	V _{POR}	确保内部上电复位信号的 V _{DD} 起始电压	V _{SS}	—	0.7	V	
DC17	SV _{DD}	确保内部上电复位信号的 V _{DD} 上升速率	0.05	—	—	V/ms	0.1s 内电压变化范围为 0-3.3V 60 ms 内电压变化范围为 0-2.5V

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数仅供设计参考, 未经测试。

2: 这是在不丢失 RAM 数据的前提下, V_{DD} 的下限值。

PIC24FV32KA304 系列

表 29-4: 高 / 低电压检测特性

标准工作条件:		1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)						
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
DC18	VHLVD	VDD 跳变时的 HLVD 电压	HLVDL<3:0> = 0000 ⁽²⁾	—	—	1.90	V	
			HLVDL<3:0> = 0001	1.86	—	2.13	V	
			HLVDL<3:0> = 0010	2.08	—	2.35	V	
			HLVDL<3:0> = 0011	2.22	—	2.53	V	
			HLVDL<3:0> = 0100	2.30	—	2.62	V	
			HLVDL<3:0> = 0101	2.49	—	2.84	V	
			HLVDL<3:0> = 0110	2.73	—	3.10	V	
			HLVDL<3:0> = 0111	2.86	—	3.25	V	
			HLVDL<3:0> = 1000	3.00	—	3.41	V	
			HLVDL<3:0> = 1001	3.16	—	3.59	V	
			HLVDL<3:0> = 1010 ⁽¹⁾	3.33	—	3.79	V	
			HLVDL<3:0> = 1011 ⁽¹⁾	3.53	—	4.01	V	
			HLVDL<3:0> = 1100 ⁽¹⁾	3.74	—	4.26	V	
			HLVDL<3:0> = 1101 ⁽¹⁾	4.00	—	4.55	V	
			HLVDL<3:0> = 1110 ⁽¹⁾	4.28	—	4.87	V	

注 1: 这些跳变点不应用在 PIC24FXXKA30X 器件上。

2: 该跳变点不应用在 PIC24FVXXKA30X 器件上。

表 29-5: BOR 跳变点

标准工作条件:		1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)						
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
DC15		BOR 迟滞		—	5	—	mV	
DC19		VDD 跳变时的 BOR 电压	BORV<1:0> = 00	—	—	—	—	对于 LPBOR 和 DSBOR 有效, (注 1)
			BORV<1:0> = 01	2.90	3	3.38	V	
			BORV<1:0> = 10	2.53	2.7	3.07	V	
			BORV<1:0> = 11	1.75	1.85	2.05	V	(注 2)
			BORV<1:0> = 11	1.95	2.05	2.16	V	(注 3)

注 1: LPBOR 重新激活 POR 电路, 但不会引起 BOR。

2: 对于 PIC24F (3.3V) 器件有效。

3: 对于 PIC24FV (5V) 器件有效。

PIC24FV32KA304 系列

表 29-6: 直流特性: 工作电流 (IDD)

直流特性		标准工作条件:				1.8V 至 3.6V (PIC24F32KA3XX)	
						2.0V 至 5.5V (PIC24FV32KA3XX)	
		工作温度: -40°C ≤ Ta ≤ +85°C (工业级)					
		-40°C ≤ Ta ≤ +125°C (扩展级)					
参数编号	器件	典型值	最大值	单位	条件		
IDD 电流							
D20	PIC24FV32KA3XX	269	450	μA	2.0V	0.5 MIPS, Fosc = 1 MHz ⁽¹⁾	
		465	830	μA	5.0V		
	PIC24F32KA3XX	200	330	μA	1.8V		
		410	750	μA	3.3V		
DC22	PIC24FV32KA3XX	490	—	μA	2.0V	1 MIPS, Fosc = 2 MHz ⁽¹⁾	
		880	—	μA	5.0V		
	PIC24F32KA3XX	407	—	μA	1.8V		
		800	—	μA	3.3V		
DC24	PIC24FV32KA3XX	13.0	20.0	mA	5.0V	16 MIPS, Fosc = 32 MHz ⁽¹⁾	
	PIC24F32KA3XX	12.0	18.0	mA	3.3V		
DC26	PIC24FV32KA3XX	2.0	—	mA	2.0V	FRC (4 MIPS) , Fosc = 8 MHz	
		3.5	—	mA	5.0V		
	PIC24F32KA3XX	1.80	—	mA	1.8V		
		3.40	—	mA	3.3V		
DC30	PIC24FV32KA3XX	48.0	250	μA	2.0V	LPRC (15.5 KIPS) , Fosc = 31 kHz	
		75.0	450	μA	5.0V		
	PIC24F32KA3XX	8.1	28	μA	1.8V		
		13.50	150	μA	3.3V		

图注: 无阴影行代表 PIC24F32KA3XX 器件, 而阴影行代表 PIC24FV32KA3XX 器件。

注 1: 振荡器处于外部时钟模式 (FOSCSEL<2:0> = 010, FOSC<1:0> = 00)。

PIC24FV32KA304 系列

表 29-7：直流特性：空闲电流（IDLE）

直流特性		标准工作条件： 1.8V 至 3.6V （PIC24F32KA3XX） 2.0V 至 5.5V （PIC24FV32KA3XX） 工作温度：-40℃ ≤ Ta ≤ +85℃ （工业级） -40℃ ≤ Ta ≤ +125℃ （扩展级）				
参数编号	器件	典型值	最大值	单位	条件	
空闲电流（I _{IDLE} ）						
DC40	PIC24FV32KA3XX	120	200	μA	2.0V	0.5 MIPS, Fosc = 1 MHz ⁽¹⁾
		160	430	μA	5.0V	
	PIC24F32KA3XX	50	100	μA	1.8V	
		90	370	μA	3.3V	
DC42	PIC24FV32KA3XX	165	—	μA	2.0V	1 MIPS, Fosc = 2 MHz ⁽¹⁾
		260	—	μA	5.0V	
	PIC24F32KA3XX	95	—	μA	1.8V	
		180	—	μA	3.3V	
DC44	PIC24FV32KA3XX	3.1	6.5	mA	5.0V	16 MIPS, Fosc = 32 MHz ⁽¹⁾
	PIC24F32KA3XX	2.9	6.0	mA	3.3V	
DC46	PIC24FV32KA3XX	0.65	—	mA	2.0V	FRC（4 MIPS）， Fosc = 8 MHz
		1.0	—	mA	5.0V	
	PIC24F32KA3XX	0.55	—	mA	1.8V	
		1.0	—	mA	3.3V	
DC50	PIC24FV32KA3XX	60	200	μA	2.0V	LPRC（15.5 KIPS）， Fosc = 31 kHz
		70	350	μA	5.0V	
	PIC24F32KA3XX	2.2	18	μA	1.8V	
		4.0	60	μA	3.3V	

图注：无阴影行代表 PIC24F32KA3XX 器件，而阴影行代表 PIC24FV32KA3XX 器件。

注 1：振荡器处于外部时钟模式（FOSCSEL<2:0> = 010，FOSC<1:0> = 00）。

PIC24FV32KA304 系列

表 29-8: 直流特性: 掉电电流 (IPD)

直流特性		标准工作条件:					
		1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)					
		工作温度					
		-40°C ≤ Ta ≤ +85°C (工业级) -40°C ≤ Ta ≤ +125°C (扩展级)					
参数编号	器件	典型值 ⁽¹⁾	最大值	单位	条件		
掉电电流 (IPD)							
DC60	PIC24FV32KA3XX	6.0	—	μA	-40°C	2.0V	休眠模式 ⁽²⁾
			8.0		+25°C		
			8.5		+60°C		
			9.0		+85°C		
			15		+125°C		
		6.0	—	μA	-40°C	5.0V	
			8.0		+25°C		
			9.0		+60°C		
			10.0		+85°C		
			15		+125°C		
	PIC24F32KA3XX	0.025	—	μA	-40°C	1.8V	
			0.80		+25°C		
			1.5		+60°C		
			2.0		+85°C		
7.5	+125°C						
0.040	—	μA	-40°C	3.3V			
	1.0		+25°C				
	2.0		+60°C				
	3.0		+85°C				
	7.5		+125°C				
DC61	PIC24FV32KA3XX	0.25	—	μA	-40°C	2.0V	低电压休眠模式 ⁽²⁾
		0.35	3.0	μA	+85°C	5.0V	
		—	7.5	μA	+125°C	5.0V	
DC70	PIC24FV32KA3XX	0.03	—	μA	-40°C	2.0V	深度休眠模式
		0.10	2.0	μA	+85°C	5.0V	
		—	6.0	μA	+125°C	5.0V	
	PIC24F32KA3XX	0.02	—	μA	-40°C	1.8V	
		0.08	1.2	μA	+85°C	3.3V	
		—	1.2	μA	+125°C	3.3V	

图注: 无阴影行代表 PIC24F32KA3XX 器件, 而阴影行代表 PIC24FV32KA3XX 器件。

- 注 1: 除非另外声明, 否则“典型值”栏中的数据都是在以下条件下给出的: 3.3V、25°C (适用于 PIC24F32KA3XX); 5.0V、+25°C (适用于 PIC24FV32KA3XX)。这些参数仅供设计参考, 未经测试。
- 2: 基本 IPD 是在关闭所有外设和时钟的条件下测得的。所有 I/O 引脚均配置为输出且被设置为低电平, PMSLP 被设置为 0, 且 WDT 等外设均关闭。
- 3: Δ 电流为模块使能时额外消耗的电流。器件工作时外设模块的电流消耗是这一电流与基本 IPD 电流之和。
- 4: 电流仅适用于休眠模式。
- 5: 电流适用于休眠和深度休眠模式。
- 6: 电流仅适用于深度休眠模式。

PIC24FV32KA304 系列

表 29-8: 直流特性: 掉电电流 (IPD) (续)

直流特性		标准工作条件:						
		1.8V 至 3.6V (PIC24F32KA3XX)						
		2.0V 至 5.5V (PIC24FV32KA3XX)						
		工作温度						
		-40°C ≤ TA ≤ +85°C (工业级)						
		-40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	器件	典型值 ⁽¹⁾	最大值	单位	条件			
模块差分电流 (ΔIPD) ⁽³⁾								
DC71	PIC24FV32KA3XX	0.50	—	μA	-40°C	2.0V	看门狗定时器电流: ΔI _{WDT} ⁽⁴⁾	
		0.70	1.5	μA	+85°C	5.0V		
		—	1.5	μA	+125°C	5.0V		
	PIC24F32KA3XX	0.50	—	μA	-40°C	1.8V		
		0.70	1.5	μA	+85°C	3.3V		
		—	1.5	μA	+125°C	3.3V		
DC72	PIC24FV32KA3XX	0.80	—	μA	-40°C	2.0V		与 RTCC、DSWDT 或 Timer1 配合工作的 32 kHz 晶振: ΔI _{SOSC} (SOSCSEL = 0) ⁽⁵⁾
		1.50	2.0	μA	+85°C	5.0V		
		—	2.0	μA	+125°C	5.0V		
	PIC24F32KA3XX	0.70	—	μA	-40°C	1.8V		
		1.0	1.5	μA	+85°C	3.3V		
		—	1.5	μA	+125°C	3.3V		
DC75	PIC24FV32KA3XX	5.4	—	μA	-40°C	2.0V	ΔI _{HLVD} ⁽⁴⁾	
		8.1	14.0	μA	+85°C	5.0V		
		—	14.0	μA	+125°C	5.0V		
	PIC24F32KA3XX	4.9	—	μA	-40°C	1.8V		
		7.5	14.0	μA	+85°C	3.3V		
		—	14.0	μA	+125°C	3.3V		
DC76	PIC24FV32KA3XX	5.6	—	μA	-40°C	2.0V		ΔI _{BOR} ⁽⁴⁾
		6.5	11.2	μA	-40°C	5.0V		
		—	11.2	μA	+125°C	5.0V		
	PIC24F32KA3XX	5.6	—	μA	-40°C	1.8V		
		6.0	11.2	μA	+85°C	3.3V		
		—	11.2	μA	+125°C	3.3V		

- 图注: 无阴影行代表 PIC24F32KA3XX 器件, 而阴影行代表 PIC24FV32KA3XX 器件。
- 注 1: 除非另外声明, 否则“典型值”栏中的数据都是在以下条件下给出的: 3.3V、25°C (适用于 PIC24F32KA3XX); 5.0V、+25°C (适用于 PIC24FV32KA3XX)。这些参数仅供设计参考, 未经测试。
- 2: 基本 IPD 是在关闭所有外设和时钟的条件下测得的。所有 I/O 引脚均配置为输出且被设置为低电平, PMSLP 被设置为 0, 且 WDT 等外设均关闭。
- 3: Δ 电流为模块使能时额外消耗的电流。器件工作时外设模块的电流消耗是这一电流与基本 IPD 电流之和。
- 4: 电流仅适用于休眠模式。
- 5: 电流适用于休眠和深度休眠模式。
- 6: 电流仅适用于深度休眠模式。

PIC24FV32KA304 系列

表 29-8: 直流特性: 掉电电流 (IPD) (续)

直流特性		标准工作条件:					
		1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)					
		工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	器件	典型值 ⁽¹⁾	最大值	单位	条件		
模块差分电流 (ΔIPD) ⁽³⁾							
DC78	PIC24FV32KA3XX	0.03	—	μA	-40°C	2.0V	深度休眠 BOR: ΔILPBOR ⁽⁵⁾
		0.05	0.20	μA	+85°C	5.0V	
		—	0.30	μA	+125°C	5.0V	
	PIC24F32KA3XX	0.03	—	μA	-40°C	1.8V	
		0.05	0.20	μA	+85°C	3.3V	
		—	0.30	μA	+125°C	3.3V	
DC80	PIC24FV32KA3XX	0.20	—	μA	-40°C	2.0V	深度休眠 WDT: ΔIDSWDT (LPRC) ⁽⁶⁾
		0.70	1.5	μA	+85°C	5.0V	
		—	1.5	μA	+125°C	5.0V	
	PIC24F32KA3XX	0.20	—	μA	-40°C	1.8V	
		0.35	0.8	μA	+85°C	3.3V	
		—	1.5	μA	+125°C	3.3V	

图注: 无阴影行代表 PIC24F32KA3XX 器件, 而阴影行代表 PIC24FV32KA3XX 器件。

- 注 1: 除非另外声明, 否则“典型值”栏中的数据都是在以下条件下给出的: 3.3V、25°C (适用于 PIC24F32KA3XX); 5.0V、+25°C (适用于 PIC24FV32KA3XX)。这些参数仅供设计参考, 未经测试。
- 2: 基本 IPD 是在关闭所有外设和时钟的条件下测得的。所有 I/O 引脚均配置为输出且被设置为低电平, PMSLP 被设置为 0, 且 WDT 等外设均关闭。
- 3: Δ 电流为模块使能时额外消耗的电流。器件工作时外设模块的电流消耗是这一电流与基本 IPD 电流之和。
- 4: 电流仅适用于休眠模式。
- 5: 电流适用于休眠和深度休眠模式。
- 6: 电流仅适用于深度休眠模式。

PIC24FV32KA304 系列

表 29-9: 直流特性: I/O 引脚输入规范

直流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
DI10	V _{IL}	输入低电压 ⁽⁴⁾					
		I/O 引脚	V _{SS}	—	0.2 V _{DD}	V	
DI15		MCLR	V _{SS}	—	0.2 V _{DD}	V	
DI16		OSC1 (XT 模式)	V _{SS}	—	0.2 V _{DD}	V	
DI17		OSC1 (HS 模式)	V _{SS}	—	0.2 V _{DD}	V	
DI18		具有 I ² C™ 缓冲器的 I/O 引脚	V _{SS}	—	0.3 V _{DD}	V	禁止 SMBus
DI19		具有 SMBus 缓冲器的 I/O 引脚	V _{SS}	—	0.8	V	使能 SMBus
DI20	V _{IH}	输入高电压 ⁽⁴⁾					
		I/O 引脚:					
		带模拟功能	0.8 V _{DD}	—	V _{DD}	V	
		仅数字功能	0.8 V _{DD}	—	V _{DD}	V	
DI25		MCLR	0.8 V _{DD}	—	V _{DD}	V	
DI26		OSC1 (XT 模式)	0.7 V _{DD}	—	V _{DD}	V	
DI27		OSC1 (HS 模式)	0.7 V _{DD}	—	V _{DD}	V	
DI28		具有 I ² C 缓冲器的 I/O 引脚					
		带模拟功能	0.7 V _{DD}	—	V _{DD}	V	
		仅数字功能	0.7 V _{DD}	—	V _{DD}	V	
DI29		具有 SMBus 缓冲器的 I/O 引脚	2.1	—	V _{DD}	V	2.5V ≤ V _{PIN} ≤ V _{DD}
DI30	ICNPU	CNx 上拉电流	50	250	500	mA	V _{DD} = 3.3V, V _{PIN} = V _{SS}
DI50	I _{IL}	输入泄漏电流 ^(2,3)					
		I/O 端口	—	0.05	0.1	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态
DI55		MCLR	—	—	0.1	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD}
DI56		OSCI	—	—	5	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , XT 和 HS 模式

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数仅供设计参考, 未经测试。

2: MCLR 引脚上的泄漏电流主要取决于所施加的电压。指定的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: 负电流定义为自引脚流出的电流。

4: I/O 引脚缓冲器类型请参见表 1-3。

PIC24FV32KA304 系列

表 29-10: 直流特性: I/O 引脚输出规范

直流特性			标准工作条件:					
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)					
			工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件	
DO10	VOL	输出低电压 所有 I/O 引脚	—	—	0.4	V	IOL = 8.0 mA	VDD = 4.5V
			—	—	0.4	V	IOL = 4.0 mA	VDD = 3.6V
			—	—	0.4	V	IOL = 3.5 mA	VDD = 2.0V
DO16		OSC2/CLKO	—	—	0.4	V	IOL = 2.0 mA	VDD = 4.5V
			—	—	0.4	V	IOL = 1.2 mA	VDD = 3.6V
			—	—	0.4	V	IOL = 0.4 mA	VDD = 2.0V
DO20	VOH	输出高电压 所有 I/O 引脚	3.8	—	—	V	IOH = -3.5 mA	VDD = 4.5V
			3	—	—	V	IOH = -3.0 mA	VDD = 3.6V
			1.6	—	—	V	IOH = -1.0 mA	VDD = 2.0V
DO26		OSC2/CLKO	3.8	—	—	V	IOH = -2.0 mA	VDD = 4.5V
			3	—	—	V	IOH = -1.0 mA	VDD = 3.6V
			1.6	—	—	V	IOH = -0.5 mA	VDD = 2.0V

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 +25°C 条件下给出的。这些参数仅供设计参考, 未经测试。

表 29-11: 直流特性: 程序存储器

直流特性			标准工作条件:					
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)					
			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	
D130	EP	闪存程序存储器 单元耐擦写能力	10,000(2)	—	—	E/W	VMIN = 最小工作电压 假定未违反其他规范	
D131	VPR	读操作使用的 VDD	VMIN	—	3.6	V		
D133A	TIW	自定时写周期时间	—	2	—	ms		
D134	TRETD	特性保持时间	40	—	—	年		
D135	IDDP	编程时的供电电流	—	10	—	mA		

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。

2: 自写和块擦除操作。

PIC24FV32KA304 系列

表 29-12: 直流特性: 数据 EEPROM 存储器

直流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)				
			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
D140	EPD	数据 EEPROM 存储器 单元耐擦写能力	100,000	—	—	E/W	V _{MIN} = 最小工作电压
D141	VPRD	读操作使用的 V _{DD}	V _{MIN}	—	3.6	V	
D143A	T _{IWD}	自定时写周期时间	—	4	—	ms	
D143B	T _{REF}	刷新前总的 写 / 擦除周期数	—	10M	—	E/W	
D144	T _{RETDD}	特性保持时间	40	—	—	年	假定未违反其他规范
D145	IDDPD	编程时的供电电流	—	7	—	mA	

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。

表 29-13: 直流特性: 比较器规范

工作条件: 2.0V < V _{DD} < 3.6V, -40°C < T _A < +125°C (除非另外声明)							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
D300	V _{IOFF}	输入失调电压	—	20	40	mV	
D301	V _{ICM}	输入共模电压	0	—	V _{DD}	V	
D302	CMRR	共模抑制比 *	55	—	—	dB	

表 29-14: 直流特性: 比较器参考电压规范

工作条件: 2.0V < V _{DD} < 3.6V, -40°C < T _A < +125°C (除非另外声明)							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
VRD310	CV _{RES}	分辨率	—	—	V _{DD} /32	LSb	
VRD311	CV _{RAA}	绝对精度	—	—	AV _{DD} - 1.5	LSb	
VRD312	CV _{RuR}	单位电阻值 (R)	—	2k	—	Ω	

PIC24FV32KA304 系列

表 29-15: 内部稳压器规范

工作条件: $-40^{\circ}\text{C} < T_A < +125^{\circ}\text{C}$ (除非另外声明)							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
DVR10	VBG	带隙参考电压	0.973	1.024	1.075	V	
DVR11	TBG	带隙参考启动时间	—	1	—	ms	
DVR20	VRGOUT	稳压器输出电压	3.1	3.3	3.6	V	$-40^{\circ}\text{C} < T_A < +85^{\circ}\text{C}$
			3.0	3.19	3.6	V	$-40^{\circ}\text{C} < T_A < +125^{\circ}\text{C}$
DVR21	CEFC	外部滤波器电容值	4.7	10	—	μF	建议串联一个阻抗 $< 3\Omega$ 的电阻; 要求阻抗 $< 5\Omega$ 。
DVR30	VLVR	保持稳压器输出电压	—	2.6	—	V	

表 29-16: CTMU 电流源规范

直流特性			标准工作条件:					
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级) $-40^{\circ}\text{C} < T_A < +125^{\circ}\text{C}$ (扩展级)					
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	备注	条件
DCT10	IOUT1	CTMU 电流源, 基本范围	—	550	—	nA	CTMUICON<9:8> = 01	$2.5\text{V} < V_{DD} < V_{DDMAX}$
DCT11	IOUT2	CTMU 电流源, 10x 范围	—	5.5	—	μA	CTMUICON<9:8> = 10	
DCT12	IOUT3	CTMU 电流源, 100x 范围	—	55	—	μA	CTMUICON<9:8> = 11	
DCT13	IOUT4	CTMU 电流源, 1000x 范围	—	550	—	μA	CTMUICON<9:8> = 00 (注 2)	
DCT20	V _F	温度二极管正向电压	—	.76	—	V		
DCT21	V _Δ	每摄氏度电压变化	—	1.6	—	mV/°C		

- 注 1: 标称值为电流调整范围的中心值 (CTMUICON<7:2> = 000000)。在 PIC24F32KA 器件上, 当选择 I_{OUT4} 时, 电流输出限制为典型电流值。
- 2: 不要对热敏二极管使用该电流范围。

PIC24FV32KA304 系列

29.2 交流特性和时序参数

本节包含的信息说明了 PIC24FV32KA304 系列器件的交流特性和时序参数。

表 29-17: 温度和电压规范 —— 交流

交流特性	标准工作条件:	1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)
	工作温度:	-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)
	工作电压 VDD 范围如第 29.1 节 “直流特性” 所述。	

图 29-3: 器件时序规范的负载条件

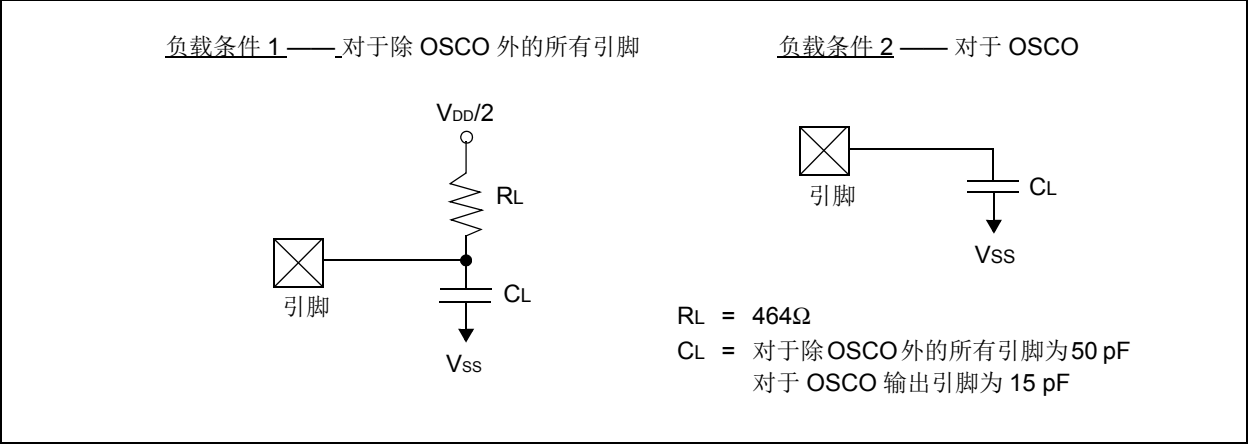


表 29-18: 输出引脚上的容性负载要求

参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
DO50	Cosc2	OSCO/CLKO 引脚	—	—	15	pF	当外部时钟用于驱动 OSC1 时，处于 XT 和 HS 模式。
DO56	Cio	所有 I/O 引脚和 OSCO	—	—	50	pF	EC 模式
DO58	CB	SCLx 和 SDAx	—	—	400	pF	在 I ² C™ 模式下

注 1: 除非另外声明，否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数仅供设计参考，未经测试。

图 29-4: 外部时钟时序

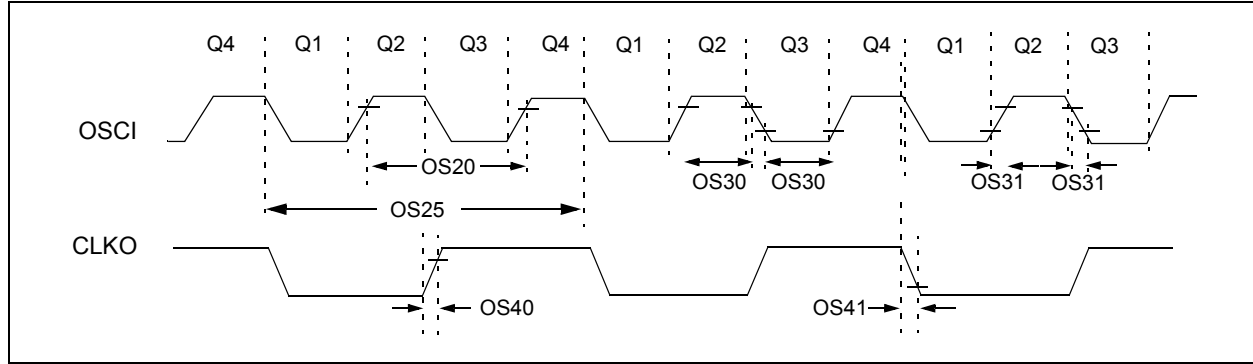


表 29-19: 外部时钟时序要求

交流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
OS10	Fosc	外部 CLKI 频率 (仅在 EC 模式下允许使用外部时钟)	DC 4	— —	32 8	MHz MHz	EC ECPLL
OS15		振荡器频率	0.2 4 4 31	— — — —	4 25 8 33	MHz MHz MHz kHz	XT HS XTPLL SOSC
OS20	Tosc	Tosc = 1/Fosc	—	—	—	—	关于 Fosc 值, 请参见参数 OS10
OS25	Tcy	指令周期 ⁽²⁾	62.5	—	DC	ns	
OS30	TosL, TosH	外部时钟输入 (OSCI) 高电平或低电平时间	62.5	—	DC	ns	EC
OS31	TosR, TosF	外部时钟输入 (OSCI) 上升或下降时间	0.45 x Tosc	—	—	ns	EC
OS40	TckR	CLKO 上升时间 ⁽³⁾	—	—	20	ns	
OS41	TckF	CLKO 下降时间 ⁽³⁾	—	6	10	ns	

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数仅供设计参考, 未经测试。

2: 指令周期 (Tcy) 等于输入振荡器时基周期的 2 倍。所有规范值均基于标准工作条件下, 器件执行代码时对应特定振荡器类型的特征数据。超过规范值可导致振荡器运行不稳定和 / 或使电流消耗超过预期值。所有器件在测试“最小值”时, 均在 OSCI/CLKI 引脚接入了外部时钟。当使用外部时钟输入时, 所有器件的“最大”周期时间限制为“DC” (无时钟)。

3: 测量在 EC 模式下进行。CLKO 信号是在 OSCO 引脚上测得的。CLKO 在 Q1-Q2 周期 (1/2 Tcy) 中为低电平, 在 Q3-Q4 周期 (1/2 Tcy) 中为高电平。

PIC24FV32KA304 系列

表 29-20: PLL 时钟时序规范

交流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)				
			工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 ⁽¹⁾	最小值	典型值 ⁽²⁾	最大值	单位	条件
OS50	FPLLI	PLL 输入频率范围	4	—	8	MHz	ECPLL 和 HSPLL 模式, -40°C ≤ TA ≤ +85°C
OS51	Fsys	PLL 输出频率范围	16	—	32	MHz	-40°C ≤ TA ≤ +85°C
OS52	TLOCK	PLL 起振时间 (锁定时间)	—	1	2	ms	
OS53	DCLK	CLKO 稳定性 (去抖动性能)	-2	1	2	%	在 100 ms 周期内测得

- 注 1: 这些参数为特征值, 未经生产测试。
 2: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数仅供设计参考, 未经测试。

表 29-21: 交流特性: 内部 RC 精度

交流特性		标准工作条件:					
		1.8V 至 3.6V (PIC24F32KA3XX)					
		2.0V 至 5.5V (PIC24FV32KA3XX)					
		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	
F20	内部 FRC 精度 @ 8 MHz ⁽¹⁾						
	FRC	-2	—	+2	%	+25°C	3.0V ≤ VDD ≤ 3.6V, F 器件 3.2V ≤ VDD ≤ 5.5V, FV 器件
		-5	—	+5	%	-40°C ≤ TA ≤ +85°C	1.8V ≤ VDD ≤ 3.6V, F 器件 2.0V ≤ VDD ≤ 5.5V, FV 器件
F21	LPRC @ 31 kHz ⁽²⁾						
		-15	—	15	%		

- 注 1: 已在 +25°C、3.3V 条件下对频率进行了校准。OSCTUN 位可用来补偿温度漂移。
 2: VDD 变化时 LPRC 频率的变化。

表 29-22: 内部 RC 振荡器规范

交流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)				
			工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 ⁽¹⁾	最小值	典型值	最大值	单位	条件
	TFRC	FRC 起振时间	—	5	—	μs	
	TLPRC	LPRC 起振时间	—	70	—	μs	

- 注 1: 这些参数为特征值, 未经生产测试。

图 29-5: CLKO 和 I/O 时序特性

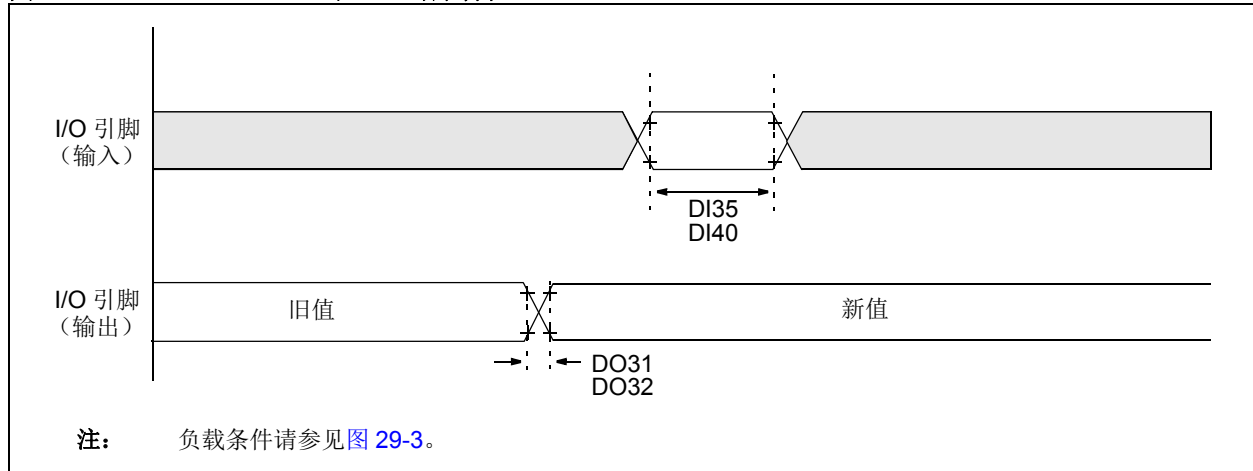


表 29-23: CLKO 和 I/O 时序要求

交流特性			标准工作条件：				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)				
			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
DO31	TioR	端口输出上升时间	—	10	25	ns	
DO32	TioF	端口输出下降时间	—	10	25	ns	
DI35	TINP	INTx 引脚高电平或低电平时间 (输出)	20	—	—	ns	
DI40	TRBP	CNx 高电平或低电平时间 (输入)	2	—	—	Tcy	

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在以下条件下给出的: 3.3V、+25°C (适用于 PIC24F32KA3XX), 5.0V、+25°C (适用于 PIC24FV32KA3XX)。

PIC24FV32KA304 系列

表 29-24: 比较器时序

参数编号	符号	特性	最小值	典型值	最大值	单位	备注
300	TRESP	响应时间 ^{*(1)}	—	150	400	ns	
301	TMC2OV	比较器模式变为输出有效的时间 [*]	—	—	10	μs	

^{*} 参数为特征值，未经测试。

注 1: 响应时间是在一个比较器输入为 $(V_{DD} - 1.5)/2$ ，而另一个比较器输入从 V_{SS} 跳变到 V_{DD} 时测得的。

表 29-25: 比较器参考电压稳定时间规范

参数编号	符号	特性	最小值	典型值	最大值	单位	备注
VR310	TSET	稳定时间 ⁽¹⁾	—	—	10	μs	

注 1: 稳定时间是在 $CVRSS = 1$ 和 $CVR<3:0>$ 位从 0000 跳变为 1111 时测得的。

图 29-6: 复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序特性

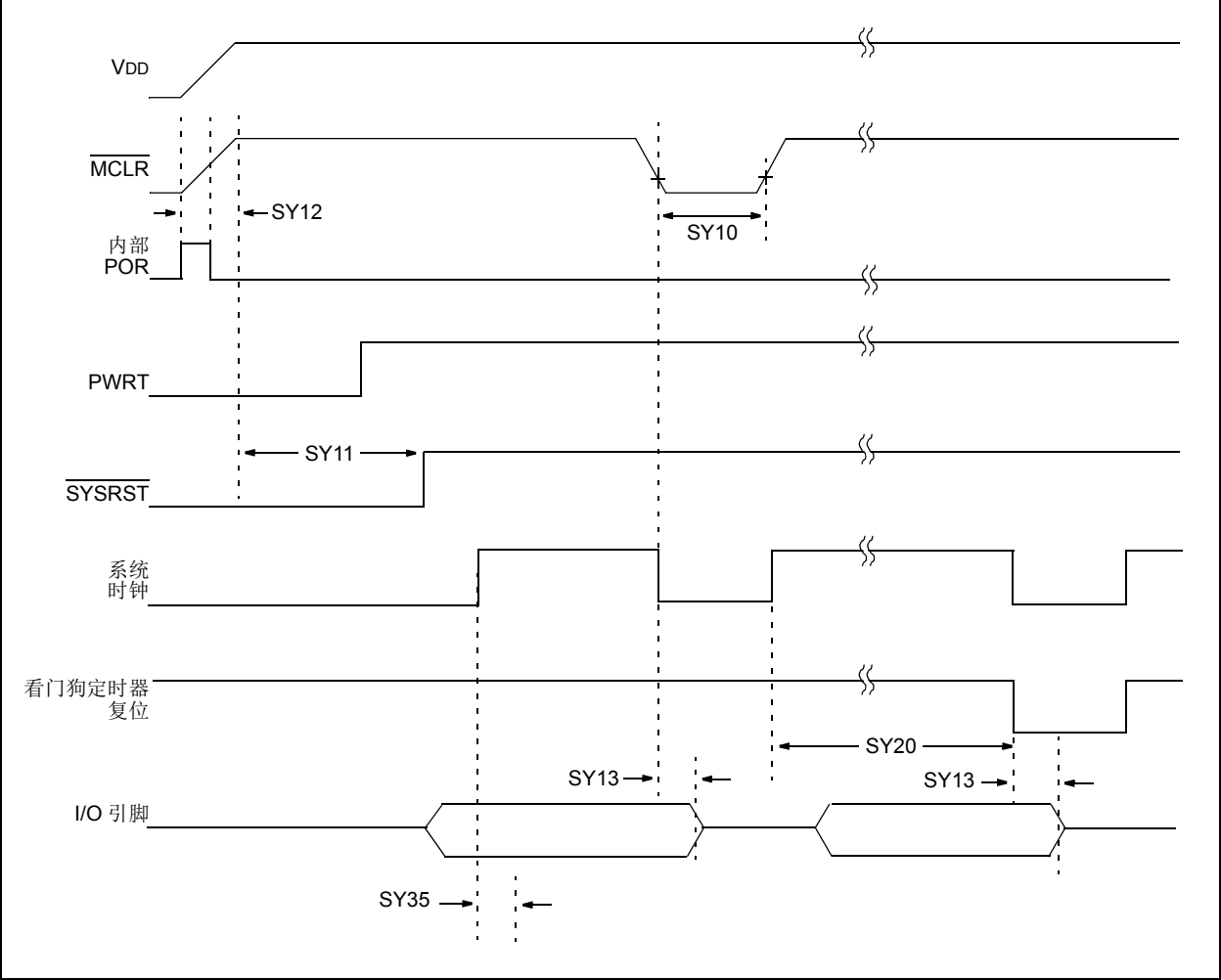


图 29-7: 欠压复位特性

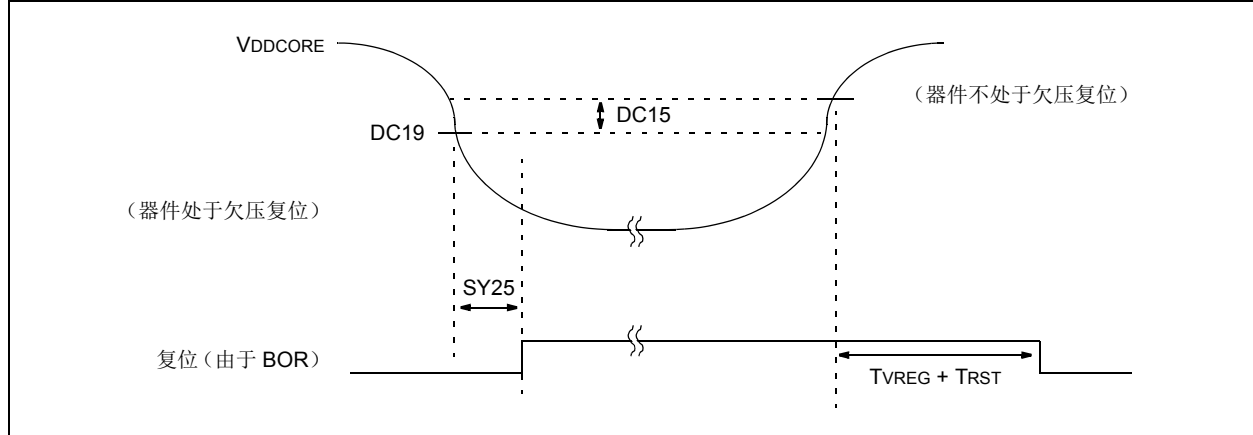


表 29-26: 复位、看门狗定时器、振荡器起振定时器、上电复位定时器和欠压复位时序要求

交流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX)				
			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
SY10	TmCL	MCLR 脉宽 (低电平)	2	—	—	μs	
SY11	TPWRT	上电延时定时器周期	50	64	90	ms	
SY12	TPOR	上电复位延时	1	5	10	μs	
SY13	TIOZ	从 MCLR 为低电平或看门狗定时器复位到 I/O 高阻态的时间	—	—	100	ns	
SY20	TWDT	看门狗定时器超时周期	0.85	1.0	1.15	ms	1.32 预分频比
			3.4	4.0	4.6	ms	1:128 预分频比
SY25	TBOR	欠压复位脉宽	1	—	—	μs	
SY35	TFSCM	故障保护时钟监视器延时	—	2.0	2.3	μs	
SY45	TRST	内部状态复位时间	—	5	—	μs	
SY50	TVREG	片内稳压器输出延时	—	10	—	μs	(注 2)
SY55	TLOCK	PLL 启动时间	—	100	—	μs	
SY65	TOST	振荡器起振时间	—	1024	—	TOSC	
SY70	TDSWU	从深度休眠唤醒的时间	—	100	—	μs	基于 VCAP 上的 10 μF 电容完全放电。包括 TPOR 和 TRST
SY71	TPM	程序存储器唤醒时间	—	1	—	μs	PMSLP = 0 时从休眠模式唤醒
SY72	TLVR	保持稳压器唤醒时间	—	250	—	μs	
SY73	THVLD	HVLD 中断响应时间	—	2	—	μs	

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。

2: 仅适用于 PIC24FV32KA3XX 器件。

PIC24FV32KA304 系列

图 29-8: TIMER1/2/3/4/5 外部时钟输入时序

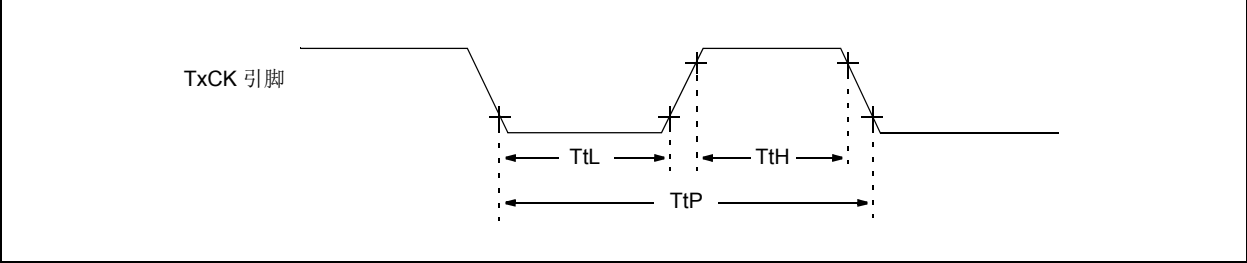


表 29-27: TIMER1/2/3/4/5 外部时钟输入要求

参数编号	符号	特性		最小值	最大值	单元	条件
	TtH	TxCK 高电平脉冲时间	同步, 带预分频器	$T_{CY} + 20$	—	ns	还必须满足参数 Ttp
			异步, 带预分频器	10	—	ns	
			异步计数器	20	—	ns	
	TtL	TxCK 低电平脉冲时间	同步, 带预分频器	$T_{CY} + 20$	—	ns	还必须满足参数 Ttp
			异步, 带预分频器	10	—	ns	
			异步计数器	20	—	ns	
	TtP	TxCK 外部输入周期	同步, 带预分频器	$2 * T_{CY} + 40$	—	ns	N = 预分频值 (1、4、8 和 16)
			异步, 带预分频器	以下两者中的较大值: 20 或 $\frac{2 * T_{CY} + 40}{N}$	—	ns	
			异步计数器	40	—	ns	
		出现输入边沿到定时器递增之间的延时	同步	1	2	T_{CY}	
			异步	—	20	ns	

图 29-9: 输入捕捉 x 时序

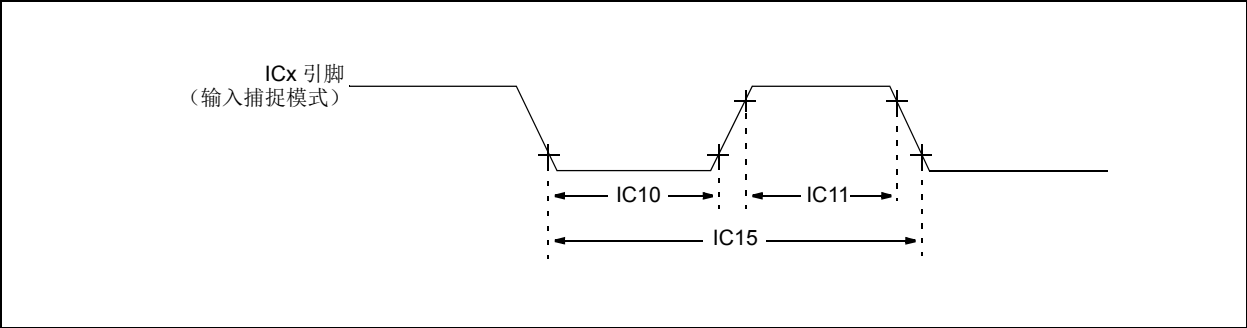


表 29-28: 输入捕捉 x 要求

参数编号	符号	特性		最小值	最大值	单位	条件
IC10	TccL	ICx 输入低电平时间 —— 同步定时器	有预分频器	$T_{CY} + 20$	—	ns	还必须满足参数 IC15
			无预分频器	20	—	ns	
IC11	TccH	ICx 输入高电平时间 —— 同步定时器	无预分频器	$T_{CY} + 20$	—	ns	还必须满足参数 IC15
			有预分频器	20	—	ns	
IC15	TccP	ICx 输入周期 —— 同步定时器		$\frac{2 * T_{CY} + 40}{N}$	—	ns	N = 预分频值 (1, 4, 16)

图 29-10: 输出比较 x 时序

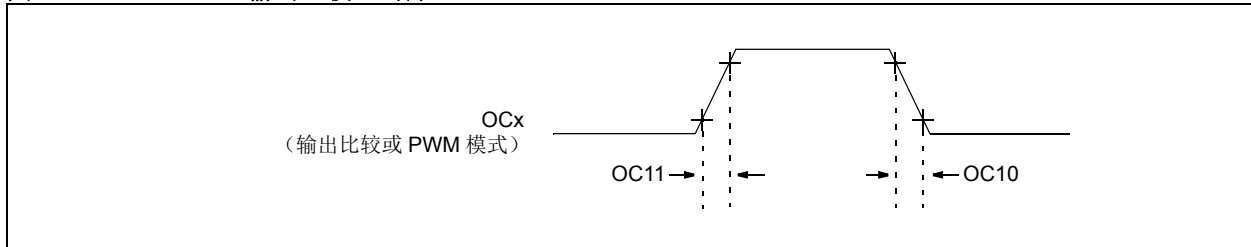


表 29-29: 输出比较要求

参数编号	符号	特性	最小值	最大值	单位	条件
OC11	TccR	OC1 输出上升时间	—	10	ns	
			—	—	ns	
OC10	TccF	OC1 输出下降时间	—	10	ns	
			—	—	ns	

图 29-11: PWM 模块时序要求

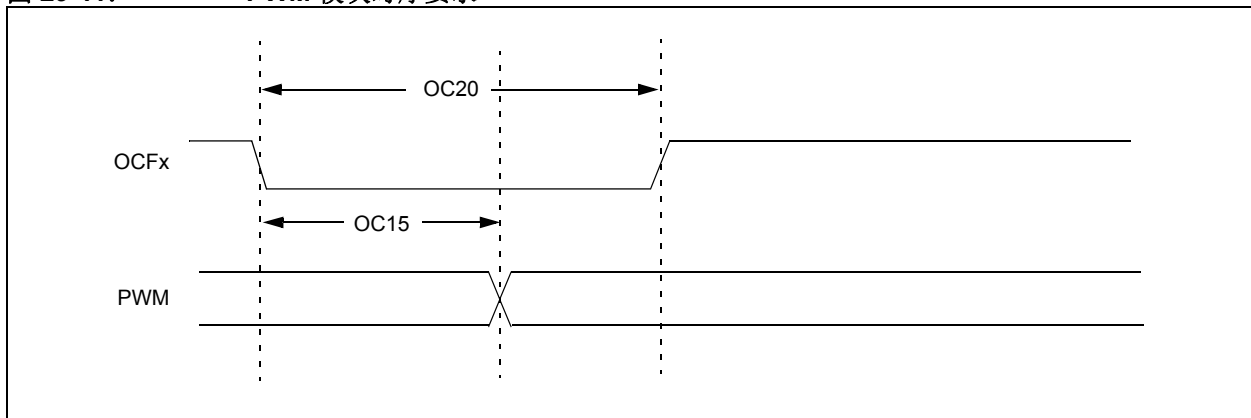


表 29-30: PWM 时序要求

参数编号	符号	特性	最小值	典型值†	最大值	单位	条件
OC15	TfD	故障输入至 PWM I/O 改变的时间	—	—	25	ns	VDD = 3.0V, -40°C 至 +125°C
OC20	TfH	故障输入脉冲宽度	50	—	—	ns	VDD = 3.0V, -40°C 至 +125°C

† 除非另外声明，否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数未经测试，仅供设计参考。

PIC24FV32KA304 系列

图 29-12: I²C™ 总线启动 / 停止位时序特性（主模式）

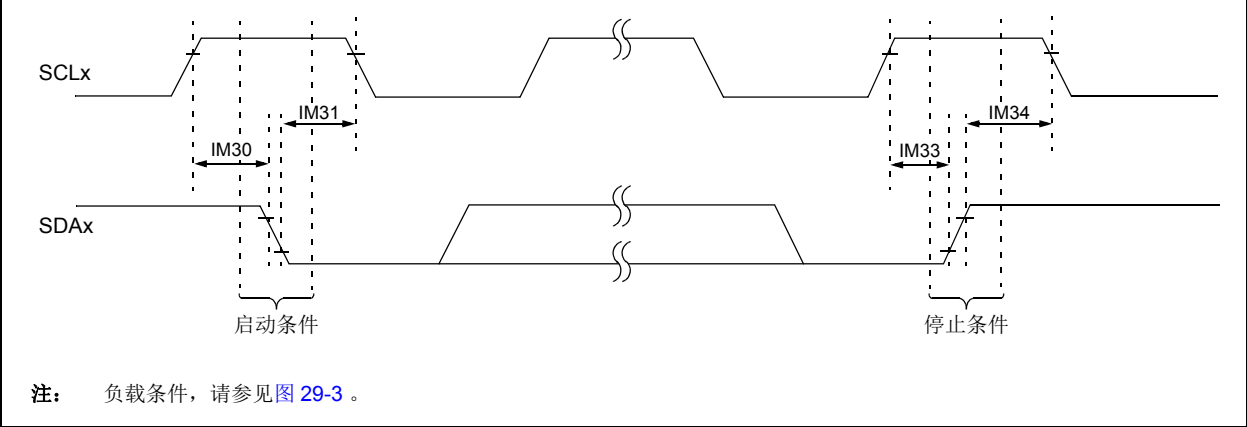


表 29-31: I²C™ 总线启动 / 停止位时序要求（主模式）

交流特性				标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度			
				-40°C ≤ Ta ≤ +85°C (工业级) -40°C ≤ Ta ≤ +125°C (扩展级)			
参数编号	符号	特性		最小值 ⁽¹⁾	最大值	单位	条件
IM30	TSU:STA	启动条件建立时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	仅与重复启动条件相关
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	Tcy/2 (BRG + 1)	—	μs	
IM31	THD:STA	启动条件保持时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	这个周期后，产生第一个时钟脉冲
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	Tcy/2 (BRG + 1)	—	μs	
IM33	TSU:STO	停止条件建立时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	Tcy/2 (BRG + 1)	—	μs	
IM34	THD:STO	停止条件保持时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	ns	
			400 kHz 模式	Tcy/2 (BRG + 1)	—	ns	
			1 MHz 模式 ⁽²⁾	Tcy/2 (BRG + 1)	—	ns	

注 1: BRG 是 I²C™ 波特率发生器的值。欲知详情，请参见第 17.3 节“设置用作总线主器件时的波特率”。
2: 所有 I²C 引脚的最大引脚电容 = 10 pF（仅 1 MHz 模式）。

图 29-13: I²C™ 总线数据时序特性（主模式）

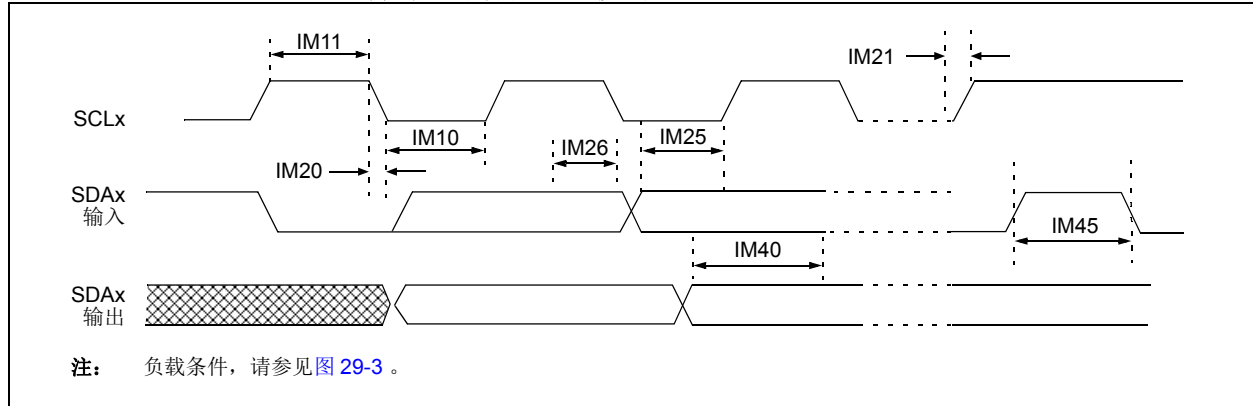


表 29-32: I²C™ 总线数据时序要求（主模式）

交流特性				标准工作条件：2.0V 至 3.6V (除非另外说明) 工作温度 -40°C ≤ T _A ≤ +85°C (工业级) -40°C ≤ T _A ≤ +125°C (扩展级)			
参数编号	符号	特性		最小值 ⁽¹⁾	最大值	单位	条件
IM10	TLO:SCL	时钟低电平时间	100 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			400 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	T _{cy} /2 (BRG + 1)	—	μs	
IM11	THI:SCL	时钟高电平时间	100 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			400 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	T _{cy} /2 (BRG + 1)	—	μs	
IM20	TF:SCL	SDAx 和 SCLx 下降时间	100 kHz 模式	—	300	ns	C _B 值规定在 10 至 400 pF 之间
			400 kHz 模式	20 + 0.1 C _B	300	ns	
			1 MHz 模式 ⁽²⁾	—	100	ns	
IM21	TR:SCL	SDAx 和 SCLx 上升时间	100 kHz 模式	—	1000	ns	C _B 值规定在 10 至 400 pF 之间
			400 kHz 模式	20 + 0.1 C _B	300	ns	
			1 MHz 模式 ⁽²⁾	—	300	ns	
IM25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	
			400 kHz 模式	100	—	ns	
			1 MHz 模式 ⁽²⁾	100	—	ns	
IM26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	ns	
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 ⁽²⁾	0	—	ns	
IM40	TAA:SCL	时钟输出有效的 时间	100 kHz 模式	—	3500	ns	
			400 kHz 模式	—	1000	ns	
			1 MHz 模式 ⁽²⁾	—	—	ns	
IM45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	μs	在开始新的传输前， 总线必须保持空闲的 时间
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 ⁽²⁾	0.5	—	μs	
IM50	C _B	总线容性负载		—	400	pF	

图注： TBD = 待定

注 1: BRG 是 I²C™ 波特率发生器的值。欲知详情，请参见第 17.3 节“设置用作总线主器件时的波特率”。

2: 所有 I²C 引脚的最大引脚电容 = 10 pF（仅 1 MHz 模式）。

PIC24FV32KA304 系列

图 29-14: I²C™ 总线数据时序特性（从模式）

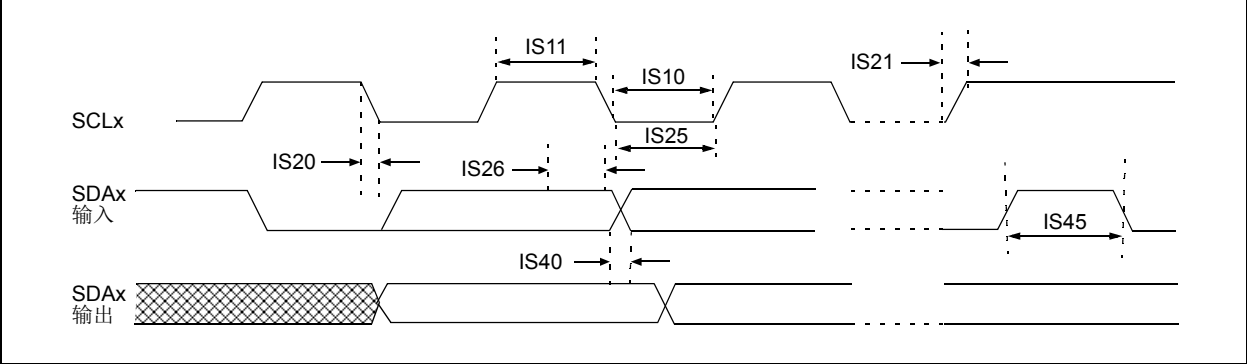


表 29-33: I²C™ 总线数据时序要求（从模式）

交流特性				标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度			-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)	
参数编号	符号	特性		最小值	最大值	单位	条件	
IS10	TLO:SCL	时钟低电平时间	100 kHz 模式	4.7	—	μs	器件必须工作在至少 1.5 MHz 频率下	
			400 kHz 模式	1.3	—	μs	器件必须工作在至少 10 MHz 频率下	
			1 MHz 模式 ⁽¹⁾	0.5	—	μs		
IS11	THI:SCL	时钟高电平时间	100 kHz 模式	4.0	—	μs	器件必须工作在至少 1.5 MHz 频率下	
			400 kHz 模式	0.6	—	μs	器件必须工作在至少 10 MHz 频率下	
			1 MHz 模式 ⁽¹⁾	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 至 400 pF 范围内	
			400 kHz 模式	20 + 0.1 CB	300	ns		
			1 MHz 模式 ⁽¹⁾	—	100	ns		
IS21	TR:SCL	SDAx和SCLx上升时间	100 kHz 模式	—	1000	ns	CB 值规定在 10 至 400 pF 范围内	
			400 kHz 模式	20 + 0.1 CB	300	ns		
			1 MHz 模式 ⁽¹⁾	—	300	ns		
IS25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns		
			400 kHz 模式	100	—	ns		
			1 MHz 模式 ⁽¹⁾	100	—	ns		
IS26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	ns		
			400 kHz 模式	0	0.9	μs		
			1 MHz 模式 ⁽¹⁾	0	0.3	μs		
IS40	TAA:SCL	时钟输出有效的时间	100 kHz 模式	0	3500	ns		
			400 kHz 模式	0	1000	ns		
			1 MHz 模式 ⁽¹⁾	0	350	ns		
IS45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	μs	在开始一个新的传输前，总线必须保持空闲的时间	
			400 kHz 模式	1.3	—	μs		
			1 MHz 模式 ⁽¹⁾	0.5	—	μs		
IS50	CB	总线容性负载		—	400	pF		

注 1: 所有 I²C 引脚的最大引脚电容 = 10 pF（仅 1 MHz 模式）。

图 29-15: I²C™ 总线启动 / 停止位时序特性（从模式）

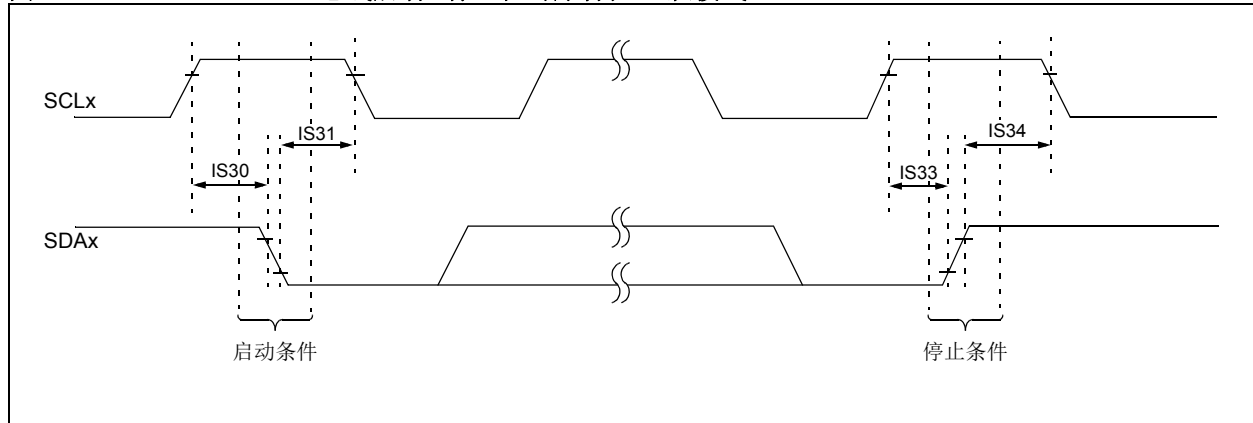


表 29-34: I²C™ 总线启动 / 停止位时序要求（从模式）

交流特性				标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度			条件
参数编号	符号	特性		最小值	最大值	单位	
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 ⁽²⁾	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 ⁽²⁾	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 ⁽²⁾	0.6	—	μs	
IS34	THD:STO	停止条件保持时间	100 kHz 模式	4000	—	ns	
			400 kHz 模式	600	—	ns	
			1 MHz 模式 ⁽²⁾	250	—	ns	

注 1: 所有 I²C 引脚的最大引脚电容 = 10 pF (仅 1 MHz 模式)。

PIC24FV32KA304 系列

图 29-16: UARTx 波特率发生器输出时序

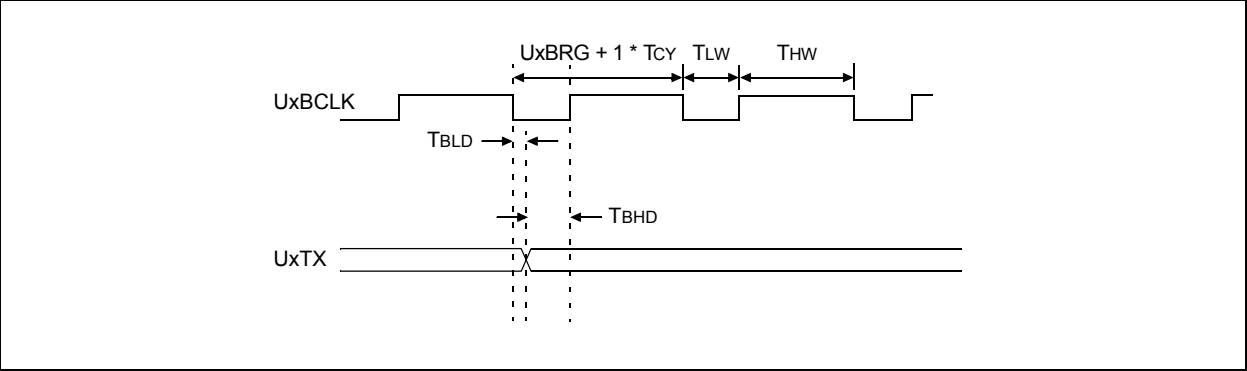


图 29-17: UARTx 启动位边沿检测

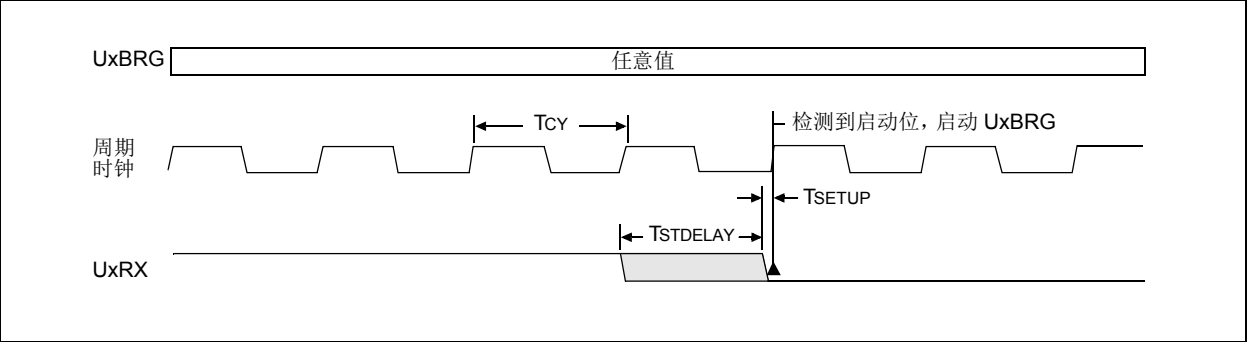


表 29-35: UARTx 时序要求

交流特性		标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)			
符号	特性	最小值	典型值	最大值	条件
TLW	UxBCLK 高电平时间	20	Tcy/2	—	ns
THW	UxBCLK 低电平时间	20	(Tcy * UxBRG) + Tcy/2	—	ns
TBLD	从 UxTX 有效到 UxBCLK 下降沿的延时	-50	—	50	ns
TBHD	从 UxTX 有效到 UxBCLK 上升沿的延时	Tcy/2 - 50	—	Tcy/2 + 50	ns
TWAK	导致唤醒, UxRX 线上的最低电平	—	1	—	μs
TCTS	启动传输, UxCTS 线上的最低电平	Tcy	—	—	ns
TSETUP	启动位下降沿到系统时钟上升沿的建立时间	3	—	—	ns
TSTDELAY	检测到启动位上升沿的最大延时	—	—	Tcy + TSETUP	ns

图 29-18: SPIx 模块主模式时序特性 (CKE = 0)

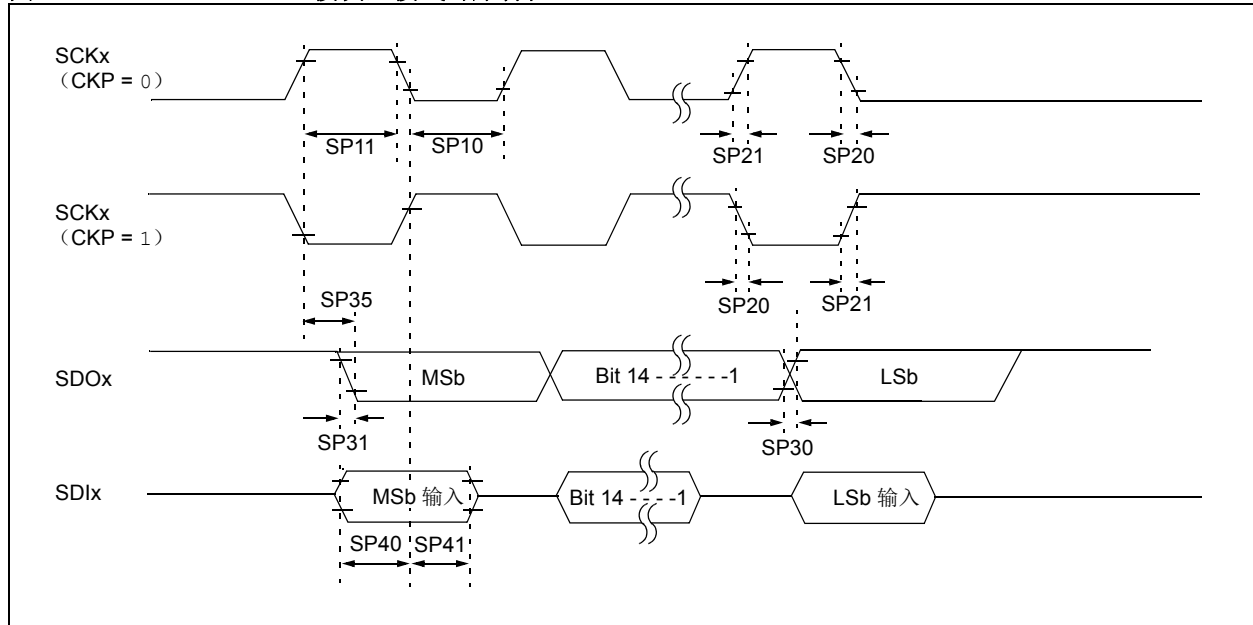


表 29-36: SPIx 模块主模式时序要求 (CKE = 0)

交流特性		标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间 ⁽²⁾	Tcy/2	—	—	ns	
SP11	TscH	SCKx 输出高电平时间 ⁽²⁾	Tcy/2	—	—	ns	
SP20	TscF	SCKx 输出下降时间 ⁽³⁾	—	10	25	ns	
SP21	TscR	SCKx 输出上升时间 ⁽³⁾	—	10	25	ns	
SP30	TdoF	SDOx 数据输出下降时间 ⁽³⁾	—	10	25	ns	
SP31	TdoR	SDOx 数据输出上升时间 ⁽³⁾	—	10	25	ns	
SP35	TscH2doV, TscL2doV	SCKx 边沿后, SDOx 数据输出有效的的时间	—	—	30	ns	
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数未经测试, 仅供设计参考。

2: SCKx 的最小时钟周期为 100 ns; 因此主模式下产生的时钟不得违反此规定。

3: 假设所有 SPIx 引脚上均有 50 pF 负载。

PIC24FV32KA304 系列

图 29-19: SPIx 模块主模式时序特性 (CKE = 1)

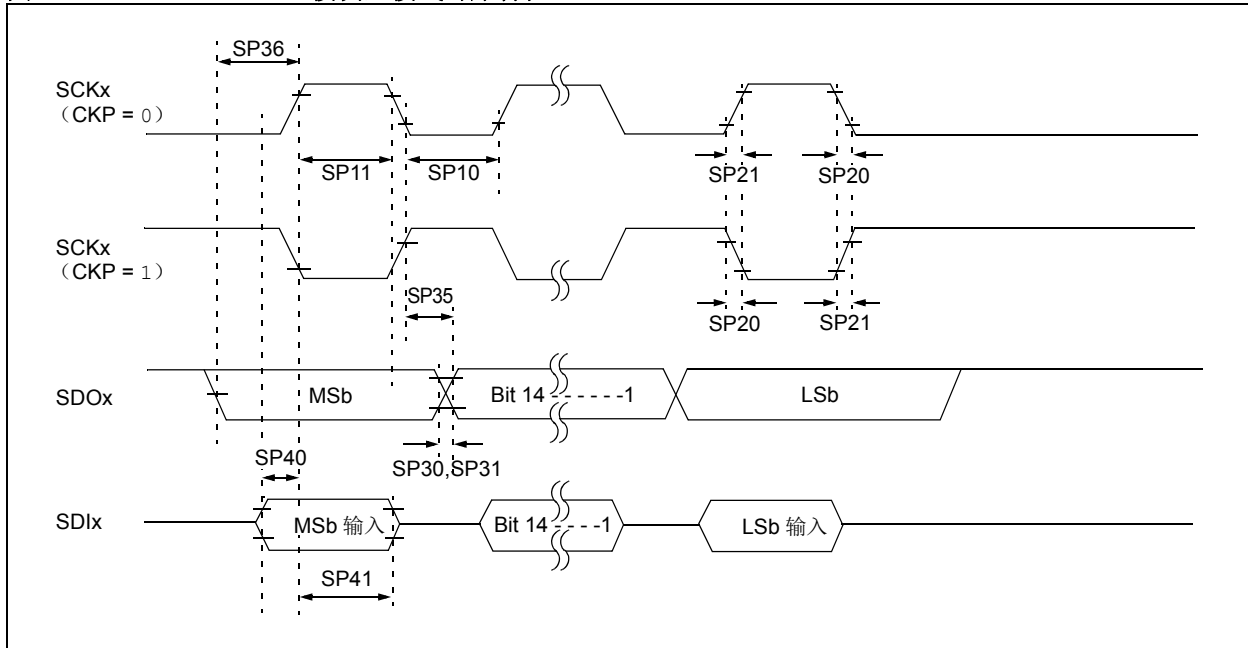


表 29-37: SPIx 模块主模式时序要求 (CKE = 1)

交流特性				标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)			
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间 (2)	Tcy/2	—	—	ns	
SP11	TscH	SCKx 输出高电平时间 (2)	Tcy/2	—	—	ns	
SP20	TscF	SCKx 输出下降时间 (3)	—	10	25	ns	
SP21	TscR	SCKx 输出上升时间 (3)	—	10	25	ns	
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	10	25	ns	
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	10	25	ns	
SP35	Tsch2doV, TscL2doV	SCKx 边沿后, SDOx 数据输出有效的 时间	—	—	30	ns	
SP36	TdoV2sc, TdoV2scL	SDOx 数据输出建立到第一个 SCKx 边沿的时间	30	—	—	ns	
SP40	TdiV2sch, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建 立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保 持时间	20	—	—	ns	

- 注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数未经测试, 仅供设计参考。
- 2: SCKx 的最小时钟周期为 100 ns; 因此主模式下产生的时钟不得违反此规定。
- 3: 假设所有 SPIx 引脚上均有 50 pF 负载。

图 29-20: SPIx 模块从模式时序特性 (CKE = 0)

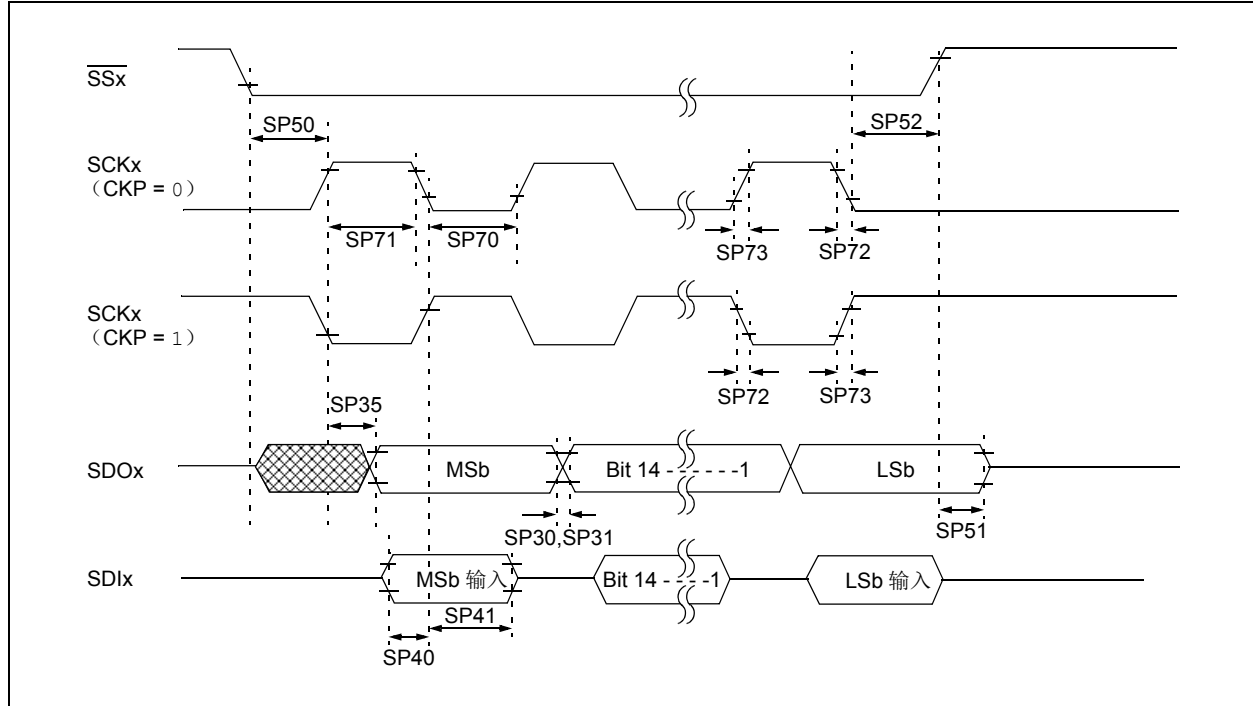


表 29-38: SPIx 模块从模式时序特性 (CKE = 0)

交流特性		标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
SP70	TscL	SCKx 输出低电平时间 ⁽²⁾	30	—	—	ns	
SP71	TscH	SCKx 输出高电平时间 ⁽²⁾	30	—	—	ns	
SP72	TscF	SCKx 输出下降时间 ⁽³⁾	—	10	25	ns	
SP73	TscR	SCKx 输出上升时间 ⁽³⁾	—	10	25	ns	
SP30	TdoF	SDOx 数据输出下降时间 ⁽³⁾	—	10	25	ns	
SP31	TdoR	SDOx 数据输出上升时间 ⁽³⁾	—	10	25	ns	
SP35	TscH2doV, TscL2doV	SCKx 边沿后, SDOx 数据输出有效的 时间	—	—	30	ns	
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	
SP50	TssL2scH, TssL2scL	SSx 到 SCKx ↑ 或 SCKx 输入的时间	120	—	—	ns	
SP51	TssH2doZ	SSx ↑ 到 SDOx 输出呈现高阻态的时间 ⁽³⁾	10	—	50	ns	
SP52	TscH2ssH, TscL2ssH	SCKx 边沿后 SSx 有效的的时间	1.5 Tcy + 40	—	—	ns	

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数未经测试, 仅供设计参考。

2: SCKx 的最小时钟周期为 100 ns; 因此主模式下产生的时钟不得违反此规定。

3: 假设所有 SPIx 引脚上均有 50 pF 负载。

PIC24FV32KA304 系列

图 29-21: SPIx 模块从模式时序特性 (CKE = 1)

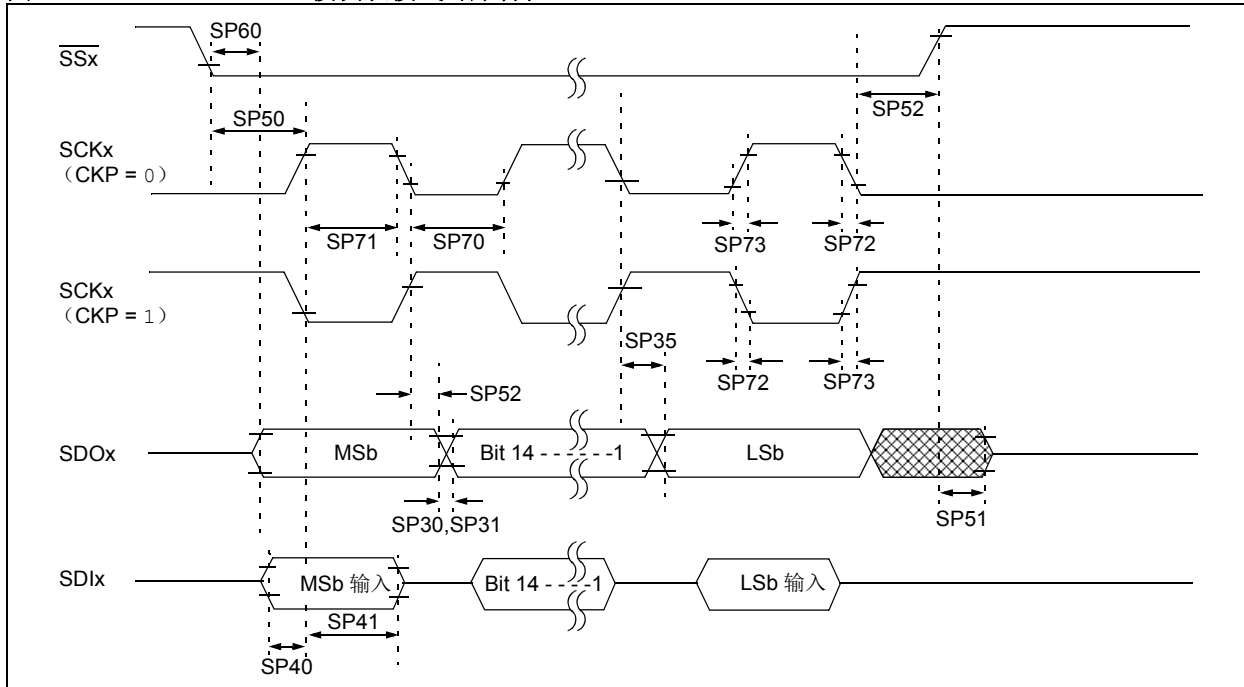


表 29-39: SPIx 模块从模式时序要求 (CKE = 1)

交流特性				标准工作条件: 2.0V 至 3.6V (除非另外说明) 工作温度			
				-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)			
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间 ⁽²⁾	30	—	—	ns	
SP71	TscH	SCKx 输入高电平时间 ⁽²⁾	30	—	—	ns	
SP72	TscF	SCKx 输入下降时间 ⁽³⁾	—	10	25	ns	
SP73	TscR	SCKx 输入上升时间 ⁽³⁾	—	10	25	ns	
SP30	TdoF	SDOx 数据输出下降时间 ⁽³⁾	—	10	25	ns	
SP31	TdoR	SDOx 数据输出上升时间 ⁽³⁾	—	10	25	ns	
SP35	TscH2doV, TscL2doV	SCKx 边沿后, SDOx 数据输出有效的 时间	—	—	30	ns	
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	
SP50	TssL2scH, TssL2scL	SSx↓ 到 SCKx↓ 或 SCKx↑ 输入的时间	120	—	—	ns	
SP51	TssH2doZ	SSx↑ 到 SDOx 输出呈现高阻态的时间 ⁽³⁾	10	—	50	ns	
SP52	TscH2ssH, TscL2ssH	SCKx 边沿后 SSx↑ 的时间	1.5 Tcy + 40	—	—	ns	
SP60	TssL2doV	SSx 边沿后, SDOx 数据输出有效的 时间	—	—	50	ns	

注 1: 除非另外声明, 否则“典型值”栏中的数据都是在 3.3V、+25°C 条件下给出的。这些参数未经测试, 仅供设计参考。

2: SCKx 的最小时钟周期为 100 ns; 因此主模式下产生的时钟不得违反此规定。

3: 假设所有 SPIx 引脚上均有 50 pF 负载。

PIC24FV32KA304 系列

表 29-40: ADC 模块规范

交流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX) 2.0V 至 5.5V (PIC24FV32KA3XX) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
器件电源							
AD01	AVDD	模块电源 VDD	取 VDD - 0.3 或 1.8 中的较大值	—	取 VDD + 0.3 或 3.6 中的较小值	V	PIC24FXXKA30X 器件
			取 VDD - 0.3 或 2.0 中的较大值	—	取 VDD + 0.3 或 5.5 中的较小值	V	PIC24FVXXKA30X 器件
AD02	AVSS	模块电源 VSS	VSS - 0.3	—	VSS + 0.3	V	
参考输入							
AD05	VREFH	参考电压高电平	AVSS + 1.7	—	AVDD	V	
AD06	VREFL	参考电压低电平	AVSS	—	AVDD - 1.7	V	
AD07	VREF	绝对参考电压	AVSS - 0.3	—	AVDD + 0.3	V	
AD08	IVREF	参考电压输入电流	—	1.25	—	mA	
AD09	ZVREF	参考源输入阻抗	—	10k	—	Ω	
模拟输入							
AD10	VINH-VINL	满量程输入范围	VREFL	—	VREFH	V	(注 2)
AD11	VIN	绝对输入电压	AVSS - 0.3	—	AVDD + 0.3	V	
AD12	VINL	绝对 VINL 输入电压	AVSS - 0.3	—	AVDD/2	V	
AD17	RIN	模拟电压源的推荐阻抗	—	—	1k	Ω	12 位
A/D 精度							
AD20b	NR	分辨率	—	12	—	位	
AD21b	INL	积分非线性误差	—	±1	±9	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD22b	DNL	微分非线性误差	—	±1	±5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD23b	GERR	增益误差	—	±1	±9	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD24b	Eoff	失调误差	—	±1	±5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD25b		单调性 ⁽¹⁾	—	—	—	—	保证

注 1: A/D 转换结果将不会随着输入电压的增加而减小。

2: 测量是在使用外部 VREF+ 和 VREF- 作为 A/D 参考电压的情况下进行的。

PIC24FV32KA304 系列

图 29-22: A/D 转换时序

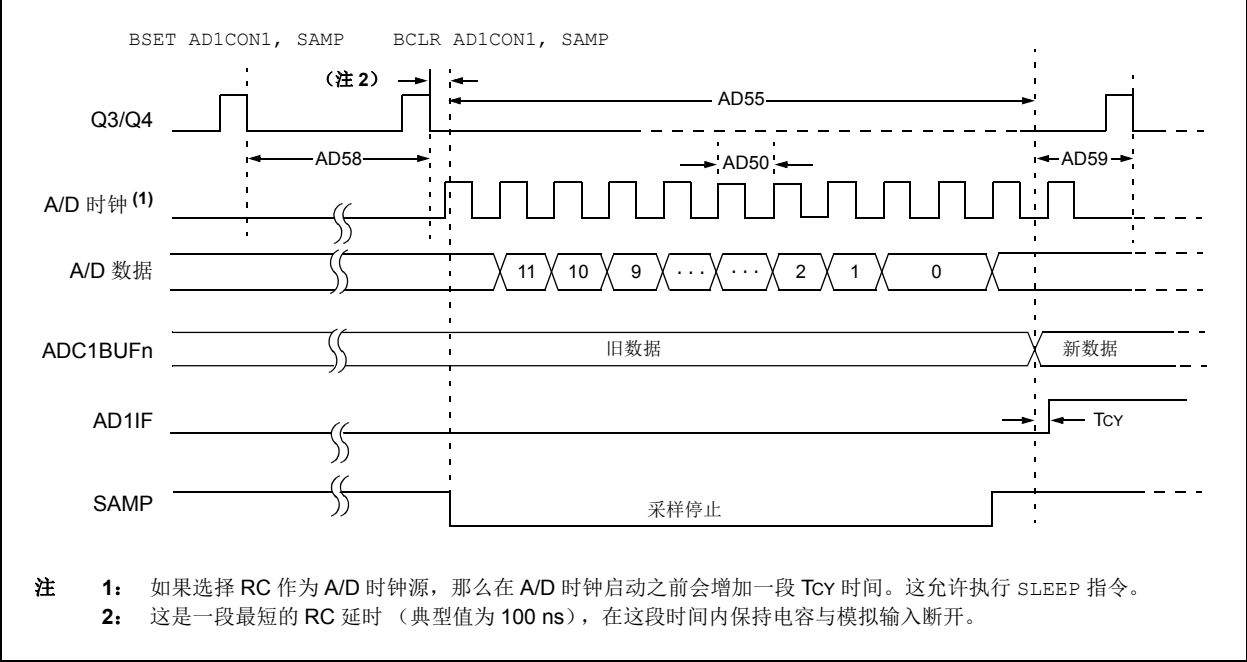


表 29-41: A/D 转换时序要求 (1)

交流特性			标准工作条件:				
			1.8V 至 3.6V (PIC24F32KA3XX)				
			2.0V 至 5.5V (PIC24FV32KA3XX)				
			-40°C ≤ Ta ≤ +85°C (工业级)				
			-40°C ≤ Ta ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
时钟参数							
AD50	TAD	A/D 时钟周期	600	—	—	ns	Tcy = 75 ns, AD1CON3 处于默认状态
AD51	TRC	A/D 内部 RC 振荡器周期	—	1.67	—	μs	
转换速率							
AD55	TCONV	转换时间	—	12	—	TAD	10 位结果
			—	14	—	TAD	12 位结果
AD56	FCNV	吞吐率	—	—	100	ksps	
AD57	TSAMP	采样时间	—	1	—	TAD	
AD58	TACQ	采集时间	750	—	—	ns	(注 2)
AD59	TSWC	从转换切换到采样的时间	—	—	(注 3)		
AD60	TDIS	放电时间	12	—	—	TAD	
时钟参数							
AD61	TPSS	从置 1 采样位 (SAMP) 到采样启动的延时	2	—	3	TAD	

- 注 1**: 因为采样电容最终将释放电荷，因此低于 10 kHz 的时钟频率可能会影响线性性能，尤其是在温度较高时。
- 注 2**: 转换完成后当电压满量程变化时 (VDD 至 VSS 或 VSS 至 VDD)，保持电容采集一个“新”输入电压所需的时间。
- 注 3**: 在器件时钟的下一个周期。

30.0 直流和交流特性图表

注： 本注释下方的图表是根据有限数量的样本所做的统计，仅用于示意。在此列出的性能特性未经测试，不做任何担保。在某些图表中，呈现的数据超出规定的工作范围（即，超出规定的电源范围），当然也不在担保范围内。

V_{DD} 大于 3.3V 时的数据仅适用于 PIC24FV32KA304 系列器件。

30.1 工业级温度器件的特性图表（-40°C 至 +85°C）

图 30-1： I_{DD} 的典型值和最大值与 F_{OSC} 的关系曲线
(EC 模式, 2 MHz 至 32 MHz, -40°C 至 +85°C)

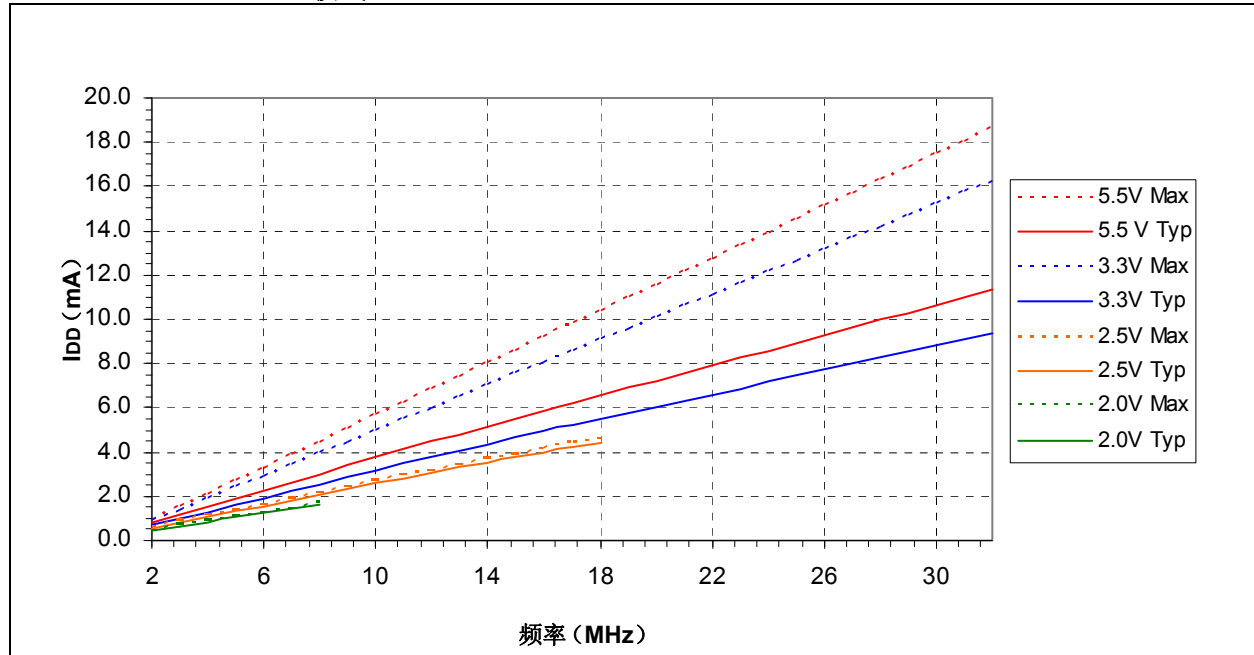
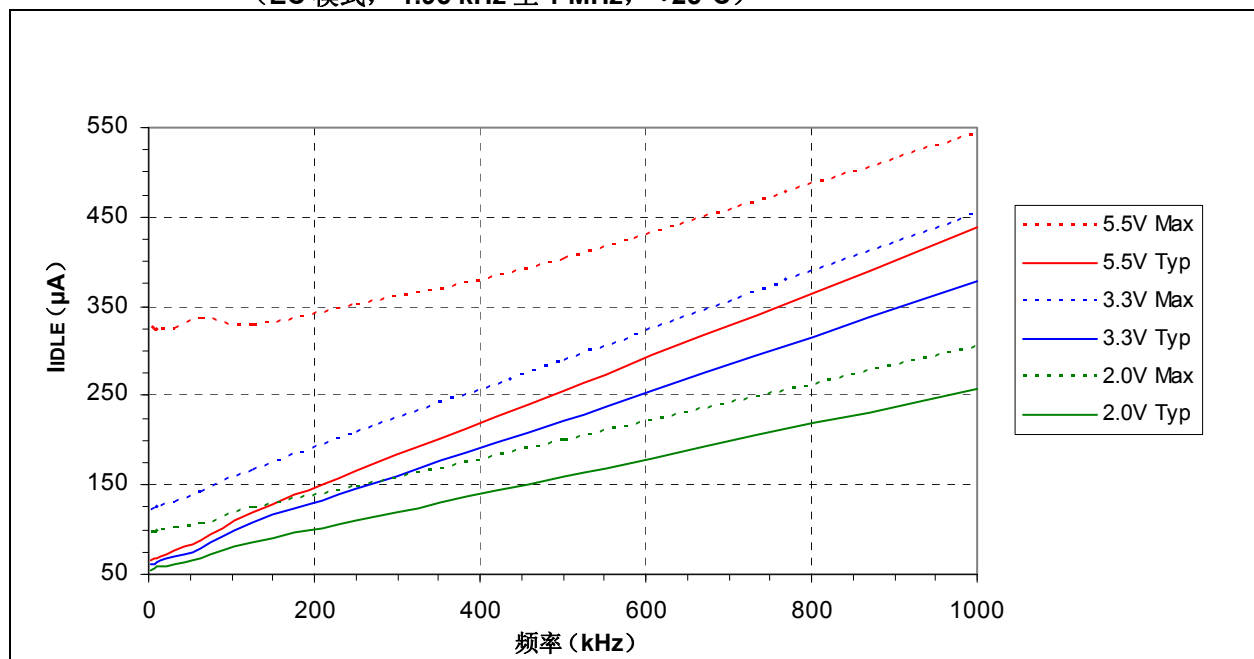


图 30-2： I_{DD} 的典型值和最大值与 F_{OSC} 的关系曲线
(EC 模式, 1.95 kHz 至 1 MHz, +25°C)



PIC24FV32KA304 系列

图 30-3: IDLE 的典型值和最大值与频率的关系曲线 (EC 模式, 2 MHz 至 32 MHz)

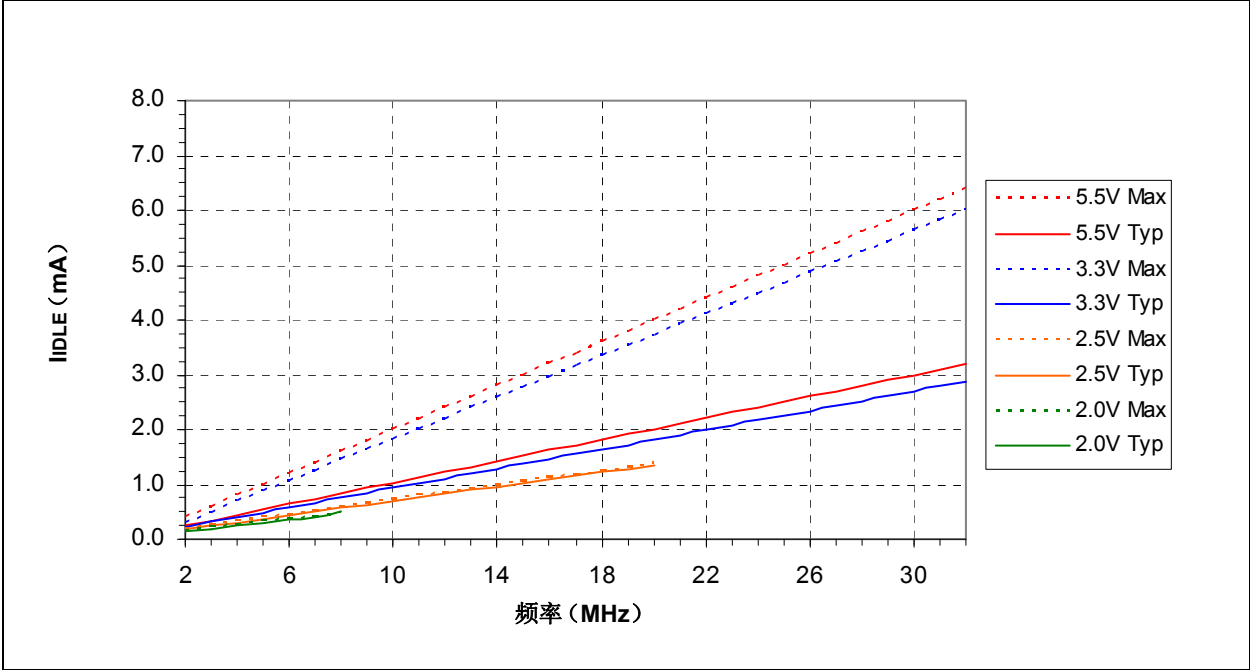


图 30-4: IDLE 的典型值和最大值与频率的关系曲线 (EC 模式, 1.95 kHz 至 1 MHz)

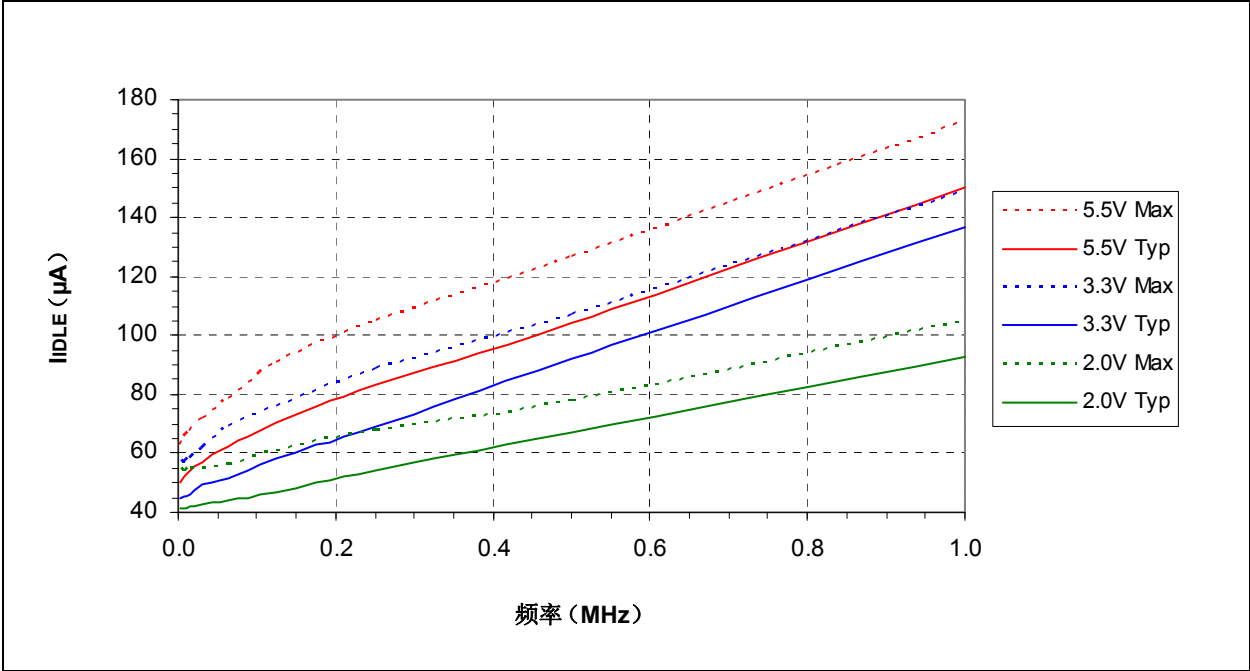


图 30-5: I_{DD} 的典型值与 V_{DD} 的关系曲线 (8 MHz 和 EC 模式)

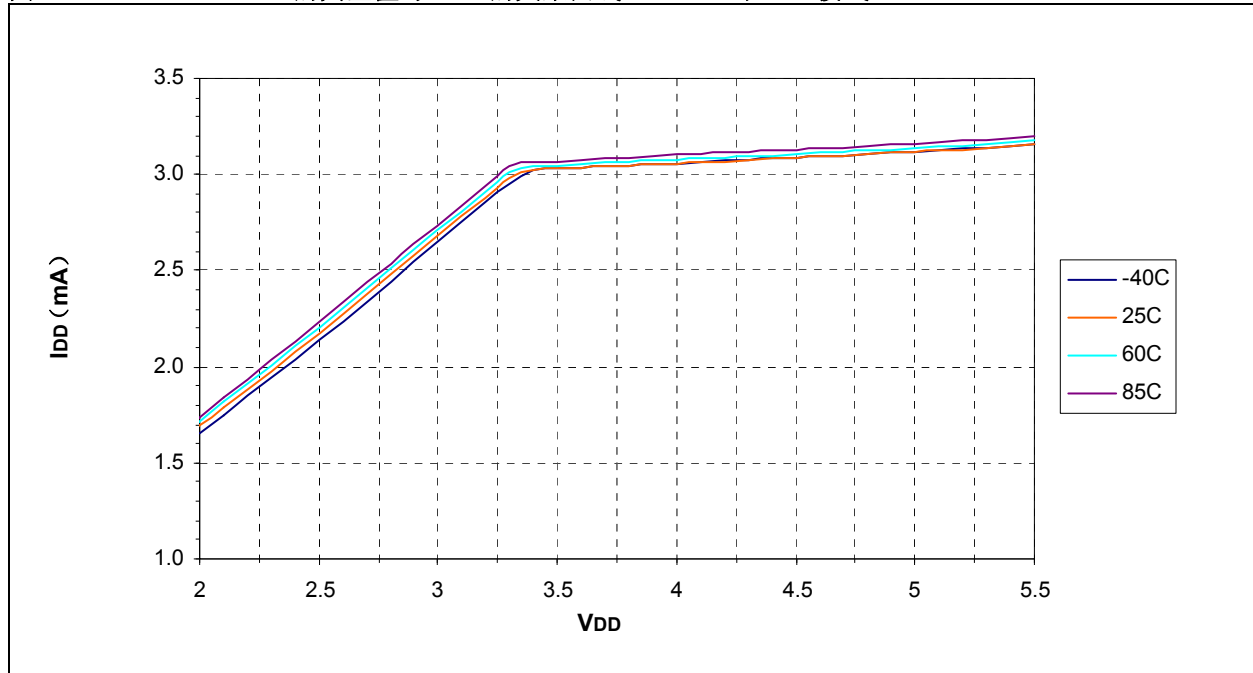
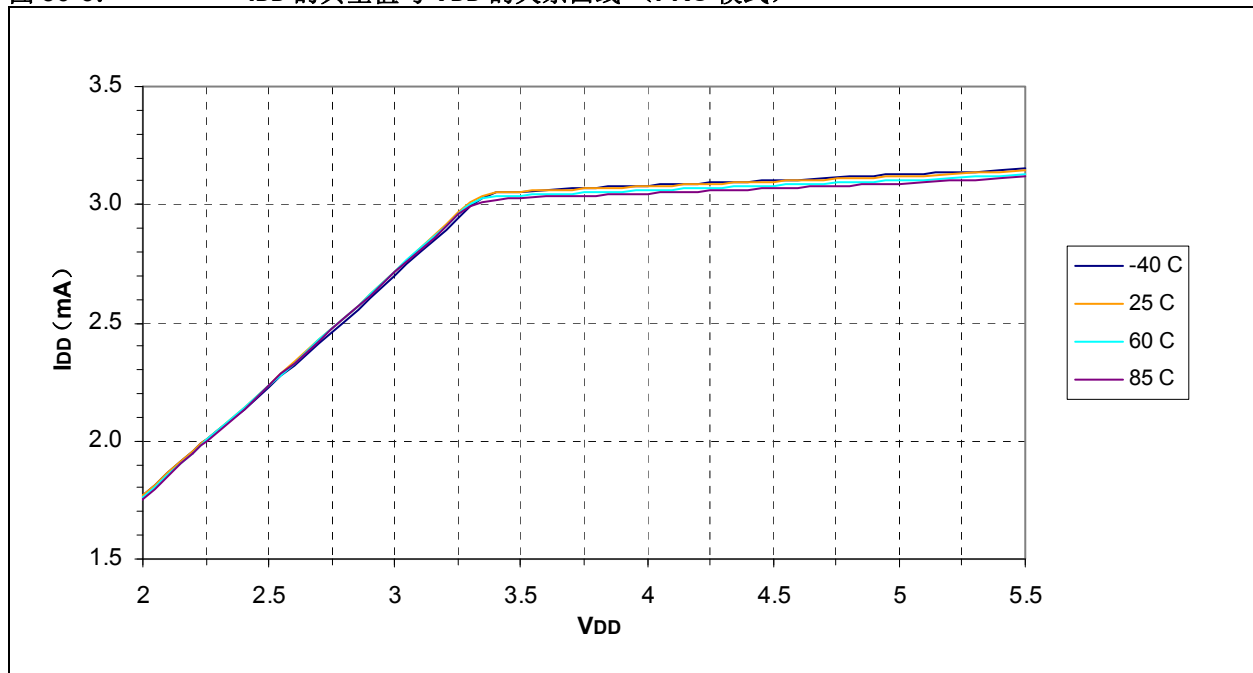


图 30-6: I_{DD} 的典型值与 V_{DD} 的关系曲线 (FRC 模式)



PIC24FV32KA304 系列

图 30-7: I_{DD} 的典型值和最大值与温度的关系曲线 (FRC 模式)

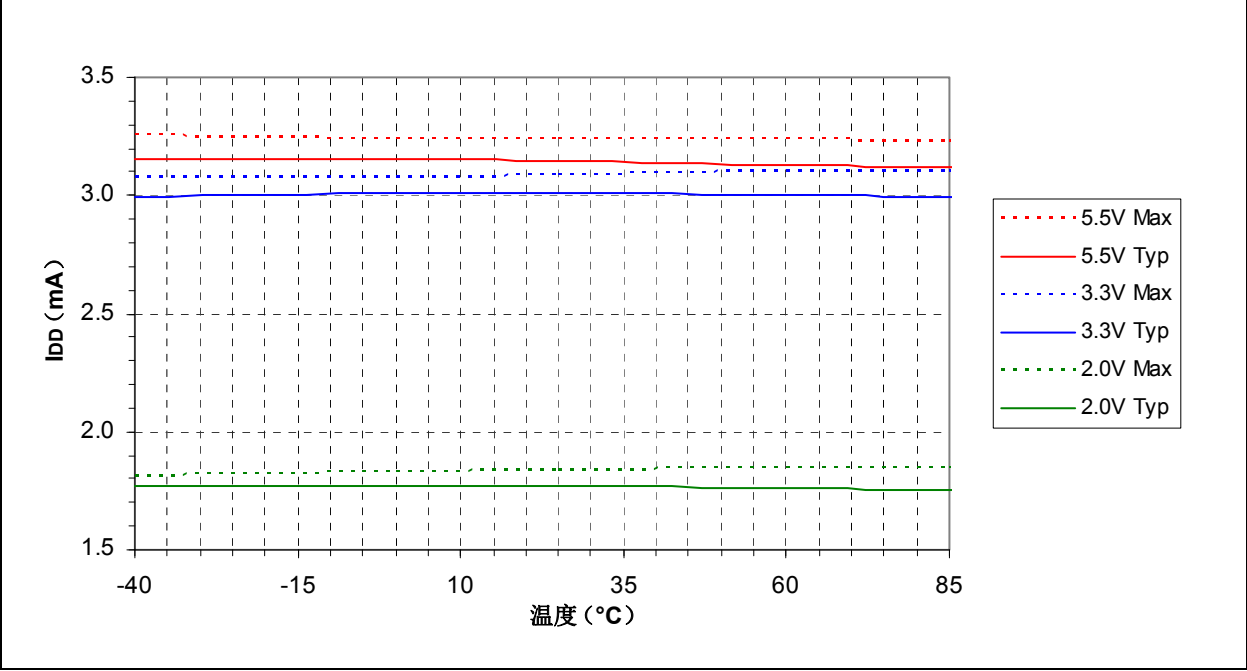


图 30-8: I_{IDLE} 的典型值和最大值与 V_{DD} 的关系曲线 (FRC 模式)

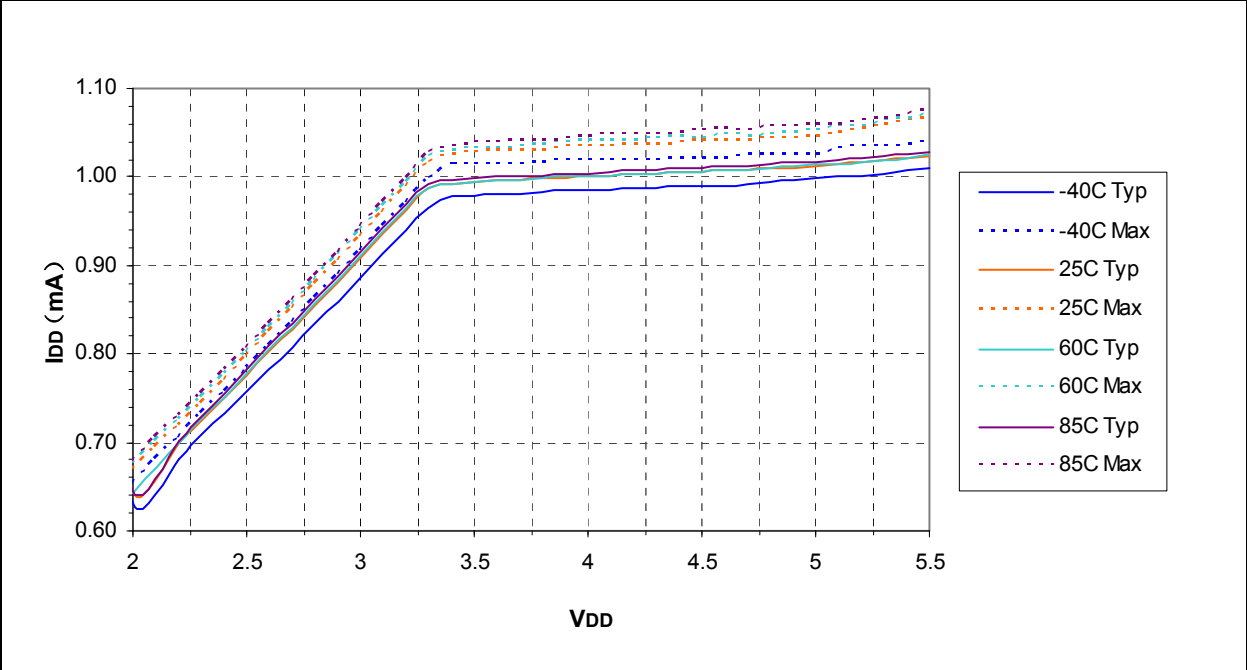
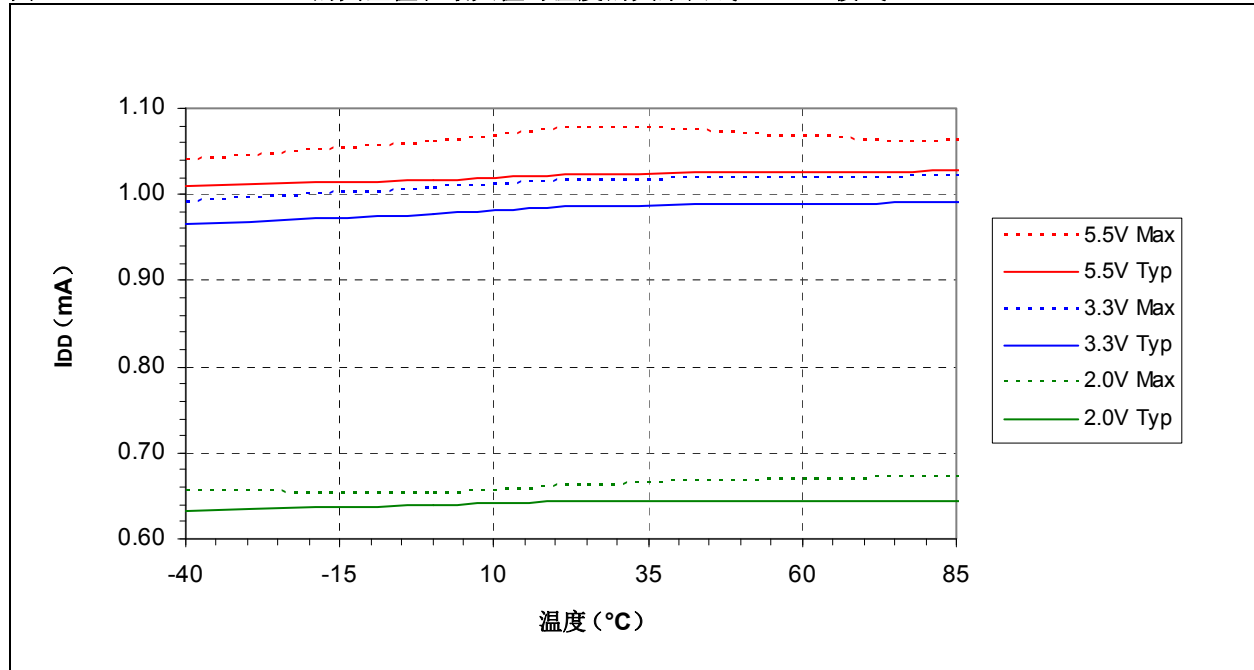


图 30-9: IDLE 的典型值和最大值与温度的关系曲线 (FRC 模式)



PIC24FV32KA304 系列

图 30-10: FRC 频率精度与 VDD 关系曲线

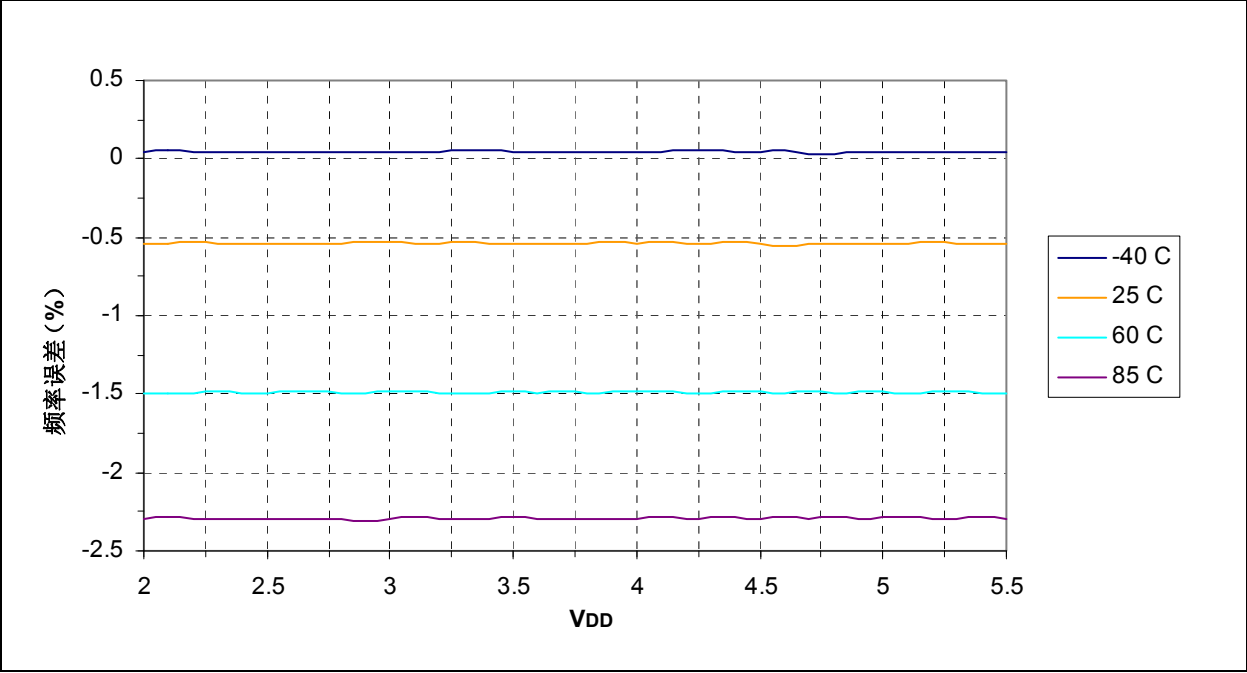


图 30-11: FRC 频率精度与温度的关系曲线 (2.0V ≤ VDD ≤ 5.5V)

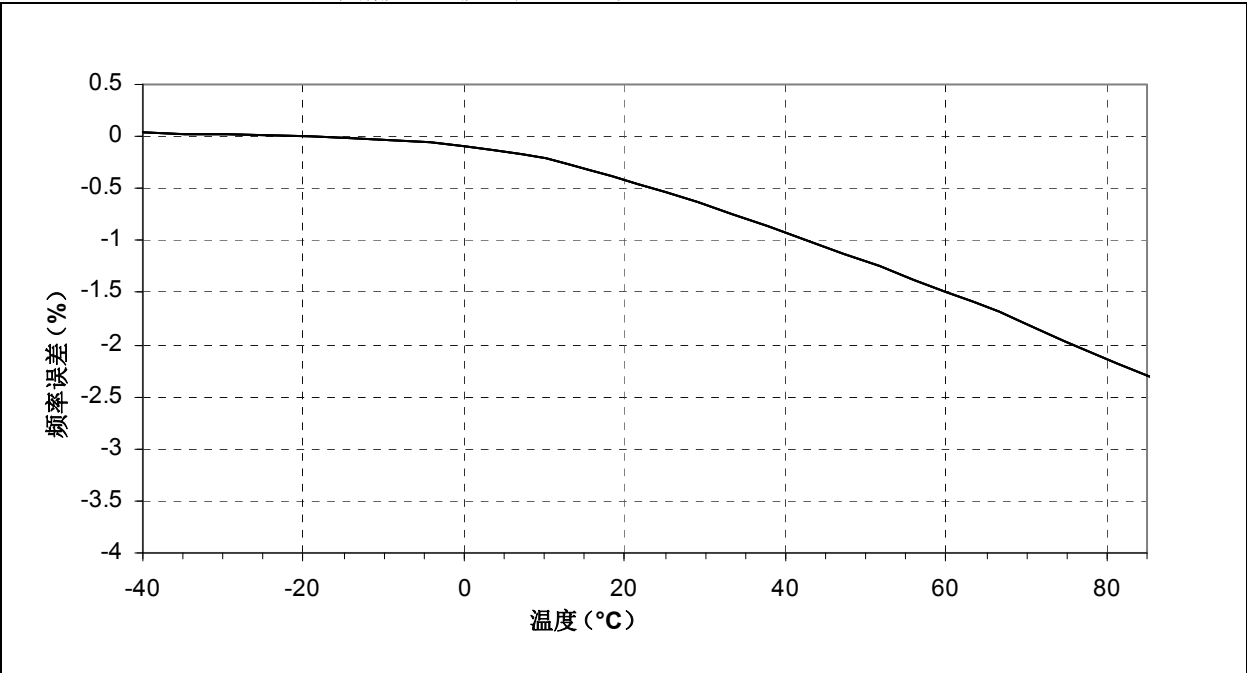


图 30-12: LPRC 频率精度与 VDD 关系曲线

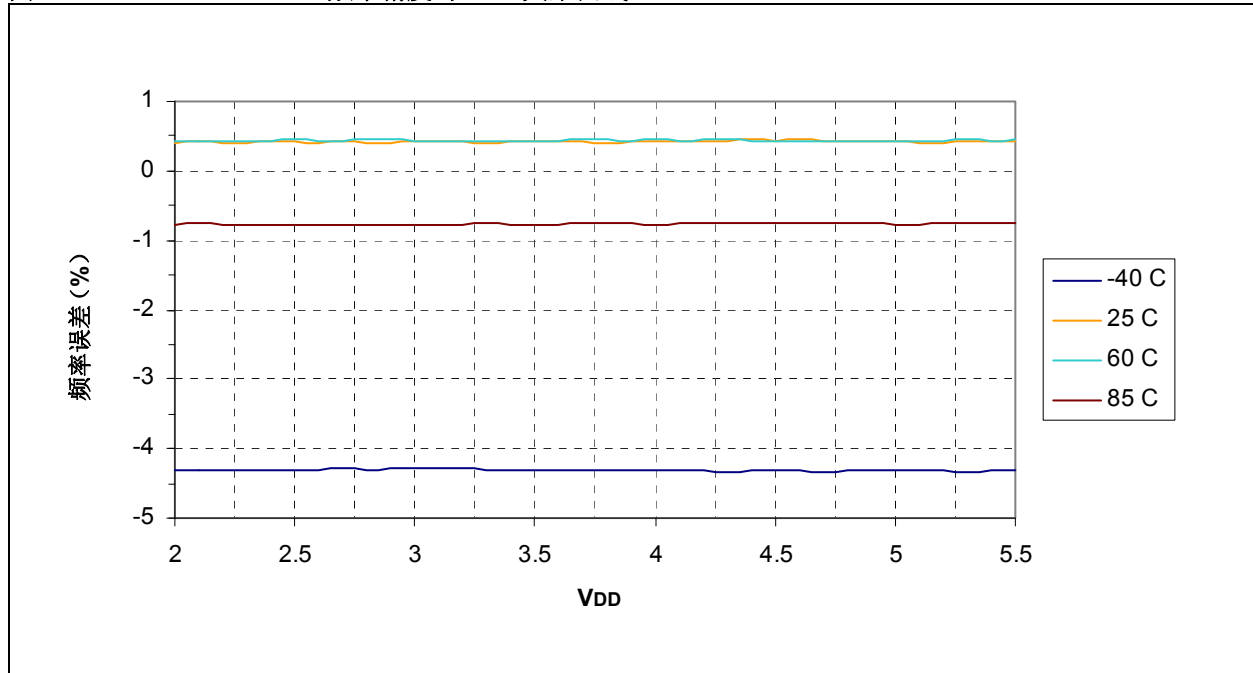
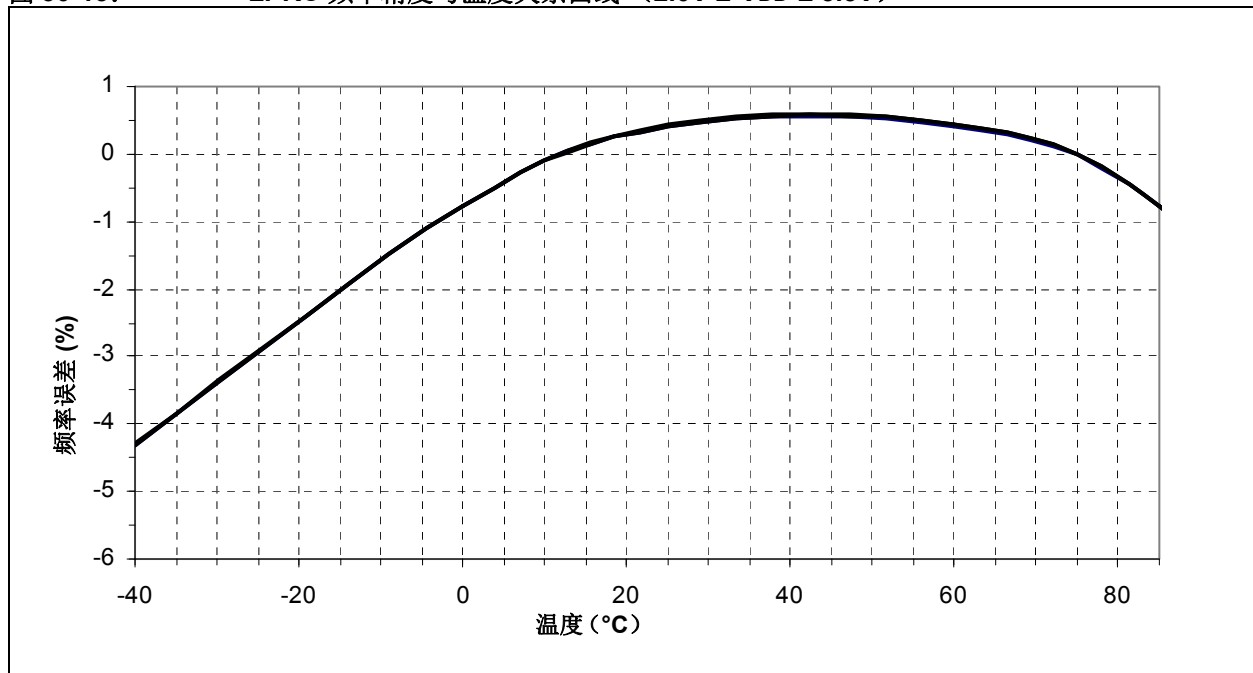


图 30-13: LPRC 频率精度与温度关系曲线 (2.0V ≤ VDD ≤ 5.5V)



PIC24FV32KA304 系列

图 30-14: IPD 的典型值和最大值与 VDD 的关系曲线

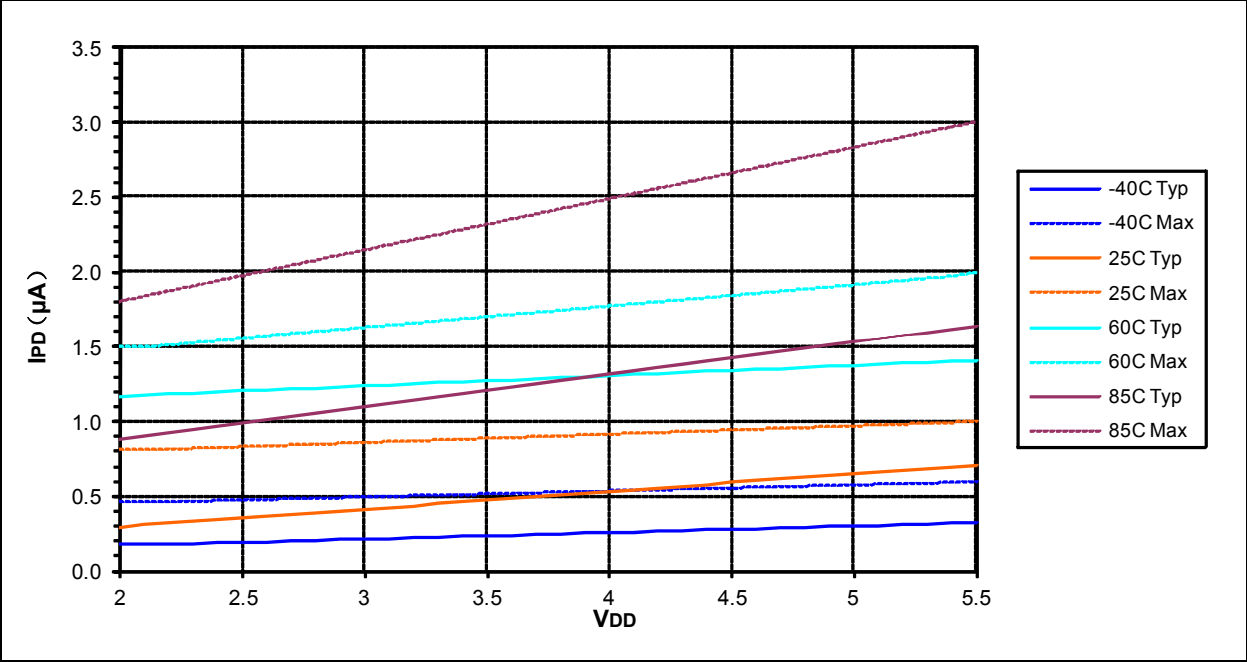


图 30-15: IPD 的典型值和最大值与温度的关系曲线

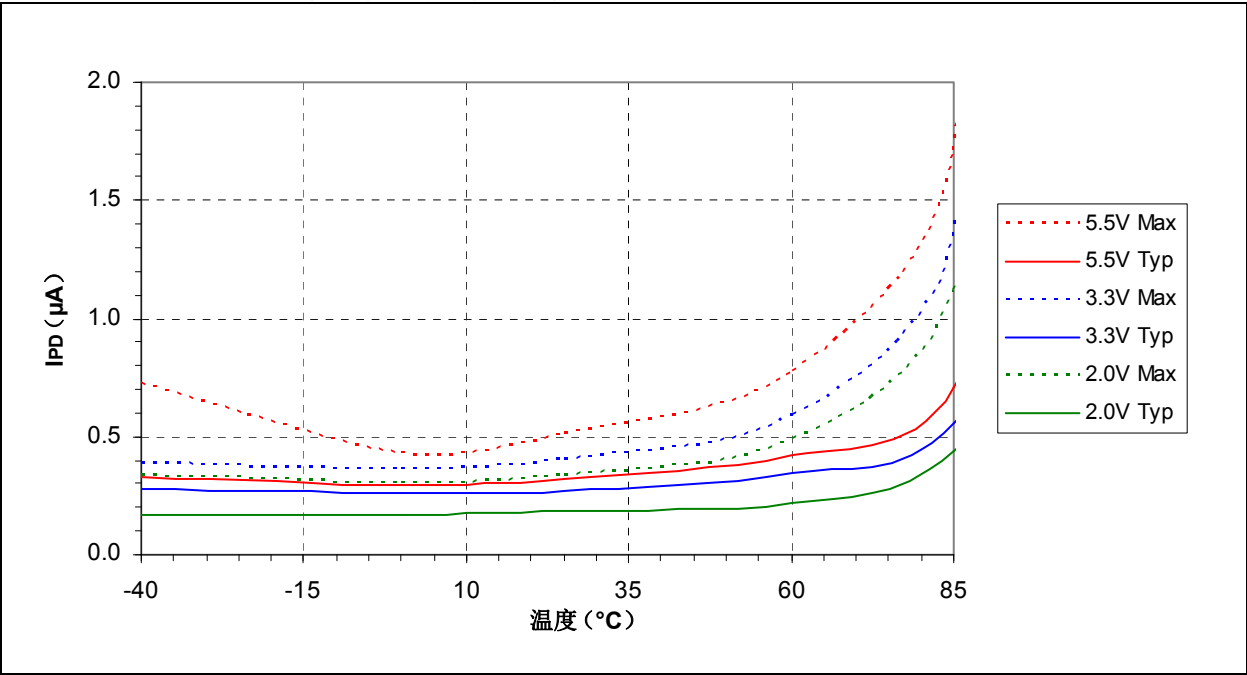


图 30-16: IPD 的典型值和最大值与 VDD 的关系曲线（深度休眠模式）

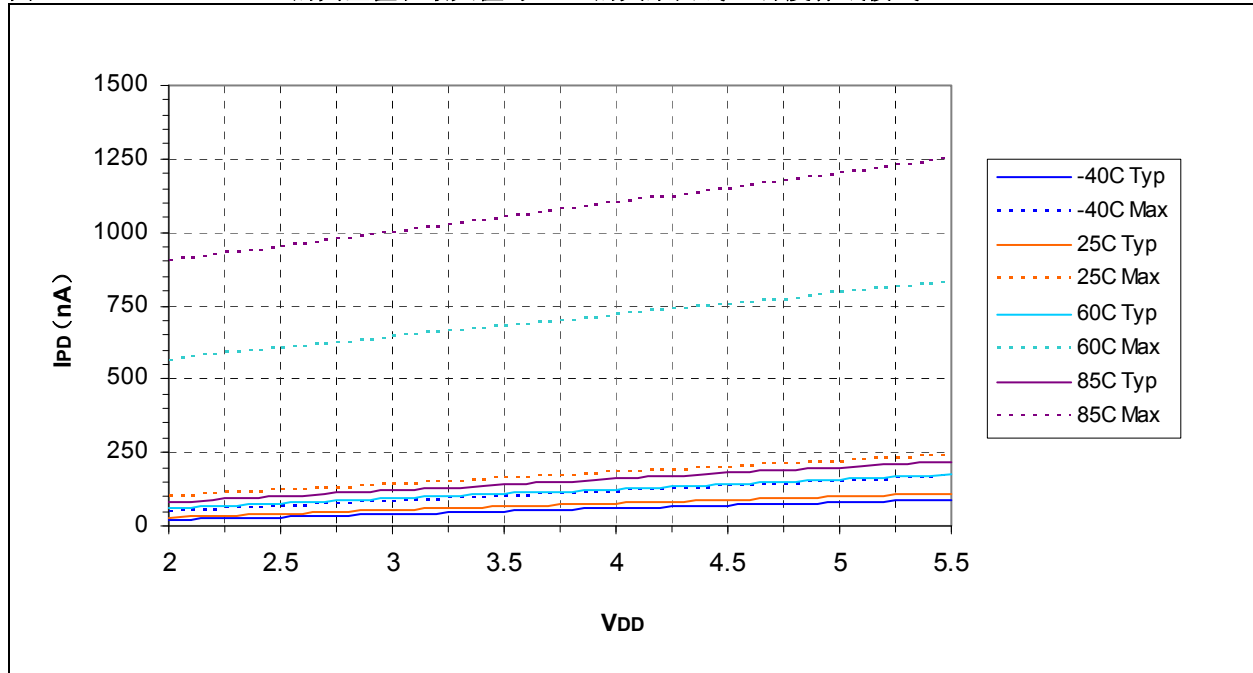
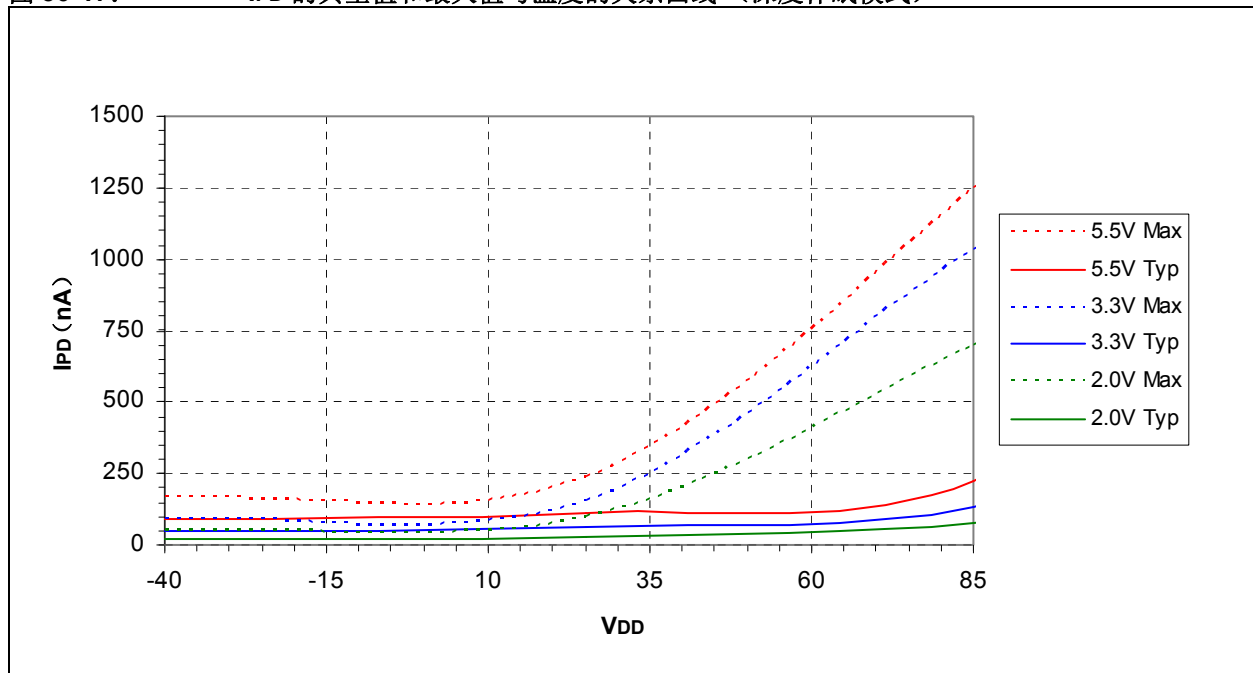


图 30-17: IPD 的典型值和最大值与温度的关系曲线（深度休眠模式）



PIC24FV32KA304 系列

图 30-18: ΔI_{BOR} 的典型值与 V_{DD} 的关系曲线

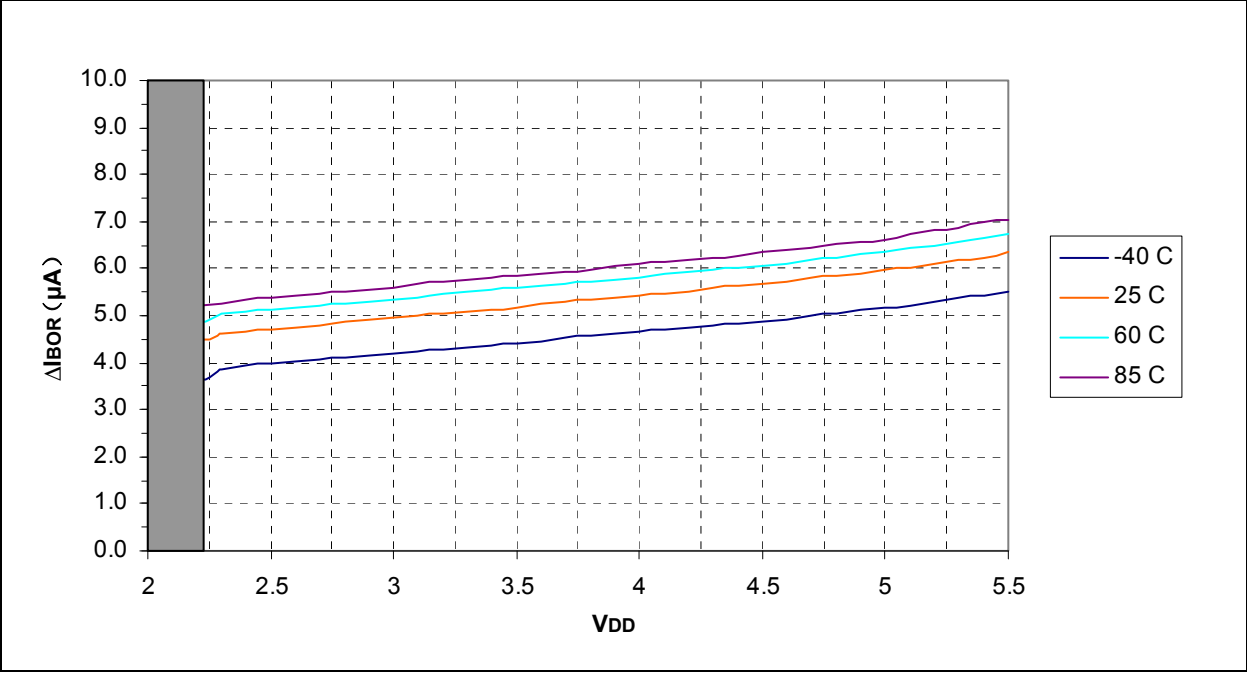


图 30-19: ΔI_{WDT} 的典型值与 V_{DD} 的关系曲线

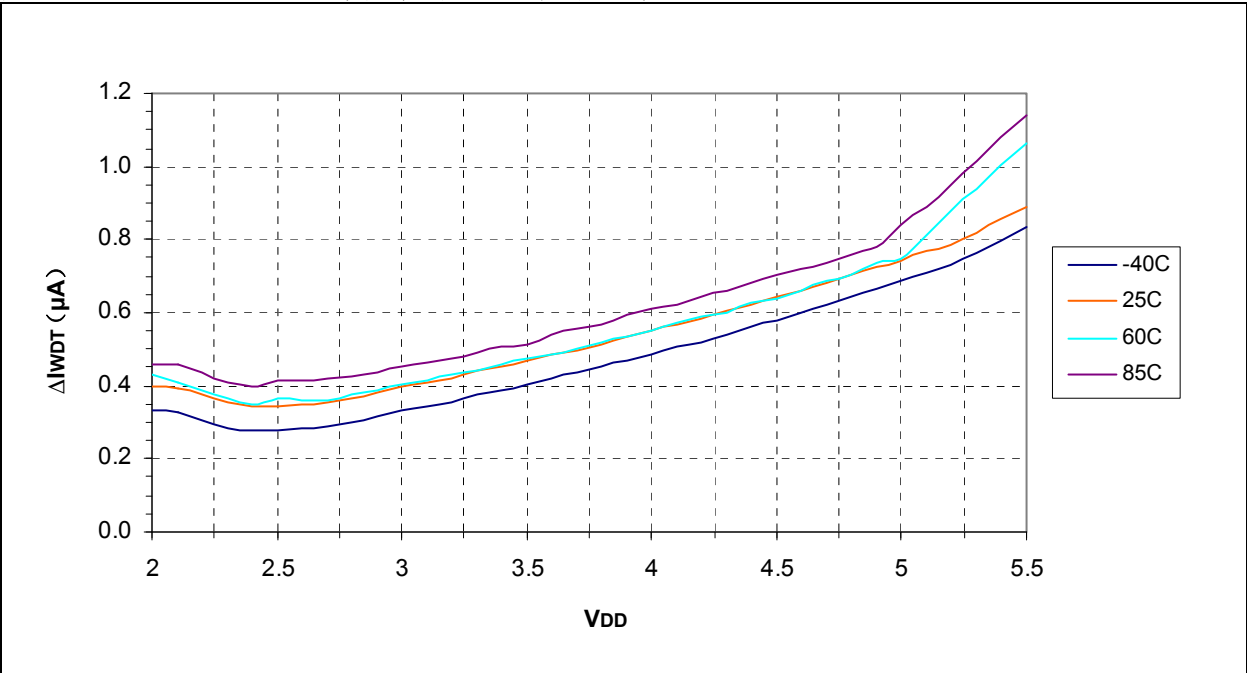


图 30-20: ΔI_{DSBOR} 的典型值与 V_{DD} 的关系曲线

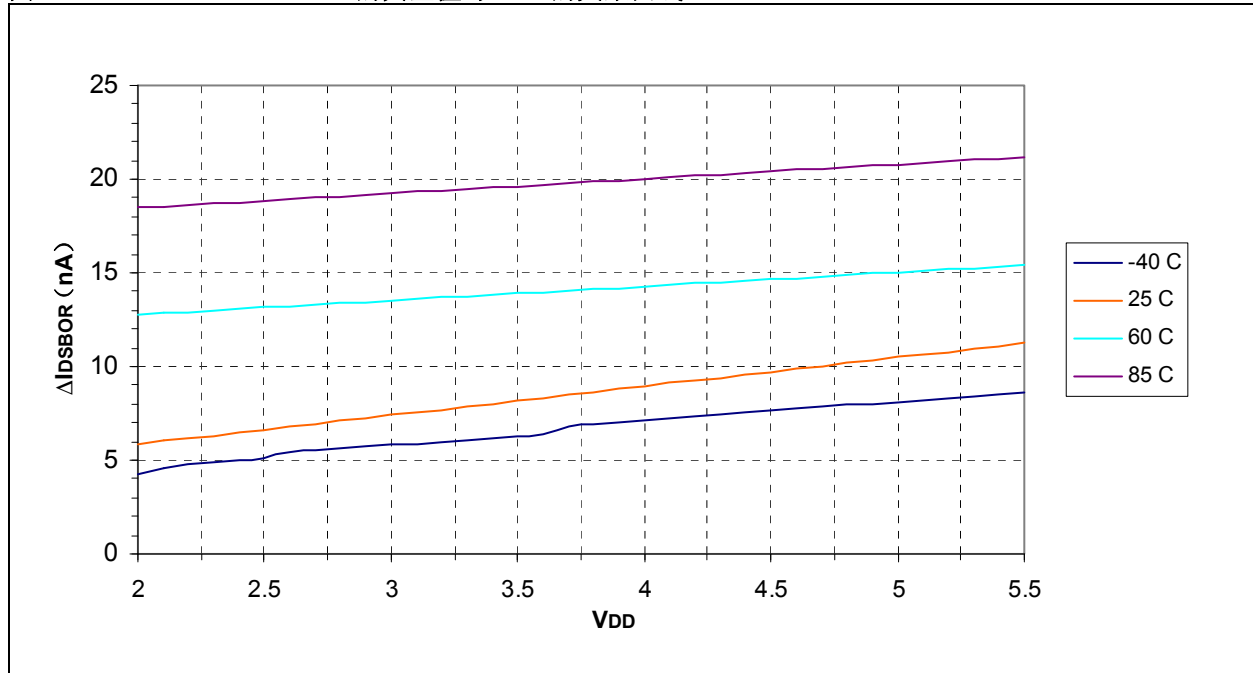
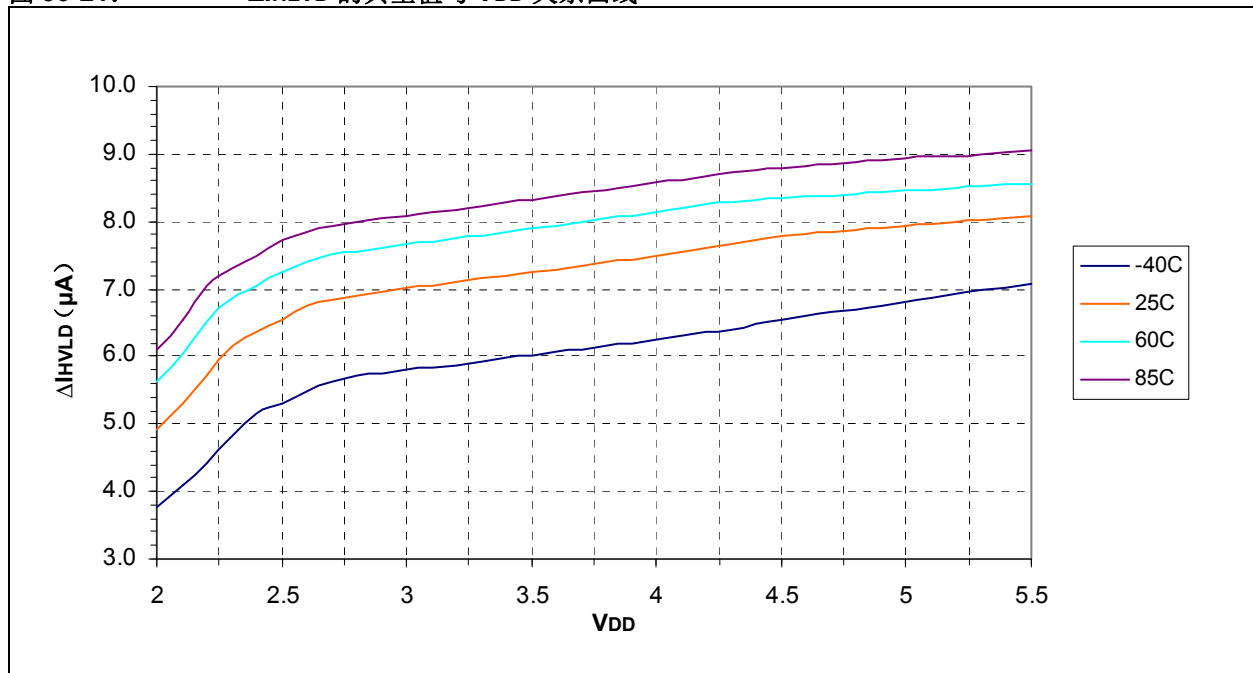


图 30-21: ΔI_{HLVD} 的典型值与 V_{DD} 关系曲线



PIC24FV32KA304 系列

图 30-22: ΔI_{DSWDT} 的典型值与 V_{DD} 关系曲线

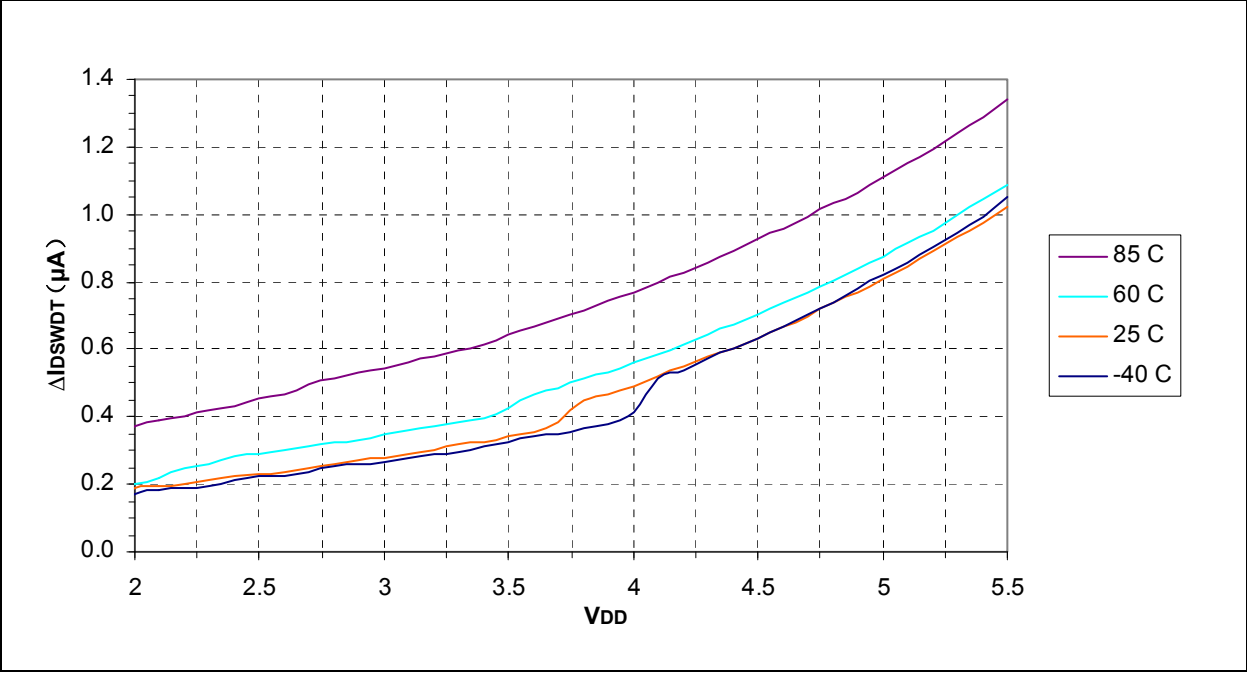


图 30-23: V_{BOR} 的典型值与温度的关系曲线 (BOR 跳变点 3)

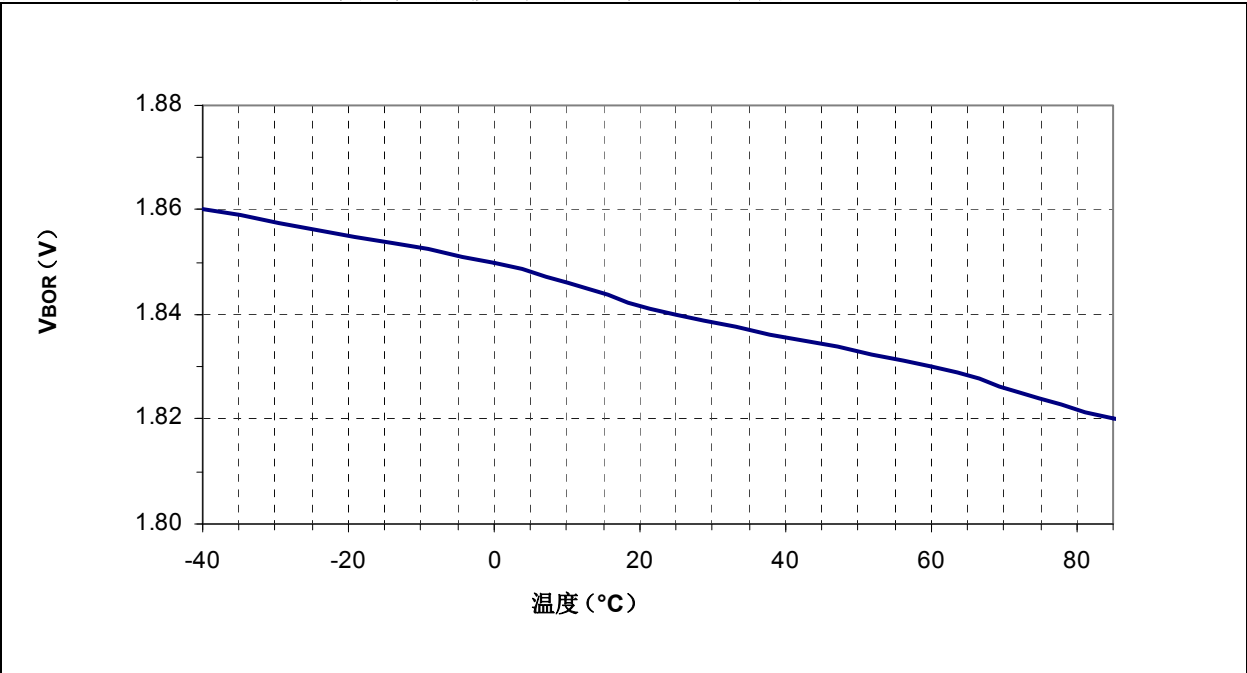


图 30-24: V_{OH} 的典型值与 I_{OH} 的关系曲线 (通用 I/O, 随 V_{DD} 变化)

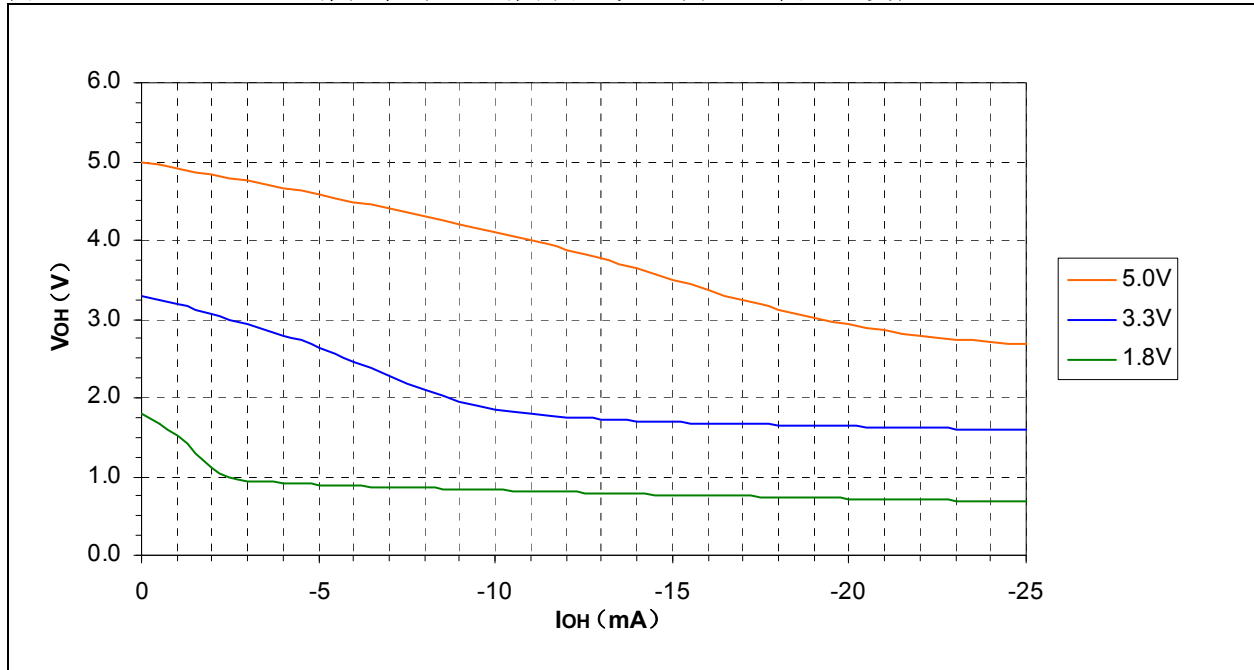
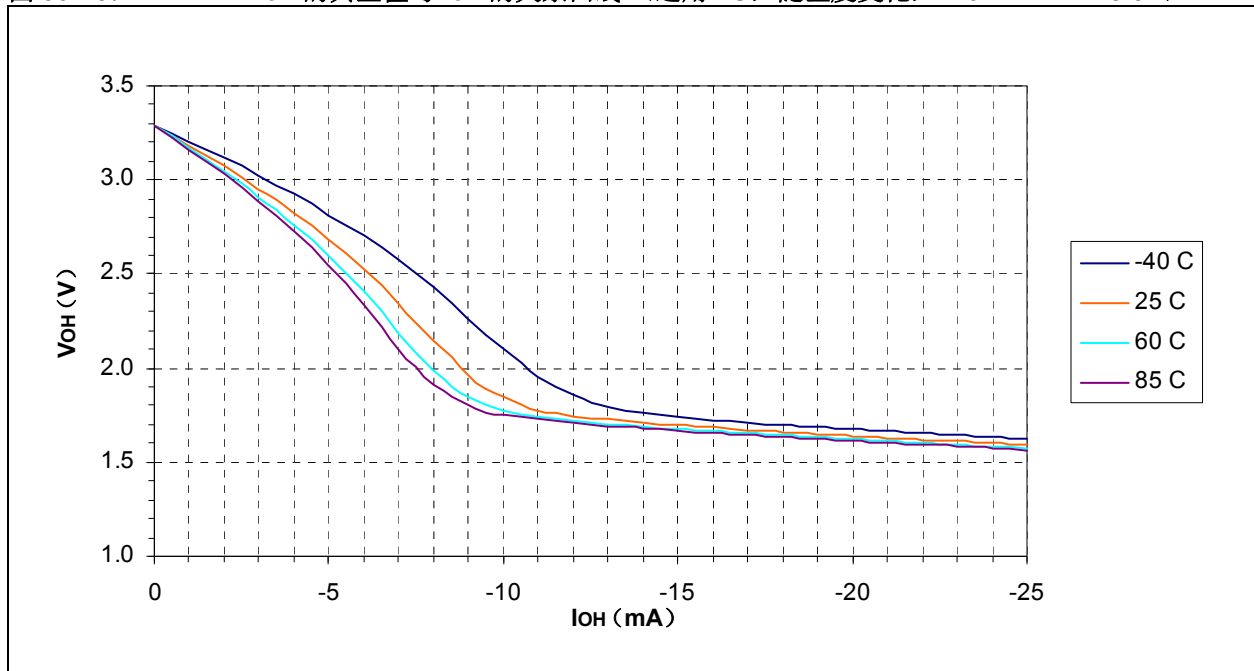


图 30-25: V_{OH} 的典型值与 I_{OH} 的关系曲线 (通用 I/O, 随温度变化, $2.0V \leq V_{DD} \leq 5.5V$)



PIC24FV32KA304 系列

图 30-26: V_{OL} 的典型值与 I_{OL} 的关系曲线（通用 I/O，随 V_{DD} 变化）

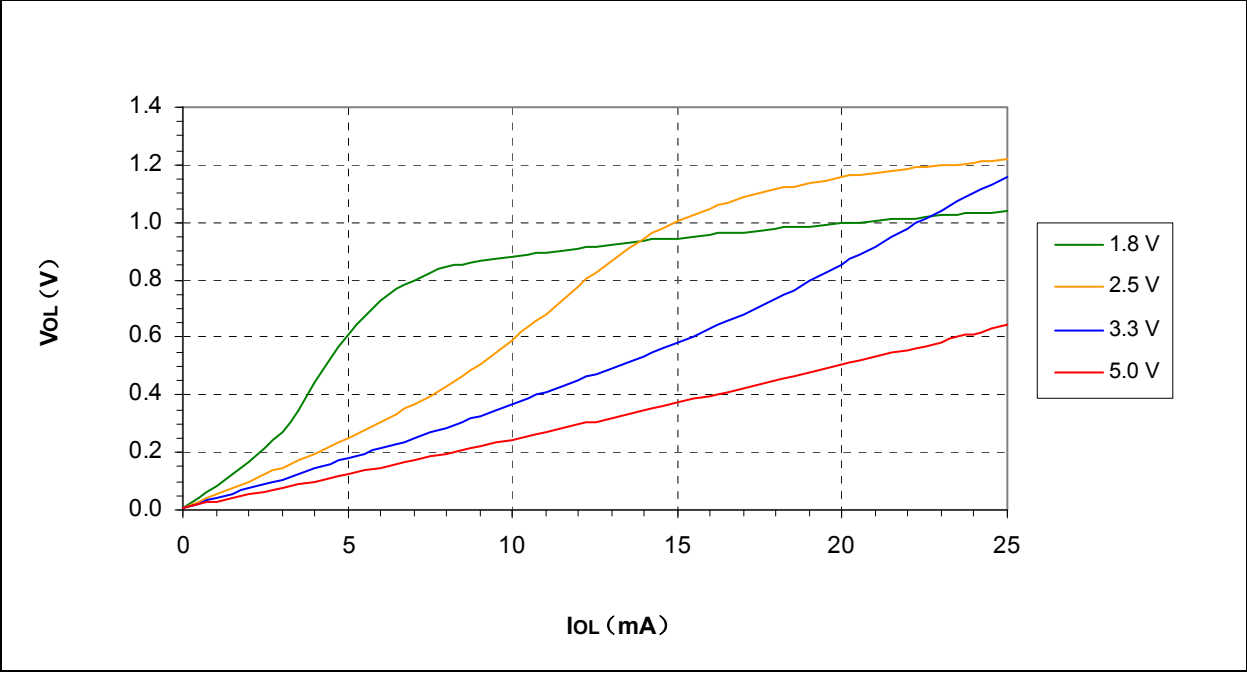


图 30-27: V_{OL} 的典型值与 I_{OL} 关系曲线（通用 I/O，随温度变化， $2.0V \leq V_{DD} \leq 5.5V$ ）

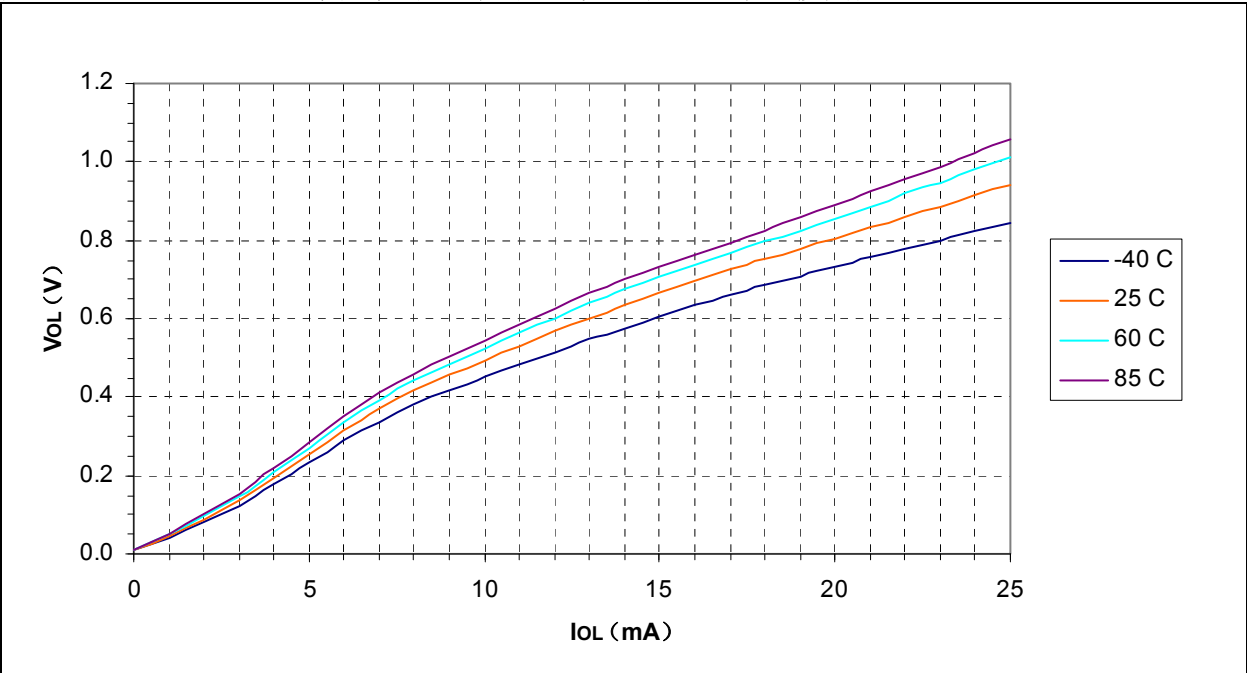


图 30-28: V_{IL}/V_{IH} 与 V_{DD} 的关系曲线 (通用 I/O, 温度如标注所示)

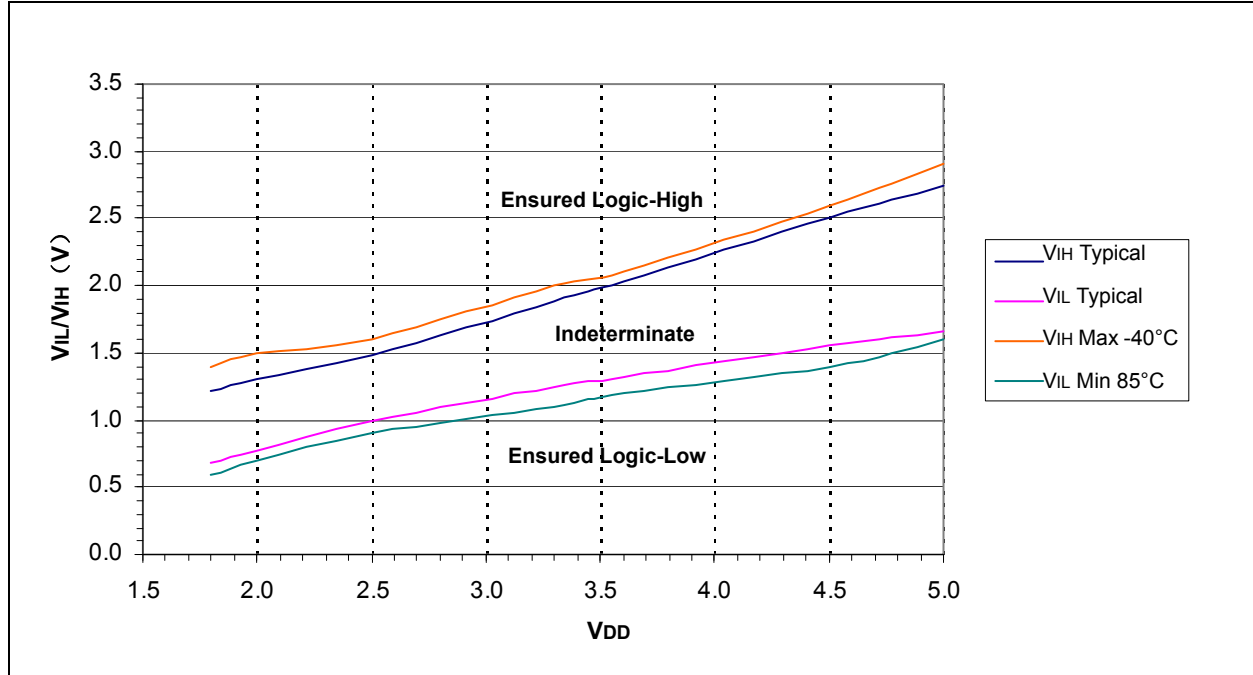
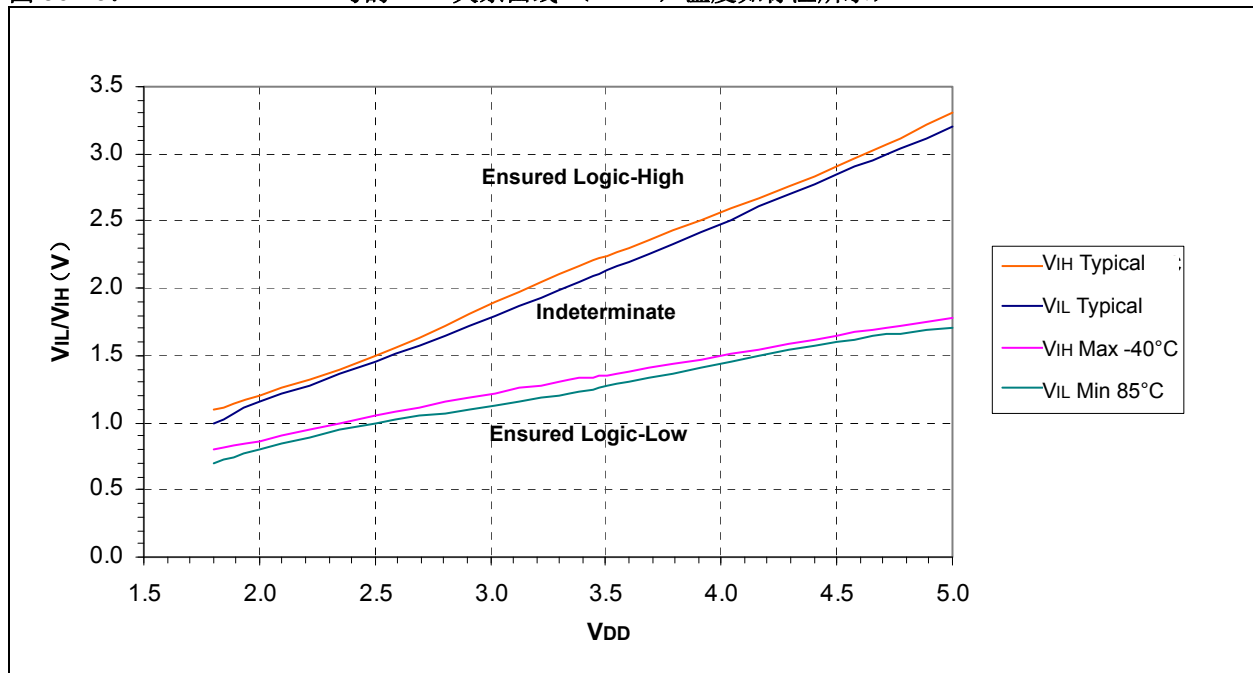


图 30-29: V_{IL}/V_{IH} 与的 V_{DD} 关系曲线 (I^2C^{TM} , 温度如标注所示)



PIC24FV32KA304 系列

图 30-30: V_{IL}/V_{IH} 与 V_{DD} 的关系曲线（OSCO，温度如标注所示）

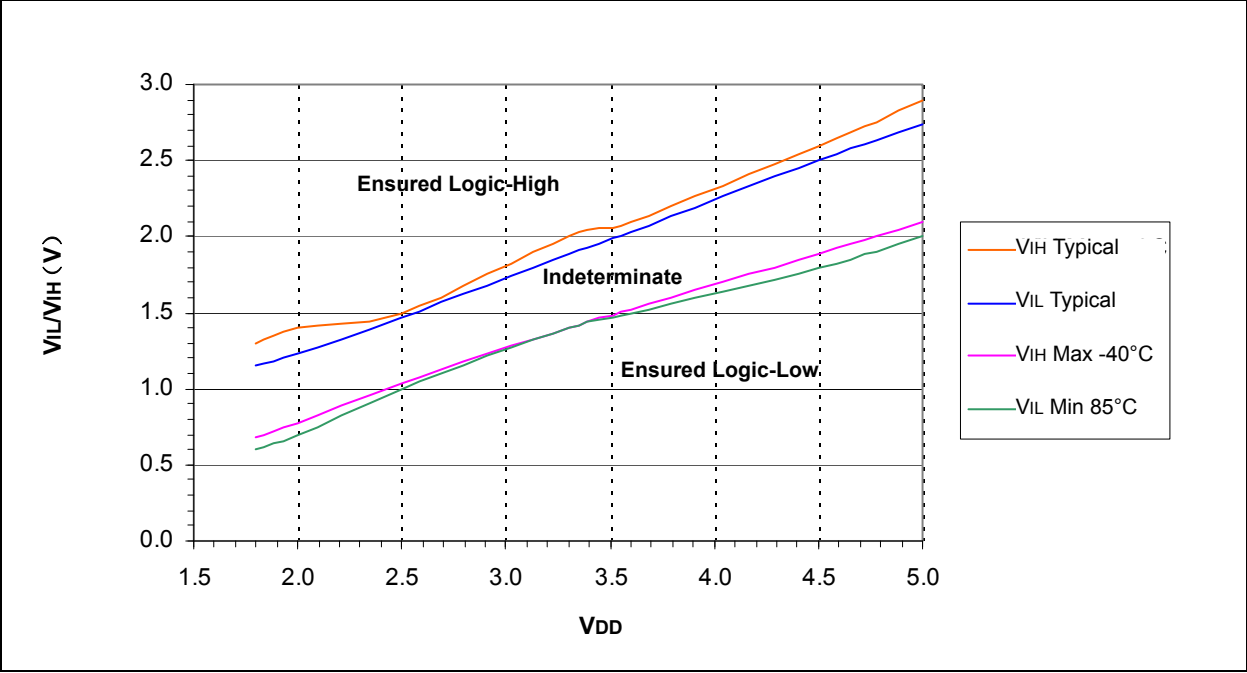


图 30-31: V_{IL}/V_{IH} 与 V_{DD} 的关系曲线（MCLR，温度如标注所示）

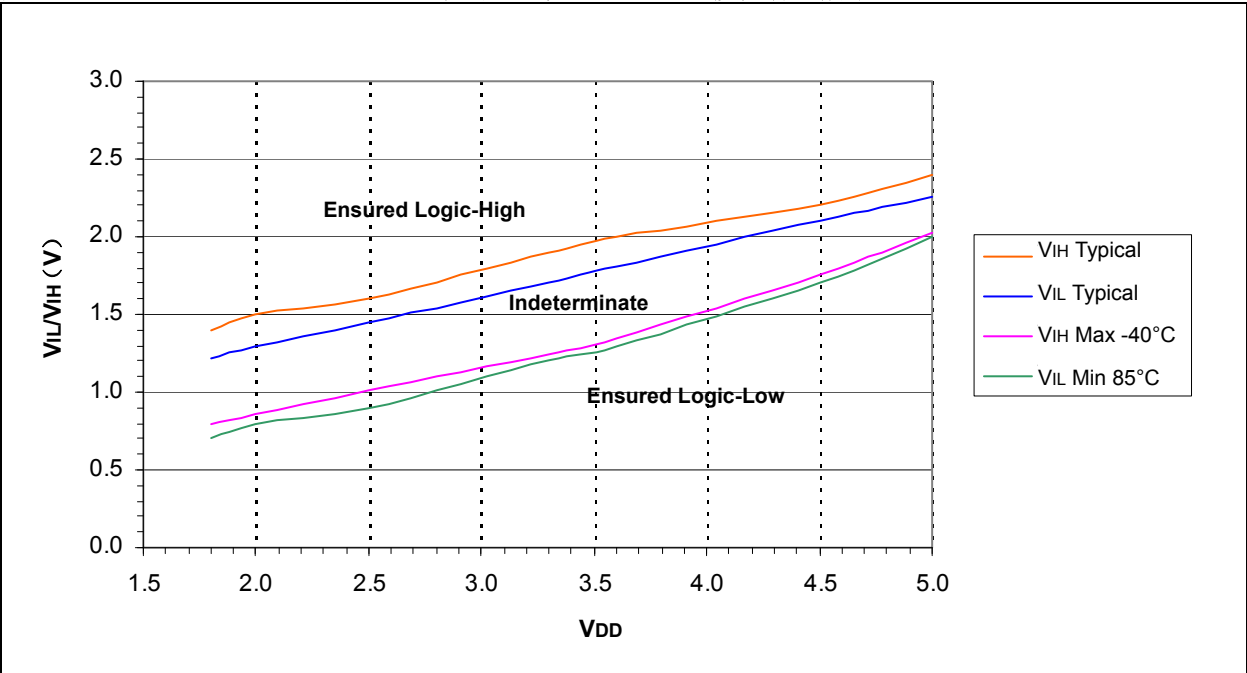


图 30-32: 典型带隙电压与 VDD 的关系曲线

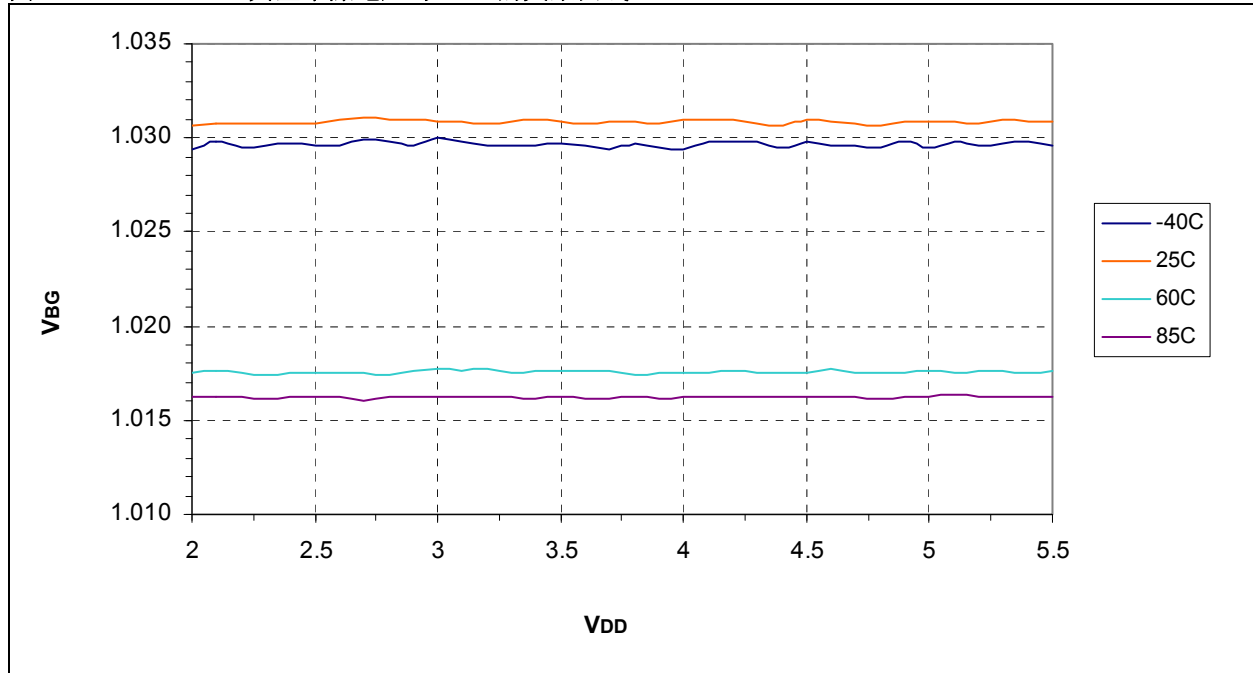
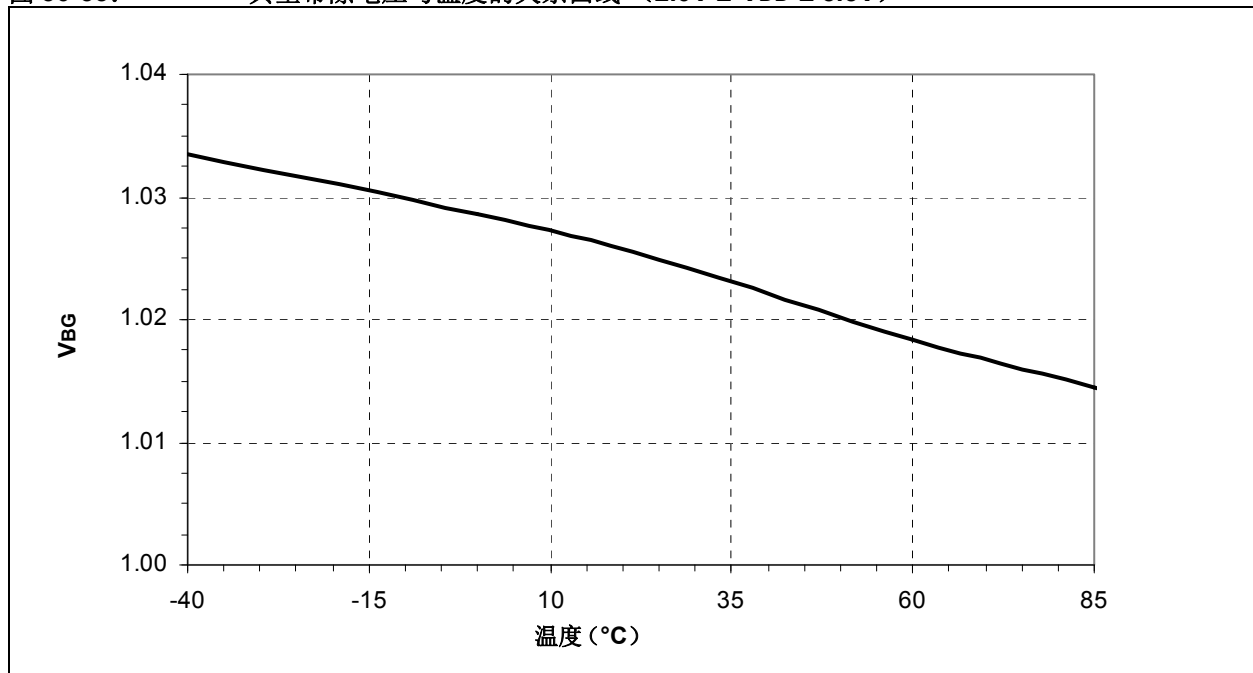


图 30-33: 典型带隙电压与温度的关系曲线 (2.0V ≤ VDD ≤ 5.5V)



PIC24FV32KA304 系列

图 30-34: 典型稳压器输出与 VDD 关系曲线

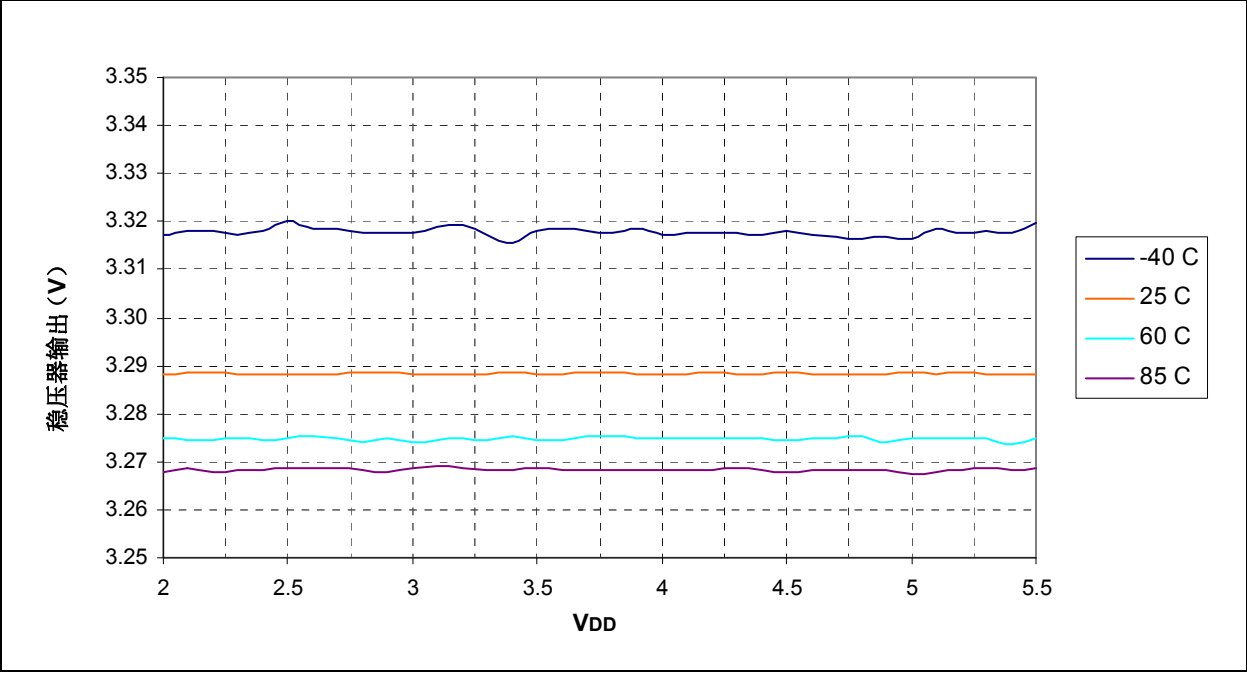


图 30-35: 典型稳压器输出与温度关系曲线

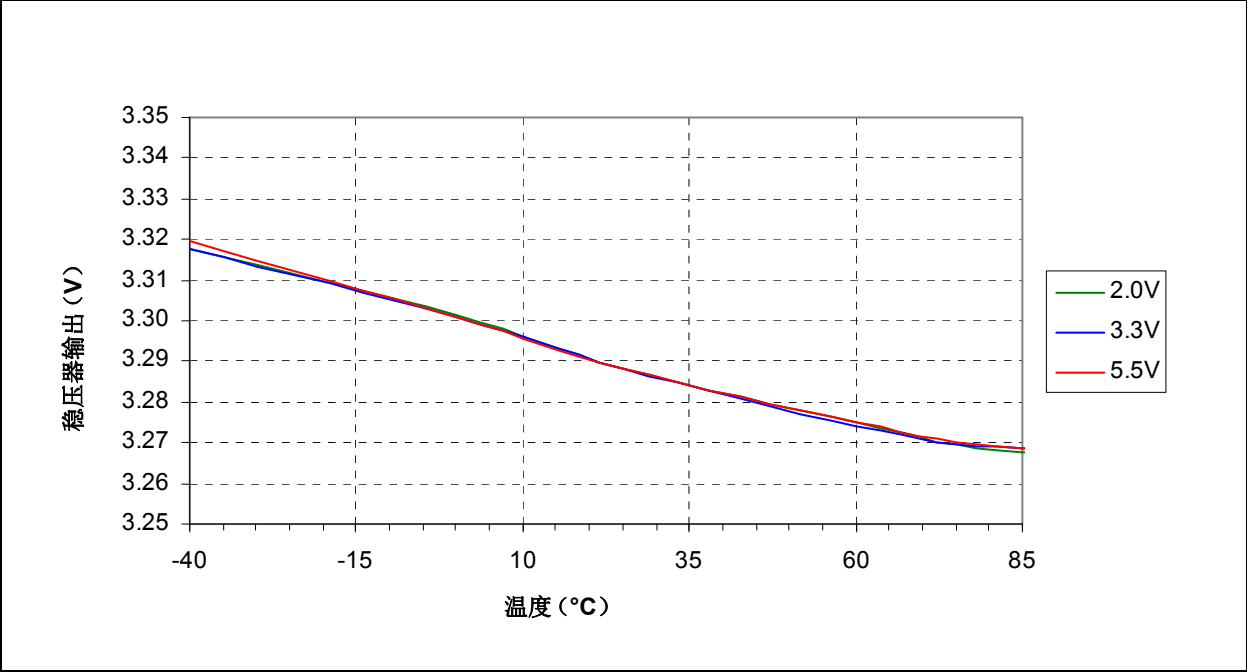


图 30-36: HLVD 跳变点电压与温度关系曲线 (HLVDL<3:0> = 0000, 仅限 PIC24F32KA304 系列器件)

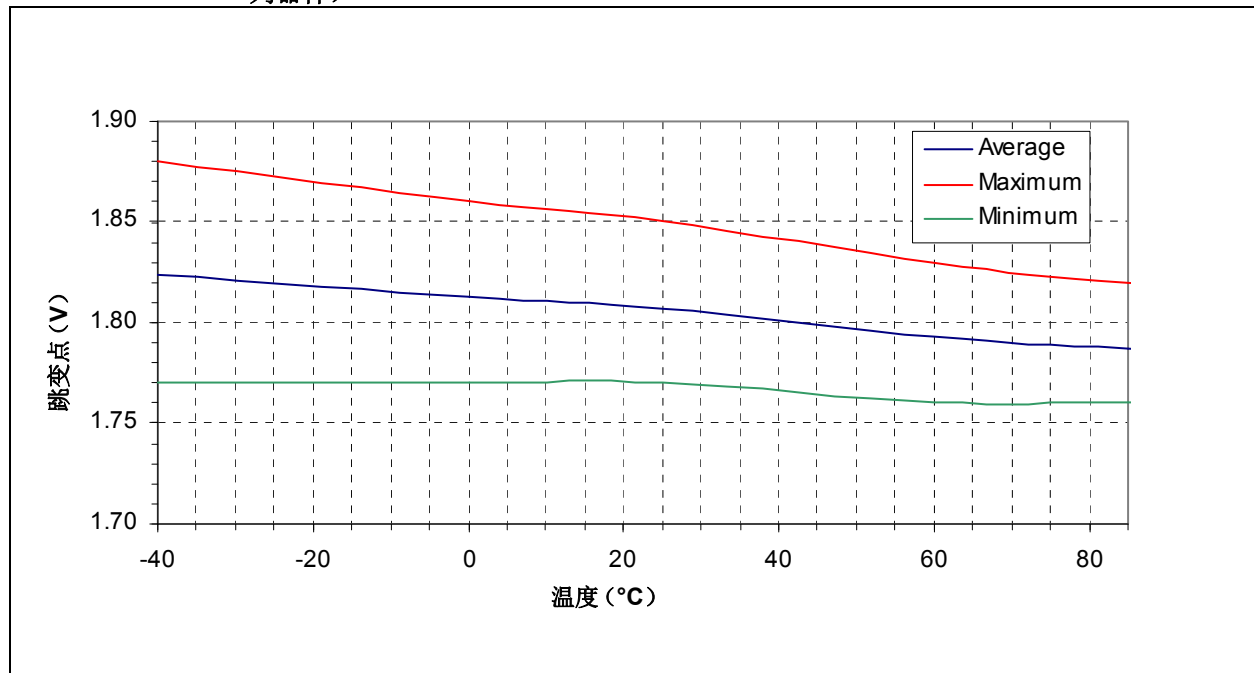
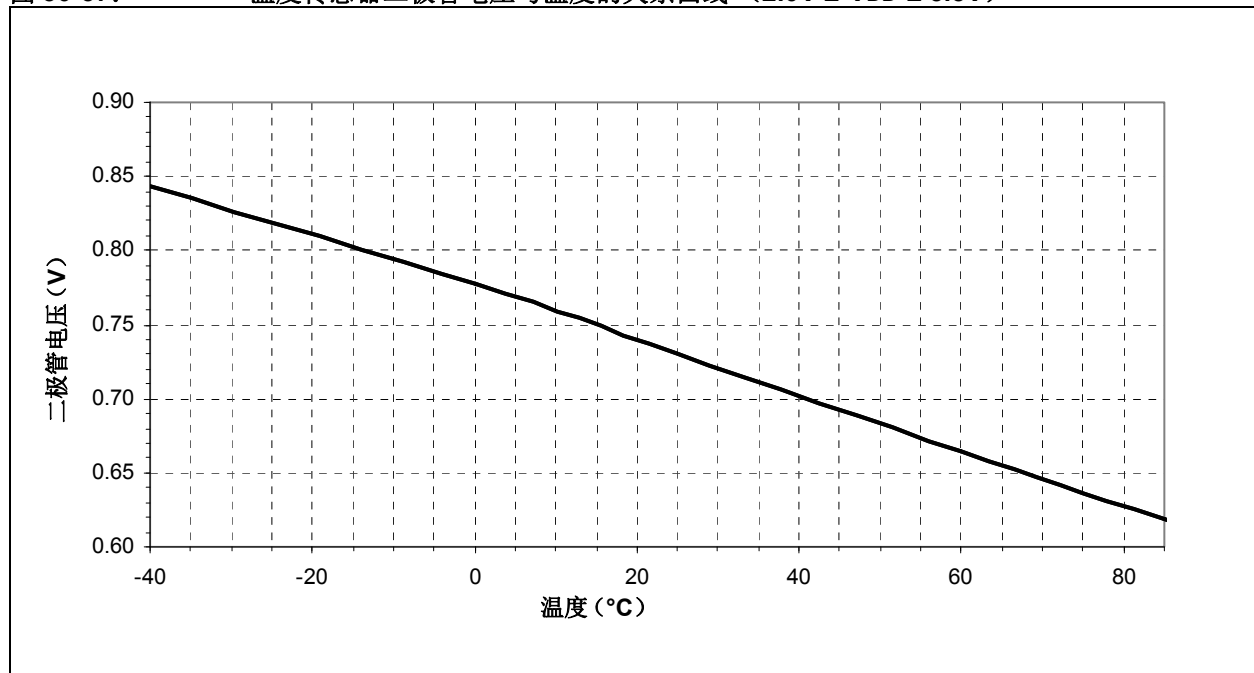


图 30-37: 温度传感器二极管电压与温度的关系曲线 ($2.0V \leq V_{DD} \leq 5.5V$)



PIC24FV32KA304 系列

图 30-38: CTMU 输出电流与温度的关系曲线 (IRNG<1:0> = 01, 2.0V ≤ VDD ≤ 5.5V)

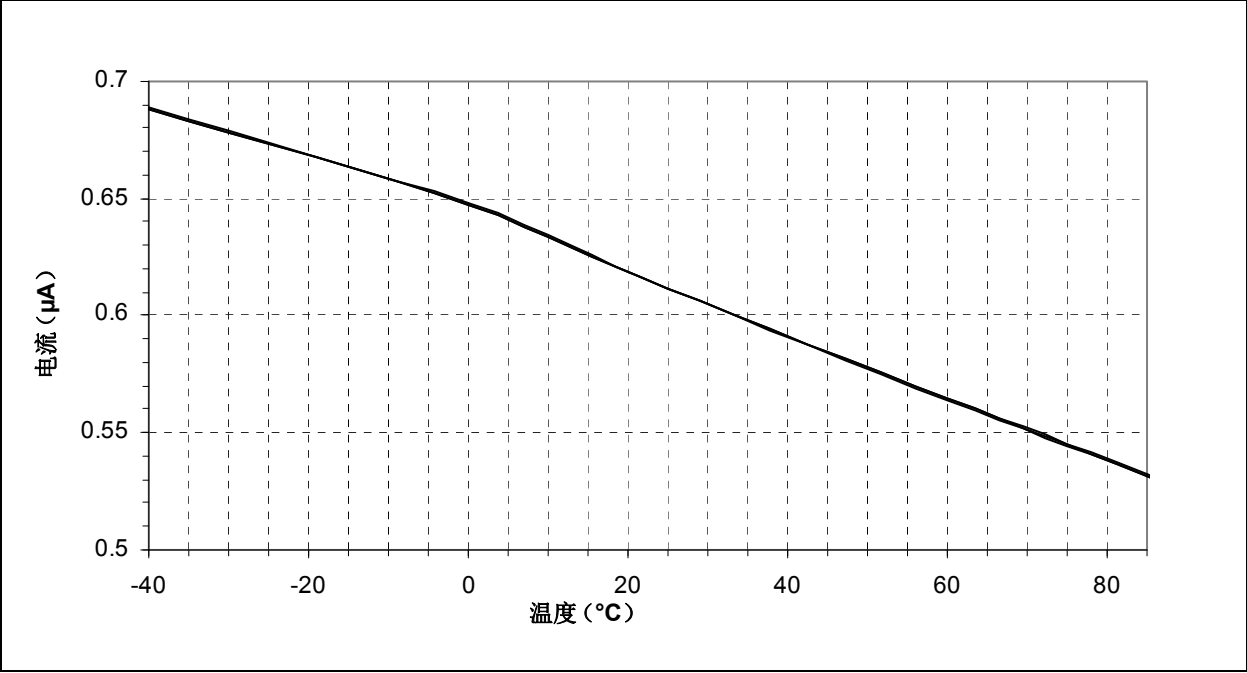
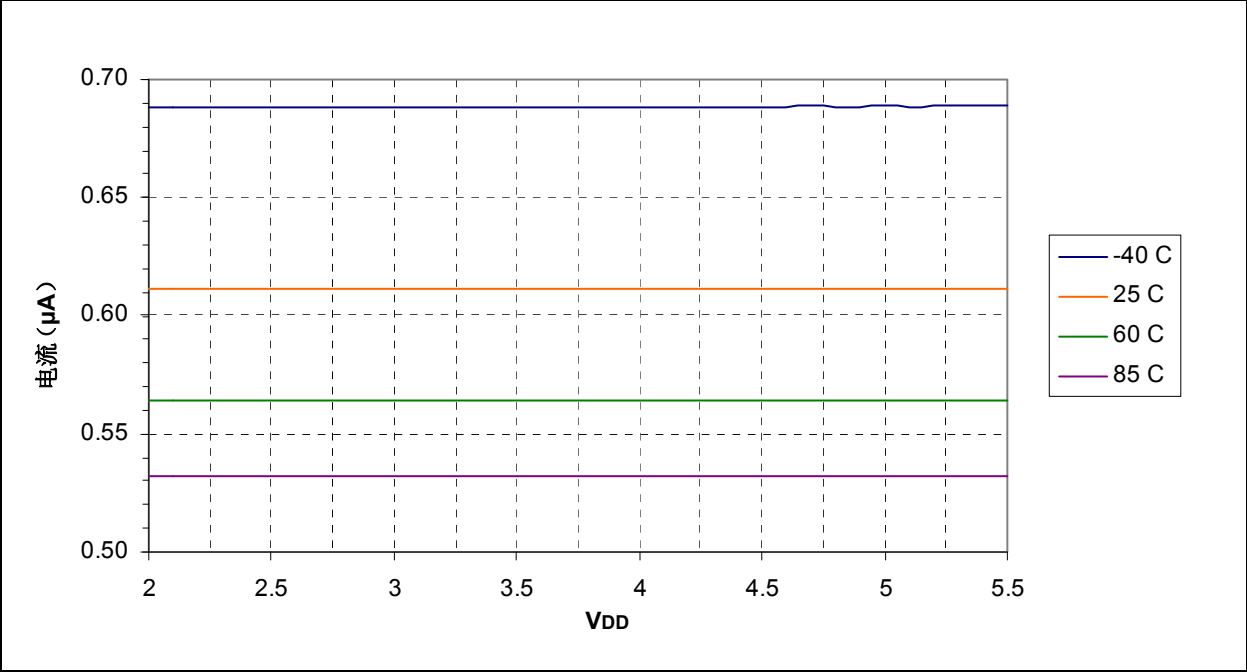


图 30-39: CTMU 输出电流与 VDD 的关系曲线 (IRNG<1:0> = 01)



30.2 扩展级温度器件的特性 (-40°C 至 +125°C)

注：VDD 电平高于 3.3V 的数据仅适用于 PIC24FV32KA304 系列器件。

图 30-40: 典型和最大 I_{IDLE} 与 VDD 的关系曲线 (FRC 模式)

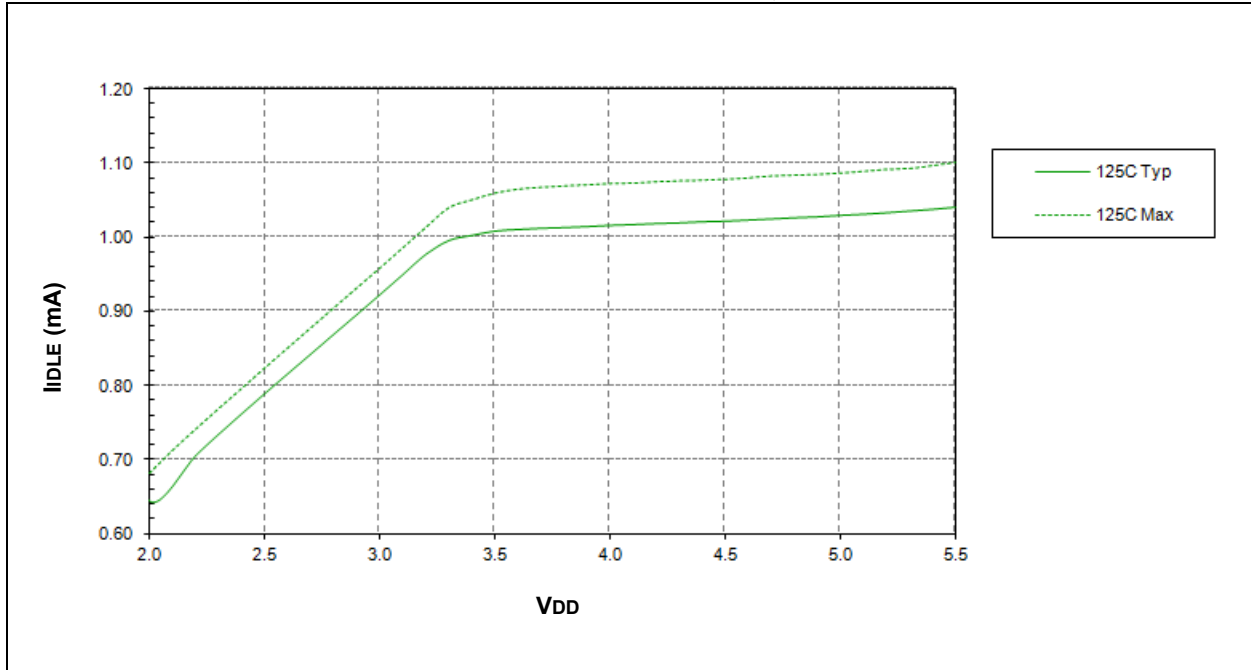
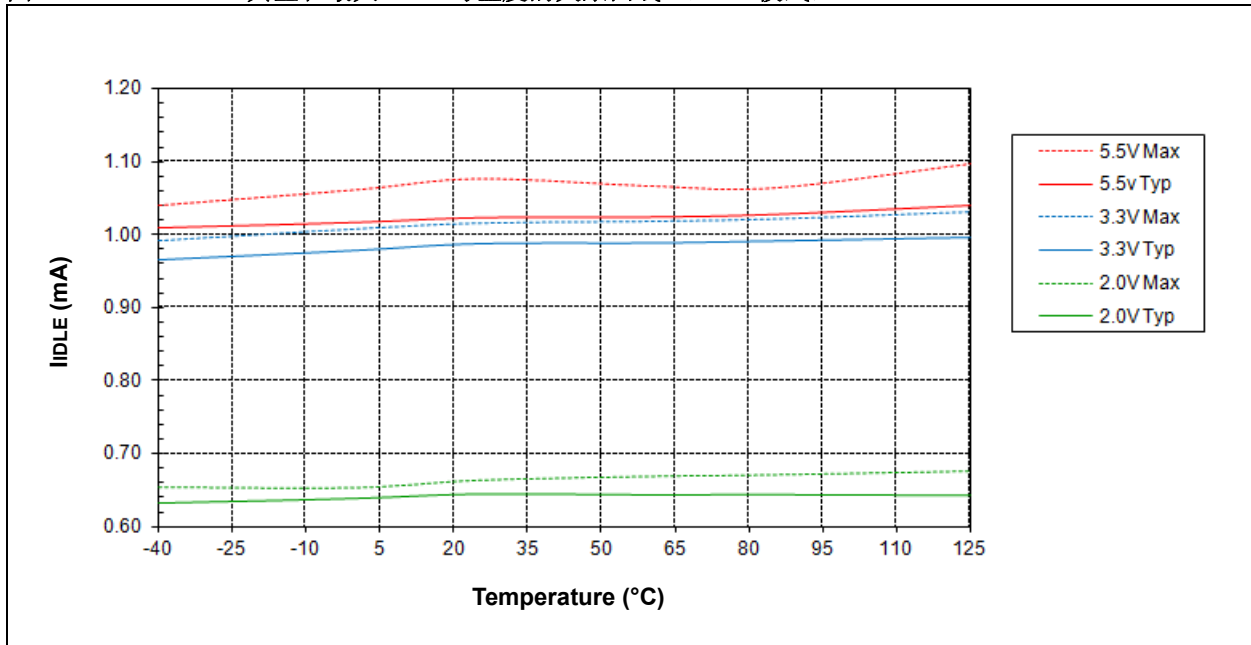


图 30-41: 典型和最大 I_{IDLE} 与温度的关系曲线 (FRC 模式)



PIC24FV32KA304 系列

图 30-42: 典型和最大 IPD 与 VDD 的关系曲线

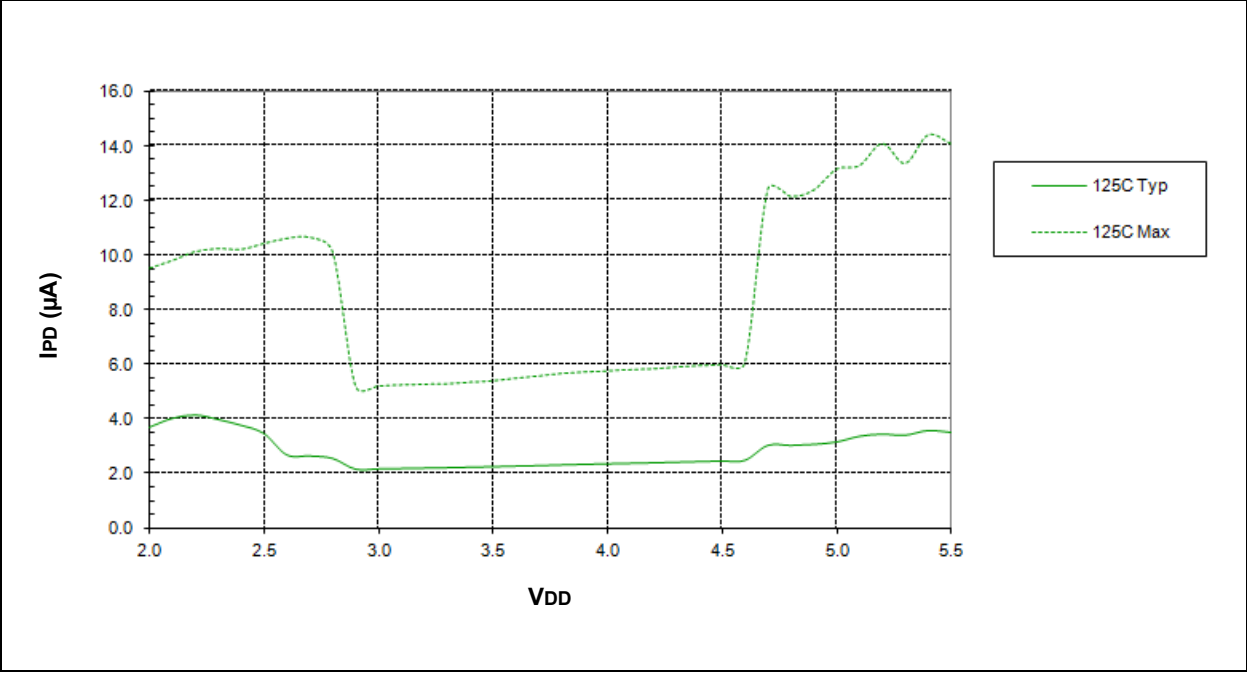


图 30-43: 典型和最大 IPD 与温度的关系曲线

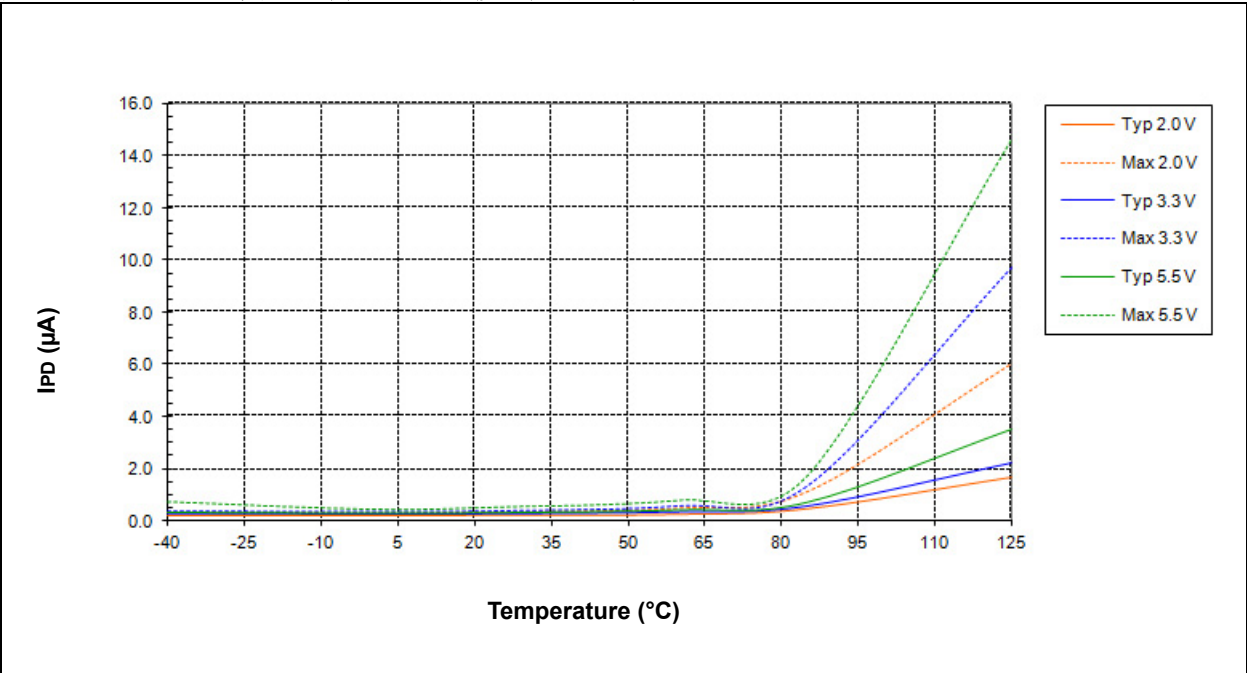


图 30-44: 典型和最大 I_{PD} 与 V_{DD} 的关系曲线（深度休眠模式）

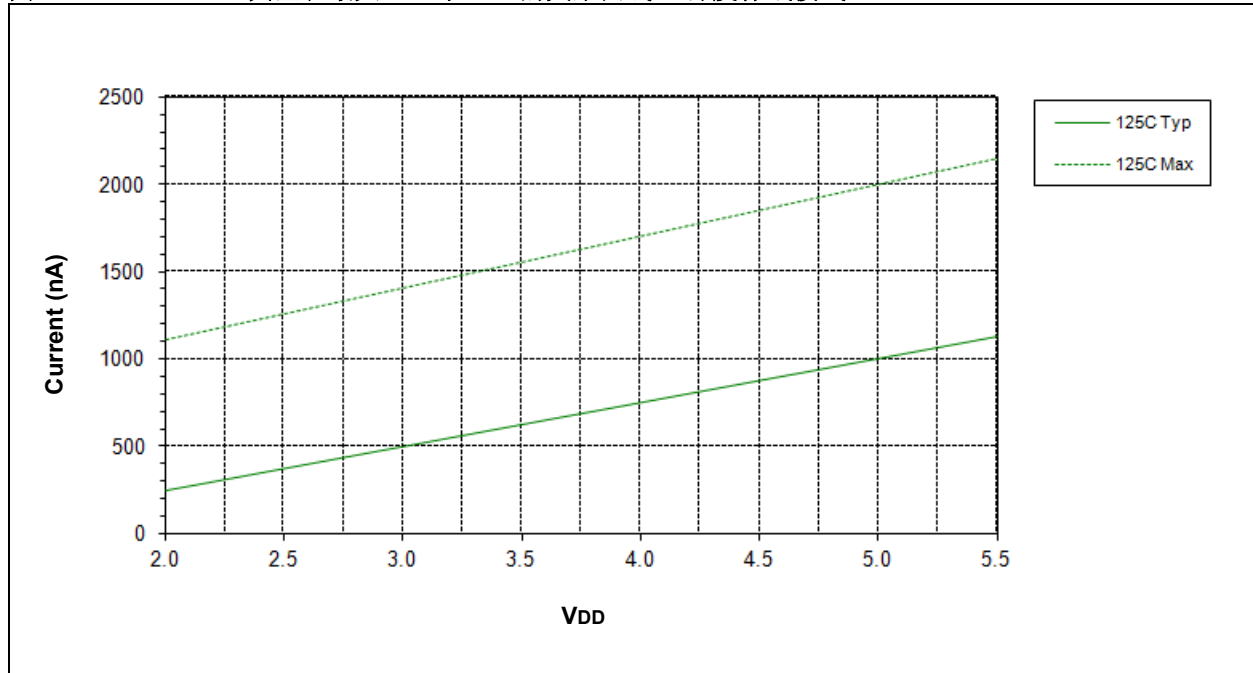
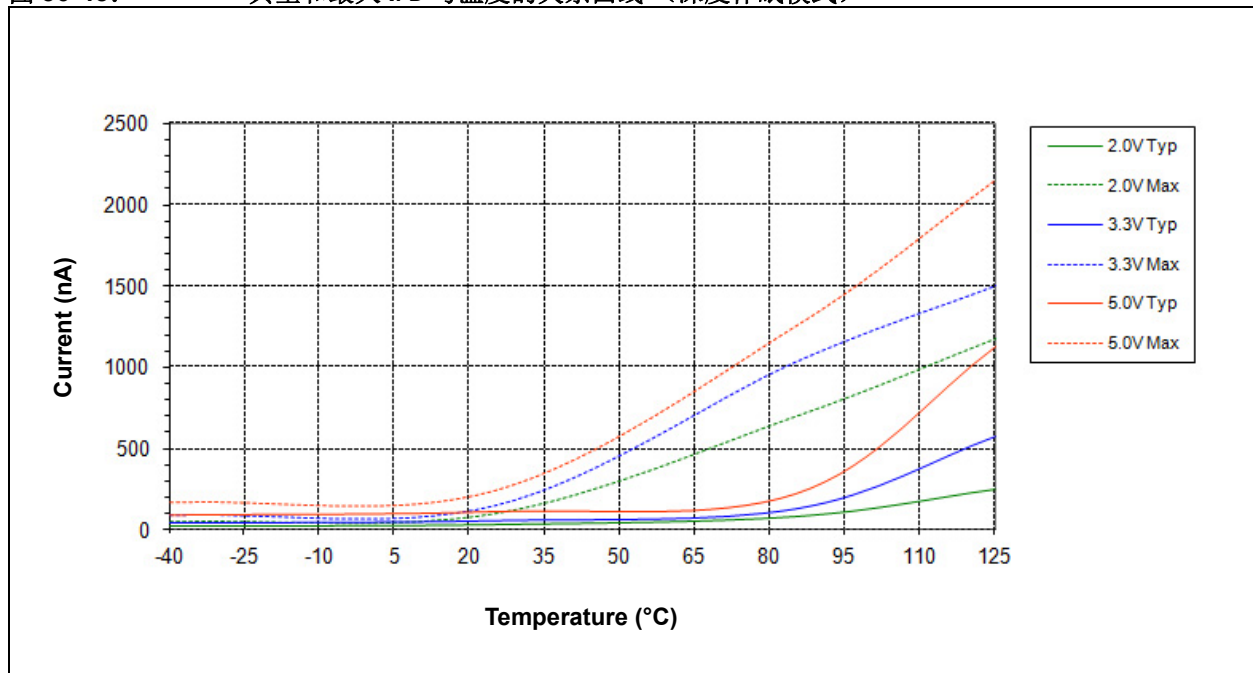


图 30-45: 典型和最大 I_{PD} 与温度的关系曲线（深度休眠模式）



PIC24FV32KA304 系列

图 30-46: 典型 ΔI_{WDT} 与 V_{DD} 的关系曲线

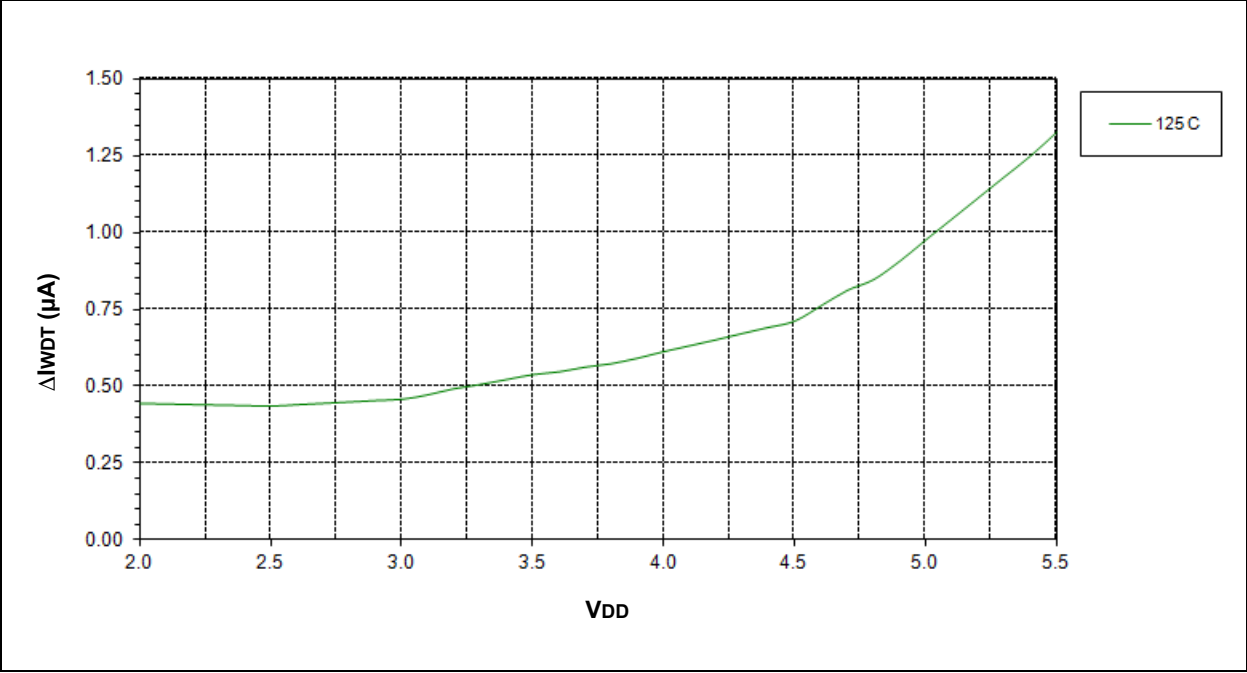


图 30-47: 典型 ΔI_{DSBOR} 与 V_{DD} 的关系曲线

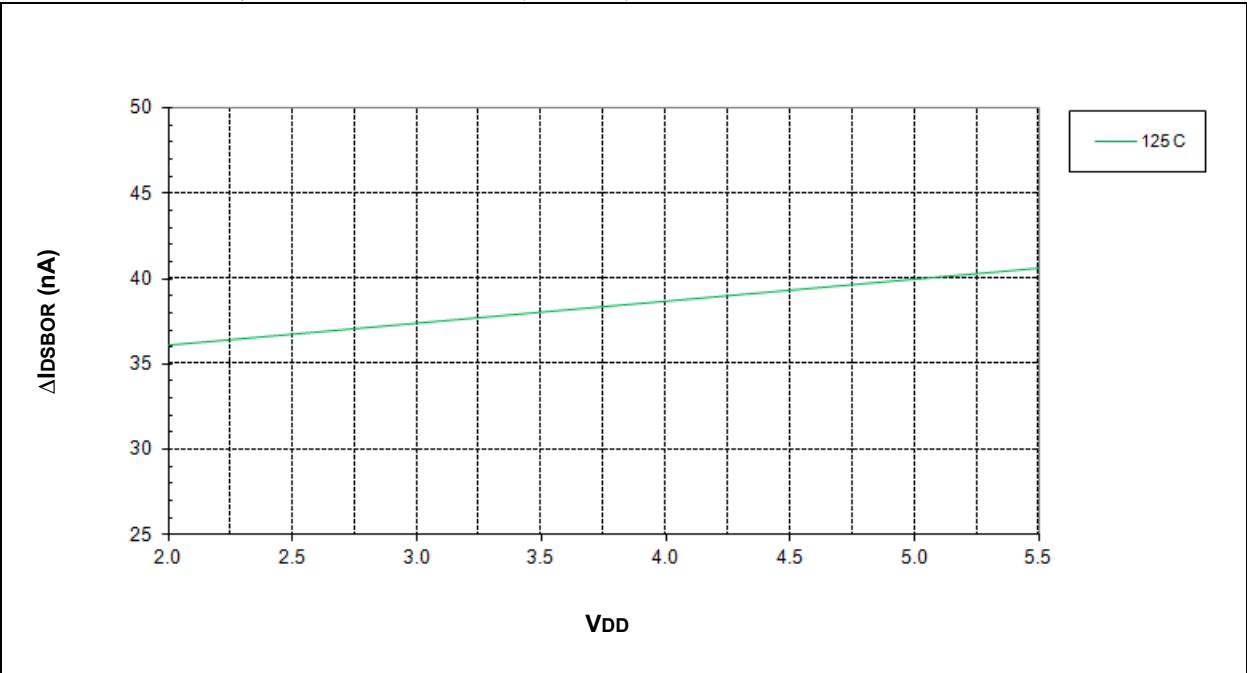
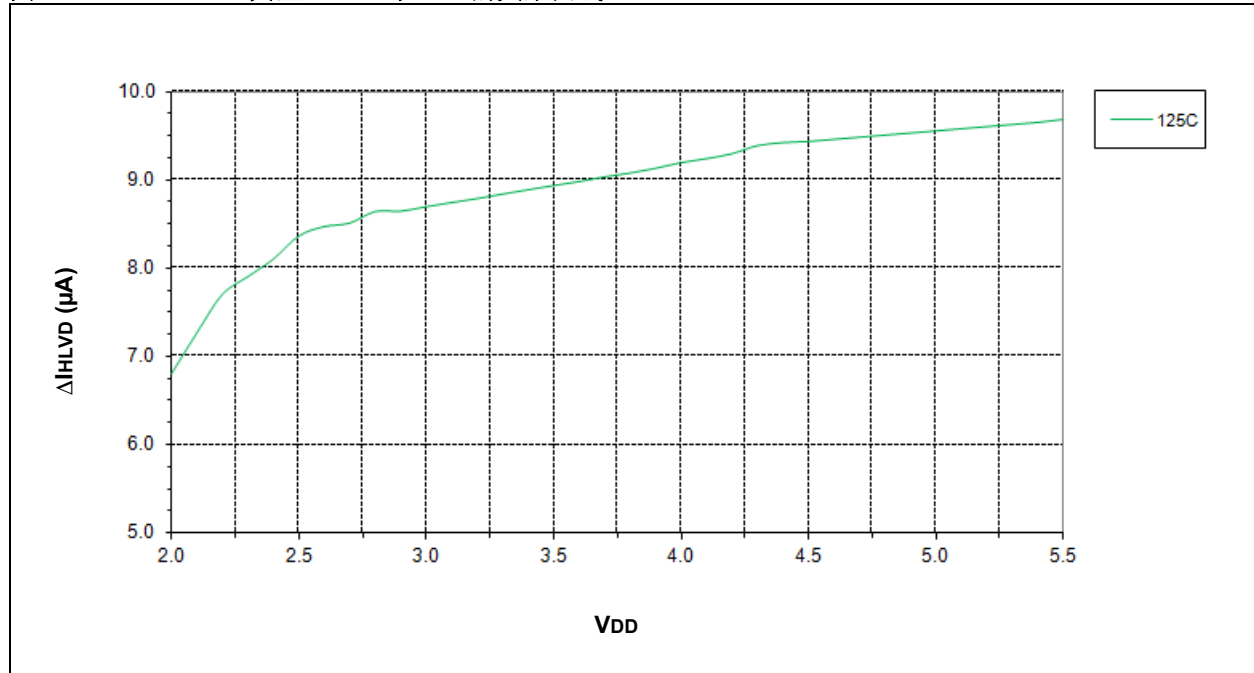


图 30-48: 典型 ΔI_{HLV} 与 V_{DD} 的关系曲线



PIC24FV32KA304 系列

图 30-49: 典型 V_{OL} 与 I_{OL} 的关系曲线 (通用 I/O, $2.0V \leq V_{DD} \leq 5.5V$)

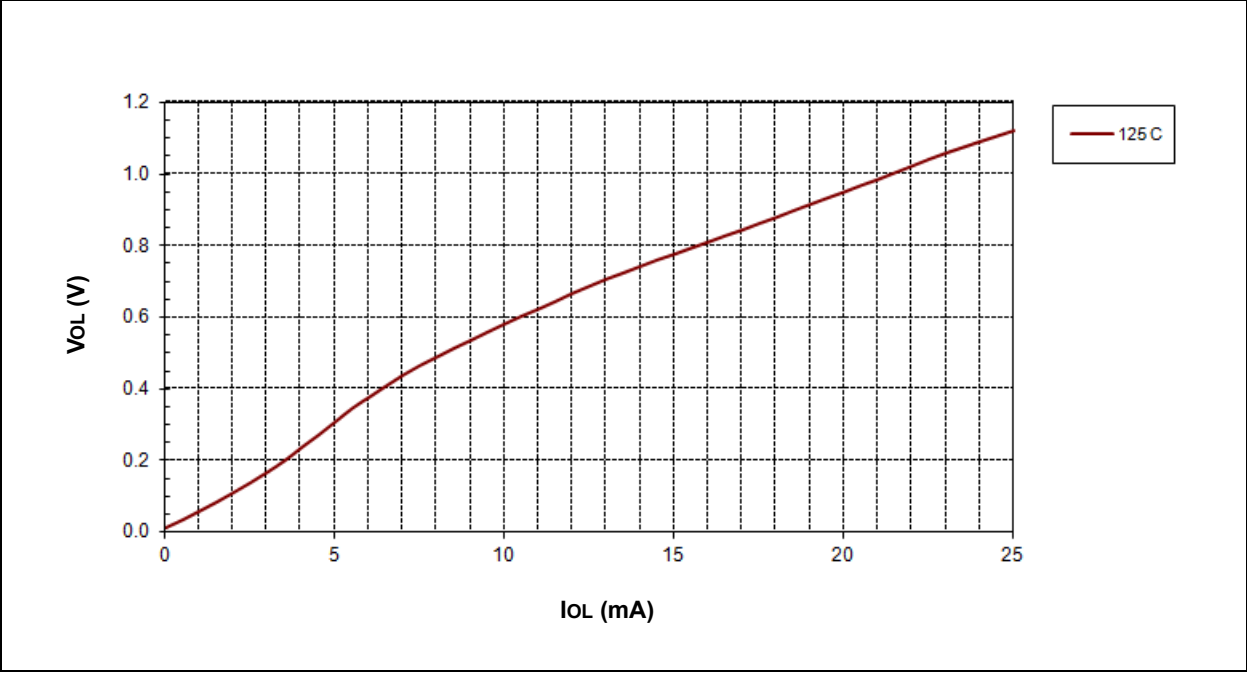


图 30-50: 典型 V_{OH} 与 I_{OH} 的关系曲线 (通用 I/O, 随温度变化, $2.0V \leq V_{DD} \leq 5.5V$)

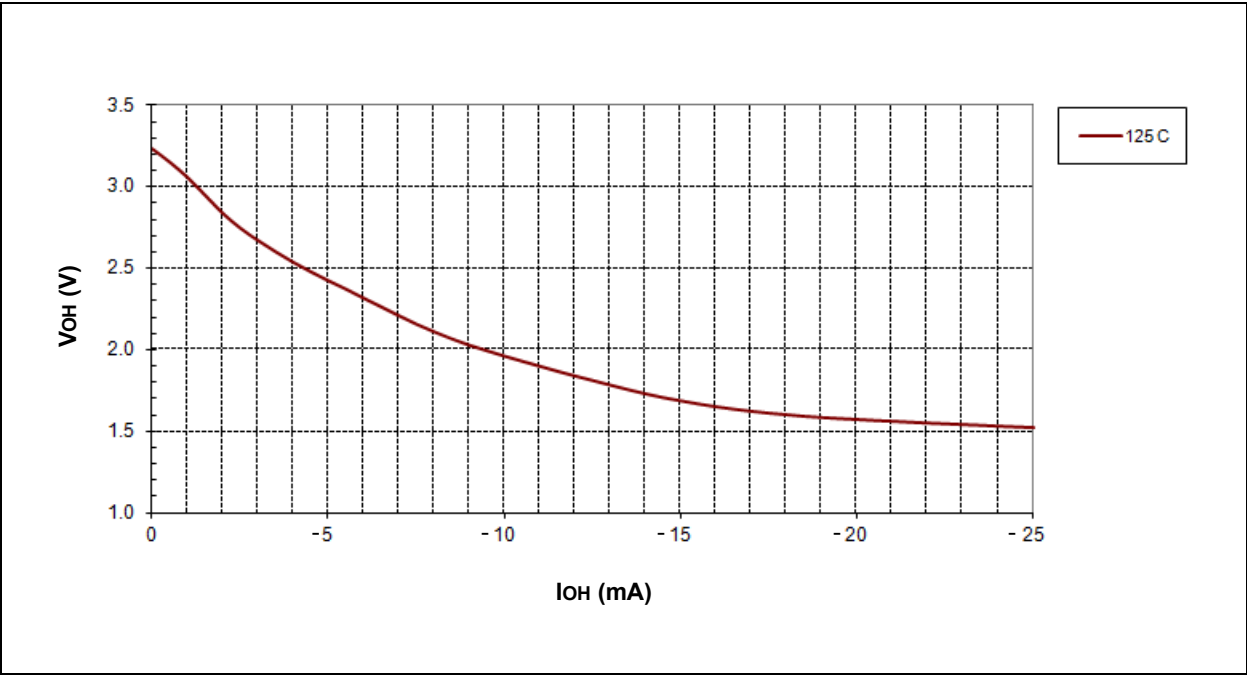


图 30-51: V_{IL}/V_{IH} 与 V_{DD} 的关系曲线 (通用 I/O, 标注温度下)

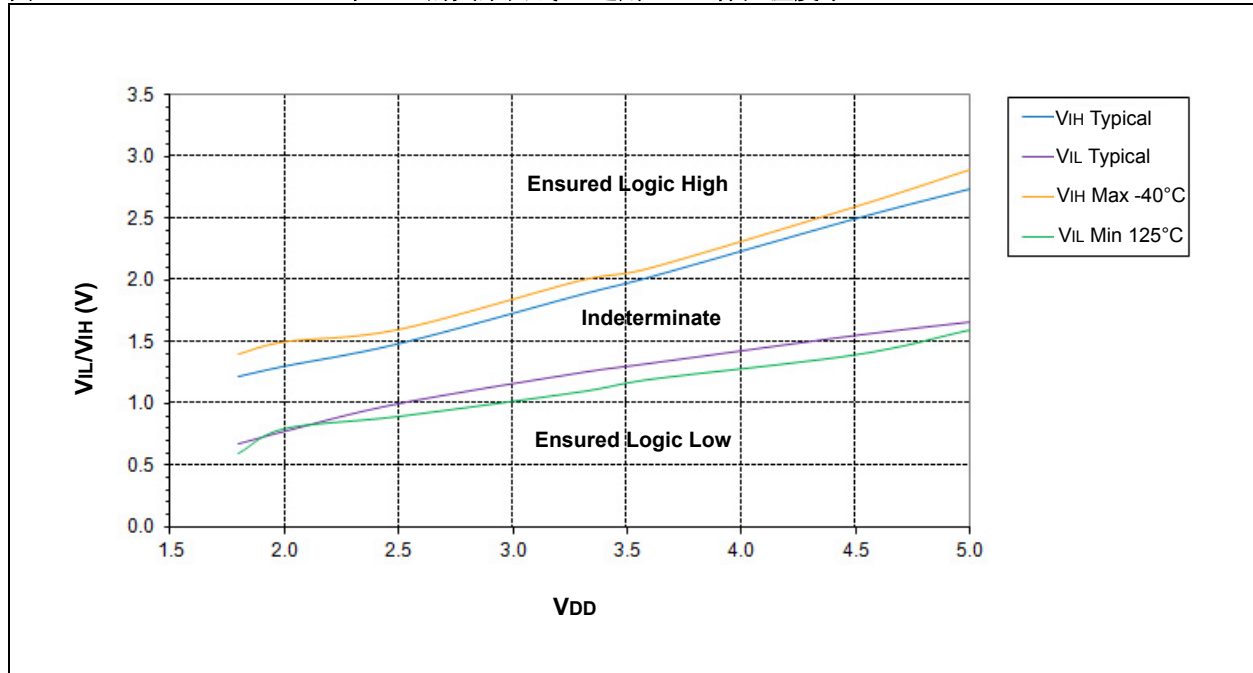
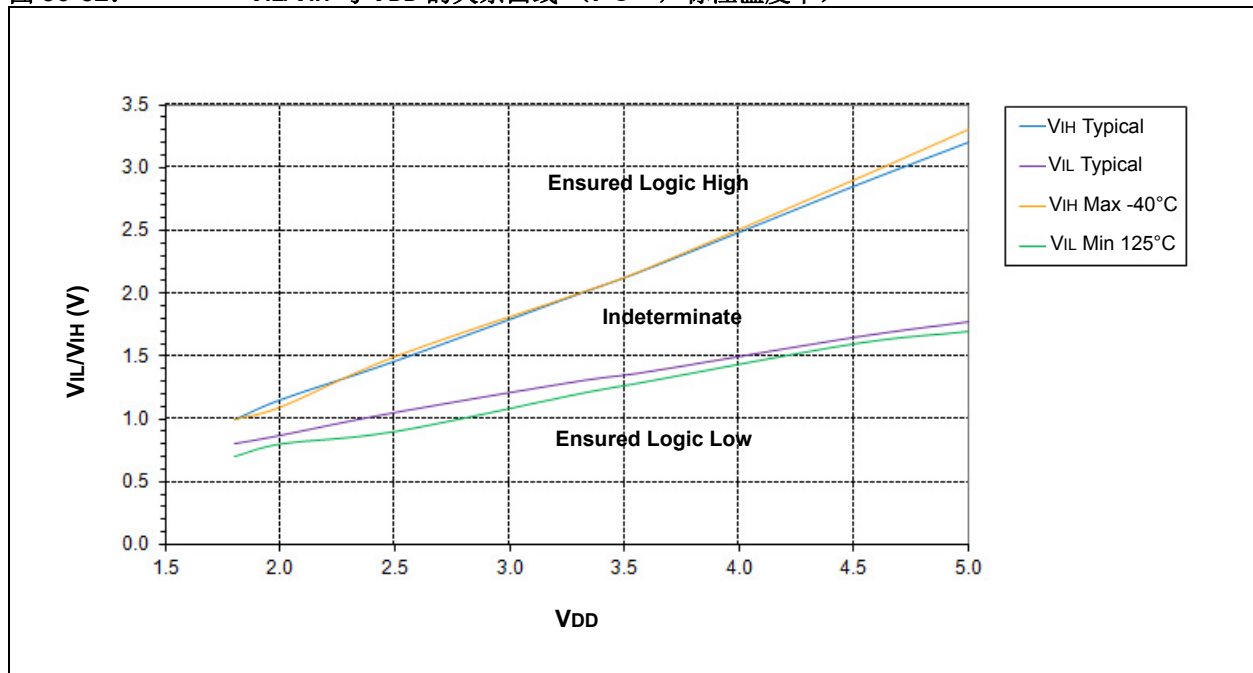


图 30-52: V_{IL}/V_{IH} 与 V_{DD} 的关系曲线 (I^2C^{TM} , 标注温度下)



PIC24FV32KA304 系列

图 30-53: V_{IL}/V_{IH} 与 V_{DD} 的关系曲线（OSCO，标注温度下）

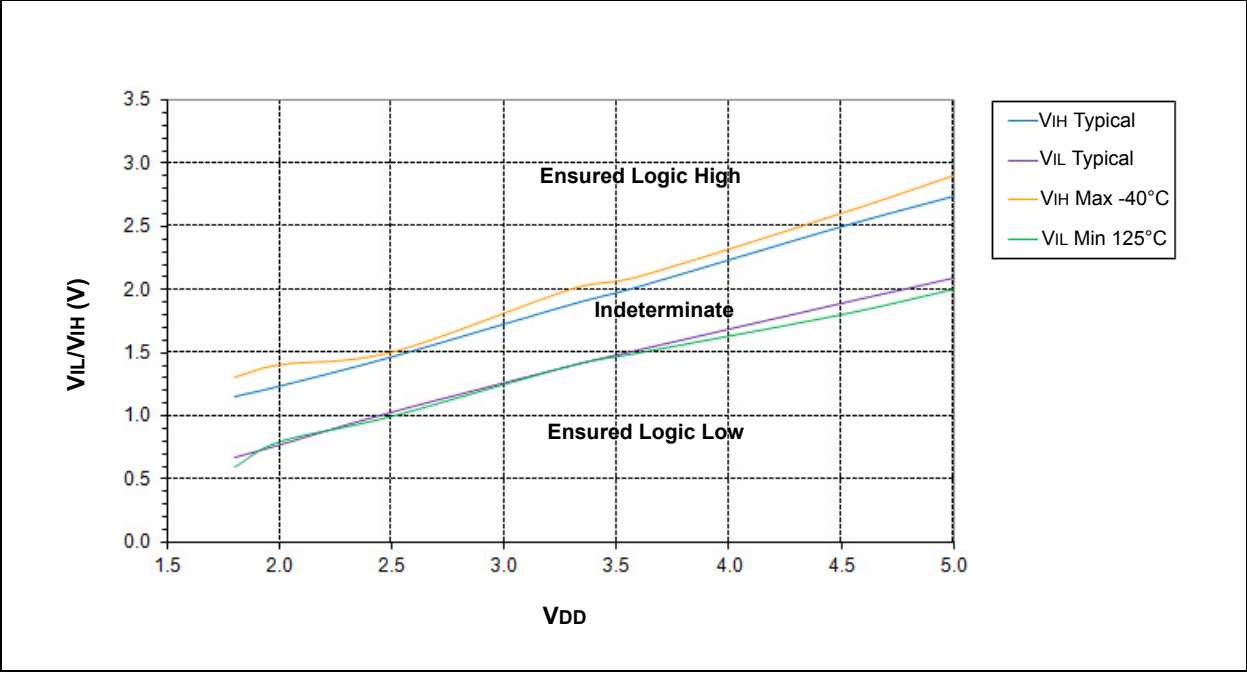


图 30-54: V_{IL}/V_{IH} 与 V_{DD} 的关系曲线（MCLR，标注温度下）

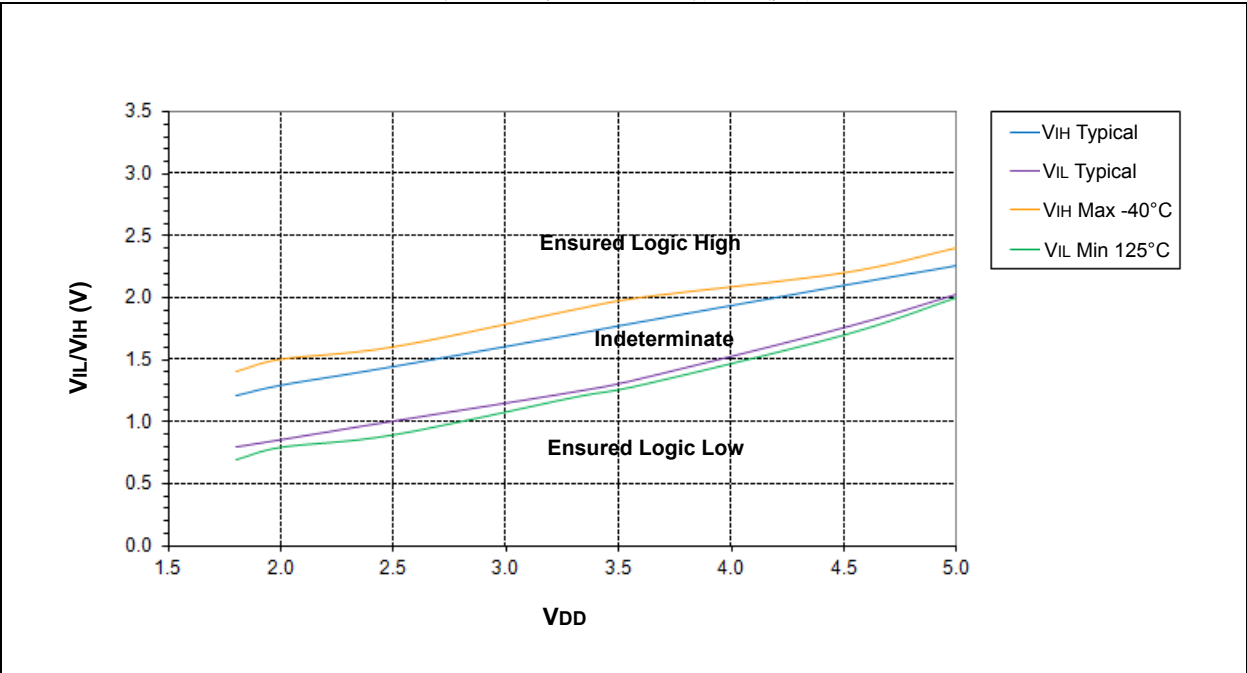


图 30-55: 典型带隙电压与温度的关系曲线 ($2.0V \leq V_{DD} \leq 5.5V$)

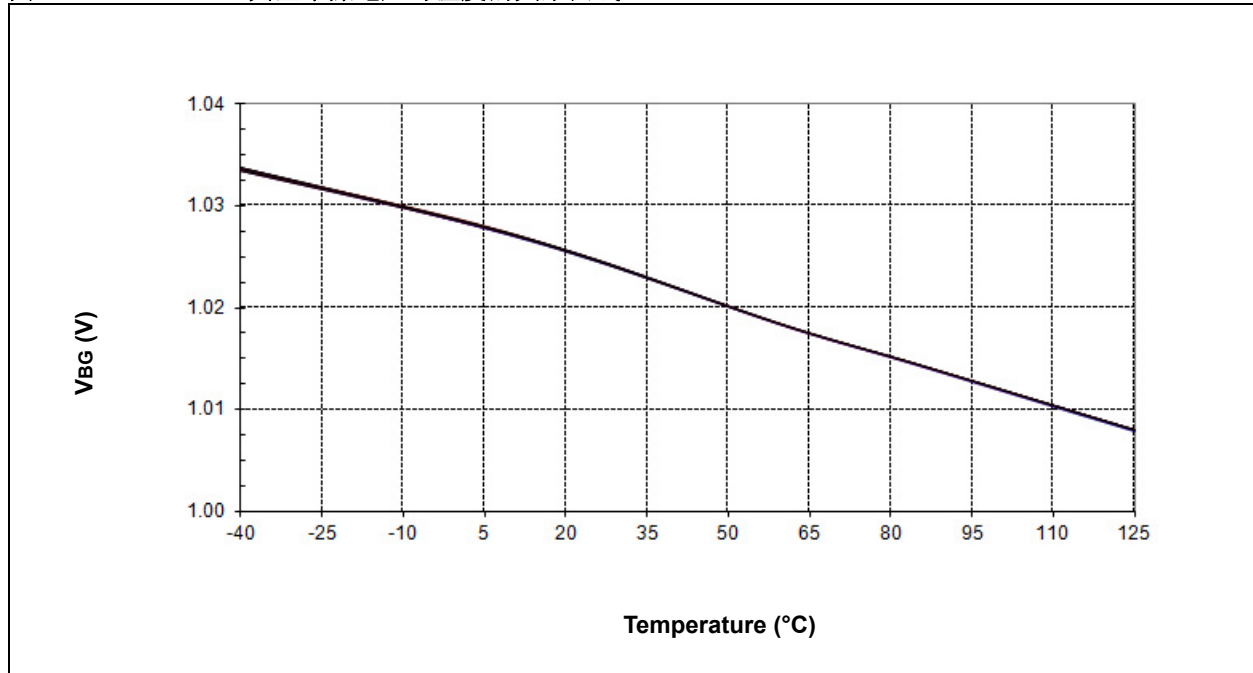
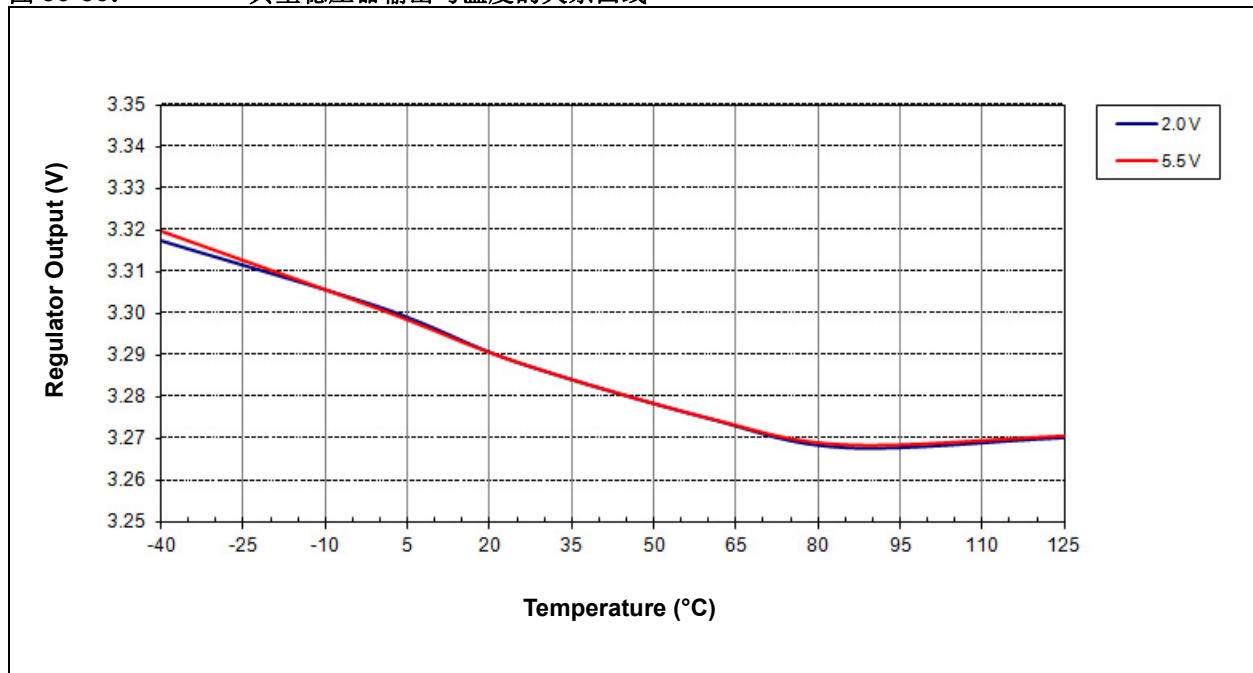


图 30-56: 典型稳压器输出与温度的关系曲线



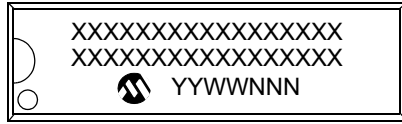
PIC24FV32KA304 系列

注:

31.0 封装信息

31.1 封装标识信息

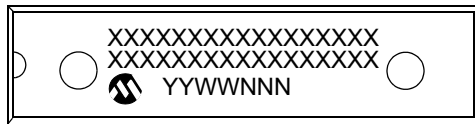
20 引脚 PDIP (300 mil)



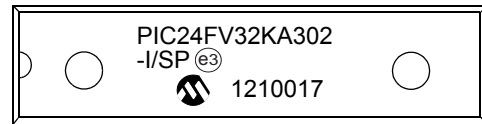
示例



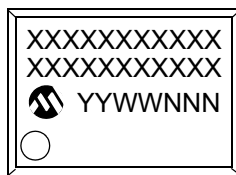
28 引脚 SPDIP (.300")



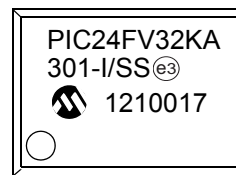
示例



20 引脚 SSOP (5.30 mm)



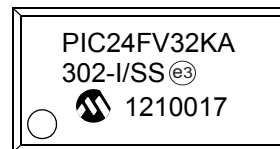
示例



28 引脚 SSOP (5.30 mm)



示例



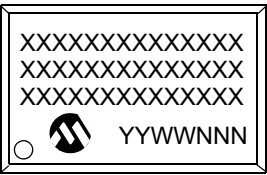
图注:

XX...X	产品信息
Y	年份代码 (日历年的最后一位数字)
YY	年份代码 (日历年的最后两位数字)
WW	星期代码 (一月一日的星期代码为“01”)
NNN	以字母数字排序的追踪代码
(e3)	雾锡 (Matte Tin, Sn) 的 JEDEC 无铅标志
*	表示无铅封装。JEDEC 无铅标志 ((e3)) 标示于此种封装的外包装上。

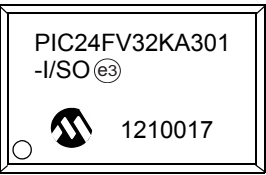
注: Microchip 元器件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户信息的字符数。

PIC24FV32KA304 系列

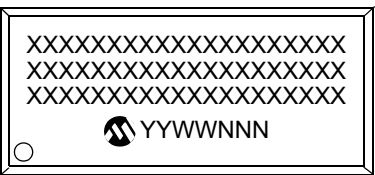
20 引脚 SOIC (7.50 mm)



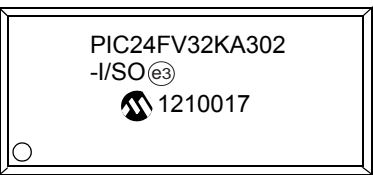
示例



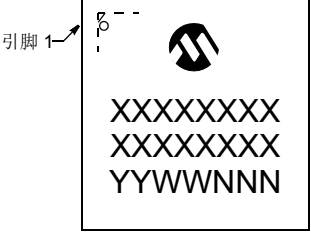
28 引脚 SOIC (7.50 mm)



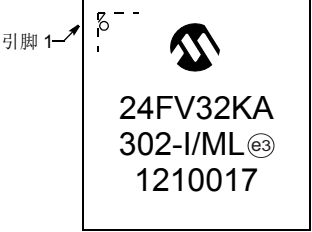
示例



28 引脚 QFN (6x6 mm)

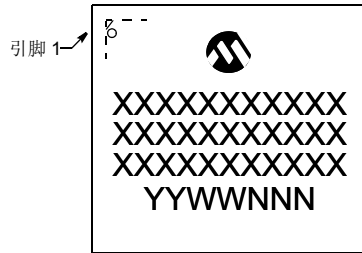


示例

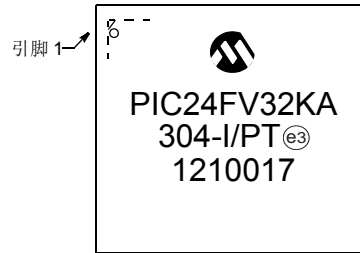


PIC24FV32KA304 系列

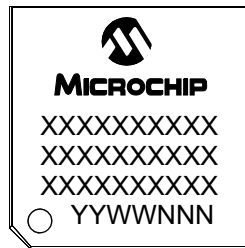
44 引脚 QFN (8x8x0.9 mm)



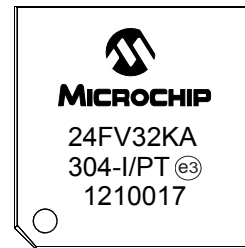
示例



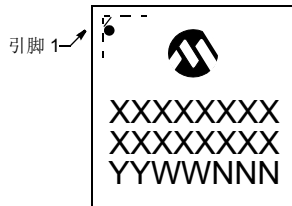
44 引脚 TQFP (10x10x1 mm)



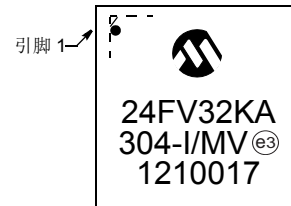
示例



48 引脚 UQFN (6x6x0.5 mm)



示例



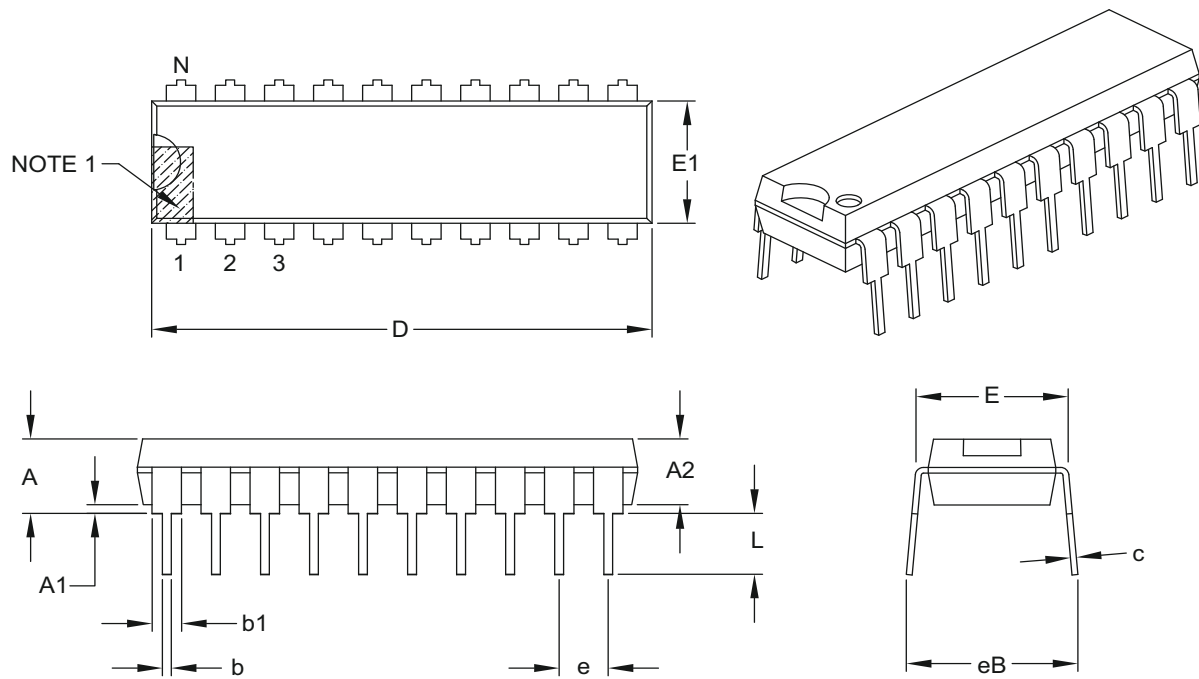
PIC24FV32KA304 系列

31.2 封装详细信息

以下部分将介绍各种封装的技术细节。

20 引脚塑封双列直插式封装（P）—— 主体 300 mil [PDIP]

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	INCHES		
		MIN	NOM	MAX
Number of Pins	N	20		
Pitch	e	.100 BSC		
Top to Seating Plane	A	—	—	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	—	—
Shoulder to Shoulder Width	E	.300	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.980	1.030	1.060
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	—	—	.430

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.

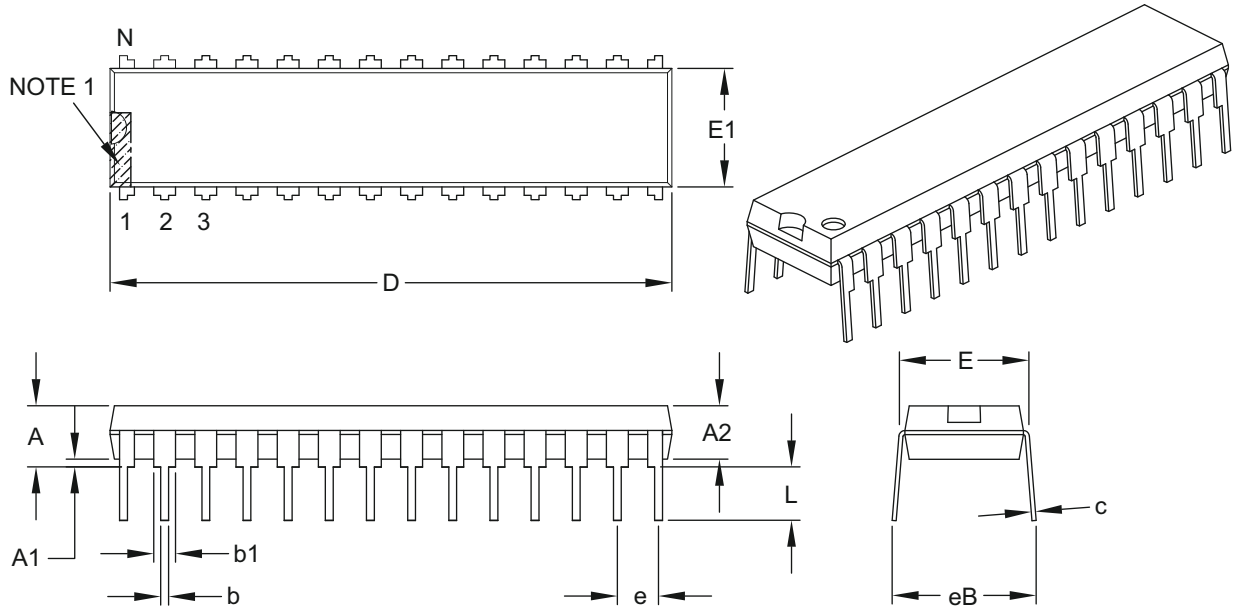
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-019B

PIC24FV32KA304 系列

28 引脚窄型塑封双列直插式封装（SP）—— 主体 300 mil [SPDIP]

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	.100 BSC		
Top to Seating Plane	A	—	—	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	—	—
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	—	—	.430

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.

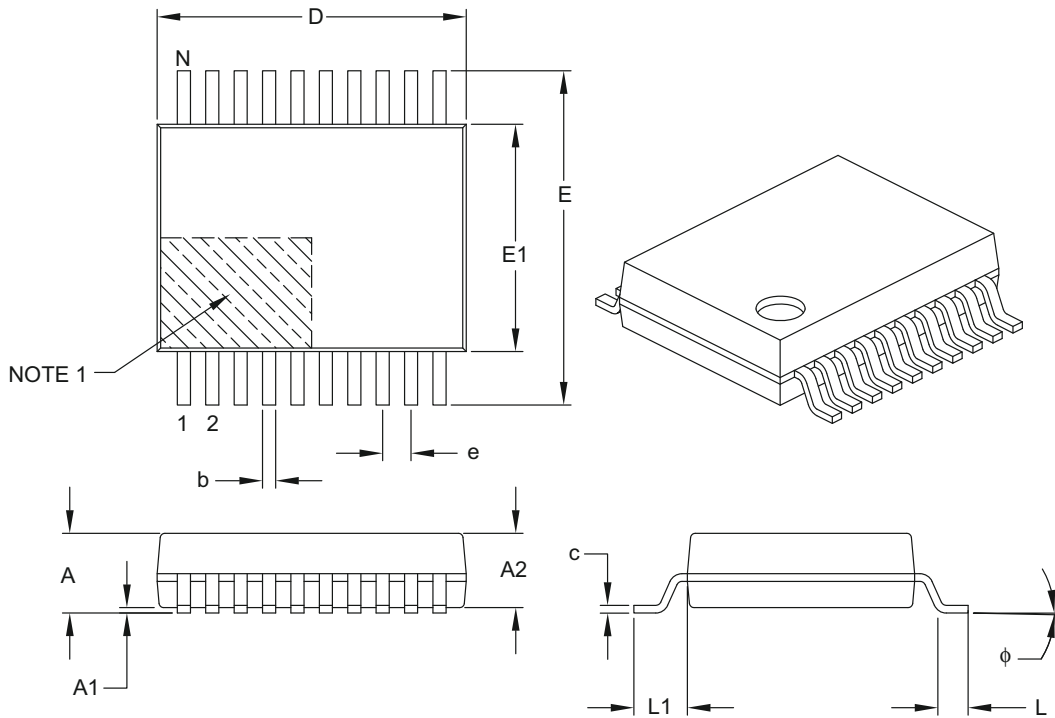
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-070B

PIC24FV32KA304 系列

20 引脚塑封缩小外形封装（SS）—— 主体 5.30 mm [SSOP]

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
Number of Pins	N		20		
Pitch	e		0.65 BSC		
Overall Height	A		—	—	2.00
Molded Package Thickness	A2		1.65	1.75	1.85
Standoff	A1		0.05	—	—
Overall Width	E		7.40	7.80	8.20
Molded Package Width	E1		5.00	5.30	5.60
Overall Length	D		6.90	7.20	7.50
Foot Length	L		0.55	0.75	0.95
Footprint	L1		1.25 REF		
Lead Thickness	c		0.09	—	0.25
Foot Angle	φ		0°	4°	8°
Lead Width	b		0.22	—	0.38

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

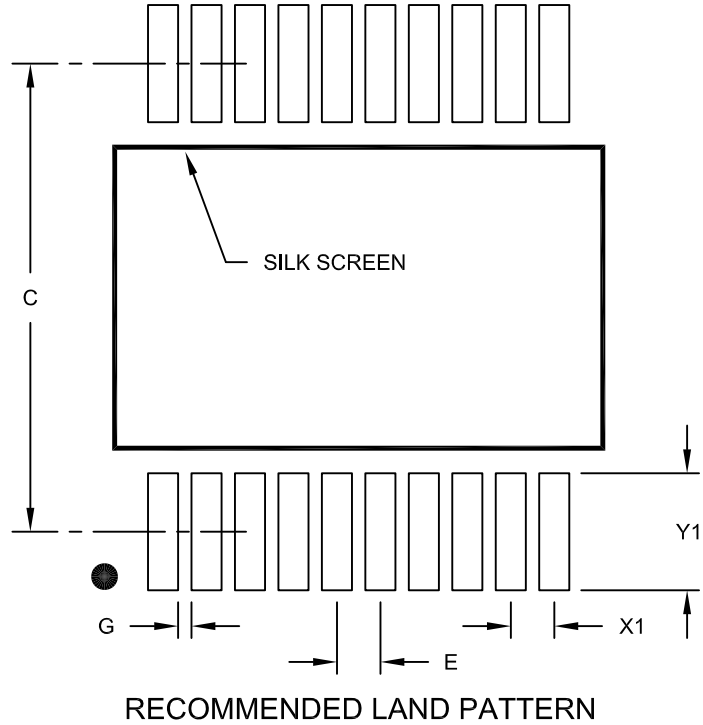
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-072B

PIC24FV32KA304 系列

20 引脚塑封缩小外形封装 (SS) —— 主体 5.30 mm [SSOP]

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	0.65 BSC		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

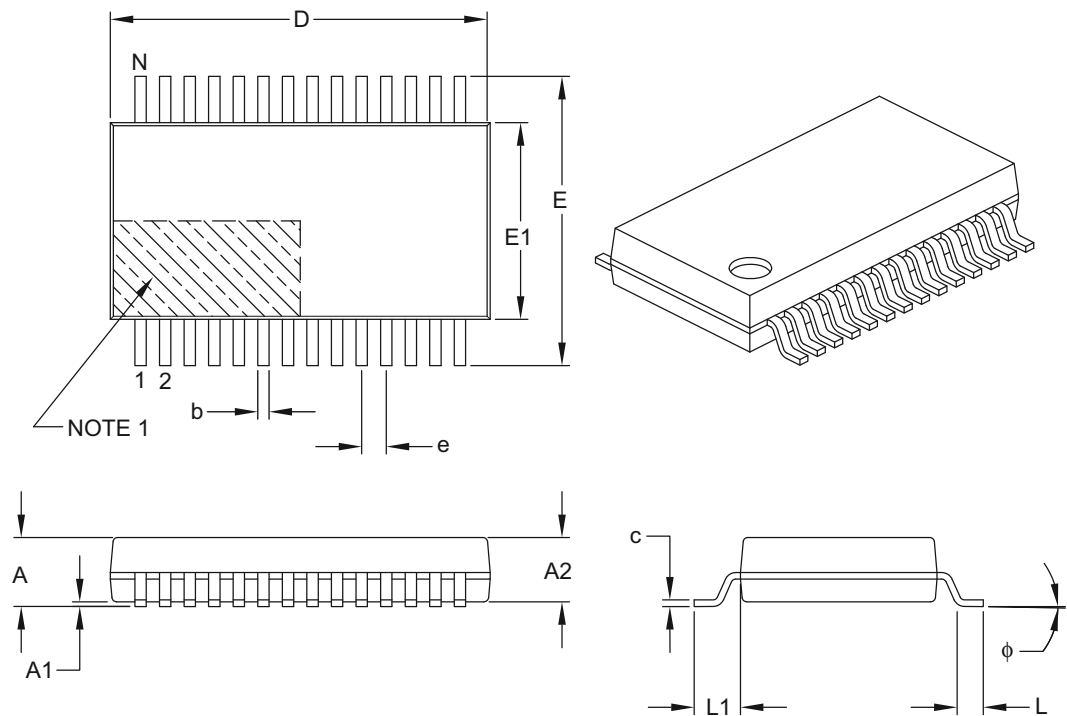
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2072A

PIC24FV32KA304 系列

28 引脚塑封缩小外形封装（SS）—— 主体 5.30 mm [SSOP]

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
Number of Pins	N	28		
Pitch	e	0.65 BSC		
Overall Height	A	—	—	2.00
Molded Package Thickness	A2	1.65	1.75	1.85
Standoff	A1	0.05	—	—
Overall Width	E	7.40	7.80	8.20
Molded Package Width	E1	5.00	5.30	5.60
Overall Length	D	9.90	10.20	10.50
Foot Length	L	0.55	0.75	0.95
Footprint	L1	1.25 REF		
Lead Thickness	c	0.09	—	0.25
Foot Angle	φ	0°	4°	8°
Lead Width	b	0.22	—	0.38

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
- 3. Dimensioning and tolerancing per ASME Y14.5M.

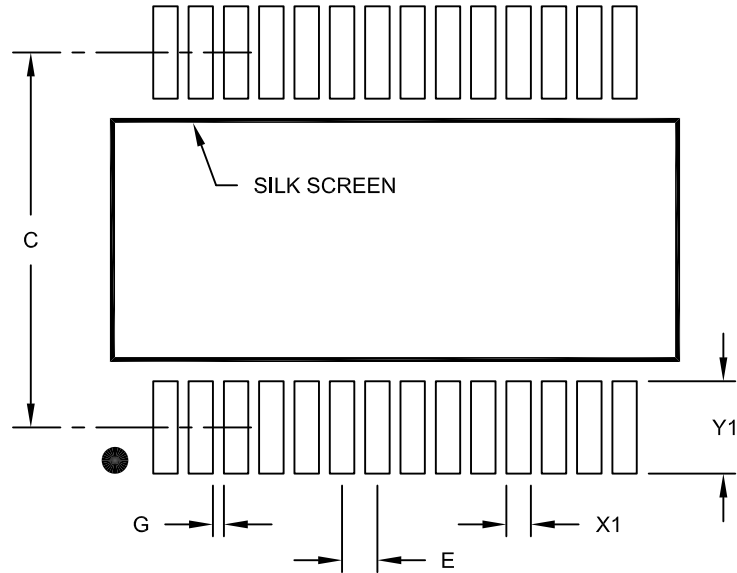
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-073B

PIC24FV32KA304 系列

28 引脚塑封缩小外形封装 (SS) —— 主体 5.30 mm [SSOP]

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		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X28)	X1			0.45
Contact Pad Length (X28)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

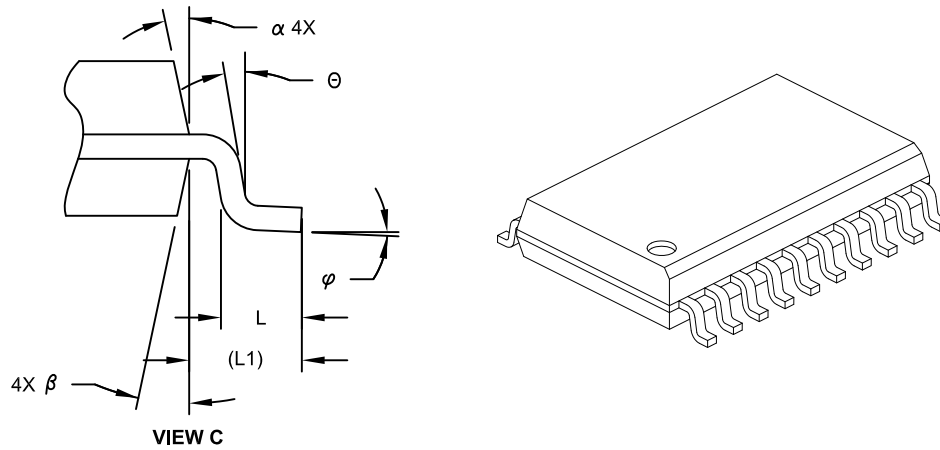
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2073A

20 引脚塑封宽条小外形封装 (SO) —— 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。



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	20		
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	12.80 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	0.20	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

Notes:

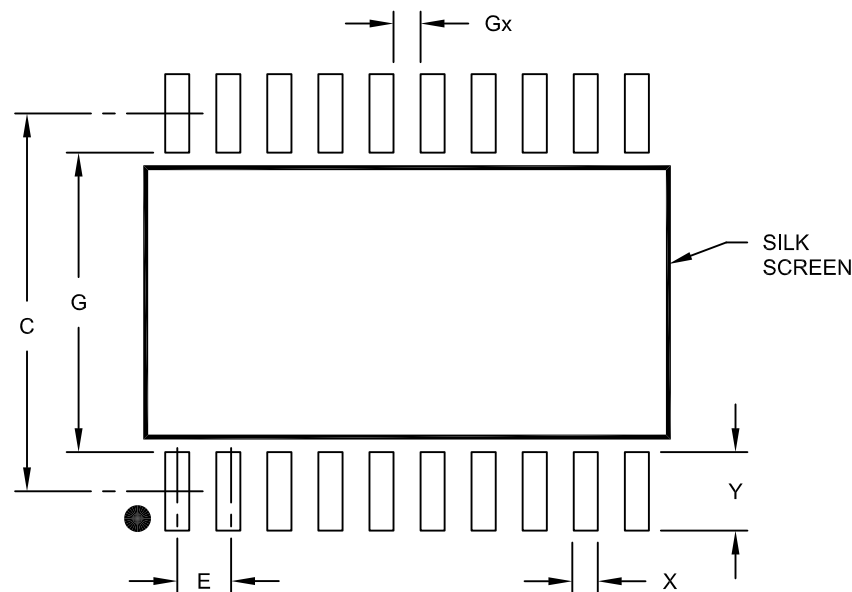
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not 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 No. C04-094C Sheet 2 of 2

PIC24FV32KA304 系列

20 引脚塑封宽条小外形封装（SO）—— 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

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C		9.40	
Contact Pad Width (X20)	X			0.60
Contact Pad Length (X20)	Y			1.95
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.45		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

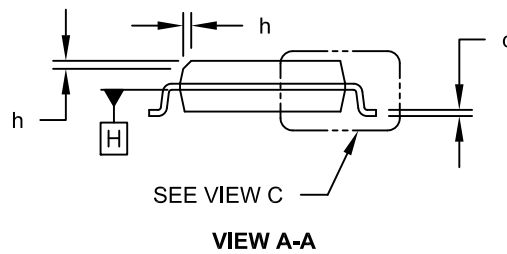
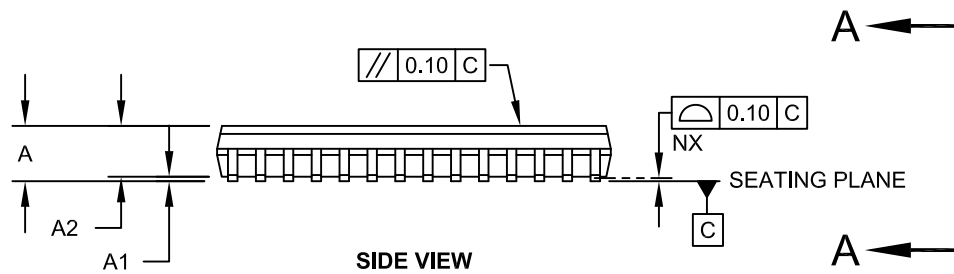
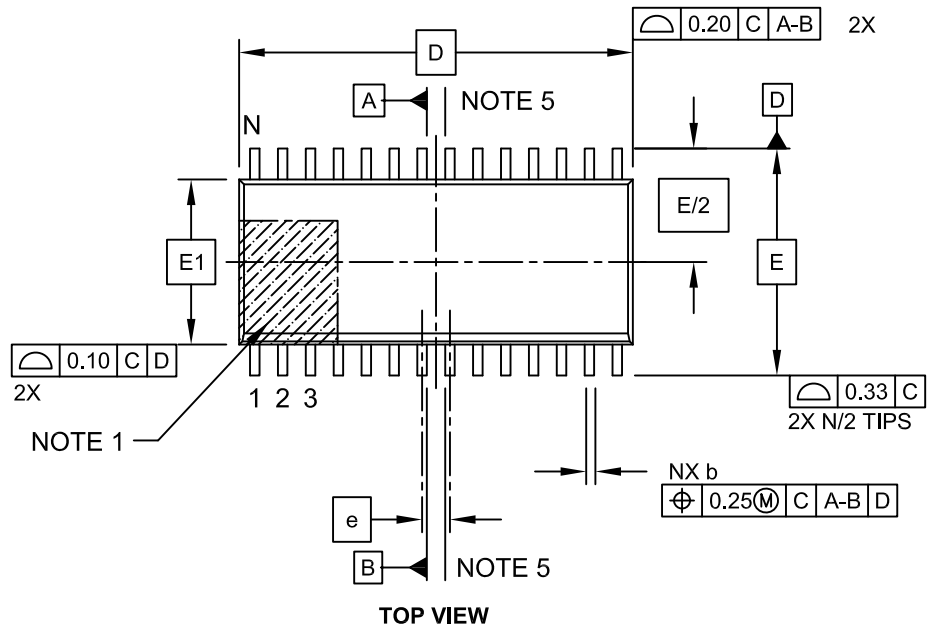
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2094A

PIC24FV32KA304 系列

28 引脚塑封宽条小外形封装 (SO) —— 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。

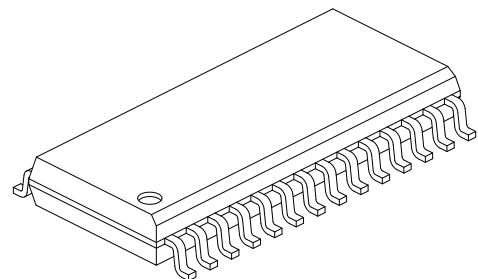
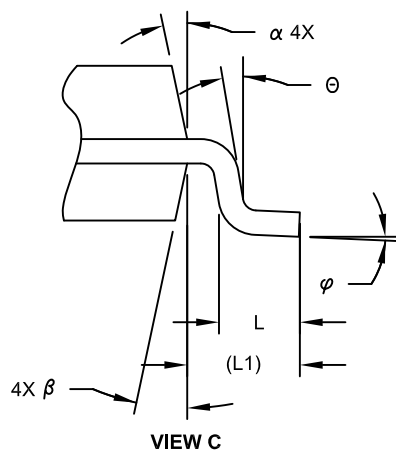


Microchip Technology Drawing C04-052C Sheet 1 of 2

PIC24FV32KA304 系列

28 引脚塑封宽条小外形封装（SO）—— 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。



Units		MILLIMETERS		
Dimension Limits		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:

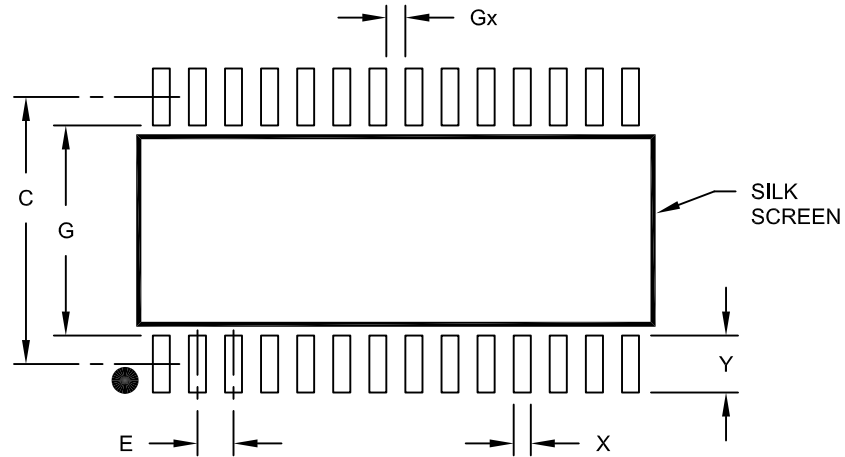
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic
- Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not 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

PIC24FV32KA304 系列

28 引脚塑封宽条小外形封装 (SO) —— 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

Units		MILLIMETERS		
Dimension	Limits	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

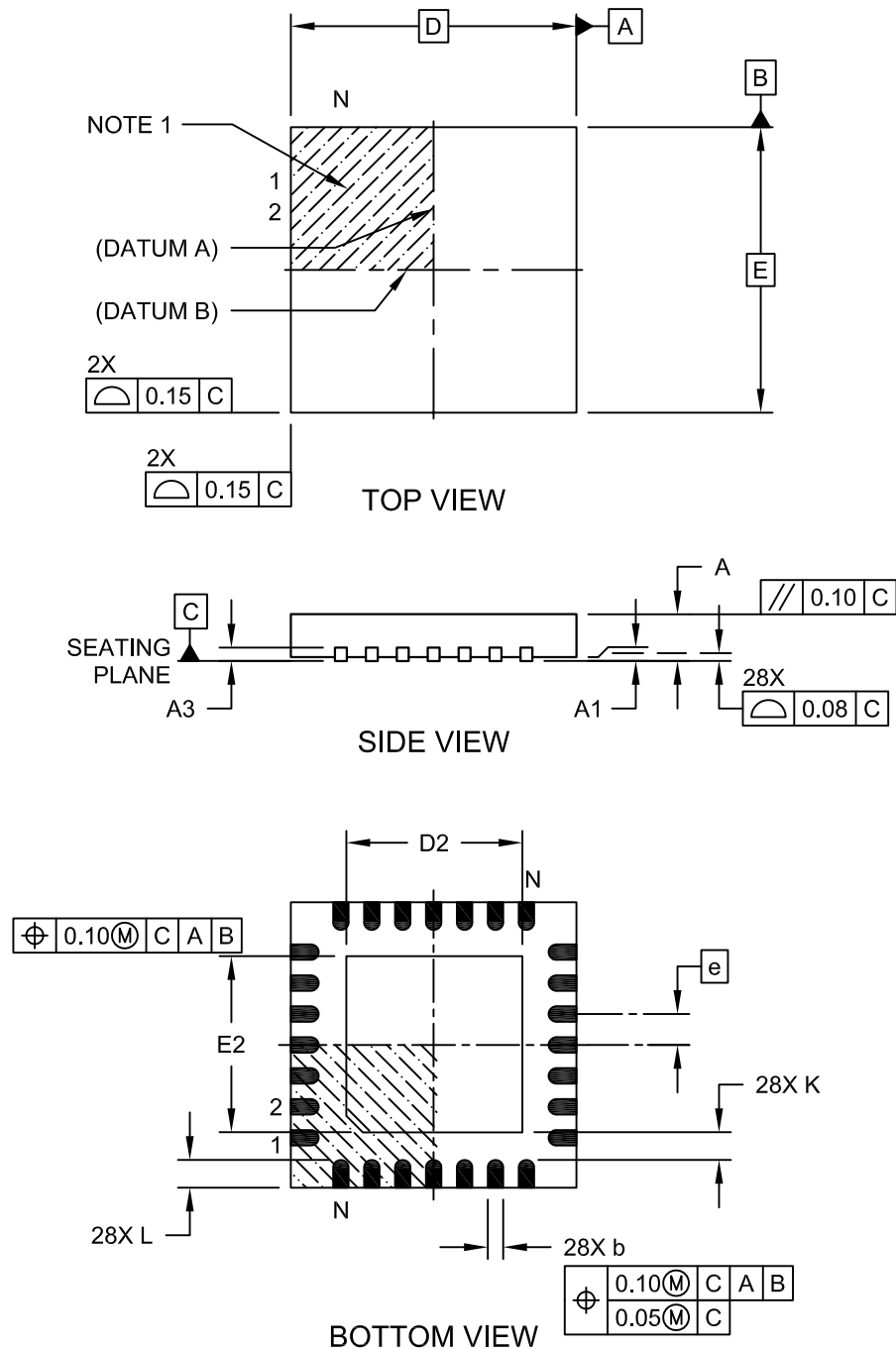
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A

PIC24FV32KA304 系列

28 引脚塑封四方扁平无脚封装（ML）—— 主体 6x6 mm [QFN]，触点长度为 0.55 mm

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

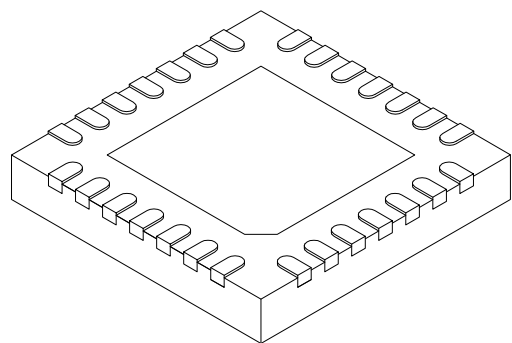


Microchip Technology Drawing C04-105C Sheet 1 of 2

PIC24FV32KA304 系列

28 引脚塑封四方扁平无脚封装（ML）—— 主体 6x6 mm [QFN]，触点长度为 0.55 mm

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
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	4.20
Overall Length	D	6.00 BSC		
Exposed Pad Length	D2	3.65	3.70	4.20
Terminal Width	b	0.23	0.30	0.35
Terminal Length	L	0.50	0.55	0.70
Terminal-to-Exposed Pad	K	0.20	-	-

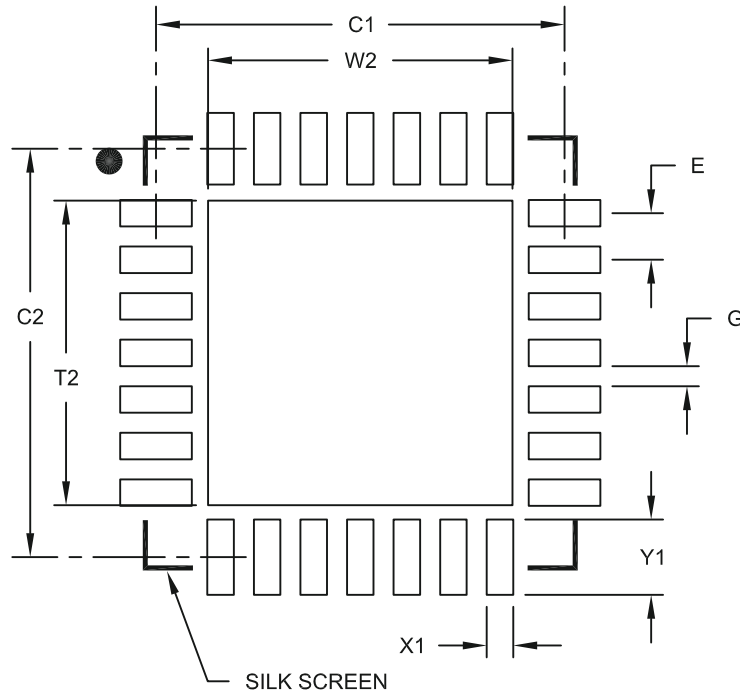
Notes:

- 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.

PIC24FV32KA304 系列

28 引脚塑封四方扁平无脚封装（ML）—— 主体 6x6 mm [QFN]，触点长度为 0.55 mm

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	W2			4.25
Optional Center Pad Length	T2			4.25
Contact Pad Spacing	C1		5.70	
Contact Pad Spacing	C2		5.70	
Contact Pad Width (X28)	X1			0.37
Contact Pad Length (X28)	Y1			1.00
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

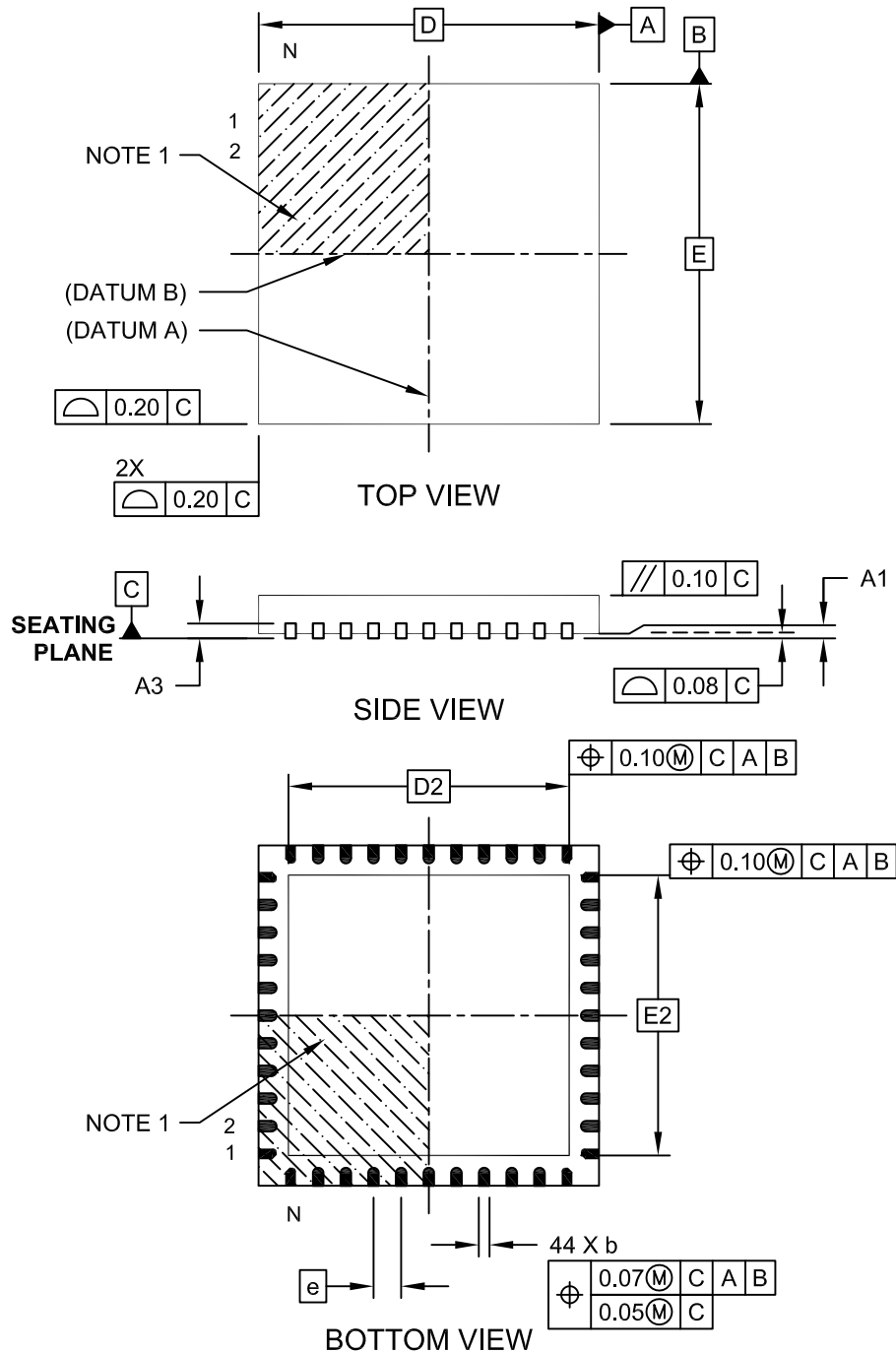
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2105A

PIC24FV32KA304 系列

44 引脚塑封四方扁平无脚封装（ML）—— 主体 8x8 mm [QFN]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

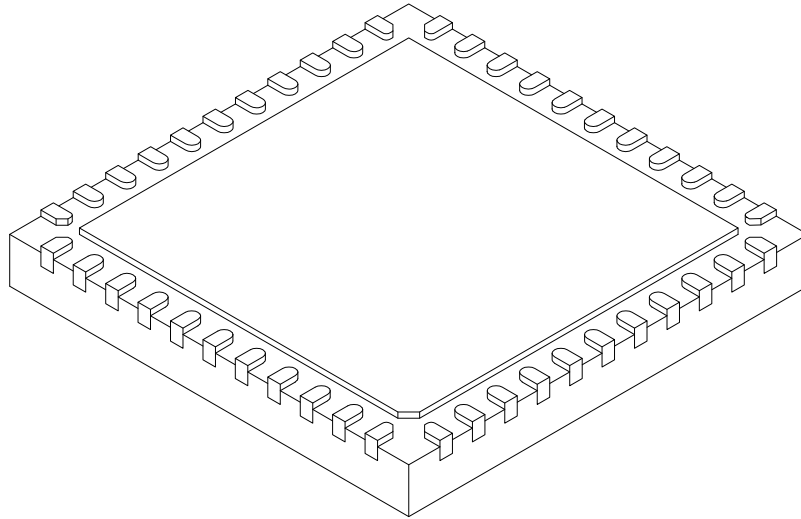


Microchip Technology Drawing C04-103C Sheet 1 of 2

PIC24FV32KA304 系列

44 引脚塑封四方扁平无脚封装（ML）—— 主体 8x8 mm [QFN]

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Units		MILLIMETERS		
Dimension Limits		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.

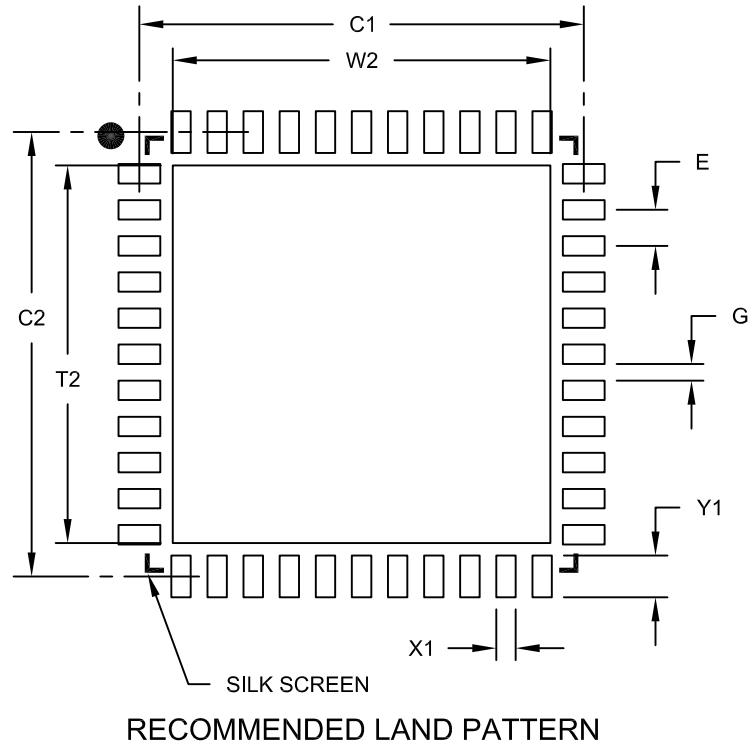
REF: Reference Dimension. usually without tolerance. for information purposes only.

Microchip Technology Drawing C04-103C Sheet 2 of 2

PIC24FV32KA304 系列

44 引脚塑封四方扁平无脚封装（ML）—— 主体 8x8 mm [QFN]

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	W2			6.60
Optional Center Pad Length	T2			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
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

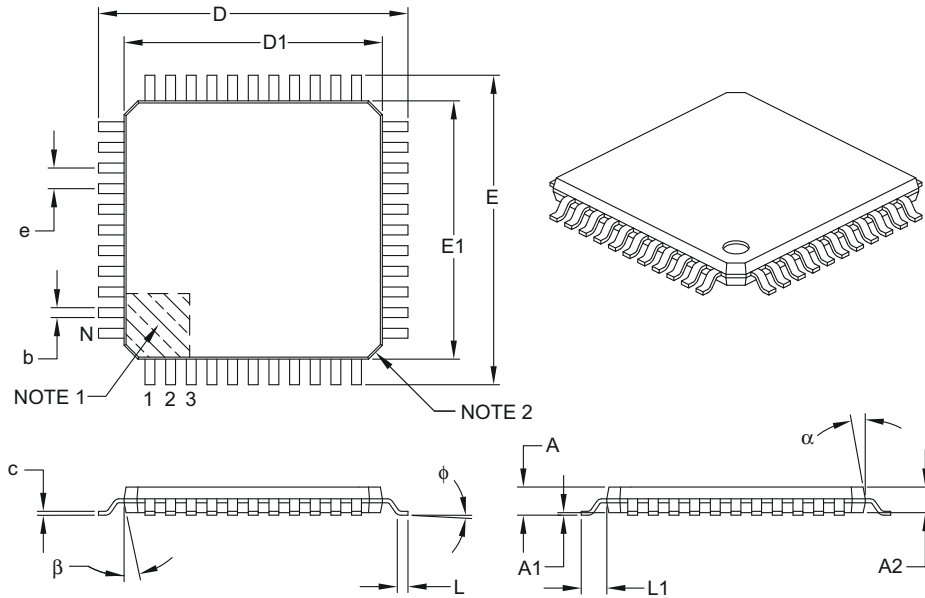
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2103B

PIC24FV32KA304 系列

44 引脚塑封薄型四方扁平封装（PT）—— 主体 10x10x1 mm，引脚占位长度 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
Number of Leads	N	44		
Lead Pitch	e	0.80 BSC		
Overall Height	A	—	—	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	—	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	ϕ	0°	3.5°	7°
Overall Width	E	12.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Thickness	c	0.09	—	0.20
Lead Width	b	0.30	0.37	0.45
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

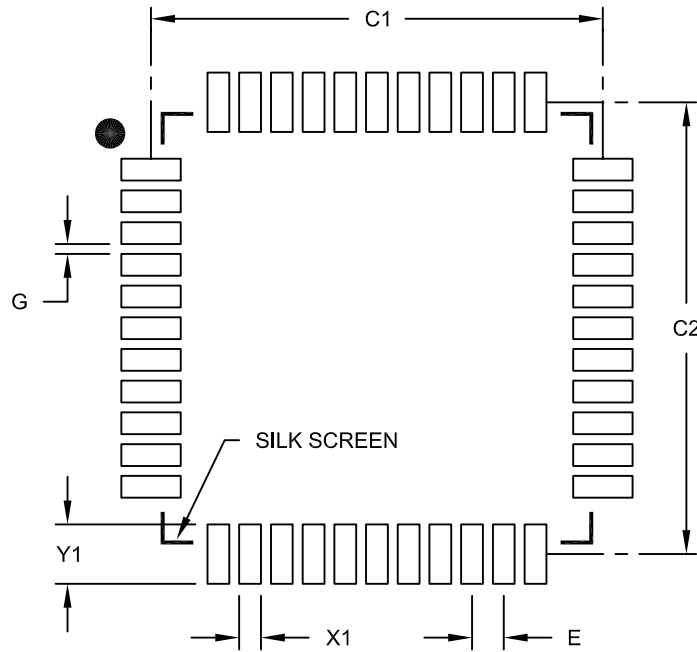
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-076B

44 引脚塑封薄型四方扁平封装（PT）—— 主体 10x10x1 mm，引脚占位长度 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

		Units	MILLIMETERS		
Dimension Limits			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

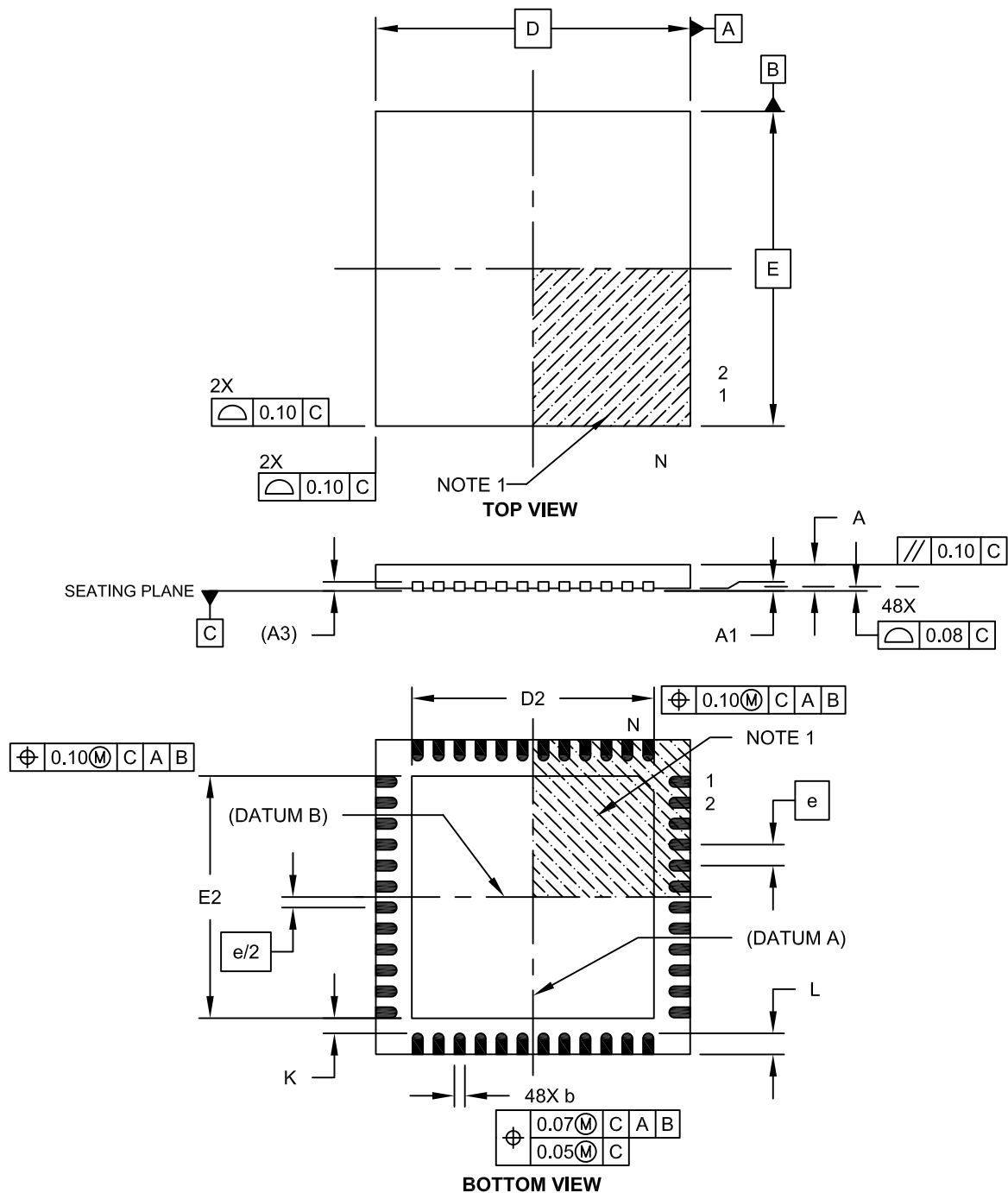
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2076B

PIC24FV32KA304 系列

48 引脚塑封超薄型四方扁平无脚封装 (MV) —— 主体 6x6x0.5 mm [UQFN]

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。

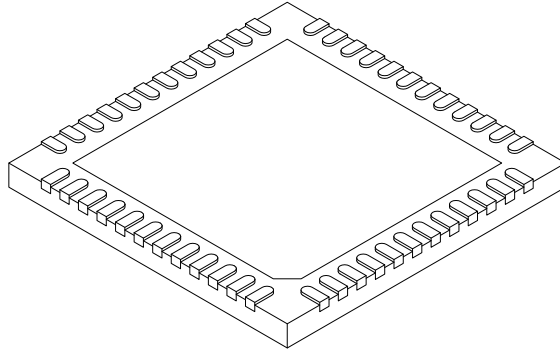


Microchip Technology Drawing C04-153A Sheet 1 of 2

PIC24FV32KA304 系列

48 引脚塑封超薄型四方扁平无脚封装（MV）—— 主体 6x6x0.5 mm [UQFN]

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
Number of Pins	N		48		
Pitch	e		0.40 BSC		
Overall Height	A		0.45	0.50	0.55
Standoff	A1		0.00	0.02	0.05
Contact Thickness	A3		0.127 REF		
Overall Width	E		6.00 BSC		
Exposed Pad Width	E2		4.45	4.60	4.75
Overall Length	D		6.00 BSC		
Exposed Pad Length	D2		4.45	4.60	4.75
Contact Width	b		0.15	0.20	0.25
Contact Length	L		0.30	0.40	0.50
Contact-to-Exposed Pad	K		0.20	-	-

Notes:

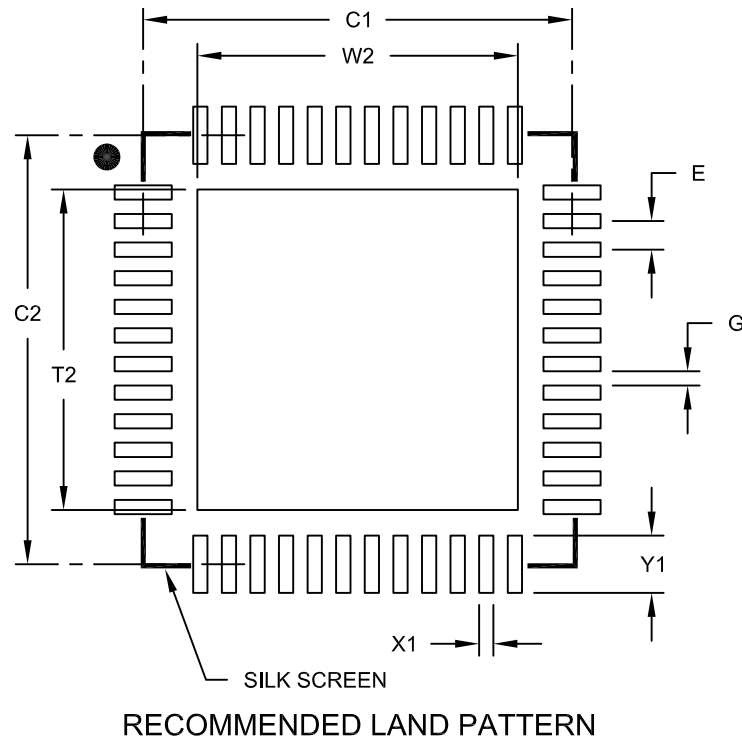
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-153A Sheet 2 of 2

PIC24FV32KA304 系列

48 引脚塑封超薄型四方扁平无脚封装（MV）—— 主体 6x6 mm [UQFN]，触点长度 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
Contact Pitch	E	0.40 BSC		
Optional Center Pad Width	W2			4.45
Optional Center Pad Length	T2			4.45
Contact Pad Spacing	C1		6.00	
Contact Pad Spacing	C2		6.00	
Contact Pad Width (X28)	X1			0.20
Contact Pad Length (X28)	Y1			0.80
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. C04-2153A

附录 A： 版本历史

版本 A（2011 年 3 月）

PIC24FV32KA304 系列器件的初始数据手册。

版本 B（2011 年 4 月）

对第 25.0 节“充电时间测量单元 (CTMU)”进行了修订，更改了 CTMUICON（寄存器 25-3）中 IRNG 位的说明。设置 01 是基本电流级别（标称值为 0.55 μA ），设置 00 是 1000 倍基本电流。

对第 29.0 节“电气特性”进行了修订，更改了以下典型 IPD 规范：

- DC20h/i/j/k，从 204 μA 更改为 200 μA
- DC60h/i/j/k，从 0.15 μA 更改为 0.025 μA
- DC60l/m/n/o，从 0.25 μA 更改为 0.040 μA
- DC72h/i/j/k，从 0.80 μA 更改为 0.70 μA

版本 C（2012 年 4 月）

更新了第 3 页至第 7 页的引脚图，将所有的“LVDIN”均改为“HLVDIN”，并更新了某些功能引脚的位置。

更新了表 1-3，删除了未实现的封装类型，更正了一些错误的引脚分配，还删除了一些备用但未实现的引脚分配。

在第 5.0 节“闪存程序存储器”中，采用新的表偏移函数更新了例 5-2、例 5-3 和例 5-4。

更新了图 12-1 以正确显示已实现的 Timer1 输入选项。

在第 22.0 节“带阈值检测的 12 位 A/D 转换器”中：

- 更新了寄存器 22-1 以添加 MODE12 位
- 更新了寄存器 22-2 中 PVCFGx 和 CSCNA 位的描述
- 更新了寄存器 22-4，将 VRSREQ 位所在的位置替换成保留位
- 更改了寄存器 22-5 的脚注文本文本
- 更正了图 22-2 中的 CHOLD

在第 25.0 节“充电时间测量单元 (CTMU)”中：

- 更新了第 25.1 节“测量电容”和第 25.3 节“脉冲生成和延时”中的文字，以更好地反映模块的实现
- 更新了图 25-3 给出了脉冲生成的更多详细信息

在第 29.0 节“电气特性”中添加了以下时序图和时序要求表：

- 图 29-6（复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序特性）
- 图 29-7（欠压复位特性）
- 图 29-9（输入捕捉 x 时序）至图 29-21（SPIx 模块从模式时序特性（CKE = 1））
- 表 29-28（输入捕捉 x 要求）至表 29-39（SPIx 模块从模式时序要求（CKE = 1））
- 图 29-22（A/D 转换时序）

更新了表 29-5，添加了参数 DC15。

使用新的简化的表更新了表 29-6、表 29-7 和表 29-8，删除了未实现的温度选项。（在这个过程中，没有修改任何现有的规范值。）

采用正确的值更新了表 29-16 中 CTMUICON 位的设置。

将表 29-21 和表 29-22 合并，创建一个新的表 29-21）。既有的后续所有表均相应按顺序重新编号。

更新了表 29-26，添加了参数 SY35 和 SY55。

更新了表 29-40：

- 将 AD01 分为两条，分别表示“F”和“FV”器件系列
- 添加了 AD08（IVREF）和 AD09（ZVREF）
- 更改了 AD17（将最大值 2.5 k Ω 更改为 1 k Ω ）

更新了表 29-41：

- 更改了 AD50（将最小值 75 ns 更改为 600 ns）
- 更改了 AD51（将典型值 250 ns 更改为 1.67 μs ）
- 更改了 AD60（将最小值 0.5 TAD 更改为 2 TAD）
- 将 AD55 分为两条，分别表示 10 位和 12 位转换

添加了第 30.0 节“直流和交流特性图表”，包含图 30-1 至图 30-39。

采用新修改的图替换了第 31.0 节“封装信息”中的部分封装图。

全篇做了其他一些小的排版修改。

PIC24FV32KA304 系列

版本 D（2013 年 3 月）

对本数据手册通篇做了以下修改：将 RCON 寄存器的 bit 12 更名为 RETEN（之前为 LVREN），使之与其他 PIC24F 器件一致。此外，更改了 RCON 寄存器的位描述（[寄存器 7-1](#)）以阐明它在控制保持稳压器方面的功能。

对本数据手册通篇做了以下修改：将 FPOR 配置寄存器的 bit 2 更名为 RETCFG（之前为 LVRCFG），使之与其他 PIC24F 器件一致。此外，更改了 FPOR 配置寄存器的位描述（[寄存器 26-6](#)）以阐明它在使能保持稳压器方面的功能。

对于 [第 10.4 节“基于稳压器的节能特性”](#)：

- 删除了所有有关快速唤醒睡眠模式的引用，该模式在本器件中未实现
- 将所有有关高电压稳压器的引用更改为片上稳压器
- 删除了所有有关低电压稳压器的引用，在大多数情况下，由保持稳压器替代
- 在 [第 10.4.3 节“保持休眠模式”](#)（之前的“低电压休眠模式”）中声明了保持稳压器的操作
- 修改了 [表 10-1](#)，使之与上述变更统一

更正了 [第 26.2 节“片内稳压器”](#) 以阐明片上稳压器在“F”和“FV”系列中的操作，并提供了直流参数和规范。

对于 [第 29.0 节“电气特性”](#)：

- 更新了所有规范表的标题以包含扩展级温度数据
- 更改了 [表 29-8](#)，使包含所有现有参数在 +125°C 下的数据
- 新增了 [表 29-27](#) 和 [图 29-8](#)，给出了通用定时器外部时钟输入规范的特征值（所有后续图表均相应重新编号）
- 给多张表中先前没有编号的参数添加了参数编号

更新了 [第 30.0 节“直流和交流特性图表”](#)：

- 添加了一些说明扩展级温度器件的图表（[第 30.2 节“扩展级温度器件的特性（-40°C 至 +125°C）”](#)，[图 30-40](#) 到 [图 30-56](#)）
- 用更新的图表替换了 [图 30-32](#)

用更新的图替换了 [第 31.0 节“封装信息”](#) 中的部分封装图。

更新了产品标识信息以在关键信息中包含扩展级温度器件。

通篇做了其他少量排版修订。

索引

数字

16 位同步 Timer2	143
---------------------	-----

A

A/D

采样要求	219
缓冲区数据格式	221
结果缓冲器	210
控制寄存器	210
AD1CHITH/L	210
AD1CHS	210
AD1CON1	210
AD1CON2	210
AD1CON3	210
AD1CON5	210
AD1CSSL/H	210
AD1CTMUENH/L	210
模块规范	293
传递函数	220
转换时序要求	294

B

版本历史	351
保留地址	171
比较器	225
比较器参考电压	229
配置	229
变更通知客户服务	359
波特率发生器	
设置为总线主器件	171

C

C 编译器

MPLAB C18	252
-----------------	-----

CPU

ALU	33
编程模型	29
控制寄存器	32
内核寄存器	30

CRC

典型操作	201
寄存器	201
用户接口	200
多项式	200
数据	200
数据移位方向	201
中断操作	201

CTMU

测量电容	231
测量时间	232
脉冲生成和延时	233

产品标识体系	361
--------------	-----

程序存储空间与数据存储空间

寻址空间	51
程序空间可视性	55
使用表指令访问	53

程序存储空间与数据存储空间的接口	51
------------------------	----

程序存储器

存储器映射	35
地址空间	35
构成	36
器件配置字	36
硬存储器向量	36

程序校验	250
充电时间测量单元。见 CTMU。	
串行外设接口。见 SPI。	

D

代码保护	250
------------	-----

代码示例

擦除程序存储器行，	
“C”语言代码	61
擦除程序存储器行，	
汇编语言代码	60
超低功耗唤醒初始化	131
单字擦除	66
I/O 端口写 / 读	138
将 RTCWREN 位置 1	186
解锁序列	126
启动编程序列，	
“C”语言代码	62
启动编程序列，	
汇编语言代码	62
使用 TBLRD 命令读取数据 EEPROM	68
时钟切换的基本代码序列	122
数据 EEPROM 解锁序列	63
数据 EEPROM 批量擦除	67
装载写缓冲器，“C”语言代码	62
装载写缓冲器，汇编语言代码	61
带阈值检测的 12 位 A/D 转换器	207
单字写数据 EEPROM	67
电气特性	
封装热阻特性	265
绝对最大值	263
热工作条件	265
V/F 关系图	264

F

复位

器件时间	72
欠压复位 (BOR)	73
RCON 标志操作	71
SFR 状态	73
深睡眠 BOR (DSBOR)	73
时钟源选择	72
延时时间	72

封装

详细信息	328
标识	325

G

高 / 低电压检测	205
高 / 低电压检测 (HLVD)	205

公式

A/D 转换时钟周期	219
BRGH = 0 时的 UARTx 波特率	178
BRGH = 1 时的 UART 波特率	178
波特率重载值	171
计算 PWM 周期	155
计算 PWM 周期和占空比	155
计算最大 PWM 分辨率	155
器件速率和 SPIx 时钟速率之间的关系	168

H

汇编器

MPASM 汇编器	252
-----------------	-----

PIC24FV32KA304 系列

I

I/O 端口

并行 (PIO)	135
漏极开路配置	136
模拟端口配置	136
模拟选择寄存器	136
输入电平变化通知	138

I²C

从地址掩码	171
时钟速率	171
引脚重新映射选项	169
作为主器件在单主器件环境中通信	169

I²C 模块

.....	170
-------	-----

J

寄存器

AD1CHITH (A/D 扫描比较命中, 低字)	216
AD1CHITH (A/D 扫描比较命中, 高字)	216
AD1CHS (A/D 采样选择)	215
AD1CON1 (A/D 控制 1)	211
AD1CON2 (A/D 控制 2)	212
AD1CON3 (A/D 控制 3)	213
AD1CON5 (A/D 控制 5)	214
AD1CTMUENH (CTMU 使能, 高字)	218
AD1CTMUENL (使能 CTMU, 低字)	218
ADCSSH (A/D 输入扫描选择, 高字)	217
ADCSSL (A/D 输入扫描选择, 低字)	217
ALCFGRPT (闹钟配置)	190
ALMINSEC (闹钟分钟和秒值)	194
ALMTHDY (闹钟月和日值)	193
ALWDHR (闹钟星期和小时值)	193
ANSA (模拟功能选择, PORTA)	136
ANSB (模拟功能选择, PORTB)	137
ANSC (模拟功能选择, PORTC)	137
CLKDIV (时钟分频器)	119
CMSTAT (比较器状态)	228
CMxCON (比较器 x 控制)	227
CORCON (CPU 内核控制)	80
CORCON (CPU 控制)	33
CRCCON1 (CRC 控制 1)	202
CRCCON2 (CRC 控制 2)	203
CRCXORH (CRC 多项式异或操作, 高字节)	204
CRCXORL (CRC 多项式异或操作, 低字节)	203
CTMUCON (CTMU 控制寄存器 1)	234
CTMUCON2 (CTMU 控制寄存器 2)	235
CTMUICON (CTMU 电流控制)	237
CVRCON (比较器参考电压控制)	230
DEVID (器件 ID)	246
DEVREV (器件版本)	247
DSCON (深度休眠控制)	129
DSWAKE (深度休眠唤醒源)	130
FBS (启动段配置)	239
FBS (通用段配置)	240
FDS (深睡眠配置)	245
FICD (在线调试器配置)	244
FOSC (振荡器配置)	241
FOSSEL (振荡器选择配置)	240
FPOR (复位配置)	243

FWDT (看门狗定时器配置)	242
HLVDCON (高 / 低电压检测控制)	206
I2CxMSK (I2Cx 从模式地址掩码)	176
I2CxSTAT (I2Cx 状态)	174
I2CxxCON (I2Cx 控制)	172
ICxCON1 (输入捕捉 x 控制 1)	149
ICxCON2 (输入捕捉 x 控制 2)	150
IEC0 (中断允许控制 0)	90
IEC1 (中断允许控制 1)	92
IEC2 (中断允许控制 2)	93
IEC3 (中断允许控制 3)	94
IEC4 (中断允许控制 4)	95
IEC5 (中断允许控制 5)	96
IFS0 (中断标志状态 0)	83
IFS1 (中断标志状态 1)	85
IFS2 (中断标志状态 2)	86
IFS3 (中断标志状态 3)	87
IFS4 (中断标志状态 4)	88
IFS5 (中断标志状态 5)	89
INTCON1 (中断控制 1)	81
INTCON2 (中断控制寄存器 2)	82
INTTREG (中断控制和状态)	112
IPC0 (中断优先级控制 0)	97
IPC1 (中断优先级控制 1)	98
IPC12 (中断优先级控制 12)	107
IPC15 (中断优先级控制 15)	108
IPC16 (中断优先级控制 16)	109
IPC18 (中断优先级控制 18)	110
IPC19 (中断优先级控制 19)	110
IPC2 (中断优先级控制 2)	99
IPC20 (中断优先级控制 20)	111
IPC3 (中断优先级控制 3)	100
IPC4 (中断优先级控制 4)	101
IPC5 (中断优先级控制 5)	102
IPC6 (中断优先级控制 6)	103
IPC7 (中断优先级控制 7)	104
IPC8 (中断优先级控制寄存器 8)	105
IPC9 (中断优先级控制 9)	106
MINSEC (RTCC 分钟和秒值)	192
MTHDY (RTCC 月和日值)	191
NVMCON (非易失性存储器控制)	64
NVMCON (闪存控制)	59
OCxCON1 (输出比较 x 控制 1)	157
OCxCON2 (输出比较 x 控制 2)	159
PADCFG1 (填充配置控制)	176
RCFGCAL (RTCC 校准和配置)	187
REFOCN (参考振荡器控制)	123
RTCCSWT (控制 / 采样窗口定时器)	195
RTCPWC (RTCC 配置 2)	189
SPIxCON1 (SPIx 控制 1)	166
SPIxCON2 (SPIx 控制 2)	167
SPIxSTAT (SPIx 状态和控制)	164
SR (ALU 状态)	32, 79
T1CON (Timer1 控制)	140
TxCON (Timer2/4 控制)	144
TxCON (Timer3/5 控制)	145
WKDYHR (RTCC 星期和小时值)	192
ULPWCON (ULPWU 控制)	132
UxMODE (UARTx 模式)	180
UxRXREG (UARTx 接收)	184
UxSTA (UARTx 状态和控制)	182

PIC24FV32KA304 系列

UxTXREG (UARTx 发送)	184
YEAR (RTCC 年值)	191
寄存器映射	
A/D 转换器 (ADC)	47
CPU 内核	39
CRC	49
CTMU	48
超低功耗唤醒	50
定时器	42
I2Cx	44
ICN	40
模拟选择	48
NVM	50
PMD	50
PORTA	45
PORTB	45
PORTC	46
SPIx	45
深睡眠	49
实时时钟和日历 (RTCC)	48
时钟控制	49
输出比较	43
输入捕捉	42
UARTx	44
引脚配置	46
中断控制器	41
交流特性	
比较器	280
比较器参考电压建立时间	280
复位、看门狗定时器、 振荡器起振定时器、上电复位定时器 和欠压复位时序要求	281
负载条件和要求	276
内部 RC 振荡器规范	278
内部 RC 振荡器精度	278
输出比较	283
输出引脚上的容性负载要求	276
温度和电压规范	276
节能	134
节能特性	125
电压，基于稳压器	
保持休眠模式	133
超低功耗唤醒	131
打盹模式	134
基于稳压器	133
低电压休眠模式	133
深休眠模式	133
休眠（待机）模式	133
运行模式	133
基于指令的模式	125
空闲	126
深睡眠	126
休眠	125
时钟频率和时钟切换	125
同时中断	126
选择性外设模块控制	134
K	
开发支持	251
看门狗定时器	
深睡眠 (DSWDT)	250
看门狗定时器 (WDT)	248
窗口操作	249
勘误表	9
客户通知服务	359
客户支持	359

框图

12 位 A/D 转换器	208
12 位 A/D 转换器模拟输入模型	219
16 位 Timer1 模块	139
16 位异步 Timer3 和 Timer5	143
比较器参考电压	229
比较器模块	225
表寄存器寻址	57
CALL 堆栈帧	51
CPU 编程模型	31
CRC 模块	199
CRC 移位引擎	199
串联电阻	131
电容测量的 CTMU 连接和内部配置	232
访问程序空间内的数据的地址生成方式	52
复位系统	69
高 / 低电压检测 (HLVD)	205
各个比较器配置	226
共用 I/O 端口的结构	135
简化 UARTx	177
看门狗定时器 (WDT)	249
PIC24F CPU 内核	30
PIC24FV32KA304 系列（通用）	15
PSV 操作	55
片内稳压器的连接	248
RTCC	185
SPI1 模块（标准缓冲模式）	162
SPI1 模块（增强型缓冲模式）	163
生成脉冲延时的 CTMU 典型连接和内部配置	233
时间测量的 CTMU 典型连接 和内部配置	232
使用表指令访问程序存储器	54
使用 TBLPAG 和 NVM 地址寄存器 对数据 EEPROM 进行寻址	65
输出比较 x（16 位模式）	152
输出比较 x（双缓冲， 16 位 PWM 模式）	154
输入捕捉 x	147
Timer2/3 和 Timer4/5（32 位）	142
系统时钟	115

M

Microchip 因特网网站	359
MPLAB ASM30 汇编器、链接器和库管理器	252
MPLAB PM3 器件编程器	253
MPLAB REAL ICE 在线仿真器系统	253
MPLAB 集成开发环境软件	251
MPLINK 目标链接器 / MPLIB 目标库管理器	252
脉宽调制。见 PWM。	
脉宽调制（PWM）模式	154

N

Near 数据空间	38
内核特性	11

O

OSCCON（振荡器控制寄存器）	117
OSCTUN（FRC 振荡器调节寄存器）	120

P

PIC24F32KA304 系列器件特性（汇总）	14
PIC24FV32KA304 系列器件特性（汇总）	13
PWM	
占空比和周期	155
PWRSV 指令语法	125

PIC24FV32KA304 系列

配置位	239
片内稳压器	248

Q

欠压复位	
跳变点	266

R

RCON（复位控制寄存器）	70
RTCC	185
模块寄存器	186
写锁定	186
映射	186
时钟源选择	186
闹钟配置	196
闹钟掩码设置（图）	197
时钟源	185
校准	196
软件堆栈	51
软件模拟器（MPLAB SIM）	253

S

SFR 空间	38
闪存程序存储器	
编程操作	58
编程算法	60
表指令	57
控制寄存器	58
RTSP 操作	58
增强型 ICSP 操作	58

示例

波特率误差计算（BRGH = 0）	178
-------------------------	-----

时序图

A/D 转换	294
CLKO 和 I/O 时序	279
复位、看门狗定时器、振荡器起振定时器	
和上电延时定时器	280
I ² C™ 总线启动 / 停止位	
（主模式）	284
I ² C™ 总线数据	
（主模式）	285
I ² C™ 总线启动 / 停止位	
（从模式）	287
I ² C™ 总线数据	
（从模式）	286
PWM 模块要求	283
欠压复位特性	281
SPIx 模块从模式（CKE = 0）	291
SPIx 模块从模式（CKE = 1）	292
SPIx 模块主模式（CKE = 0）	289
SPIx 模块主模式（CKE = 1）	290
输出比较	283
输入捕捉	282
UARTx 波特率发生器输出	288
UARTx 启动位边沿检测	288
外部时钟	277

时序要求

I ² C 总线启动 / 停止位要求	
（主模式）	284
A/D 转换	294
CLKO 和 I/O	279
I ² C 总线启动 / 停止位（从模式）	287
I ² C 总线数据（从模式）	286
I ² C 总线数据（主模式）	285
I ² C™ 总线启动 / 停止位	284

PLL 时钟规范	278
PWM	283
SPIx 从模式（CKE = 0）	291
SPIx 主模式（CKE = 0）	289
UARTx	288
外部时钟	277
Timer1/2/3/4/5 扩展时钟时序	282

使用专用定时器的输入捕捉	147
--------------------	-----

输出比较

32 位模式	151
操作	153
同步和触发模式	151
子循环分辨率	156

数据存储

存储器映射	37
地址空间	37
构成	38
空间宽度	37
Near 数据空间	38
软件堆栈	51
SFR 空间	38

数据 EEPROM 存储器	63
---------------------	----

编程

单字写	67
读数据 EEPROM	68
批量擦除	67

编程控制寄存器

NVMADR（U）	65
NVMCON	63
NVMKEY	63
擦除	66
操作	65

输入捕捉

32 位模式	148
操作	148
同步和触发模式	147

T

Timer2/3	141
Timer2/3 和 Timer4/5	141
Timer1	139

W

UART	177
8 位或 9 位数据模式下的接收	179
8 位数据模式下的发送	179
9 位数据模式下的发送	179
波特率发生器（BRG）	178
IrDA 支持	179
间隔和同步发送序列	179
UxCTS 和 UxRTS 控制引脚的操作	179
WWW 地址	359

Y

引脚配置说明	16
因特网地址	359

Z

在线串行编程（ICSP）	250
在线调试器	250
振荡器配置	
CPU 时钟机制	116
参考时钟输出	122
上电复位时的初始配置	116
时钟切换	121
序列	121

PIC24FV32KA304 系列

用于时钟选择的配置位值	116	I/O 引脚输入规范	272
指令集		空闲电流 (IDLE)	268
操作码符号	256	内部稳压器规范	275
汇总	255, 257	数据 EEPROM 存储器	274
直流和交流特性图表	295	温度和电压规范	265
扩展级温度	315	中断	
工业级温度	295	备用中断向量表 (AIVT)	75
直流特性		复位过程	75
BOR 跳变点	266	控制和状态寄存器	78
比较器	274	设置过程	113
比较器参考电压	274	实现的向量	77
CTMU 电流源	275	陷阱向量	77
程序存储器	273	向量表	76
掉电电流 (IPD)	269	中断向量表 (IVT)	75
高 / 低电压检测	266		
工作电流 (IDD)	267		
I/O 引脚输出规范	273		

PIC24FV32KA304 系列

注:

MICROCHIP 网站

Microchip 网站 (www.microchip.com) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的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归档软件
- **一般技术支持**——常见问题解答 (FAQ)、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

变更通知客户服务

Microchip 的变更通知客户服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请登录 Microchip 网站 www.microchip.com。在“支持” (Support) 下，点击“变更通知客户 (Customer Change Notification)” 服务后按照注册说明完成注册。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过 <http://microchip.com/support> 获得网上技术支持。

PIC24FV32KA304 系列

注:

PIC24FV32KA304 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《数字器件千年版权法案（Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

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

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、适销性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

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

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

Silicon Storage Technology 为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

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

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

GestIC 和 ULPP 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。

© 2011-2013, Microchip Technology Inc. 版权所有。

ISBN: 978-1-62077-360-4

**QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
= ISO/TS 16949 =**

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。

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

亚特兰大 Atlanta
Duluth, GA
Tel: 1-678-957-9614
Fax: 1-678-957-1455

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

克里夫兰 Cleveland
Independence, OH
Tel: 1-216-447-0464
Fax: 1-216-447-0643

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

印第安纳波利斯 Indianapolis
Noblesville, IN
Tel: 1-317-773-8323
Fax: 1-317-773-5453

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara
Santa Clara, CA
Tel: 1-408-961-6444
Fax: 1-408-961-6445

加拿大多伦多 Toronto
Mississauga, Ontario, Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 **Asia Pacific Office**
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 北京
Tel: 86-10-8569-7000
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 - 杭州
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

中国 - 香港特别行政区
Tel: 852-2943-5100
Fax: 852-2401-3431

中国 - 南京
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳
Tel: 86-755-8864-2200
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 - 珠海
Tel: 86-756-321-0040
Fax: 86-756-321-0049

亚太地区

台湾地区 - 高雄
Tel: 886-7-213-7828
Fax: 886-7-330-9305

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: 91-20-2566-1512
Fax: 91-20-2566-1513

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

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Druenen
Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820