

## 数字增强型电源模拟同步下桥臂 PWM 控制器

### 特性

- 输入电压：4.5V 至 42V
- 可配置多种拓扑结构，包括但不限于：
  - 反激式
  - Ćuk
  - 升压
  - SEPIC（单端初级电感变换器）
- 能够工作在准谐振模式或固定频率下
- 低静态电流：5 mA（典型值）
- 低休眠电流：30  $\mu$ A（典型值）
- 下桥臂栅极驱动器：
  - +5V 栅极驱动
  - 0.5A 灌 / 拉电流
  - +10V 栅极驱动
  - 1A 灌 / 拉电流
- 峰值电流模式控制
- 差分远程输出检测
- 多输出系统：
  - 主器件或从器件
- 符合 AEC-Q100 标准
- 可配置参数：
  - $V_{REF}$ ，高精度  $I_{OUT}/V_{OUT}$  设定值（DAC）
  - 输入欠压锁定（Undervoltage Lockout, UVLO）
  - 输入过压锁定（Overvoltage Lockout, OVLO）
  - 检测和保护的
  - 初级电流前沿消隐（0、50 ns、100 ns 和 200 ns）
  - 栅极驱动死区（16 ns 至 256 ns）
  - 固定开关频率范围：31.25 kHz 至 2.0 MHz
  - 斜率补偿
  - 准谐振配置，具有内置比较器和可编程失调电压调节
  - 初级电流失调调节
  - 可配置 GPIO 引脚选项
- 集成下桥臂差分电流检测放大器
- $\pm 5\%$  电流调节
- 热关断

### 单片机特性

- 高精度 8 MHz 内部振荡器模块：
  - 出厂校准为  $\pm 1\%$ （典型值）
- 中断功能
  - 固件
  - 电平变化中断引脚
- 仅需学习 35 条指令
- 4096 字片上程序存储器
- 高耐擦写闪存：
  - 闪存耐写次数达 100,000 次
  - 闪存数据保存期：>40 年
- 带独立振荡器的看门狗定时器（Watchdog Timer, WDT），确保可靠工作
- 可编程代码保护
- 通过两个引脚进行在线串行编程（In-Circuit Serial Programming™, ICSP™）
- 8 个 I/O 引脚和 1 个仅用作输入的引脚
  - 2 个漏极开路引脚
- 模数转换器（Analog-to-Digital Converter, ADC）：
  - 10 位分辨率
  - 5 个外部通道
- Timer0：带 8 位预分频器的 8 位定时器 / 计数器
- 增强型 Timer1：
  - 带预分频器的 16 位定时器
  - 2 个可选时钟源
- Timer2：带预分频器的 8 位定时器
  - 8 位周期寄存器
- I<sup>2</sup>C™ 通信：
  - 7 位地址掩码
  - 2 个专用地址寄存器

# MCP19114/5

引脚图—— 24 引脚 QFN (MCP19114)

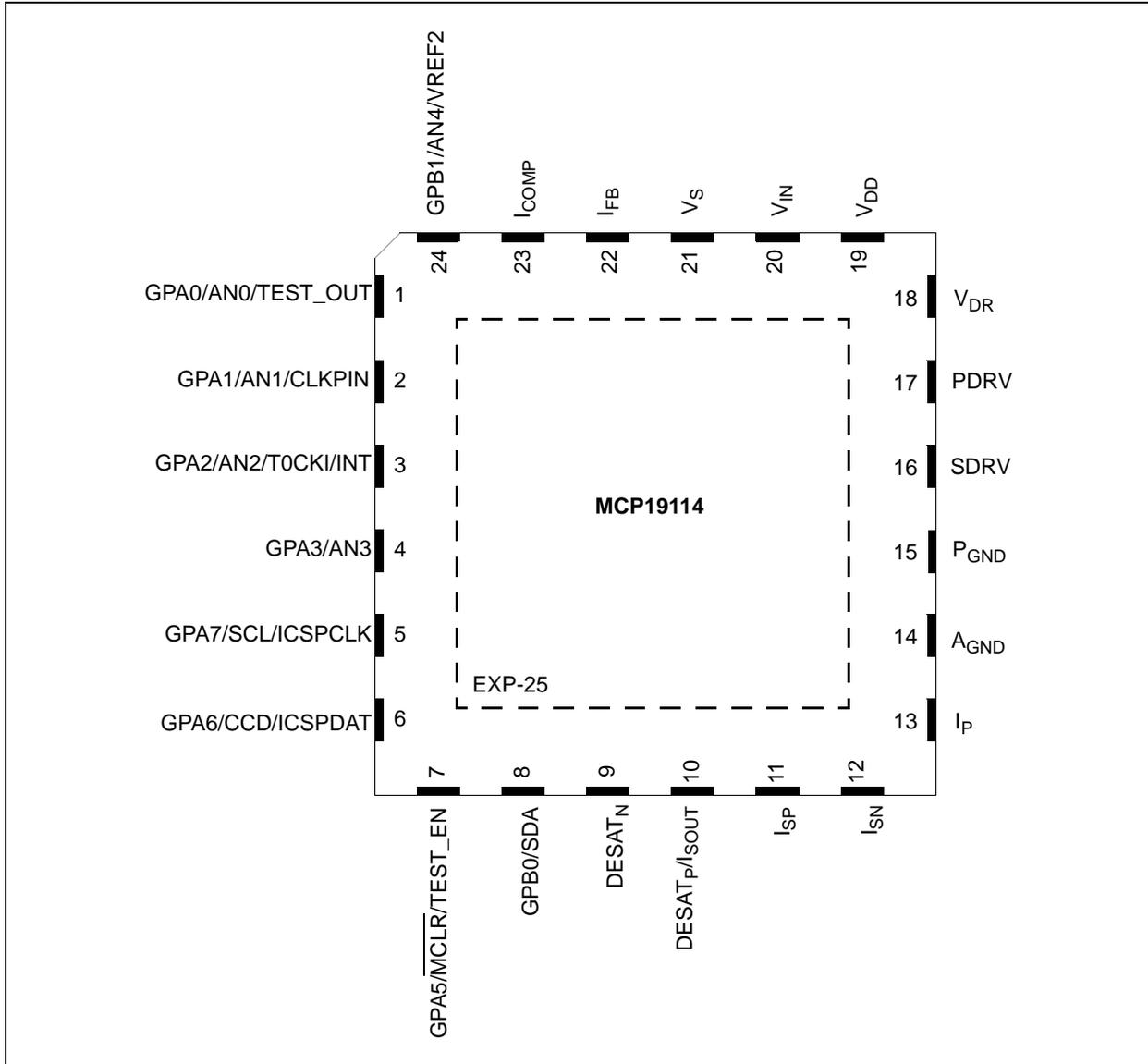


表 1: 24 引脚汇总

I/O	24 引脚 QFN	ANSEL	A/D	定时器	MSSP	中断	上拉	基本功能	其他
GPA0	1	有	AN0	—	—	IOC	有	—	模拟 / 数字调试输出 <sup>(1)</sup>
GPA1	2	有	AN1	—	—	IOC	有	—	同步信号输入 / 输出 <sup>(2)</sup>
GPA2	3	有	AN2	TOCKI	—	IOC INT	有	—	—
GPA3	4	有	AN3	—	—	IOC	有	—	—
GPA5	7	无	—	—	—	IOC <sup>(4)</sup>	有 <sup>(5)</sup>	MCLR	测试使能输入
GPA6	6	无	—	—	—	IOC	有	ICSPDAT	双捕捉 / 比较输入
GPA7	5	无	—	—	SCL	IOC	无	ICSPCLK	—
GPB0	8	无	—	—	SDA	IOC	无	—	—
GPB1	24	有	AN4	—	—	IOC	有	—	V <sub>REF2</sub> <sup>(3)</sup>
DESAT <sub>N</sub>	9	无	—	—	—	—	—	—	DESAT 反相输入
DESAT <sub>P</sub> / I <sub>SOUT</sub>	10	无	—	—	—	—	—	—	DESAT <sub>P</sub> 输入或 I <sub>SOUT</sub> 输出 <sup>(6)</sup>
I <sub>SP</sub>	11	无	—	—	—	—	有	—	电流检测放大器同相输入
I <sub>SN</sub>	12	无	—	—	—	—	—	—	电流检测放大器反相输入
I <sub>P</sub>	13	无	—	—	—	—	—	—	初级输入电流检测
A <sub>GND</sub>	14	无	—	—	—	—	—	A <sub>GND</sub>	小信号地
P <sub>GND</sub>	15	无	—	—	—	—	—	P <sub>GND</sub>	大信号地
SDRV	16	无	—	—	—	—	—	—	次级 LS 栅极驱动输出
PDRV	17	无	—	—	—	—	—	—	初级 LS 栅极驱动输出
V <sub>DR</sub>	18	无	—	—	—	—	—	V <sub>DR</sub>	栅极驱动电源电压
V <sub>DD</sub>	19	无	—	—	—	—	—	V <sub>DD</sub>	V <sub>DD</sub> 输出
V <sub>IN</sub>	20	无	—	—	—	—	—	V <sub>IN</sub>	输入电源电压
V <sub>S</sub>	21	无	—	—	—	—	—	—	输出电压检测
I <sub>FB</sub>	22	无	—	—	—	—	—	—	误差放大器反馈输入
I <sub>COMP</sub>	23	无	—	—	—	—	—	—	误差放大器输出

- 注 1: 模拟 / 数字调试输出通过控制 ABECON 寄存器进行选择。
- 2: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件用作主器件或从器件时, 选择此项。
- 3: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件配置为主器件时, 选择 V<sub>REF2</sub> 输出。
- 4: 使能 MCLR 时, 禁止 IOC。
- 5: 使能 MCLR 时, 将始终使能弱上拉; 否则, 上拉由用户控制。
- 6: 当 MODECON<5> (RFB) 位 = 0 时, 使能内部反馈电阻和 DESAT<sub>P</sub> 输入。当 RFB = 1 时, 使能 I<sub>SOUT</sub>。

# MCP19114/5

引脚图——28 引脚 QFN (MCP19115)

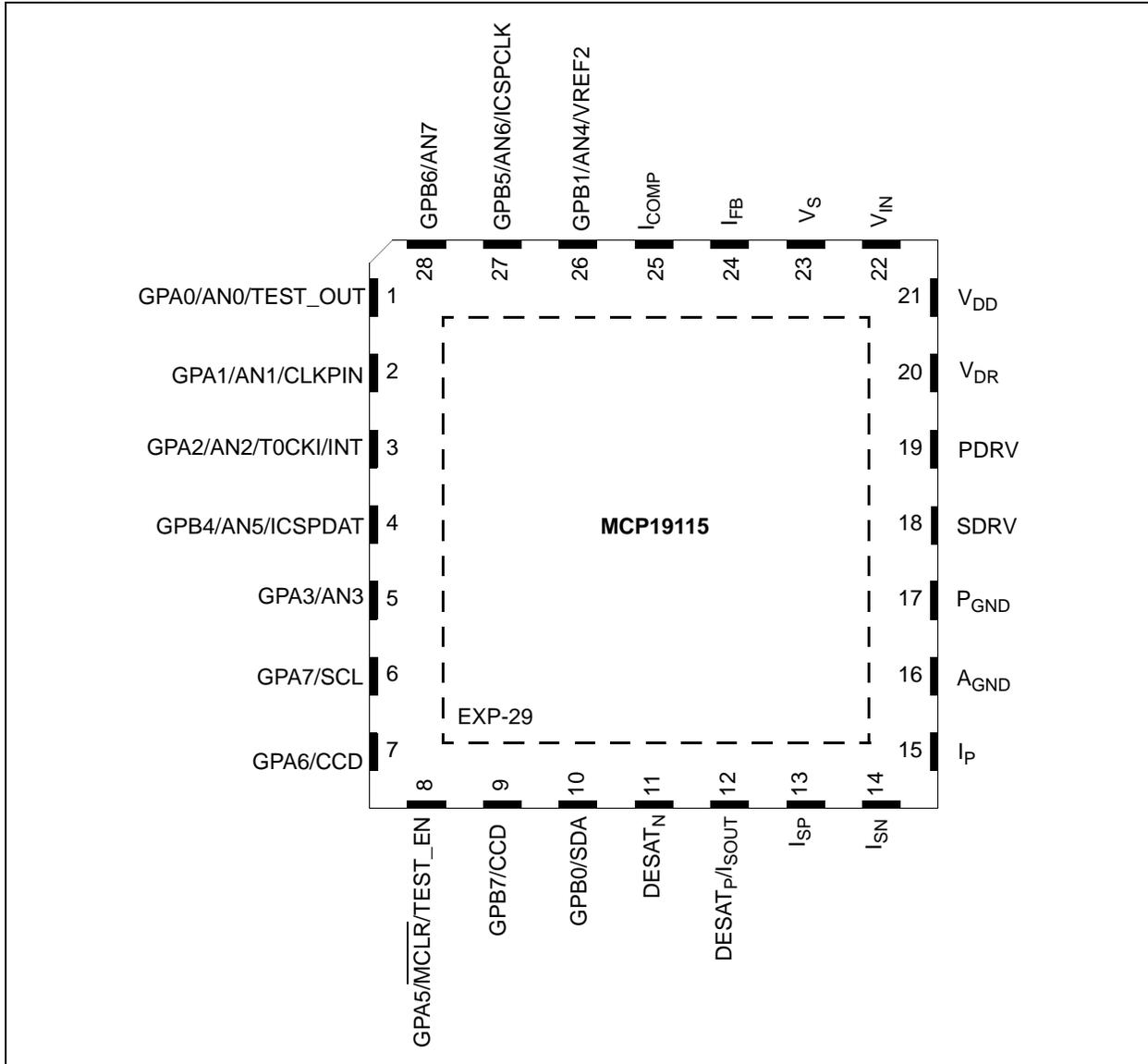


表 2: 28 引脚汇总

I/O	28 引脚 QFN	ANSEL	A/D	定时器	MSSP	中断	上拉	基本功能	其他
GPA0	1	有	AN0	—	—	IOC	有	—	模拟 / 数字调试输出 (1)
GPA1	2	有	AN1	—	—	IOC	有	—	同步信号输入 / 输出 (2)
GPA2	3	有	AN2	TOCKI	—	IOC INT	有	—	—
GPA3	5	有	AN3	—	—	IOC	有	—	—
GPA5	8	无	—	—	—	IOC(4)	有 (5)	$\overline{\text{MCLR}}$	测试使能输入
GPA6	7	无	—	—	—	IOC	有	—	双捕捉 / 单比较 1 输入
GPA7	6	无	—	—	SCL	IOC	无	—	—
GPB0	10	无	—	—	SDA	IOC	无	—	—
GPB1	26	有	AN4	—	—	IOC	有	—	$V_{\text{REF2}}$ (3)
GPB4	4	有	AN5	—	—	IOC	有	ICSPDAT	—
GPB5	27	有	AN6	—	—	IOC	有	ICSPCLK	—
GPB6	28	有	AN7	—	—	IOC	有	—	—
GPB7	9	有	—	—	—	IOC	有	—	单比较 2 输入
DESAT <sub>P</sub> / I <sub>SOUT</sub>	12	无	—	—	—	—	—	—	DESAT <sub>P</sub> 输入或 I <sub>SOUT</sub> 输出 (6)
DESAT <sub>N</sub>	11	无	—	—	—	—	—	—	DESAT 反相输入
I <sub>SP</sub>	13	无	—	—	—	—	有	—	电流检测放大器同相输入
I <sub>SN</sub>	14	无	—	—	—	—	—	—	电流检测放大器反相输入
I <sub>P</sub>	15	无	—	—	—	—	—	—	初级输入电流检测
A <sub>GND</sub>	16	无	—	—	—	—	—	A <sub>GND</sub>	小信号地
P <sub>GND</sub>	17	无	—	—	—	—	—	P <sub>GND</sub>	大信号地
SDRV	18	无	—	—	—	—	—	—	次级 LS 栅极驱动输出
PDRV	19	无	—	—	—	—	—	—	初级 LS 栅极驱动输出
V <sub>DR</sub>	20	无	—	—	—	—	—	V <sub>DR</sub>	栅极驱动电源电压
V <sub>DD</sub>	21	无	—	—	—	—	—	V <sub>DD</sub>	V <sub>DD</sub> 输出
V <sub>IN</sub>	22	无	—	—	—	—	—	V <sub>IN</sub>	输入电源电压
V <sub>S</sub>	23	无	—	—	—	—	—	—	输出电压检测
I <sub>FB</sub>	24	无	—	—	—	—	—	—	误差放大器反馈输入
I <sub>COMP</sub>	25	无	—	—	—	—	—	—	误差放大器输出

注 1: 模拟 / 数字调试输出通过控制 ABECON 寄存器进行选择。

2: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件用作主器件或从器件时, 选择此项。

3: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件配置为主器件时, 选择 VREF2 输出。

4: 使能  $\overline{\text{MCLR}}$  时, 禁止 IOC。

5: 使能  $\overline{\text{MCLR}}$  时, 将始终使能弱上拉; 否则, 上拉由用户控制。

6: 当 MODECON<5> (RFB) 位 = 0 时, 使能内部反馈电阻和 DESAT<sub>P</sub> 输入。当 RFB = 1 时, 使能 I<sub>SOUT</sub>。

# MCP19114/5

---

## 目录

1.0	器件概述	9
2.0	引脚说明	13
3.0	功能说明	19
4.0	电气特性	22
5.0	数字电气特性	29
6.0	配置 MCP19114/5	37
7.0	典型性能曲线	53
8.0	系统基准测试	57
9.0	器件校准	59
10.0	存储器构成	69
11.0	器件配置	81
12.0	振荡器模式	83
13.0	复位	85
14.0	中断	93
15.0	掉电模式（休眠）	101
16.0	看门狗定时器（WDT）	103
17.0	闪存程序存储器控制	105
18.0	I/O 端口	111
19.0	电平变化中断	121
20.0	内部温度指示器模块	125
21.0	模数转换器（ADC）模块	127
22.0	Timer0 模块	137
23.0	带门控的 Timer1 模块	139
24.0	Timer2 模块	143
25.0	增强型 PWM 模块	145
26.0	双捕捉/比较（CCD）模块	149
27.0	PWM 控制逻辑	153
28.0	主同步串行口（MSSP）模块	155
29.0	指令集汇总	199
30.0	在线串行编程（ICSP™）	209
31.0	开发支持	211
32.0	封装信息	215
附录 A:	版本历史	221
索引		223
Microchip 网站		229
变更通知客户服务		229
客户支持		229
产品标识体系		231

## 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需求。出版物的质量将随新文档及更新版本的推出而得到提升。

如果您对本出版物有任何问题和建议，请通过电子邮件联系我公司 TRC 经理，电子邮件地址为 [CTRC@microchip.com](mailto:CTRC@microchip.com)。我们期待您的反馈。

### 最新数据手册

欲获得本数据手册的最新版本，请访问我公司的网站：

<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。文献编号中紧跟数字串后的字母是版本号，例如：DS30000000A\_CN 是文档的 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 <http://www.microchip.com> 上注册。

# MCP19114/5

---

注:

## 1.0 器件概述

MCP19114/5 是高度集成的混合信号下桥臂同步控制器，工作电压为 4.5V 到 42V。该系列器件具有一个模拟 PWM 控制器，集成了适合 LED 照明系统、电池充电器和其他下桥臂开关 PWM 应用的单片机内核。MCP19114/5 器件具有一个类似于 MCP1631 的内部模拟 PWM 控制器和一个类似于 PIC12F617 的 PIC<sup>®</sup> 单片机。

通过使用 Microchip 的 MPLAB<sup>®</sup> X 集成开发环境软件和许多在线调试器和器件编程器中的其中一款可实现器件工作参数、启动或关闭曲线、保护等级以及故障处理过程的完全定制。

MCP19114/5 混合信号下桥臂同步控制器具有集成可编程输入 UVLO/OVLO、可编程输出过压 (OverVoltage, OV)、两个具有独立可编程死区的下桥臂栅极驱动输出、可编程前沿消隐 (四步)、6 位可编程斜率补偿和适合固定频率应用的内部集成可编程振荡器。集成的 8 位参考电压 ( $V_{REF}$ ) 用于设置输出电压或电流。内部比较器支持准谐振应用。集成了额外的捕捉和比较模块实现更多控制，包括增强型调光功能。MCP19114/5 器件包含两个内部 LDO。5V LDO 用于向内部处理器供电，并提供 5V 外部输出。该 5V 外部输出可用于为栅极驱动供电。通过  $V_{DD}$  供电实现 5V 栅极驱动时，建议在  $V_{DD}$  输出和  $V_{DR}$  输入之间放置一个模拟滤波器。建议尽可能靠近  $V_{DD}$  和  $V_{DR}$  分别放置一个 4.7  $\mu$ F 电容，两者通过一个 10 $\Omega$  隔离电阻隔开。 $V_{DD}$  上的电容不要超过 10  $\mu$ F。需要一个外部电源来实现更高的栅极驱动电压。通过使用 Microchip 的 TC1240A 倍压器 (由  $V_{DD}$  供电) 来提供  $V_{DR}$ ，可实现 10V 栅极驱动。4V LDO 用于向内部模拟电路供电。两个下桥臂驱动器可使电源转换器工作在双向模式下，可实现 LED 应用中 LED 调光电流的成形或开发适合电池供电应用的双向电源转换器。

MCP19114 的封装形式为 24 引脚 4 mm x 4 mm QFN，并提供备用结合 28 引脚 5 mm x 5 mm QFN 封装。MCP19115 的封装形式为 28 引脚 5 mm x 5 mm QFN。

MCP19114/5 使系统设计人员能够配置特定于应用的功能，从而可提供比现有集成器件更小的封装。可将 MCP19114/5 的通用输入 / 输出 (General Purpose Input/Output, GPIO) 配置为提供状态输出；器件使能以控制外部开关；开关频率同步输出或输入、甚至器件状态，以及“脉冲信号”指示。该灵活性允许 MCP19114/5 封装和完整解决方案更小，从而减小系统印刷电路板的尺寸并节约成本。

因输出电流调节和动态输出电压配置之类的集成特性，MCP19114/5 系列器件在同类中具有最佳性能且目前集成级别最高。

该架构支持的电源包括但不限于升压、反激、准谐振反激、SEPIC、Ćuk 等。两个下桥臂栅极驱动器在 10V  $V_{DR}$  下的拉 / 灌电流能力为 1A。使用 5V 栅极驱动时，驱动器的拉 / 灌电流能力为 0.5A。用户可选择将 UVLOEN 位置 1 允许  $V_{IN}$  UVLO 关闭驱动器。当该位未置 1 时，器件驱动器将穿越 UVLO 条件并继续工作直至  $V_{DR}$  达到栅极驱动 UVLO 值。该值可选择为 2.7V 或 5.4V 且始终使能。单片机内核的内部复位电压设置为 2.0V。内部比较器模块用于检测反激变压器的退饱和以同步准谐振应用的开关。器件正常工作时的输入电压范围为从 4.5V 到 42V，绝对最大值为 44V。最大瞬态电压为 48V，持续 500 ms。I<sup>2</sup>C 串行总线用于从 PWM 控制器到系统的设备通信。

图 1-1: MCP19114/5 反激同步准谐振框图

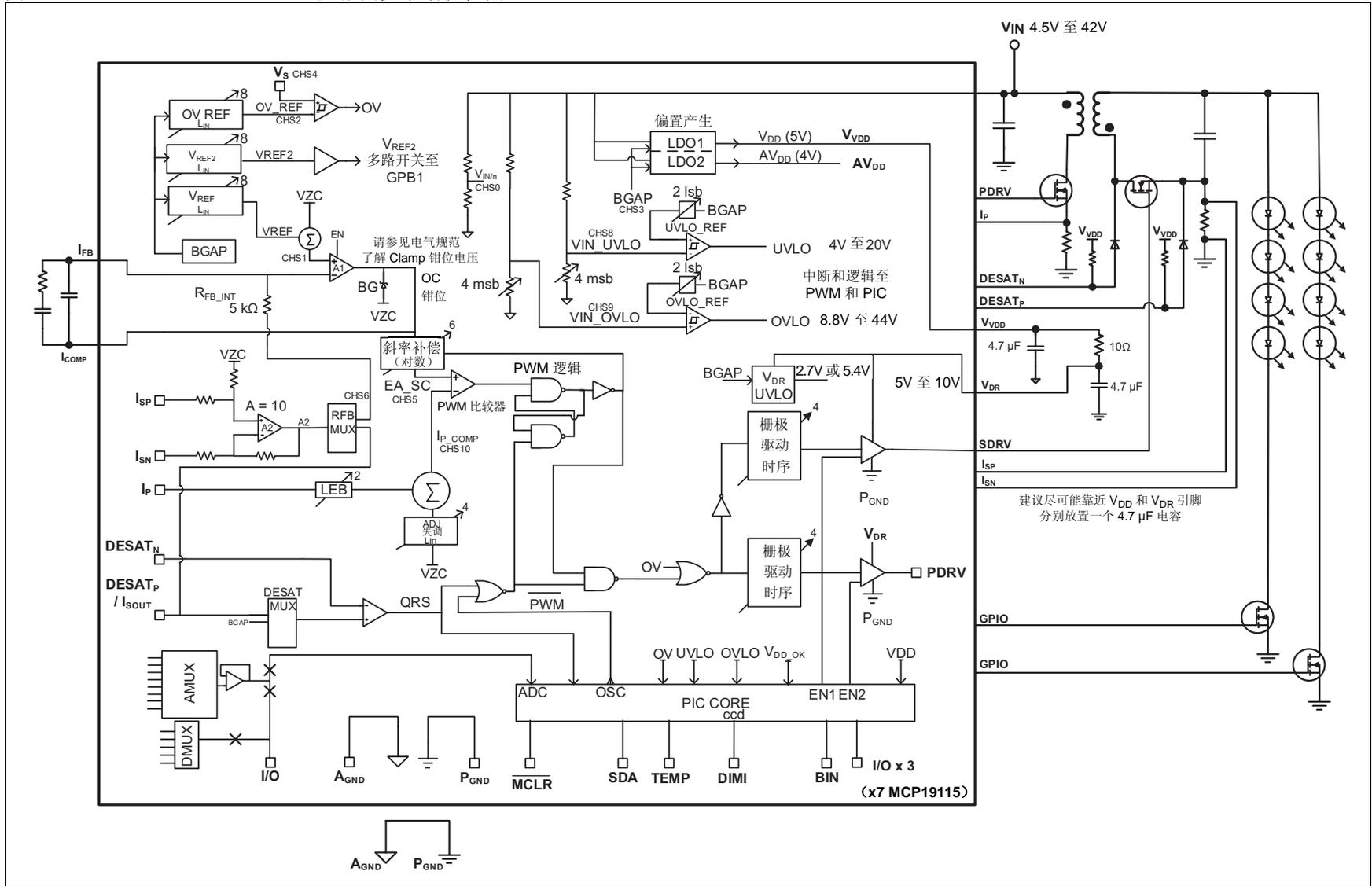


图 1-2: MCP19114 ĆUK 同步同相输出应用图

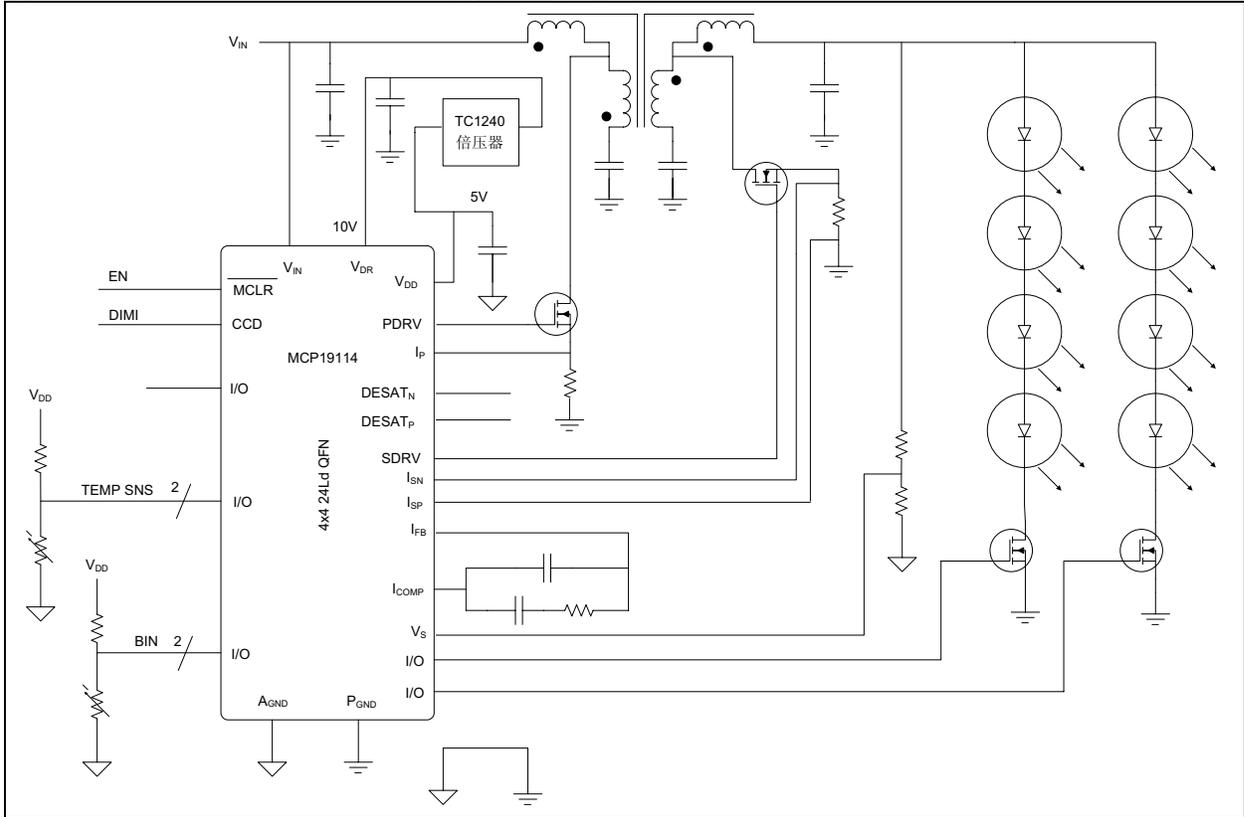
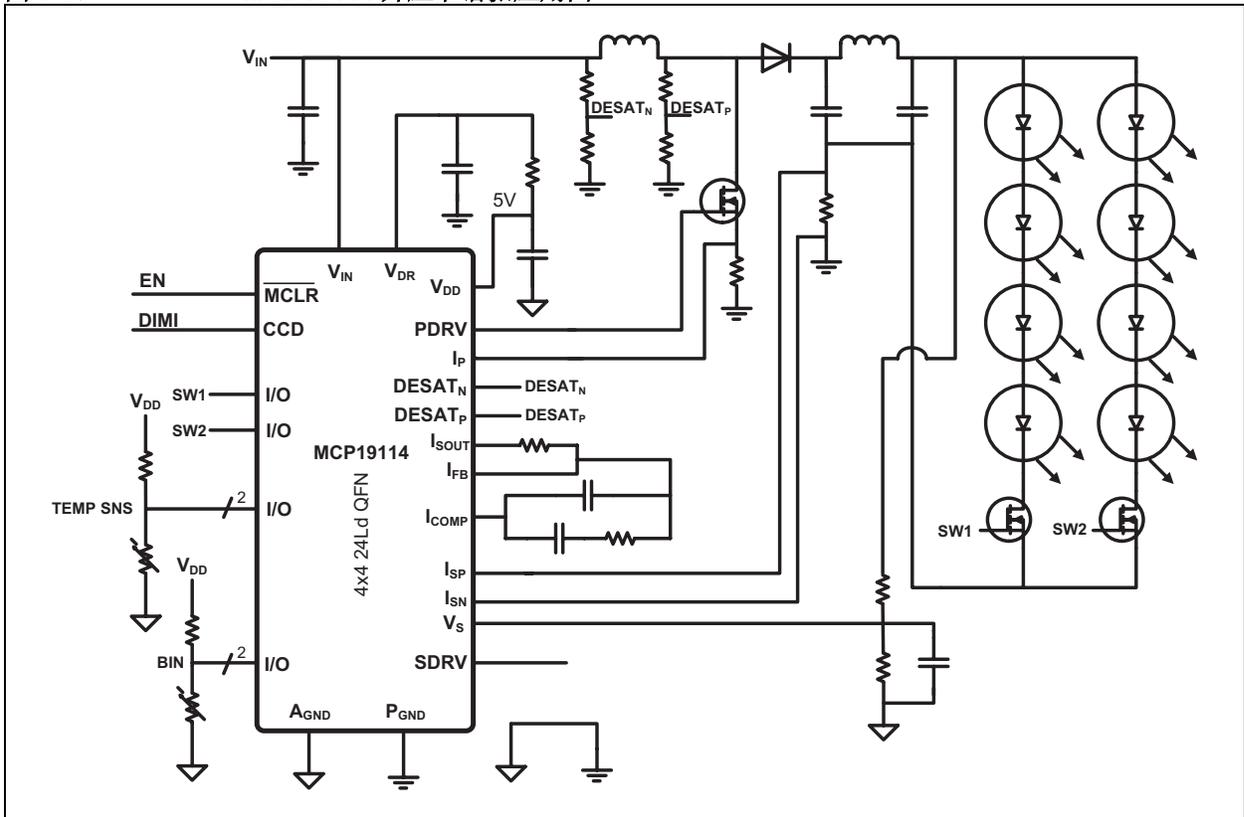
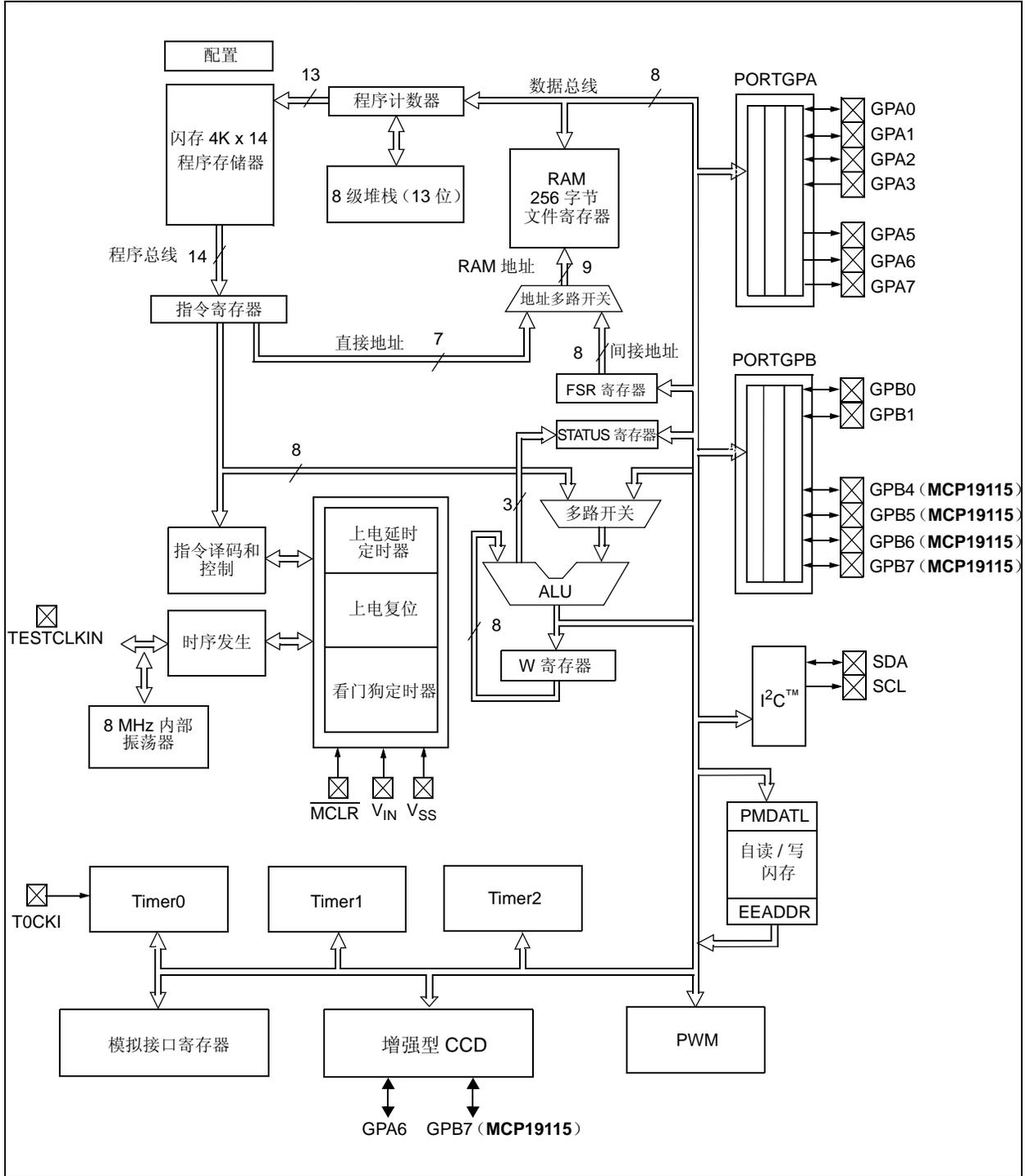


图 1-3: MCP19114 升压准谐振应用图



# MCP19114/5

图 1-4: 单片机内核框图



## 2.0 引脚说明

24 引脚 MCP19114 和 28 引脚 MCP19115 器件的一些引脚具有多个功能。表 2-1 介绍了这些不同功能。更多详细信息，请参见第 2.1 节“详细的引脚功能说明”。

表 2-1: MCP19114/5 引脚排列说明

名称	功能	输入类型	输出类型	说明
GPA0/AN0/TEST_OUT	GPA0	TTL	CMOS	通用 I/O
	AN0	AN	—	A/D 通道 0 输入
	TEST_OUT	—	—	内部模拟 / 数字信号多路开关输出 <sup>(1)</sup>
GPA1/AN1/CLKPIN	GPA1	TTL	CMOS	通用 I/O
	AN1	AN	—	A/D 通道 1 输入
	CLKPIN	ST	CMOS	开关频率时钟输入或输出 <sup>(2)</sup>
GPA2/AN2/T0CKI/INT	GPA2	ST	CMOS	通用 I/O
	AN2	AN	—	A/D 通道 2 输入
	T0CKI	ST	—	Timer0 时钟输入
	INT	ST	—	外部中断
GPA3/AN3	GPA3	TTL	CMOS	通用 I/O
	AN3	AN	—	A/D 通道 3 输入
GPA5/MCLR	GPA5	TTL	—	仅通用输入
	MCLR	ST	—	带内部上拉的主复位
GPA6/CCD/ICSPDAT	GPA6	ST	CMOS	通用 I/O
	ICSPDAT	ST	CMOS	串行编程数据 I/O
	CCD	ST	CMOS	单比较输出。双捕捉输入
GPA7/SCL/ICSPCLK	GPA7	ST	OD	通用漏极开路 I/O
	SCL	I <sup>2</sup> C™	OD	I <sup>2</sup> C 时钟
	ICSPCLK	ST	—	串行编程时钟
GPB0/SDA	GPB0	TTL	OD	通用 I/O
	SDA	I <sup>2</sup> C	OD	I <sup>2</sup> C 数据输入 / 输出
GPB1/AN4/VREF2	GPB1	TTL	CMOS	通用 I/O
	AN4	AN	—	A/D 通道 4 输入
	VREF2	—	AN	VREF2 DAC 输出 <sup>(3)</sup>
GPB4/AN5/ICSPDAT (仅 MCP19115)	GPB4	TTL	CMOS	通用 I/O
	AN5	AN	—	A/D 通道 5 输入
	ICSPDAT	ST	CMOS	主串行编程数据 I/O
GPB5/AN6/ICSPCLK (仅 MCP19115)	GPB5	TTL	CMOS	通用 I/O
	AN6	AN	—	A/D 通道 6 输入
	ICSPCLK	ST	—	主串行编程时钟

图注: AN = 模拟输入或输出      CMOS = CMOS 兼容输入或输出      OD = 漏极开路  
TTL = TTL 兼容输入      ST = 带 CMOS 电平的施密特触发器输入      I<sup>2</sup>C = 带 I<sup>2</sup>C 的施密特触发器输入

- 注 1: 模拟 / 数字调试输出通过控制 ABECON 寄存器进行选择。  
 2: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件用作主器件或从器件时，选择此项。  
 3: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件配置为主器件时，选择 VREF2 输出。

# MCP19114/5

表 2-1: MCP19114/5 引脚排列说明 (续)

名称	功能	输入类型	输出类型	说明
GPB6/AN7 (仅 MCP19115)	GPB6	TTL	CMOS	通用 I/O
	AN7	AN	—	A/D 通道 7 输入
GPB7/CCD (仅 MCP19115)	GPB7	TTL	CMOS	通用 I/O
	CCD	ST	CMOS	单比较输出。双捕捉输入
V <sub>IN</sub>	V <sub>IN</sub>	—	—	器件输入电源电压
V <sub>DD</sub>	V <sub>DD</sub>	—	—	内部 +5V LDO 输出引脚
V <sub>DR</sub>	V <sub>DR</sub>	—	—	栅极驱动电源电压
A <sub>GND</sub>	A <sub>GND</sub>	—	—	小信号“安静”地
P <sub>GND</sub>	P <sub>GND</sub>	—	—	大信号电源地
PDRV	PDRV	—	—	初级下桥臂 MOSFET 栅极驱动
SDRV	SDRV	—	—	次级下桥臂 MOSFET 栅极驱动
I <sub>P</sub>	I <sub>P</sub>	—	—	初级输入电流检测
I <sub>SN</sub>	I <sub>SN</sub>	—	—	次级电流检测放大器反相输入
I <sub>SP</sub>	I <sub>SP</sub>	—	—	次级电流检测放大器同相输入
V <sub>S</sub>	V <sub>S</sub>	—	—	检测电压与过压 DAC 作比较
I <sub>FB</sub>	I <sub>FB</sub>	—	—	误差放大器反馈输入
I <sub>COMP</sub>	I <sub>COMP</sub>	—	—	误差放大器输出
DESAT <sub>P</sub> /I <sub>SOUT</sub>	DESAT <sub>P</sub> /I <sub>SOUT</sub>	—	—	DESAT <sub>P</sub> : DESAT 检测比较器同相输入 I <sub>SOUT</sub> : 次级电流检测放大器输出
DESAT <sub>N</sub>	DESAT <sub>N</sub>	—	—	DESAT <sub>N</sub> : DESAT 检测比较器反相输入

图注: AN = 模拟输入或输出      CMOS = CMOS 兼容输入或输出      OD = 漏极开路  
TTL = TTL 兼容输入      ST = 带 CMOS 电平的施密特触发器输入      I<sup>2</sup>C = 带 I<sup>2</sup>C 的施密特触发器输入

- 注 1: 模拟 / 数字调试输出通过控制 ABECON 寄存器进行选择。  
2: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件用作主器件或从器件时, 选择此项。  
3: 通过正确配置 MODECON 寄存器中的 MSC<1:0> 位将器件配置为主器件时, 选择 VREF2 输出。

## 2.1 详细的引脚功能说明

### 2.1.1 GPA0 引脚

GPA0 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPA 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN0 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 0 读取，必须将 TRISA0 和 ANSA0 位置 1。

可通过配置 ABECON 寄存器，将该引脚设置为 TEST\_OUT 功能。它是内部模拟或数字信号多路开关的缓冲输出。该引脚上的模拟信号由 ADCON0 寄存器控制。该引脚上的数字信号由 ABECON 寄存器控制。

### 2.1.2 GPA1 引脚

GPA1 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPA 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN1 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 1 读取，必须将 TRISA1 和 ANSA1 位置 1。

当 MCP19114/5 配置为主器件或从器件时，该引脚配置为开关频率同步输入或输出（CLKPIN）。

### 2.1.3 GPA2 引脚

GPA2 为通用 ST 输入或 CMOS 输出引脚，其数据方向在 TRISGPA 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN2 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 2 读取，必须将 TRISA2 和 ANSA2 位置 1。

当 OPTION\_REG 寄存器中的 T0CS 位置 1 时，使能 T0CKI 功能。更多信息，请参见第 22.0 节“Timer0 模块”。

也可通过 INTE 位置 1 将 GPA2 配置为外部中断。更多信息，请参见第 14.2 节“GPA2/INT 中断”。

### 2.1.4 GPA3 引脚

GPA3 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPA 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN3 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 3 读取，必须将 TRISA3 和 ANSA3 位置 1。

### 2.1.5 GPA5 引脚

GPA5 是仅用作通用 TTL 输入的引脚。该引脚还提供了内部弱上拉和电平变化中断功能。

为实现编程目的，该引脚可连接到串行编程器的 MCLR 引脚。更多信息，请参见第 30.0 节“在线串行编程 (ICSP™)”。

当 CONFIG 寄存器中的 MCLRE 位置 1 时，该引脚为 MCLR。

### 2.1.6 GPA6 引脚

GPA6 为通用 CMOS 输出或 ST 输入引脚，其数据方向在 TRISGPA 中控制。

ICSPDAT 为串行编程数据 I/O 功能。该功能与 ICSPCLK 配合使用对器件进行串行编程。

GPA6 是 CCD 模块的一部分。更多信息，请参见第 26.0 节“双捕捉 / 比较 (CCD) 模块”。

### 2.1.7 GPA7 引脚

GPA7 为真正的漏极开路通用引脚，其数据方向在 TRISGPA 中控制。该引脚与器件  $V_{DD}$  间不存在内部连接。该引脚无弱上拉，但提供电平变化中断功能。

该引脚为主 ICSPCLK 输入。该功能与 ICSPDAT 配合使用对器件进行串行编程。

如果已配置 MCP19114/5 来实现 I<sup>2</sup>C 通信（第 28.2 节“I<sup>2</sup>C 模式概述”），则 GPA7 用作 I<sup>2</sup>C 时钟（SCL）。该引脚必须配置为输入才可进行正常工作。

# MCP19114/5

## 2.1.8 GPB0 引脚

GPB0 为真正的漏极开路通用引脚，其数据方向在 TRISGPB 中控制。该引脚与器件  $V_{DD}$  间不存在内部连接。该引脚无弱上拉，但提供电平变化中断功能。如果已配置 MCP19114/5 来实现 I<sup>2</sup>C 通信（第 28.2 节“I<sup>2</sup>C 模式概述”），则 GPB0 用作 I<sup>2</sup>C 时钟（SDA）。该引脚必须配置为输入才可进行正常工作。

## 2.1.9 GPB1 引脚

GPB1 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPB 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN4 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 4 读取，必须将 TRISB1 和 ANSB1 位置 1。

当 MCP19114/5 配置为主器件时，该引脚配置为  $V_{REF2}$  DAC 输出。

## 2.1.10 GPB4 引脚（仅 MCP19115）

GPB4 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPB 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN5 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 5 读取，必须将 TRISB4 和 ANSB4 位置 1。

ICSPDAT 为主串行编程数据 I/O 功能。该功能与 ICSPCLK 配合使用对器件进行串行编程。

## 2.1.11 GPB5 引脚（仅 MCP19115）

GPB5 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPB 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN6 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 6 读取，必须将 TRISB5 和 ANSB5 位置 1。

ICSPCLK 为主串行编程时钟功能。该功能与 ICSPDAT 配合使用对器件进行串行编程。

## 2.1.12 GPB6 引脚（仅 MCP19115）

GPB6 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPB 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

AN7 是 A/D 的输入。要将该引脚配置为通过 A/D 的通道 7 读取，必须将 TRISB6 和 ANSB6 位置 1。

## 2.1.13 GPB7 引脚（仅 MCP19115）

GPB7 为通用 TTL 输入或 CMOS 输出引脚，其数据方向在 TRISGPB 中控制。该引脚还提供了内部弱上拉和电平变化中断功能。

GPA7 是 CCD 模块的一部分。更多信息，请参见第 26.0 节“双捕捉/比较（CCD）模块”。

## 2.1.14 DESAT<sub>N</sub> 引脚

内部比较器反相输入。准谐振操作期间用于退饱和和检测。

## 2.1.15 DESAT<sub>p</sub>/I<sub>SOUT</sub> 引脚

在准谐振操作期间使用内部比较器进行退饱和和检测时，该引脚连接到比较器的同相输入。远程电流检测放大器的输出配置为采用 5 k $\Omega$  内部反馈电阻。当未使用内部比较器和未配置为使用 5 k $\Omega$  内部反馈电阻时，电流检测放大器连接到该引脚且为 I<sub>SOUT</sub>。

## 2.1.16 I<sub>SP</sub> 引脚

内部电流检测放大器的同相输入，通常用于差分远程检测次级电流。通过将 PE1 寄存器中的 <ISPUEN> 位置 1，可将该引脚内部上拉至  $V_{DD}$ 。

## 2.1.17 I<sub>SN</sub> 引脚

内部电流检测放大器的反相输入，通常用于差分远程检测次级电流。

## 2.1.18 I<sub>P</sub> 引脚

初级输入电流检测，用于电流模式控制和峰值电流限制。对于电压模式控制，该引脚可连接到人为斜坡。

## 2.1.19 A<sub>GND</sub> 引脚

A<sub>GND</sub> 为小信号地连接引脚。该引脚应与封装底部裸露的焊盘相连。

## 2.1.20 P<sub>GND</sub> 引脚

将所有大信号电平地回线与 P<sub>GND</sub> 相连。这些大信号电平地走线的回路面积要小且长度尽可能短，以防止开关噪声与敏感的走线相耦合。

## 2.1.21 SDRV 引脚

下桥臂次级 MOSFET 的栅极与 SDRV 相连。连接 SDRV 与栅极的 PCB 走线必须具有尽可能短的长度和适当宽度，以处理高峰值驱动电流和快速电压转换。

## 2.1.22 PDRV 引脚

下桥臂初级 MOSFET 的栅极与 PDRV 相连。连接 PDRV 与栅极的 PCB 走线必须具有尽可能短的长度和适当宽度，以处理高峰值驱动电流和快速电压转换。

## 2.1.23 $V_{DR}$ 引脚

下桥臂驱动器的电源连接到该引脚且具有 +13.5V 的绝对最大值。该引脚可通过一个 RC 滤波器与  $V_{DD}$  引脚相连。

## 2.1.24 $V_{DD}$ 引脚

内部 +5.0V 稳压器的输出与该引脚相连。建议在该引脚和器件的 GND 引脚间放置一个最小 1.0  $\mu\text{F}$ / 最大 10  $\mu\text{F}$  的旁路电容。旁路电容应物理上靠近器件放置。

## 2.1.25 $V_{IN}$ 引脚

器件输入电源连接引脚。建议在该引脚和器件的 GND 引脚间放置电容。

## 2.1.26 $V_S$ 引脚

与过压比较器的同相输入相连的模拟输入。通常用作输出电压过压保护。过压比较器的反相输入通过 OV REF DAC 控制。

## 2.1.27 $I_{FB}$ 引脚

误差放大器反相反馈连接。

## 2.1.28 $I_{COMP}$ 引脚

误差放大器输出信号。

## 2.1.29 裸露的焊盘 (EP)

建议将裸露的焊盘连接到  $A_{GND}$ 。

**注：** 初始上电时，器件可能对可配置为模拟输入的 GPIO 焊盘施加弱上拉影响。只要 5V ( $V_{DD}$ ) 上的外部电容充电且外部 5V 稳压器超过内部 4V 稳压器 ( $AV_{DD}$ )，弱上拉影响就会消失。要防止该寄生上电影响，建议将一个下拉电阻（最大值为 10 k $\Omega$ ）连接到具有模拟输入功能的 GPIO。这适用于 GPA0、GPA1、GPA2、GPA3、GPB1、GPB4（仅 MCP19115）、GPB5（仅 MCP19115）和 GPB6（仅 MCP19115）。

# MCP19114/5

---

注:

## 3.0 功能说明

### 3.1 线性稳压器

MCP19114/5 的工作输入电压范围为 4.5V 到 42V。有 2 个内部低压差 (Low Dropout, LDO) 稳压器。5V LDO 用于向内部处理器供电, 并提供 5V 输出供外部使用。另一个 LDO ( $AV_{DD}$ ) 是 4V 稳压器, 用于向其余的内部模拟电路供电。使用 LDO 向 MCP19114/5 供电时, 通过电阻分压器监视输入电压。MCP19114/5 还包含欠压保护。详细信息, 请参见第 13.3 节“欠压复位 (BOR)”。PIC 内核将在  $V_{DD}$  为 2.0V 时复位。

### 3.2 输出驱动电路

MCP19114/5 集成了两个下桥臂驱动器, 用于驱动外部下桥臂 N 沟道功率 MOSFET 以实现同步应用, 例如同步反激和同步 Cuk 转换器。这两个转换器类型均可通过将同步 FET 替换为二极管配置为非同步控制。反激转换器还可工作在准谐振模式下。另外, MCP19114/5 还可配置为升压或 SEPIC 开关模式电源 (Switch-Mode Power Supply, SMPS)。在升压模式下, 可采用非同步固定频率或非同步准谐振控制。该器件还可用作固定频率非同步模式下的 SEPIC SMPS。在典型 SMPS 应用中, 下桥臂驱动能够以高频率开关 MOSFET。栅极驱动 ( $V_{DR}$ ) 供电电压可从 5V 到 10V。10V 栅极驱动的驱动能力为拉/灌电流高达 1A, 而 5V 栅极驱动的驱动能力为拉/灌电流低至 0.5A。可编程延时用于设置栅极导通死区。这可防止重叠电流和直通电流降低转换效率。每个驱动器应当有自己的 EN 输入受单片机内核控制。

### 3.3 电流检测

输出电流由 MCP19114/5 差分检测。在低电流应用中, 这有助于将电流检测电阻的功耗降至最低, 从而保持较高的系统效率。差分电流检测还可将外部地偏移误差降至最低。内部差分放大器的增益通常为 10 V/V。

### 3.4 峰值电流模式

MCP19114/5 是一款峰值电流模式控制的器件, 电流检测元件与初级侧 MOSFET 串联。可实现可编程前沿消隐以消隐 MOSFET 导通时产生的电流尖峰。消隐时间通过 ICLEBCON 寄存器控制。

用户可通过编程进行初级输入电流失调调节, 从而限制初级输入电流峰值。该失调调节由 ICOACON 寄存器控制。

### 3.5 退磁检测

内部比较器模块用于检测准谐振应用的电源退磁。比较器输出用作信号来同步下一个开关周期的开始。该操作不同于传统的固定频率应用。可使能 DESAT 比较器输出并传递至 PWM 电路, 或者禁止该输出以实现固定频率应用。准谐振 (Quasi-Resonant, QR) 操作期间, 使能 DESAT 比较器输出并与一对单次定时器和一个触发器结合在一起维持 PWM 工作。Timer2 (TMR2) 必须初始化并设置为以低于最小 QR 工作频率的频率运行。当 DESATCON 寄存器中的 CDSWDE 位置 1 时, TMR2 充当看门狗定时器。

工作在同步 QR 模式下的反激式 SMPS 的事件顺序示例如下:

初级栅极驱动 (PDRV) 变为高电平。DESAT 比较器的输出为高电平。初级电流增加直至  $I_p$  达到误差放大器的电平并导致 PWM 比较器输出变为低电平。PDRV 变为低电平且次级栅极驱动 (SDRV) 在编程的死区之后变为高电平。这触发第一个单次定时器以发送一个 200 ns 脉冲, 该脉冲复位触发器和 TMR2 (WDM\_RESET)。实现该 200 ns 单次脉冲设计以屏蔽开关噪声导致的 DESAT 比较器输出处的任何假性转换。SDRV 保持高电平直至次级绕组耗尽能量, 此时输出电容开始通过该绕组和次级 MOSFET 汲回电流。DESAT 比较器检测到该情况, 其输出变为低电平。这设置了触发器并触发第二个单次定时器, 发送一个 33 ns 脉冲至控制逻辑, 从而导致在编程的死区之后 SDRV 变为低电平而 PDRV 变为高电平。然后重复该周期。如果由于任何原因未触发复位单次定时器, 则 WDM\_RESET 信号保持低电平且允许 TMR2 运行直至 PWM 信号开始新的周期。

退饱和比较器模块由 DESATCON 寄存器控制。

## 3.6 启动

为了控制启动期间的输出电流，MCP19114/5 能够单调递增系统电流，任凭用户决定。该功能通过控制参考电压 DAC ( $V_{REF}$ ) 实现。整个启动配置文件由用户通过软件控制。

## 3.7 驱动器控制电路

与 MCP1631 相似，MCP19114/5 的内部驱动器控制电路由一个误差放大器 (Error Amplifier, EA)、一个高速比较器和一个锁存器组成。

误差放大器生成高速 PWM 比较器使用的控制电压。有一个内部生成的参考电压  $V_{REF}$ 。该内部参考电压和实际反馈电压之间的差值或误差即是控制电压。某些应用将实现暂停时间，在此期间栅极驱动处于不活动状态。例如，在 LED 灯串变化时和电压调整之后，用户可禁止栅极驱动并将误差放大器输出暂设为低电平。EA 暂停期间，其输出钳位为低电平 ( $1 * BG$ )，这样在重新激活时其状态已知。在重新使能输出开关之前，有时可能需要在使能输出驱动器之前重新使能 EA。使能 EA 之前的这段时间允许 EA 转向预定目标并防止次级开关导通过长时间，无意中输出电容放电并将输出电压拉低。外部补偿用于稳定控制系统。

由于 MCP19114/5 是峰值电流模式控制的，比较器将基于初级侧中流动电流的初级峰值电流波形 ( $I_P$ ) 与误差放大器控制输出电压作比较。该误差放大器控制输出电压还减去了用户可编程斜率补偿。在固定频率应用中，生成斜率补偿信号，该信号大于电感电流波形的下降斜率的 1/2 且由 SLPCRCON 寄存器控制。失调调节能力还可用于设置初级侧开关的峰值电流限制以实现过流保护。斜率补偿斜升范围是确定的。当电流检测信号达到控制电压电平减去斜率补偿时，导通周期终止，而外部开关锁存，直至开始下一个时钟周期。

为了提高低电平时的电流稳定度，应用了设置为 BG (1.23V) 的基本电压 ( $V_{ZC}$ )。该虚地充当误差放大器 (A1)、斜率补偿、电流检测放大器 (A2) 和  $I_P$  失调调节的参考地。

S-R 锁存器 (置位复位触发器) 用于阻止 PWM 电路在下一个时钟周期开始前打开外部开关。

## 3.8 固定 PWM 频率

当 MCP19114/5 的开关频率不是由 DESAT 比较器输出控制时，则由单边沿 8 MHz 内部时钟产生。用户通过配置 PR2 寄存器设置 MCP19114/5 开关频率。允许的最大 PDRV 占空比可调节且由 PWMRL 寄存器控制。开关频率的可编程范围为 31.25 kHz 到 2 MHz。低于 2 MHz 的可用开关频率定义为  $F_{SW} = 8 \text{ MHz}/N$ ，其中 N 为整数 ( $4 \leq N \leq 256$ )。详细信息，请参见第 25.0 节“增强型 PWM 模块”。

## 3.9 $V_{REF}$

该参考电压用于生成连接到误差放大器的同相输入的电压。整个模拟控制回路被提高到等于带隙电压 (1.23V) 的虚地基本电压。

## 3.10 OV REF

该参考电压用于设置输出过压设定值。它与  $V_S$  输入引脚电压作比较，该输入引脚电压通常与基于电阻分压器的输出电压成比例。当使能 OV 保护时，可将其设置为一个值来保护系统电路，也可用于“纹波”调节转换器输出电压以实现调整目的。详细信息，请参见寄存器 6-4。

## 3.11 具有可编程延时的独立栅极驱动

集成了两个独立的下桥臂栅极驱动，可实现同步应用。已实现可编程延时以提高效率并防止直通电流。每个栅极驱动都具有独立的使能输入 (由 PE1 寄存器控制) 和可编程死区 (由 DEADCON 寄存器控制)。

## 3.12 温度管理

### 3.12.1 热关断

为了防止 MCP19114/5 处于过温条件，已实现 150°C 结温热关断。当该结温达到该限制时，器件禁止输出驱动器。在关断模式下，PDRV 和 SDRV 输出均禁止，且 PIR2 寄存器中的过温标志位（OTIF）置 1。当结温降低 20°C 到 130°C 时，MCP19114/5 可恢复正常输出驱动开关。

### 3.12.2 温度报告

MCP19114/5 具有另一个片上温度监视电路，可由 ADC 通过模拟测试多路开关读取温度。关于该内部温度监视电路的详细信息，请参见第 20.0 节“[内部温度指示器模块](#)”。

# MCP19114/5

## 4.0 电气特性

### 4.1 绝对最大值†

$V_{IN} - V_{GND}$ (工作)	-0.3V 至 +44V
$V_{IN}$ (瞬态 < 500 ms)	+48V
PDRV	(GND - 0.3V) 至 ( $V_{DR} + 0.3V$ )
SDRV	(GND - 0.3V) 至 ( $V_{DR} + 0.3V$ )
$V_{DD}$ 内部产生	+6.5V
$V_{DR}$ 外部产生	+13.5V
MCLR 相对于 GND 的电压	-0.3V 至 +13.5V
最大电压: 任意其他引脚	$+(V_{GND} - 0.3V)$ 至 ( $V_{DD} + 0.3V$ )
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 GPIO 的最大灌电流	90 mA
所有 GPIO 的最大拉电流	35 mA
储存温度	-65°C 至 +150°C
最高结温	+150°C
工作结温	-40°C 至 +125°C
所有引脚上的 ESD 保护 (HBM)	2.0 kV
所有引脚上的 ESD 保护 (MM)	200V

† 注: 如果器件工作条件超过上述“绝对最大值”, 则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。

### 4.2 电气特性

电气规范: 除非另外声明, 否则 $V_{IN} = 12V$ , $F_{SW} = 150\text{ kHz}$ , $T_A = +25^\circ\text{C}$ 。粗体字规范适用的 $T_A$ 范围为 $-40^\circ\text{C}$ 到 $+125^\circ\text{C}$ 。						
参数	符号	最小值	典型值	最大值	单位	条件
<b>输入</b>						
输入电压	$V_{IN}$	<b>4.5</b>	—	<b>42</b>	V	
输入静态电流	$I_Q$	—	5	<b>6.5</b>	mA	$V_{IN} = 12V$ , 不进行开关操作
		—	5	<b>6.75</b>		$V_{IN} = 20V$ , 不进行开关操作
关断电流	$I_{SHDN}$	—	30	<b>150</b>	$\mu\text{A}$	$V_{IN} = 12V$ (注 1)
<b>线性稳压器 <math>V_{DD}</math></b>						
内部电路偏置电压	$V_{DD}$	<b>4.75</b>	5.0	<b>5.5</b>	V	$V_{IN} = 6.0V$ 至 $42V$
最大外部 $V_{DD}$ 输出电流	$I_{DD\_OUT}$	<b>35</b>	—	—	mA	$V_{IN} = 6.0V$ 至 $42V$ (注 4)
线路稳定度	$\frac{\Delta V_{DD\_OUT}}{(V_{DD\_OUT} * \Delta V_{IN})}$	<b>-0.1</b>	0.002	<b>0.1</b>	%/V	$(V_{DD} + 1.0V) \leq V_{IN} \leq 20V$ (注 4)
负载稳定度	$\frac{\Delta V_{DD\_OUT}}{V_{DD\_OUT}}$	<b>-0.65</b>	0.1	<b>+0.65</b>	%	$I_{DD\_OUT} = 1\text{ mA}$ 至 $20\text{ mA}$ (注 4)
输出短路电流	$I_{DD\_SC}$	—	50	—	mA	$V_{IN} = (V_{DD} + 1.0V)$ (注 4)

- 注 1: 请参见第 15.0 节“掉电模式 (休眠)”。
- 2: 这些参数仅为特征值, 未经生产测试。
- 3: 通过设计保证, 未经生产测试。
- 4:  $V_{DD}$  是  $V_{DD}$  引脚上的电压。
- 5: 电压差定义为输出电压降至低于其正常值 2% 时输入与输出电压的差值, 输出电压正常值是在  $V_{IN}$  和  $V_{DD}$  之间的差值为 1V 时测得。
- 6:  $V_{DD}$  LDO 将总拉电流限制为最大 35 mA。每个引脚可单独获得的最大拉电流为 15 mA。

## 4.2 电气特性 (续)

电气规范: 除非另外声明, 否则  $V_{IN} = 12V$ ,  $F_{SW} = 150\text{ kHz}$ ,  $T_A = +25^\circ\text{C}$ 。粗体字规范适用的  $T_A$  范围为  $-40^\circ\text{C}$  到  $+125^\circ\text{C}$ 。

参数	符号	最小值	典型值	最大值	单位	条件
电压差	$V_{IN} - V_{DD}$	—	0.3	<b>0.5</b>	V	$I_{DD\_OUT} = 20\text{ mA}$ , (注 4 和 5)
电源抑制比	$PSRR_{LDO}$	—	60	—	dB	$f \leq 1000\text{ Hz}$ , $I_{DD\_OUT} = 25\text{ mA}$ $C_{IN} = 0\text{ }\mu\text{F}$ , $C_{DD} = 1\text{ }\mu\text{F}$
<b>线性稳压器 <math>AV_{DD}</math></b>						
内部模拟电源电压	$AV_{DD}$	—	4.0	—	V	
带隙电压	BG	—	1.23	—	V	以 1.0% 容差微调
带隙容差	$BG_{TOL}$	<b>-2.5</b>	—	<b>+2.5</b>	%	
<b>输入 UVLO 电压</b>						
UVLO 范围	$UVLO_{ON}$	<b>4.0</b>	—	<b>20</b>	V	$V_{IN}$ 下降
UVLO <sub>ON</sub> 跳变容差	$UVLO_{TOL}$	<b>-14</b>	—	<b>14</b>	%	$V_{IN}$ 下降 UVLO 跳变设置为 9V $V_{INUVLO} = 0x21h$
UVLO 迟滞	$UVLO_{HYS}$	<b>1</b>	4	<b>8</b>	%	迟滞基于 $UVLO_{ON}$ 设置 UVLO 跳变设置为 9V $V_{INUVLO} = 0x21h$
分辨率	nbits	—	6	—	位	对数步长
<b>UVLO 比较器</b>						
输入到输出延时	TD	—	5	—	$\mu\text{s}$	100 ns 上升时间到 $V_{IN}$ 上出现 1V 过驱动 $V_{IN} > UVLO$ 至中断标志置 1
<b>输入 OVLO 电压</b>						
OVLO 范围	$OVLO_{ON}$	<b>8.8</b>	—	<b>44</b>	V	$V_{IN}$ 上升
OVLO <sub>ON</sub> 跳变容差	$OVLO_{TOL}$	<b>-14</b>	—	<b>14</b>	%	$V_{IN}$ 上升 OVLO 跳变设置为 18V $V_{INOVLO} = 0x1Fh$
OVLO 迟滞	$OVLO_{HYS}$	<b>1</b>	5	<b>8</b>	%	迟滞基于 $OVLO_{ON}$ 设置 OVLO 跳变设置为 18V $V_{INOVLO} = 0x1Fh$
分辨率	nbits	—	6	—	位	对数步长
<b>OVLO 比较器</b>						
输入到输出延时	TD	—	5	—	$\mu\text{s}$	100 ns 上升时间到 $V_{IN}$ 上出现 1V 过驱动 $V_{IN} > OVLO$ 至中断标志置 1
<b>输出 OV DAC</b>						

注 1: 请参见第 15.0 节“掉电模式 (休眠)”。

2: 这些参数仅为特征值, 未经生产测试。

3: 通过设计保证, 未经生产测试。

4:  $V_{DD}$  是  $V_{DD}$  引脚上的电压。

5: 电压差定义为输出电压降至低于其正常值 2% 时输入与输出电压的差值, 输出电压正常值是在  $V_{IN}$  和  $V_{DD}$  之间的差值为 1V 时测得。

6:  $V_{DD}$  LDO 将总拉电流限制为最大 35 mA。每个引脚可单独获得的最大拉电流为 15 mA。

# MCP19114/5

## 4.2 电气特性 (续)

电气规范: 除非另外声明, 否则 $V_{IN} = 12V$ , $F_{SW} = 150\text{ kHz}$ , $T_A = +25^\circ\text{C}$ 。粗体字规范适用的 $T_A$ 范围为 $-40^\circ\text{C}$ 到 $+125^\circ\text{C}$ 。						
参数	符号	最小值	典型值	最大值	单位	条件
分辨率	nbits	—	8	—	位	线性 DAC
满量程范围	FSR	0	—	$2 * BG$	V	
容差	$OVREF_{TOL}$	<b>-10</b>	—	<b>+10</b>	%	满量程, 编码 = 0xFF
<b>输出 OV 比较器</b>						
OV 迟滞	$OV_{HYS}$	—	50	—	mV	
输入偏置电流	$I_{BIAS}$	—	$\pm 1$	—	$\mu\text{A}$	
共模输入电压范围	$V_{CMR}$	0	—	3.0	V	注 3
输入到输出延时	TD	—	200	—	ns	注 3 100 ns 上升时间到 $V_S$ 上出现 1V 过驱动 $V_S > OV$ 至中断标志置 1
<b>参考电压 DAC (<math>V_{REF}</math>)</b>						
分辨率	nbits	—	8	—	V/V	线性 DAC
满量程范围	FSR	BG	—	$2 * BG$	V	基本电压设置为 BG
<b>参考电压 DAC (<math>V_{REF2}</math>)</b>						
分辨率	nbits	—	8	—	位	线性 DAC
满量程范围	FSR	0	—	BG	V	
灌电流	$I_{SINK}$	<b>-3</b>	—	—	mA	$V_{REF2} = 0V$ , $R_L = 300\Omega$ 至 BG
拉电流	$I_{SOURCE}$	<b>3</b>	—	—	mA	$V_{REF2} = BG$ , $R_L = 300\Omega$ 至 GND
容差	$VREF2_{TOL}$	<b>-10</b>	—	<b>+10</b>	%	满量程, 编码 = 0xFF
<b>电流检测放大器 (A2)</b>						
放大器 PSRR	PSRR	—	65	—	dB	$V_{CM} = 2 * BG$
闭环电压增益	$A2_{VCL}$	—	10	—	V/V	$R_L = 5\text{ k}\Omega$ 至 2.048V, $100\text{ mV} < A2 <$ $AV_{DD} - 100\text{ mV}$ , $V_{CM} = BG$
低电平输出	$V_{OL}$	—	300	—	mV	$R_L = 5\text{ k}\Omega$ 至 2.048V
增益带宽积	GBWP	—	10	—	MHz	$AV_{DD} = 4V$
输入阻抗	$R_{IN}$	—	10	—	k $\Omega$	
灌电流	$I_{SINK}$	<b>-3</b>	—	—	mA	$I_{SP} = I_{SN} = GND$ $R_L = 300\Omega$ 至 $2 * BG$
拉电流	$I_{SOURCE}$	<b>3</b>	—	—	mA	$I_{SP} = I_{SN} = GND$ $R_L = 300\Omega$ 至 GND

- 注 1: 请参见第 15.0 节“掉电模式 (休眠)”。
- 2: 这些参数仅为特征值, 未经生产测试。
- 3: 通过设计保证, 未经生产测试。
- 4:  $V_{DD}$  是  $V_{DD}$  引脚上的电压。
- 5: 电压差定义为输出电压降至低于其正常值 2% 时输入与输出电压的差值, 输出电压正常值是在  $V_{IN}$  和  $V_{DD}$  之间的差值为 1V 时测得。
- 6:  $V_{DD}$  LDO 将总拉电流限制为最大 35 mA。每个引脚可单独获得的最大拉电流为 15 mA。

## 4.2 电气特性 (续)

电气规范: 除非另外声明, 否则 $V_{IN} = 12V$ , $F_{SW} = 150\text{ kHz}$ , $T_A = +25^\circ\text{C}$ 。粗体字规范适用的 $T_A$ 范围为 $-40^\circ\text{C}$ 到 $+125^\circ\text{C}$ 。						
参数	符号	最小值	典型值	最大值	单位	条件
共模范围	$V_{CMR}$	$GND - 0.3$	—	$V_{BG} + 0.3$	V	注 3
共模抑制比	CMRR	—	70	—	dB	
内部反馈电阻	$R_{FB\_INT}$	—	5	—	k $\Omega$	
内部反馈电阻 Tol	$R_{FB\_INT\_TOL}$	—	2	—	%	微调
<b>基本电压</b>						
基本电压电平	VZC	—	BG	—	V	
<b>误差放大器 (EA)</b>						
输入失调电压	$V_{OS}$	—	2	—	mV	
共模抑制比	CMRR	—	65	—	dB	$V_{CM} = 0V$ 至 BG
开环电压增益	$A_{VOL}$	—	70	—	dB	注 3
低电平钳位值	$V_{OL}$	<b>BG - 0.35</b>	BG - 0.22	<b>BG - 0.1</b>	V	$R_L = 5\text{ k}\Omega$ 至 2.048V
增益带宽积	GBWP	—	3.5	—	MHz	
误差放大器灌电流	$I_{SINK}$	<b>-3</b>	—	—	mA	$V_{REF} = BG$ , $I_{FB} = I_{COMP}$ $R_L = 150\Omega$ 至 $1.5 * BG$
误差放大器拉电流	$I_{SOURCE}$	<b>3</b>	—	—	mA	$V_{REF} = 2 * BG$ , $I_{FB} = I_{COMP}$ $R_L = 150\Omega$ 至 $1.5 * BG$
误差放大器最大输出高电平钳位	$V_{EA\_MAX}$	—	2 x BG	—	V	EA 输出钳位至 2 x BG 电压
<b>峰值电流检测输入</b>						
最大初级电流检测信号电压	$V_{IP\_MAX}$	—	BG	1.5	V	注 3
<b>PWM 比较器</b>						
输入到输出延时	TD	—	11	<b>20</b>	ns	注 3
<b>峰值电流前沿消隐</b>						
分辨率	LEB	—	2	—	位	
消隐时间可调节范围	$LEB_{RANGE}$	0	—	256	ns	4 步长可编程范围 (0、50、100 和 200 ns) (注 3)

- 注 1: 请参见第 15.0 节“掉电模式 (休眠)”。
- 2: 这些参数仅为特征值, 未经生产测试。
- 3: 通过设计保证, 未经生产测试。
- 4:  $V_{DD}$  是  $V_{DD}$  引脚上的电压。
- 5: 电压差定义为输出电压降至低于其正常值 2% 时输入与输出电压的差值, 输出电压正常值是在  $V_{IN}$  和  $V_{DD}$  之间的差值为 1V 时测得。
- 6:  $V_{DD}$  LDO 将总拉电流限制为最大 35 mA。每个引脚可单独获得的最大拉电流为 15 mA。

# MCP19114/5

## 4.2 电气特性 (续)

**电气规范:** 除非另外声明, 否则  $V_{IN} = 12V$ ,  $F_{SW} = 150\text{ kHz}$ ,  $T_A = +25^\circ\text{C}$ 。粗体字规范适用的  $T_A$  范围为  $-40^\circ\text{C}$  到  $+125^\circ\text{C}$ 。

参数	符号	最小值	典型值	最大值	单位	条件
<b>失调调节 (I<sub>p</sub> 检测)</b>						
分辨率	OS <sub>ADJ</sub>	—	4	—	位	
失调调节范围	OS <sub>ADJ_RANGE</sub>	0	—	750	mV	
失调调节步长	OS <sub>ADJ_STEP</sub>	—	50	—	mV	线性步长
<b>可调节斜率补偿</b>						
分辨率	SC <sub>RES</sub>	—	6	—	位	对数步长
斜率	m	4.1	—	432.5	mV/μs	
斜率步长	SC <sub>STEP</sub>	—	8	—	%	对数步长
斜升设定值容差	m <sub>TOL</sub>	—	±1	±30	%	
<b>退饱和和检测比较器</b>						
输入失调电压	V <sub>OS</sub>	—	±1	—	mV	微调, 5 位可调节
输入偏置电流	I <sub>BIAS</sub>	—	±1	—	μA	独立于内部电路
共模输入电压范围	V <sub>CMR</sub>	GND - 0.3V	—	2.7	V	<a href="#">注 3</a>
输入到输出延时	TD	—	20	—	ns	
<b>V<sub>DR</sub> UVLO</b>						
V <sub>DR</sub> UVLO (2.7V V <sub>DR</sub> 下降)	V <sub>DR_UVLO_2.7_F</sub>	2.45	—	2.9	V	
V <sub>DR</sub> UVLO (2.7V V <sub>DR</sub> 上升)	V <sub>DR_UVLO_2.7_R</sub>	2.68	—	3.23	V	
V <sub>DR</sub> UVLO (2.7V) 滞后	V <sub>DR_UVLO_2.7_HYS</sub>	190	—	415	mV	
V <sub>DR</sub> UVLO (5.4V V <sub>DR</sub> 下降)	V <sub>DR_UVLO_5.4_F</sub>	4.7	—	5.96	V	
V <sub>DR</sub> UVLO (5.4V V <sub>DR</sub> 上升)	V <sub>DR_UVLO_5.4_R</sub>	5.15	—	6.56	V	
V <sub>DR</sub> UVLO (5.4V) 滞后	V <sub>DR_UVLO_5.4_HYS</sub>	380	—	830	mV	
<b>输出驱动器 (PDRV 和 SDRV)</b>						
PDRV/SDRV 栅极驱动 拉电流阻抗	R <sub>DR-SRC</sub>	—	—	<b>13.5</b>	Ω	V <sub>DR</sub> = 4.5V ( <a href="#">注 3</a> )
PDRV/SDRV 栅极驱动 灌电流阻抗	R <sub>DR-SINK</sub>	—	—	<b>12</b>	Ω	V <sub>DR</sub> = 4.5V ( <a href="#">注 3</a> )

- 注**
- 1: 请参见第 15.0 节“掉电模式 (休眠)”。
  - 2: 这些参数仅为特征值, 未经生产测试。
  - 3: 通过设计保证, 未经生产测试。
  - 4: V<sub>DD</sub> 是 V<sub>DD</sub> 引脚上的电压。
  - 5: 电压差定义为输出电压降低至低于其正常值 2% 时输入与输出电压的差值, 输出电压正常值是在 V<sub>IN</sub> 和 V<sub>DD</sub> 之间的差值为 1V 时测得。
  - 6: V<sub>DD</sub> LDO 将总拉电流限制为最大 35 mA。每个引脚可单独获得的最大拉电流为 15 mA。

## 4.2 电气特性 (续)

电气规范: 除非另外声明, 否则 $V_{IN} = 12V$ , $F_{SW} = 150\text{ kHz}$ , $T_A = +25^\circ\text{C}$ 。粗体字规范适用的 $T_A$ 范围为 $-40^\circ\text{C}$ 到 $+125^\circ\text{C}$ 。						
参数	符号	最小值	典型值	最大值	单位	条件
PDRV/SDRV 栅极驱动拉电流	$I_{DR-SRC}$	—	0.5	—	A	$V_{DR} = 5V$ $V_{DR} = 10V$ (注 3)
		—	1.0	—		
PDRV/SDRV 栅极驱动灌电流	$I_{DR-SINK}$	—	0.5	—	A	$V_{DR} = 5V$ $V_{DR} = 10V$ (注 3)
		—	1.0	—		
<b>死区调节</b>						
分辨率	$DT_{RES}$	—	4	—	位	
死区可调节范围	$DT_{RANGE}$	16	—	256	ns	
死区步长	$DT_{STEP}$	—	16	—	ns	线性步长
死区容差	$DT_{TOL}$	—	$\pm 8$	—	ns	
<b>振荡器 /PWM</b>						
内部振荡器频率	$F_{OSC}$	<b>7.60</b>	8.00	<b>8.40</b>	MHz	
开关频率	$F_{SW}$	—	$F_{OSC}/N$	—	MHz	
开关频率范围选择	N	4	—	255	—	$F_{MAX} = 2\text{ MHz}$
<b>A/D 转换器 (ADC) 特性</b>						
分辨率	$N_R$	—	—	10	位	
积分误差	$E_{IL}$	—	—	$\pm 1$	LSb	$V_{REF\_ADC} = AV_{DD}$
微分误差	$E_{DL}$	—	—	$\pm 1$	LSb	10 位无编码丢失, $V_{REF\_ADC} = AV_{DD}$ (注 2)
失调误差	$E_{OFF}$	—	+3.0	<b>+7</b>	LSb	$V_{REF\_ADC} = AV_{DD}$
增益误差	$E_{GN}$	—	$\pm 2$	<b><math>\pm 6</math></b>	LSb	$V_{REF\_ADC} = AV_{DD}$
参考电压	$V_{REF\_ADC}$	—	$AV_{DD}$	—	V	
满量程范围	$FSR_{A/D}$	GND	—	$AV_{DD}$	—	
<b>GPIO 引脚</b>						
GPIO 最大灌电流	$I_{SINK\_GPIO}$	—	—	90	mA	注 6
GPIO 最大拉电流	$I_{SOURCE\_GPIO}$	—	—	35	mA	注 6
GPIO 弱上拉电流	$I_{PULL-UP\_GPIO}$	50	250	400	$\mu\text{A}$	
GPIO 输入低电压	$V_{GPIO\_IL}$	<b>GND</b>	—	<b>0.8</b>	V	带 TTL 缓冲器的 I/O 端口, $V_{DD} = 5V$
		<b>GND</b>	—	<b><math>0.2V_{DD}</math></b>	V	带施密特触发缓冲器的 I/O 端口, $V_{DD} = 5V$
		<b>GND</b>	—	<b><math>0.2V_{DD}</math></b>	V	MCLR

注 1: 请参见第 15.0 节“掉电模式 (休眠)”。

2: 这些参数仅为特征值, 未经生产测试。

3: 通过设计保证, 未经生产测试。

4:  $V_{DD}$  是  $V_{DD}$  引脚上的电压。

5: 电压差定义为输出电压降至低于其正常值 2% 时输入与输出电压的差值, 输出电压正常值是在  $V_{IN}$  和  $V_{DD}$  之间的差值为 1V 时测得。

6:  $V_{DD}$  LDO 将总拉电流限制为最大 35 mA。每个引脚可单独获得的最大拉电流为 15 mA。

# MCP19114/5

## 4.2 电气特性 (续)

电气规范: 除非另外声明, 否则  $V_{IN} = 12V$ ,  $F_{SW} = 150\text{ kHz}$ ,  $T_A = +25^\circ\text{C}$ 。粗体字规范适用的  $T_A$  范围为  $-40^\circ\text{C}$  到  $+125^\circ\text{C}$ 。

参数	符号	最小值	典型值	最大值	单位	条件
GPIO 输入高电压	$V_{GPIO\_IH}$	<b>2.0</b>	—	$V_{DD}$	V	带 TTL 缓冲器的 I/O 端口, $V_{DD} = 5V$
		<b><math>0.8V_{DD}</math></b>	—	$V_{DD}$	V	带施密特触发缓冲器的 I/O 端口, $V_{DD} = 5V$
		<b><math>0.8V_{DD}</math></b>	—	$V_{DD}$	V	$\overline{MCLR}$
GPIO 输出低电压	$V_{GPIO\_OL}$	—	—	<b><math>0.12V_{DD}</math></b>	V	$I_{OL} = 7\text{ mA}$ , $V_{DD} = 5V$
GPIO 输出高电压	$V_{GPIO\_OH}$	$V_{DD} - 0.7$	—	—	V	$I_{OH} = 2.5\text{ mA}$ , $V_{DD} = 5V$
GPIO 输入泄漏电流	$GPIO\_I_{IL}$	—	$\pm 0.1$	$\pm 1$	$\mu\text{A}$	负电流定义为引脚的拉电流。
<b>POR</b>						
上电复位电压	$V_{POR}$	—	1.75	—	V	
上电复位电压迟滞	$V_{POR\_HYS}$	—	100	—	mV	
<b>热关断</b>						
热关断	$T_{SHD}$	—	150	—	$^\circ\text{C}$	
热关断迟滞	$T_{SHD\_HYS}$	—	20	—	$^\circ\text{C}$	

- 注
- 1: 请参见第 15.0 节“掉电模式 (休眠)”。
  - 2: 这些参数仅为特征值, 未经生产测试。
  - 3: 通过设计保证, 未经生产测试。
  - 4:  $V_{DD}$  是  $V_{DD}$  引脚上的电压。
  - 5: 电压差定义为输出电压降至低于其正常值 2% 时输入与输出电压的差值, 输出电压正常值是在  $V_{IN}$  和  $V_{DD}$  之间的差值为 1V 时测得。
  - 6:  $V_{DD}$  LDO 将总拉电流限制为最大 35 mA。每个引脚可单独获得的最大拉电流为 15 mA。

## 4.3 热规范

参数	符号	最小值	典型值	最大值	单位
<b>温度范围</b>					
规定温度范围	$T_A$	-40	—	+125	$^\circ\text{C}$
工作结温范围	$T_J$	-40	—	+125	$^\circ\text{C}$
最高结温	$T_J$	—	—	+150	$^\circ\text{C}$
储存温度范围	$T_A$	-65	—	+150	$^\circ\text{C}$
<b>封装热阻</b>					
热阻, 24L-QFN 4x4	$\theta_{JA}$	—	42	—	$^\circ\text{C/W}$
热阻, 28L-QFN 5x5	$\theta_{JA}$	—	35.3	—	$^\circ\text{C/W}$

## 5.0 数字电气特性

### 5.1 时序参数符号

时序参数符号采用以下格式之一进行创建：

- |             |                       |                          |
|-------------|-----------------------|--------------------------|
| 1. TppS2ppS | 3. T <sub>CC:ST</sub> | (仅 I <sup>2</sup> C™ 规范) |
| 2. TppS     | 4. Ts                 | (仅 I <sup>2</sup> C 规范)  |

<b>T</b>		
F	频率	T
		时间

小写字母 (pp) 及其含义：

<b>pp</b>		
cc	CCP1	osc
ck	CLKOUT	rd
cs	$\overline{CS}$	rw
di	SDI	sc
do	SDO	ss
dt	数据输入	t0
io	I/O 端口	wr
mc	$\overline{MCLR}$	
		OSC1
		$\overline{RD}$
		$\overline{RD}$ 或 $\overline{WR}$
		SCK
		$\overline{SS}$
		$\overline{TOCKI}$
		$\overline{WR}$

大写字母及其含义：

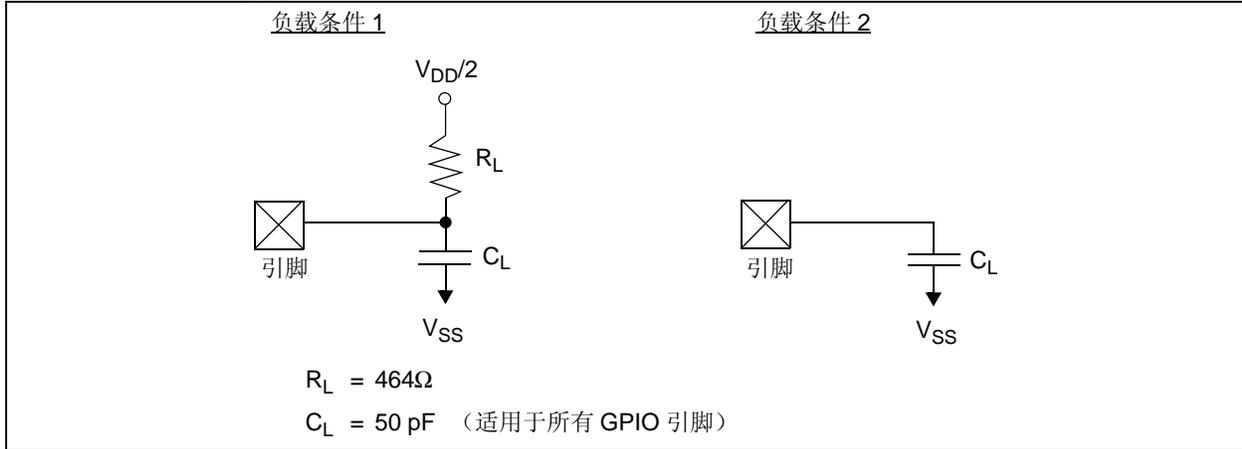
<b>S</b>		
F	下降	P
H	高	R
I	无效 (高阻态)	V
L	低	Z
仅 I <sup>2</sup> C™		
AA	输出访问	High
BUF	总线空闲	Low
		周期
		上升
		有效
		高阻态
		高
		低

T<sub>CC:ST</sub> (仅 I<sup>2</sup>C 规范)

<b>CC</b>		
HD	保持	SU
<b>ST</b>		
DAT	数据输入保持	STO
STA	启动条件	
		建立
		停止条件

# MCP19114/5

图 5-1: 负载条件



## 5.2 交流特性: MCP19114 (工业级和扩展级)

图 5-2: 外部时钟时序

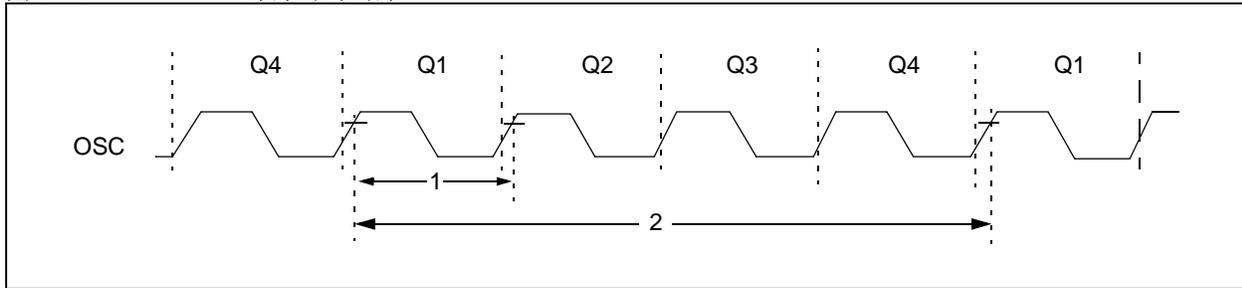


表 5-1: 外部时钟时序要求

参数编号	符号	特性	最小值	典型值 <sup>†</sup>	最大值	单位	条件
	$F_{OSC}$	振荡器频率 (1)	—	8	—	MHz	
1	$T_{OSC}$	振荡器周期 (1)	—	250	—	ns	
2	$T_{CY}$	指令周期 (1)	—	$T_{CY}$	$\infty$	ns	$T_{CY} = 4 * T_{OSC}$

\* 这些参数仅为特征值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为  $V_{IN} = 12V$  ( $V_{DD} = 5V$ ) 和  $25^\circ C$  条件下的值。这些参数仅作为设计参考，未经测试。

注 1: 指令周期 ( $T_{CY}$ ) 等于输入振荡器时基周期的 4 倍。所有规范值均基于器件在标准工作条件下执行代码时对应特定振荡器类型的特征数据。

图 5-3: I/O 时序

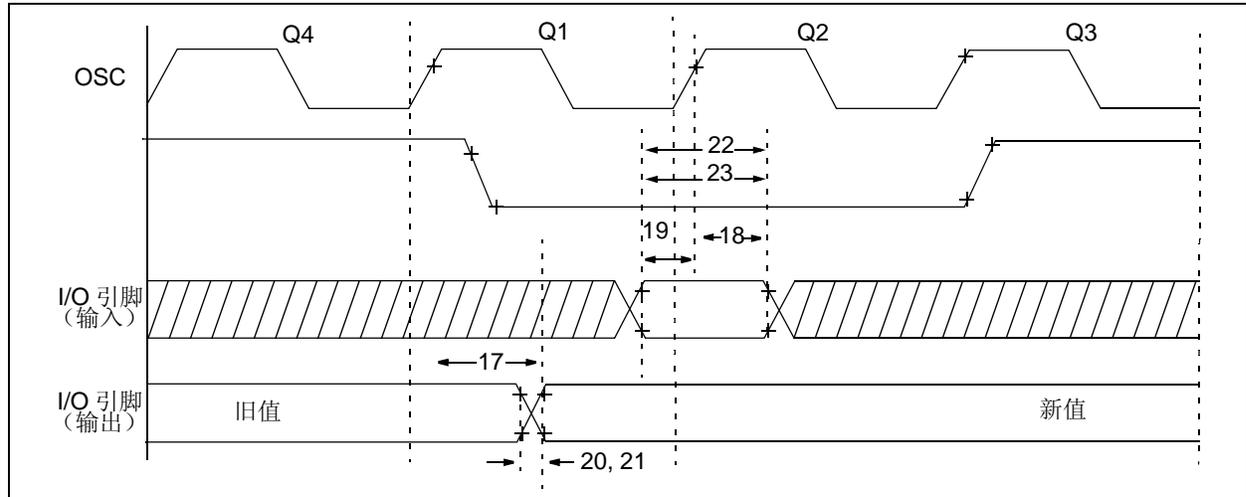


表 5-2: I/O 时序要求

参数编号	符号	特性	最小值	典型值 <sup>†</sup>	最大值	单位	条件
17	TosH2ioV	OSC1 $\uparrow$ (Q1 周期) 至端口输出有效的时间	—	50	70*	ns	
18	TosH2ioI	OSC1 $\uparrow$ (Q2 周期) 至端口输入无效的时间 (I/O 输入保持时间)	50	—	—	ns	
19	TioV2osH	端口输入有效至 OSC1 $\uparrow$ 的时间 (I/O 输入建立时间)	20	—	—	ns	
20	TioR	端口输出上升时间	—	32	40	ns	
21	TioF	端口输出下降时间	—	15	30	ns	
22*	Tinp	INT 引脚高电平时间或低电平时间	25	—	—	ns	
23*	T <sub>RABP</sub>	GPIO 电平变化中断新输入电平时间	T <sub>CY</sub>	—	—	ns	

\* 这些参数仅为特征值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为  $V_{IN} = 12V$  ( $V_{DD} = 5V$ ) 和  $25^{\circ}C$  条件下的值。

# MCP19114/5

图 5-4: 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

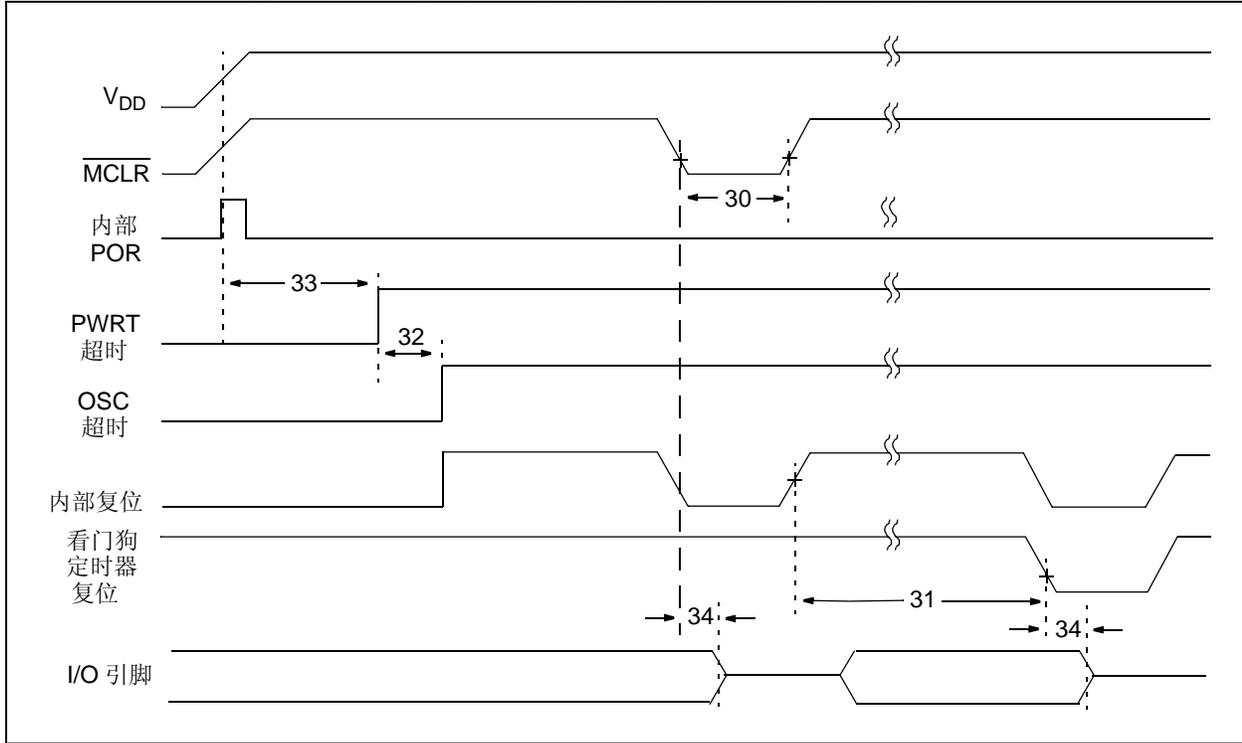


图 5-5: 欠压复位时序和特性

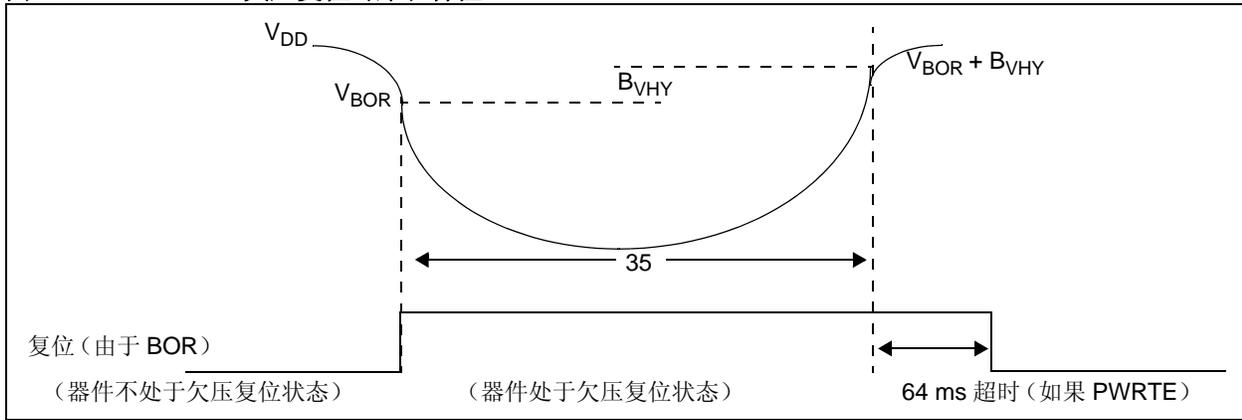


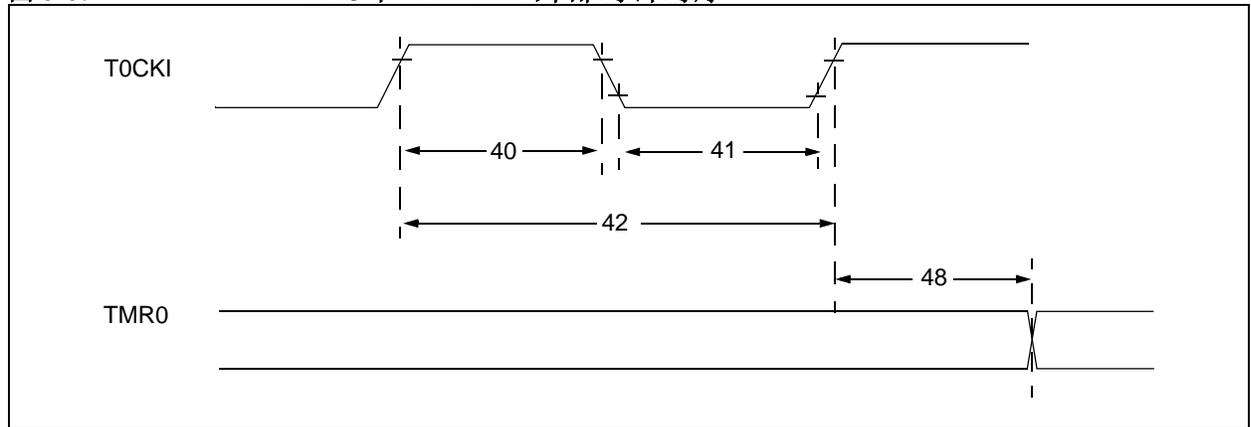
表 5-3: 复位、看门狗定时器、振荡器起振定时器和上电延时定时器要求

参数编号	符号	特性	最小值	典型值 <sup>†</sup>	最大值	单位	条件
30	$T_{MCL}$	MCLR 脉宽 (低电平)	2	—	—	$\mu\text{s}$	$V_{DD} = 5\text{ V}$ , -40°C 至 +85°C
31	$T_{WDT}$	看门狗定时器超时周期 (无预分频器)	7	18	33	ms	$V_{DD} = 5\text{ V}$ , -40°C 至 +85°C
32	$T_{OST}$	振荡器起振定时器周期	—	$1024T_{OSC}$	—	—	$T_{OSC} = \text{OSC1 周期}$
33*	$T_{PWRT}$	上电延时定时器周期 ( $4 \times T_{WDT}$ )	28	72	132	ms	$V_{DD} = 5\text{ V}$ , -40°C 至 +85°C
34	$T_{IOZ}$	自 $\overline{\text{MCLR}}$ 低电平或看门狗定时器复位起 I/O 处于高阻态的时间	—	—	2.0	$\mu\text{s}$	
	$V_{BOR}$	欠压复位电压	2.0	2.13	2.3	V	
	$V_{BHY}$	欠压复位滞后电压	—	100	—	mV	
35	$T_{BCR}$	欠压复位脉冲宽度	100*	—	—	$\mu\text{s}$	$V_{DD} \leq V_{BOR}$ (D005)
48	$TCKEZ_{TMR}$	从时钟边沿到定时器递增的延时	$2T_{OSC}$	—	$7T_{OSC}$		

\* 这些参数仅为特征值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为  $V_{IN} = 12\text{V}$  ( $V_{DD} = AV_{DD} = 5\text{V}$ ) 和 +25°C 条件下的值。这些参数仅作为设计参考，未经测试。

图 5-6: TIMER0 和 TIMER1 外部时钟时序



# MCP19114/5

表 5-4: TIMERO 外部时钟要求

参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
40*	Tt0H	T0CKI 高电平脉冲宽度	无预分频器	$0.5T_{CY} + 20$	—	—	ns
			有预分频器	10	—	—	ns
41*	Tt0L	T0CKI 低电平脉冲宽度	无预分频器	$0.5T_{CY} + 20$	—	—	ns
			有预分频器	10	—	—	ns
42*	Tt0P	T0CKI 周期	取以下两者中的较大值: 20 或 $\frac{T_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (2, 4, ..., 256)

\* 这些参数仅为特征值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为  $V_{IN} = 12V$  和  $25^{\circ}C$  条件下的值。这些参数仅作为设计参考，未经测试。

图 5-7: PWM 时序

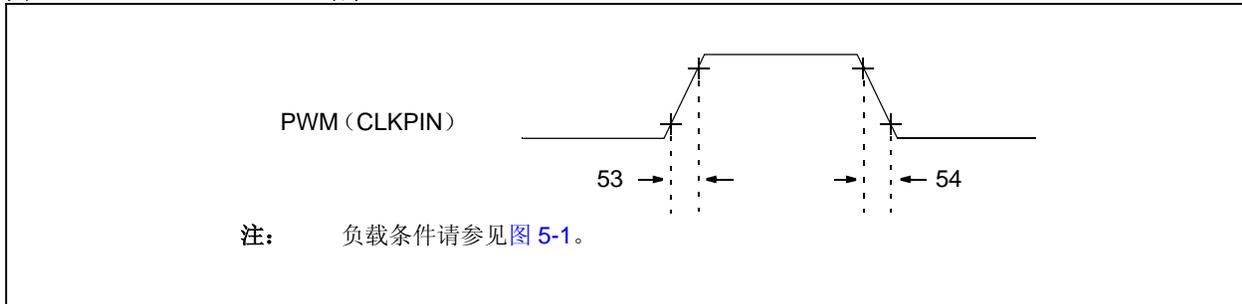


表 5-5: PWM 要求

参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
53*	TccR	PWM (CLKPIN) 输出上升时间	—	10	25	ns	
54*	TccF	PWM (CLKPIN) 输出下降时间	—	10	25	ns	

\* 这些参数仅为特征值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为  $V_{IN} = 12V$  ( $AV_{DD} = 4V$ ) 和  $25^{\circ}C$  条件下的值。这些参数仅作为设计参考，未经测试。

表 5-6: MCP19114/5 A/D 转换器 (ADC) 特性

电气特性: 除非另外声明, 否则工作温度 = $40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$							
参数编号	符号	特性	最小值	典型值 <sup>†</sup>	最大值	单位	条件
AD01	$N_R$	分辨率	—	—	10 位	位	
AD02	$E_{IL}$	积分误差	—	—	$\pm 1$	LSb	$AV_{DD} = 4.0\text{V}$
AD03	$E_{DL}$	微分误差	—	—	$\pm 1$	LSb	10 位无编码丢失 $AV_{DD} = 4.0\text{V}$
AD04	$E_{OFF}$	失调误差	—	+3.0	+7	LSb	$AV_{DD} = 4.0\text{V}$
AD07	$E_{GN}$	增益误差	—	$\pm 2$	$\pm 6$	LSb	$AV_{DD} = 4.0\text{V}$
AD07	$V_{AIN}$	满量程范围	$A_{GND}$	—	$AV_{DD}$	V	
AD08	$Z_{AIN}$	模拟电压源的推荐阻抗	—	—	10	k $\Omega$	

\* 这些参数仅为特征值, 未经测试。

† 除非另外声明, 否则“典型值”栏中的数据均为  $V_{IN} = 12\text{V}$  ( $AV_{DD} = 4\text{V}$ ) 和  $25^{\circ}\text{C}$  条件下的值。这些参数仅作为设计参考, 未经测试。

- 注 1: 总绝对误差包括积分误差、微分误差、失调误差和增益误差。  
 2: A/D 转换结果不会因输入电压的增加而减小, 并且不会丢失编码。  
 3: 当 ADC 关闭时, 它除了消耗泄漏电流外, 不消耗任何其他电流。掉电电流规范包括 ADC 模块消耗的任何泄漏电流。

表 5-7: MCP19114/5 A/D 转换要求

电气规范: 除非另外声明, 否则工作温度 = $40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$							
参数编号	符号	特性	最小值	典型值 <sup>†</sup>	最大值	单位	条件
AD130*	$T_{AD}$	A/D 时钟周期	1.6	—	9.0	$\mu\text{s}$	基于 $T_{OSC}$
		A/D 内部 RC 振荡器周期	1.6	4.0	6.0	$\mu\text{s}$	$ADCS<1:0> = 11$ (ADRC 模式)
AD131	$T_{CNV}$	转换时间 (不包括采集时间) <sup>(1)</sup>	—	11	—	$T_{AD}$	将 $\overline{GO/DONE}$ 位置 1 直至 A/D 结果寄存器中有新数据
AD132*	$T_{ACQ}$	采集时间	—	11.5	—	$\mu\text{s}$	
AD133*	$T_{AMP}$	放大器稳定时间	—	—	5	$\mu\text{s}$	
AD134	$T_{GO}$	Q4 至 A/D 时钟开始	—	$T_{OSC}/2$	—	—	

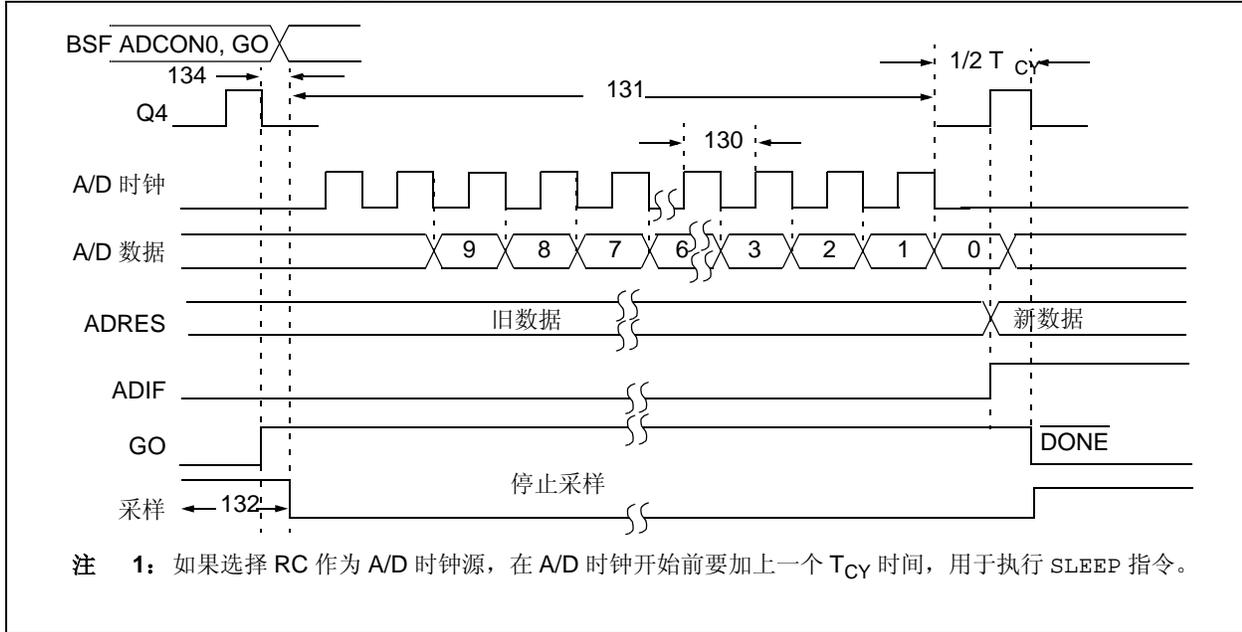
\* 这些参数仅为特征值, 未经测试。

† 除非另外声明, 否则“典型值”栏中的数据均为  $V_{IN} = 12\text{V}$  ( $V_{DD} = AV_{DD} = 5\text{V}$ ) 和  $+25^{\circ}\text{C}$  条件下的值。这些参数仅作为设计参考, 未经测试。

- 注 1: ADRESH 和 ADRESL 寄存器可在下一个  $T_{CY}$  周期被读取。

# MCP19114/5

图 5-8: A/D 转换时序



## 6.0 配置 MCP19114/5

MCP19114/5 是具有数字外设的模拟控制器。这意味着器件配置通过寄存器设置进行处理，而不是通过添加外部元件来处理。有几个内部可配置比较器模块，用作模拟电路与数字处理的接口，与当今许多 PIC 处理器（例如，PIC16F1824/1828）中的标准比较器模块非常相似。以下几节将详细介绍如何针对所有可配置参数设置模拟控制寄存器。

### 6.1 输入欠压和过压锁定 (UVLO 和 OVLO)

VINCON 是 VINUVLO 和 VINOVL0 寄存器的比较器控制寄存器。它包含这两个保护电路的使能位、极性边沿检测位和状态输出位。PIR2 寄存器中的中断标志位 <UVLOIF> 和 <OVLOIF> 与 VINCON 寄存器中的使能位 <UVLOEN> 和 <OVLOEN> 相互独立。VINCON 寄存器中的 <UVLOOUT> 欠压锁定状态输出位指示是否发生 UVLO 事件。VINCON 寄存器中的 <OVLOOUT> 过压锁定状态输出位指示是否发生 OVLO 事件。

VINUVLO 寄存器包含设置输入欠压锁定的数字值。UVLO 范围为 4V 至 20V。当  $V_{IN}$  值低于该范围和高于处理器活动电压 ( $V_{DD} = 2V$ ) 时，UVLO 比较器和 UVLOOUT 状态位将指示欠压条件。如果使用 UVLO 来确定上电  $V_{IN}$ ，建议查询 UVLOOUT 位了解状态。当 MCP19114/5 的  $V_{IN}$  引脚上的输入电压低于此编程电平并且 VINCON 寄存器中的 <UVLOEN> 位置 1 时，PDRV 和 SDRV 栅极驱动器均禁止。当 MCP19114/5 的  $V_{IN}$  电压高于此编程电平时，该位自动清零。

VINOVL0 寄存器包含设置输入过压锁定的数字值。OVLO 范围为 8.8V 至 44V。当 MCP19114/5 的  $V_{IN}$  引脚上的输入电压高于此编程电平时，VINCON 寄存器中的 <OVLOEN> 位将置 1，PDRV 和 SDRV 栅极驱动器均禁止。当 MCP19114/5 的  $V_{IN}$  电压低于此编程电平时，该位自动清零。请参见图 27-1。

**注：** 当发生中断条件时，无论相应的中断允许位或全局中断允许位（INTCON 寄存器的 GIE）的状态如何，UVLOIF 和 OVLOIF 中断标志位都将置 1。

**寄存器 6-1: VINCON: UVLO 和 OVLO 比较器控制寄存器**

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0
UVLOEN	UVLOOUT	UVLOINTP	UVLOINTN	OVLOEN	OVLOOUT	OVLOINTP	OVLOINTN
bit 7							bit 0

**图注：**

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

- bit 7      **UVLOEN:** UVLO 比较器模块逻辑使能位  
1 = 使能 UVLO 比较器模块逻辑  
0 = 禁止 UVLO 比较器模块逻辑
- bit 6      **UVLOOUT:** 欠压锁定状态输出  
1 = 发生了 UVLO 事件  
0 = 未发生 UVLO 事件
- bit 5      **UVLOINTP:** UVLO 比较器正边沿中断允许位  
1 = UVLOIF 中断标志位在出现 UVLO 的正边沿时置 1  
0 = UVLOIF 中断标志位在出现 UVLO 的正边沿时不置 1
- bit 4      **UVLOINTN:** UVLO 比较器负边沿中断允许位  
1 = UVLOIF 中断标志位在出现 UVLO 的负边沿时置 1  
0 = UVLOIF 中断标志位在出现 UVLO 的负边沿时不置 1

# MCP19114/5

## 寄存器 6-1: VINCON: UVLO 和 OVLO 比较器控制寄存器 (续)

- bit 3 **OVLOEN:** OVLO 比较器模块逻辑使能位  
1 = 使能 OVLO 比较器模块逻辑  
0 = 禁止 OVLO 比较器模块逻辑
- bit 2 **OVLOOUT:** 过压锁定状态输出位  
1 = 发生了 OVLO 事件  
0 = 未发生 OVLO 事件
- bit 1 **OVLOINTP:** OVLO 比较器正边沿中断允许位  
1 = OVLOIF 中断标志位在出现 OVLO 的正边沿时置 1  
0 = OVLOIF 中断标志位在出现 OVLO 的正边沿时不置 1
- bit 0 **OVLOINTN:** OVLO 比较器负边沿中断允许位  
1 = OVLOIF 中断标志位在出现 OVLO 的负边沿时置 1  
0 = OVLOIF 中断标志位在出现 OVLO 的负边沿时不置 1

## 寄存器 6-2: VINUVLO: 输入欠压锁定寄存器

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	UVLO5	UVLO4	UVLO3	UVLO2	UVLO1	UVLO0
bit 7							bit 0

### 图注:

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

bit 7-6 未实现: 读为 0

bit 5-0 **UVLO<5:0>:** 欠压锁定配置位  
 $UVLO(V) = 3.5472 * (1.0285^N)$ , 其中 N = 写入 VINUVLO 寄存器的十进制值 0 至 63

## 寄存器 6-3: VINOVL0: 输入过压锁定寄存器

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	OVLO5	OVLO4	OVLO3	OVLO2	OVLO1	OVLO0
bit 7							bit 0

### 图注:

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

bit 7-6 未实现: 读为 0

bit 5-0 **OVLO<5:0>:** 过压锁定配置位  
 $OVLO(V) = 7.4847 * (1.0286^N)$ , 其中 N = 写入 VINOVL0 寄存器的十进制值 0 至 63

## 6.2 输出过压保护

MCP19114/5 具有输出过压保护功能。该功能也使用类似于标准 PIC 比较器模块的比较器模块。该模块用于防止供电系统在断开负载连接时受到损坏。OVREFCON 寄存器包含设置比较器反相输入的模拟 DAC 电压的数字值。通过将连接到比较器的同相输入 ( $V_S$ ) 的分压电源输出电压与 OVREF 参考电压作比较, 用户将了解何时发生过压事件并可自动采取措施。

OVCON 寄存器包含中断标志极性和 OV 使能位以及输出状态位, 和用于输入电压 UVLO 和 OVLO 的 VINCON 作用一样。当 OVCON 寄存器中的 <OVEN> 置 1 且发生过压时, 控制逻辑自动将次级栅极驱动输出 (SDRV) 设置为高电平并将初级栅极驱动输出 (PDRV) 设置为低电平。

**注:** 当发生中断条件时, 无论相应的中断允许位或全局中断允许位 (INTCON 寄存器的 GIE) 的状态如何, OVIF 中断标志位都将置 1。

**寄存器 6-4: OVCON: 输出过压比较器控制寄存器**

U-0	U-0	U-0	U-0	R/W-0	R-0	R/W-0	R/W-0
—	—	—	—	OVEN	OVOUT	OVINTP	OVINTN
bit 7							bit 0

**图注:**

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

- bit 7-4      **未实现:** 读为 0
- bit 3      **OVEN:** OV 比较器输出使能位  
1 = 使能 OV 比较器输出  
0 = 禁止 OV 比较器输出
- bit 2      **OVOUT:** 输出过压状态输出位  
1 = 发生了输出过压  
0 = 未发生输出过压
- bit 1      **OVINTP:** OV 比较器正边沿中断允许位  
1 = OVIF 中断标志位在出现 OV 的正边沿时置 1  
0 = OVIF 中断标志位在出现 OV 的正边沿时不置 1
- bit 0      **OVINTN:** OV 比较器负边沿中断允许位  
1 = OVIF 中断标志位在出现 OV 的负边沿时置 1  
0 = OVIF 中断标志位在出现 OV 的负边沿时不置 1

**寄存器 6-5: OVREFCON: 输出过压检测电平寄存器**

R/W-x							
OOV7	OOV6	OOV5	OOV4	OOV3	OOV2	OOV1	OOV0
bit 7							bit 0

**图注:**

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

- bit 7-0      **OOV<7:0>:** 输出过压检测电平配置位  
 $V_{OV\_REF(V)} = 2 * V_{BG} * (OOV(dec)/255)$

# MCP19114/5

A/D 转换器校准字 8 可用于提高 OVREF 精度。通过将模拟多路开关缓冲器失调 (BUOFFSET) 添加到所需的 OVREF 电压 (OVREFTARGET) 并将该结果乘以 ADC 增益 (GADC) 得到 ADC 测量目标值 (例 6-1 中的目标值)。调节 OVREF 直至 ADC 读数等于或超过目标值。

OVREF 校准固件示例如下：

## 例 6-1: OVREF 校正程序示例

```
//Assumes that calibration words ADCCAL and BUFF are read from
//program memory into variables ADCC and BUOFFSET, respectively.
extern volatile unsigned int ADRES @ 0x01C;
#define OVREFTARGET (unsigned int) 0x0800 // OVREF Target = 2.0 V
unsigned long tmp = (unsigned long)ADCC*(OVREFTARGET+BUOFFSET); // ADC Reference + Buffer Offset
unsigned int target = (unsigned int)(tmp >> 15) - 3; // Subtract ADC typical offset error 3
unsigned int adc;
OVREFCON = 0x00; // Clear OVREFCON
ADCON0 = 0x09; // Enable and set channel to OVREF

do { // Adjust OVREFCON
    OVREFCON++;
    NOP(); NOP();

    adc = 0;
    for (unsigned char i = 4; i > 0; i--) {
        ADCON0bits.GO_nDONE = 1;
        while(ADCON0bits.GO_nDONE);
        adc += ADRES;
    }
    adc >>= 2;
} while ((adc < target) && (OVREFCON != 0xFF));
```

**注 1:** 在本示例中，OVREFTARGET 的 LSb 权值设置为  $1/(2^{10})V$ 。用户可根据他们的精度需求选择自己的分辨率。数字值 2.0V 根据如下公式得出： $TRUNC(2.0 \times 2^{10}) = 2048$ （十六进制为 0x0800）。

## 6.3 准谐振操作的退饱和检测

设计 MCP19114/5 时，内置了一个针对准谐振拓扑定制的退饱和检测比较器模块。这对于 LED 型应用特别有用。通过使用 MCP19114/5，可同时实现同步和异步准谐振拓扑。除了包含一些额外的可编程参数，DESAT 比较器模块具有与 UVLO/OVLO 和 OV 比较器相同的特性。

DESATCON 寄存器保存该模块的设置控制位。公共控制位有中断极性边沿触发标志位 <CDSINTP><CDSINTN>、比较器输出极性控制位 <CDSPOL>、输出使能位 <CDSOE> 和输出状态位 <CDSOUT>。与其他比较器模块一样，CDSIF 与 CDSOE 使能位相互独立。在与 DESAT 比较器同相输入相连的前端，有两个通道多路开关，可连接到 DESAT<sub>P</sub> 引脚或内部产生的固定带隙电压。此外，DESAT 比较器的输入失调电压出厂微调为在 ±1 mV 典型值内。这些出厂微调值存储在位于地址 2081h 的 CALWD2 寄存器中。固件必须将这些值读取到 DSTCAL 寄存器（196h）。如果需要更多失调，用户可根据他们的实现调节写入 DSTCAL 的值。

寄存器 6-6: DESATCON: 退饱和比较器控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CDSMUX	CDSWDE	保留	CDSPOL	CDSOE	CDSOUT	CDSINTP	CDSINTN
bit 7							bit 0

**图注:**

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

bit 7	<b>CDSMUX:</b> DESAT 比较器模块多路开关通道选择位 1 = 选择 BG 0 = 选择 DESAT <sub>P</sub> (默认设置)
bit 6	<b>CDSWDE:</b> DESAT 比较器看门狗使能位 1 = 看门狗信号使能 PWM 复位 0 = 看门狗信号不允许 PWM 复位
bit 5	保留
bit 4	<b>CDSPOL:</b> DESAT 比较器极性选择位 1 = DESAT 比较器输出反相 0 = DESAT 比较器输出不反相
bit 3	<b>CDSOE:</b> DESAT 比较器输出使能位 1 = 使能 DESAT 比较器输出 PWM 0 = 禁止 DESAT 比较器输出 PWM
bit 2	<b>CDSOUT:</b> DESAT 比较器输出状态位 <u>如果 CDSPOL = 1 (极性反相)</u> 1 = CDSVP < CDSVN (检测到 DESAT) 0 = CDSVP > CDSVN (未检测到 DESAT) <u>如果 CDSPOL = 0 (极性不反相)</u> 1 = CDSVP > CDSVN (未检测到 DESAT) 0 = CDSVP < CDSVN (检测到 DESAT)
bit 1	<b>CDSINTP:</b> CDSIF 比较器正边沿中断允许位 1 = CDSIF 中断标志位在出现正边沿时置 1 0 = CDSIF 中断标志位在出现正边沿时不置 1
bit 0	<b>CDSINTN:</b> CDSIF 比较器负边沿中断允许位 1 = CDSIF 中断标志位在出现负边沿时置 1 0 = CDSIF 中断标志位在出现负边沿时不置 1

# MCP19114/5

## 6.4 初级输入电流失调调节

初级输入电流失调调节能够将失调添加到初级输入电流信号，从而设置峰值初级电流限制。使用 ICOACON 寄存器中的四个位控制该失调调节。

**寄存器 6-7: ICOACON: 输入电流失调调节控制寄存器**

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	ICOAC3	ICOAC2	ICOAC1	ICOAC0
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n = POR 时的值

1 = 置 1

0 = 清零

bit 7-4      **未实现:** 读为 0

bit 3-0      **ICOAC<3:0>:** 输入电流失调调节配置位

- 0000 = 0 mV
- 0001 = 50 mV
- 0010 = 100 mV
- 0011 = 150 mV
- 0100 = 200 mV
- 0101 = 250 mV
- 0110 = 300 mV
- 0111 = 350 mV
- 1000 = 400 mV
- 1001 = 450 mV
- 1010 = 500 mV
- 1011 = 550 mV
- 1100 = 600 mV
- 1101 = 650 mV
- 1110 = 700 mV
- 1111 = 750 mV

## 6.5 前沿消隐

可调节前沿消隐 (Leading Edge Blanking, LEB) 用于消隐初级侧开关导通时引起的初级电流尖峰。实现可调节 LEB 允许系统忽略导通噪声最适合应用, 而无需来自 RC 滤波的初级电流检测失真。有 4 个设置可用于 LEB, 包括零。这些设置通过 ICLEBCON 寄存器中的两个位进行控制。

**寄存器 6-8: ICLEBCON: 输入电流前沿消隐控制寄存器**

U-0	U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x
—	—	—	—	—	—	ICLEBC1	ICLEBC0
bit 7						bit 0	

**图注:**

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

- bit 7-2      **未实现:** 读为 0
- bit 1-0      **ICLEBC<1:0>:** 输入电流前沿消隐配置位
  - 00 = 0 ns
  - 01 = 50 ns
  - 10 = 100 ns
  - 11 = 200 ns

# MCP19114/5

## 6.6 斜率补偿

将负电压斜率添加到误差放大器的输出中。这样可避免在以下情况下发生次谐波不稳定：

1. 工作占空比大于 50%
2. 占空比发生较大变化

添加到误差放大器输出的负斜率大小由斜率补偿压摆率控制位控制。

清零 SLPBYP 寄存器的 SLPBY 位可启用斜率补偿。

### 寄存器 6-9: SLPBYP: 斜率补偿斜升控制寄存器

U-0	R/W-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	SLPBY	SLPS5	SLPS4	SLPS3	SLPS2	SLPS1	SLPS0
bit 7							bit 0

#### 图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n = POR 时的值

1 = 置 1

0 = 清零

bit 7 未实现：读为 0

bit 6 SLPBY: 斜率补偿旁路控制位

1 = 旁路斜率补偿

0 = 不旁路斜率补偿

bit 5-0 SLP<5:0>: 斜率补偿压摆率控制位

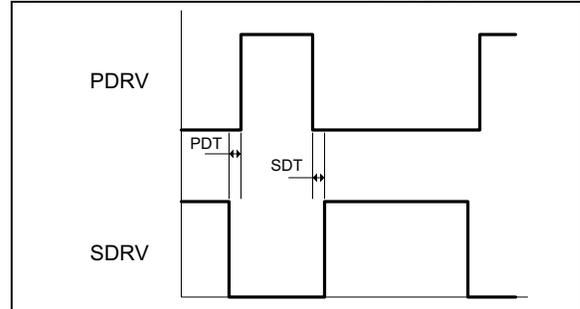
$SLPS (mV/\mu s) = 4.1505 mV/\mu s * e^{0.0739 * (dec)}$

## 6.7 MOSFET 驱动器可编程死区

可单独配置 PDRV 和 SDRV 下桥臂驱动信号的导通死区，这样便可使用不同的 MOSFET 和电路板布线来构造最优系统（见图 6-1）。

将 PE1 寄存器的 PDRVBY 和 SDRVBY 位清零可分别使能 PDRV 和 SDRV 下桥臂死区。所添加的死区量由 DEADCON 寄存器控制。

图 6-1: MOSFET 驱动器死区



寄存器 6-10: DEADCON: 驱动器死区控制寄存器

| R/W-x   |
|---------|---------|---------|---------|---------|---------|---------|---------|
| PDRVDT3 | PDRVDT2 | PDRVDT1 | PDRVDT0 | SDRVDT3 | SDRVDT2 | SDRVDT1 | SDRVDT0 |
| bit 7   |         |         |         |         |         |         | bit 0   |

**图注:**

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n = POR 时的值

1 = 置 1

0 = 清零

bit 7-4 **PDRVDT<3:0>**: PDRV 死区配置位 ( $t_{TD\_1}$ )

- 0000 = 16 ns 延时
- 0001 = 32 ns 延时
- 0010 = 48 ns 延时
- 0011 = 64 ns 延时
- 0100 = 80 ns 延时
- 0101 = 96 ns 延时
- 0110 = 112 ns 延时
- 0111 = 128 ns 延时
- 1000 = 144 ns 延时
- 1001 = 160 ns 延时
- 1010 = 176 ns 延时
- 1011 = 192 ns 延时
- 1100 = 208 ns 延时
- 1101 = 224 ns 延时
- 1110 = 240 ns 延时
- 1111 = 256 ns 延时

bit 3-0 **SDRVDT<3:0>**: SDRV 死区配置位 ( $t_{TD\_2}$ )

- 0000 = 16 ns 延时
- 0001 = 32 ns 延时
- 0010 = 48 ns 延时
- 0011 = 64 ns 延时
- 0100 = 80 ns 延时
- 0101 = 96 ns 延时
- 0110 = 112 ns 延时
- 0111 = 128 ns 延时
- 1000 = 144 ns 延时
- 1001 = 160 ns 延时
- 1010 = 176 ns 延时
- 1011 = 192 ns 延时
- 1100 = 208 ns 延时
- 1101 = 224 ns 延时
- 1110 = 240 ns 延时
- 1111 = 256 ns 延时

# MCP19114/5

## 6.8 输出稳压参考配置

VREFCON 寄存器控制误差放大器参考电压。该参考电压用于设置电流或电压调节设定值。VREFCON 保存 8 位线性 DAC 使用的数字值，该值用于设置与误差放大器的同相节点的基本电压（VZC）相加的模拟等效值。VZC 等于带隙电压（1.23V）。电流检测放大器 A2 的输出也在基本电压的基础上进行了提升，可有效消除其对输入的影响。基本电压通过模拟控制回路实现来提高低电平时的精度。VREF DAC 可调节 255 步，每步 4.8 mV。

寄存器 6-11: VREFCON: 电流 / 电压调节设定值控制寄存器

R/W-0							
VREF7	VREF6	VREF5	VREF4	VREF3	VREF2	VREF1	VREF0
bit 7							bit 0

**图注:**

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

bit 7-0 **VREF<7:0>**: 电压控制电流调节设定值位  
 $VREF(V) = V_{BG} * (VREF(dec)/255)$

要确保使用电流检测放大器（A2）时可获得最佳调节精度，必须考虑初始增益误差。8 位出厂存储的校准值 A2CAL<7:0> 已存储在 208Bh 处的 CALWD10 中。使用该值通过调节 VREF 命令来补偿 A2 增益误差。

要获得最终命令值，使用 VREF 表达式将 CALWD10 值与原始 VREF 十进制值相乘，产生一个 16 位字。循环右移该 16 位结果将产生最终的补偿命令，为最低有效字节。最高有效字节未使用。

固件示例如下：

例 6-2: A2 增益校正示例

```
//Assumes that calibration word A2CAL has been read into variable A2COMP

unsigned int VREF1_TEMP = VREFCON*A2COMP;           // A2 Gain compensate for VREFCON
VREF1_TEMP >>= 7;
VREF1_TEMP &= 0x00FF;
VREFCON = VREF1_TEMP;
```

## 6.9 V<sub>REF2</sub> 参考电压

VREF2CON 寄存器控制一个可用于外部的辅助参考 DAC。例如，它可进行片外发送并用于设置 MCP1631 脉宽调制器的电流调节设定值。必须通过设置 MODECON 寄存器中的位 MSC<0:1> = 01 将 MCP19114/5 配置为主器件模式以将 V<sub>REF2</sub> 连接到

GPB1。在独立模式下，V<sub>REF2</sub> 不可访问。VREFCON2 保存用于设置 VREF2 DAC 的数字值。由于该参考 DAC 旨在片外运行，因此没有与其相关的基本失调，并且其参考 GND。它是一个 8 位线性 DAC 且具有 0V 至 1.23V (BG) 的范围，相当于 255 个步长，每个步长为 4.8 mV。

### 寄存器 6-12: VREF2CON: V<sub>REF2</sub> 电压设定值寄存器

| R/W-0  |
|--------|--------|--------|--------|--------|--------|--------|--------|
| VREF27 | VREF26 | VREF25 | VREF24 | VREF23 | VREF22 | VREF21 | VREF20 |
| bit 7  |        |        |        |        |        |        | bit 0  |

#### 图注:

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

bit 7-0 VREF2<7:0>: 电压控制电流调节设定值位

$$V_{REF2(V)} = V_{BG} * (VREF2(dec)/255)$$

A/D 转换器校准字 8 可用于提高 V<sub>REF2</sub> 精度。将所需的 VREF2 电压 (VREF2TARGET) 乘以 ADC 增益 (ADCC) 得到 ADC 测量目标值 (例 6-3 中的目标值)。调节 V<sub>REF2</sub> 直至 ADC 读数等于或超过目标值。

V<sub>REF2</sub> 校正固件示例如下:

### 例 6-3: VREF2 校正示例

```
// Assumes that the calibration word ADCCAL has been read into variable ADCC

extern volatile unsigned int ADRES @ 0x01C;
#define VREF2TARGET (unsigned int) 0x02CC // VREF2 Target = 0.7 V(1)
unsigned long tmp = (unsigned long)ADCC*VREF2TARGET; // ADC Reference
unsigned int target = (unsigned int)(tmp >> 15) - 3; // Subtract ADC typical offset error 3
unsigned int adc;

VREF2CON = 0x00; // Clear VREF2CON
ADCON = 0x71; // Enable ADC and set channel to GPB1/VREF2

do { // Adjust VREF2CON
    VREF2CON++;
    NOP(); NOP();
    adc = 0;
    for (unsigned char i = 4; i > 0; i--) {
        ADCON0bits.GO_DONE = 1;
        while(ADCON0bits.GO_DONE);
        adc += ADRES;
    }
    adc >>= 2;
} while ((adc < target) && (VREF2CON != 0xFF));
```

**注 1:** 在本示例中，VREF2TARGET 的 LSB 权值设置为 1/(2<sup>10</sup>)V。用户可根据精度需求选择分辨率。数字值 0.7V 根据如下公式得出：TRUNC(0.7 × 2<sup>10</sup>) = 716 (十六进制为 0x02CC)。

# MCP19114/5

## 6.10 模拟外设控制

MCP19114/5 具有各种模拟外设。这些外设可配置为允许定制操作。更多信息，请参见[寄存器 6-13](#)。

### 6.10.1 MOSFET 栅极驱动器使能

MCP19114/5 可单独使能和 / 或禁止初级驱动 (PDRV) 和次级驱动 (SDRV) 的 MOSFET 栅极驱动器输出。将 PE1 寄存器的 <PDRVEN> 位置 1 可启用初级驱动。将 PE1 寄存器的 <SDRVEN> 位置 1 可启用次级驱动。详细信息，请参见[寄存器 6-13](#)。

### 6.10.2 MOSFET 驱动器死区

如[第 6.7 节 “MOSFET 驱动器可编程死区”](#) 中所述，MOSFET 驱动死区可以调节。可使用 DEADCON 寄存器为每个驱动器单独设置死区，范围为从 16 ns 到 256 ns，增量为 16 ns。可通过将 PE1 寄存器中的旁路位 <PDRVBY> 和 <SDRVBY> 置 1 单独禁止每个驱动器的死区。

### 寄存器 6-13: PE1: 模拟外设使能 1 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PDRVEN	SDRVEN	PDRVBY	SDRVBY	—	ISPUEN	PWMSTR_PEN	PWMSTR_SEN
bit 7						bit 0	

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

- bit 7      **PDRVEN:** PDRV 栅极驱动使能位  
1 = 使能  
0 = 禁止
- bit 6      **SDRVEN:** SDRV 栅极驱动使能位  
1 = 使能  
0 = 禁止
- bit 5      **PDRVBY:** PDRV 死区旁路位  
1 = 旁路 PDRV 死区  
0 = 不旁路 PDRV 死区
- bit 4      **SDRVBY:** SDRV 死区旁路位  
1 = 旁路 SDRV 死区  
0 = 不旁路 SDRV 死区
- bit 3      **未实现:** 读为 0
- bit 2      **ISPUEN:** I<sub>SP</sub> 弱上拉使能位  
1 = 使能 I<sub>SP</sub> 弱上拉  
0 = 禁止 I<sub>SP</sub> 弱上拉

### 6.10.3 次级电流正检测上拉

通过设置 PE1 寄存器的 <ISPUEN> 位可配置 I<sub>SP</sub> 引脚上的高阻态上拉。置 1 时，I<sub>SP</sub> 引脚在内部上拉至 V<sub>DD</sub>。详细信息，请参见[寄存器 6-13](#)。

### 6.10.4 PWM 控制

MCP19114/5 具有额外的控制电路来允许开环改变输出。PWMSTR\_PEN 位使能具有固定频率和占空比的仅初级 PWM 信号来提高输出电压。PWMSTR\_SEN 位使能具有固定频率和占空比的仅次级 PWM 信号来降低输出电压。降低输出电压时，输出过压保护必须处于活动状态，连同 PWMSTR\_SEN 有效以使 PWM 为 SDRV 产生脉冲。通过 TMR2 寄存器 PR2 和 TMR1L 控制频率和占空比。PWMSTPR\_PEN 和 PWMSTR\_SEN 不应同时有效，因此 PWMSTPR\_PEN 为主导位。对于开环改变期间的准谐振操作，应通过 DEADCON 寄存器中的 <CDSOE> 位禁止 DESAT 比较器输出。

## 寄存器 6-13: PE1: 模拟外设使能 1 控制寄存器 (续)

- bit 1      **PWMSTR\_PEN:** PDRV PWM 控制位  
 1 = 使能对 PDRV 的开环 PWM 控制  
 0 = 禁止对 PDRV 的开环 PWM 控制
- bit 0      **PWMSTR\_SEN:** SDRV PWM 控制位  
 1 = 使能对 SDRV 的开环 PWM 控制  
 0 = 禁止对 SDRV 的开环 PWM 控制

### 6.11 模拟模块使能控制

可使能或禁止各个模拟电路模块，如 ABECON 寄存器所示。ABECON 寄存器还包含用于控制模拟和数字测试信号的位。这些信号可配置到 GPA0。将 <DIGOEN> 位置 1 使数字测试信号连接到 GPA0。<DSEL2:0> 选择数字通道。将 <ANA0EN> 位置 1 使模拟测试信号连接到 GPA0。如果 <ANA0EN> 和 <DIGOEN> 均置 1，DIGOEN 位优先。当 ANA0EN 未置 1 时，模拟测试信号连接到内部 ADC。模拟测试通道选择通过 ADCON0 寄存器控制。

#### 6.11.1 MOSFET 驱动器欠压锁定选择

MOSFET 栅极驱动器具有内部欠压保护，由 ABECON 寄存器中的 <DRUVSEL> 位控制。由于栅极驱动电源由外部通过 V<sub>DR</sub> 引脚供给，驱动器能够驱动逻辑电平 FET 或更高的 10V（最大值 13.5V）FET。<DRUVSEL> 默认设置为清零，因此选择 2.7V 的栅极驱动 UVLO。将 <DRUVSEL> 置 1 选择更高的 5.4V 栅极驱动 UVLO。请参见第 4.2 节“电气特性”了解更多电气规范。

#### 6.11.2 误差放大器禁止

可禁止误差放大器，使其输出停在已知状态。<EADIS> 位默认为零，正常工作期间使能误差放大器。如果用户想要禁止误差放大器，将 EADIS 位置 1 以将误差放大器输出停置在略低于低钳位电压处。正常工作时，误差放大器输出介于 2 \* BG（上钳位值）和 1 \* BG - 150 mV（下钳位值）之间。模拟反馈电路使用失调基本电压（1 \* BG）来提高高低电平时的精度。

## 寄存器 6-14: ABECON: 模拟模块使能控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
DIGOEN	DSEL2	DSEL1	DSEL0	DRUVSEL	—	EADIS	ANA0EN
bit 7						bit 0	

### 图注:

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

- bit 7      **DIGOEN:** DIG 测试多路开关到 GPA0 连接控制  
 1 = DIG 测试多路开关输出不连接到外部引脚 GPA0  
 0 = DIG 测试多路开关输出不连接到外部引脚 GPA0

# MCP19114/5

---

## 寄存器 6-14: ABECON: 模拟模块使能控制寄存器 (续)

bit 6-4	<b>DSEL&lt;2:0&gt;</b> 000 = QRS (DESAT 比较器的输出) 001 = PWM_L (单稳态之后的 PWM 输出) 010 = PWM (单片机的振荡器输出) 011 = TMR2EQ (TMR2 等于 PR2 时) 100 = OV (过压比较器输出) 101 = SWFRQ (开关频率输出) 110 = SDRV_ON_ONESHOT (200 ns 单次触发信号以复位 WDM 逻辑) 111 = 未实现
bit 3	<b>DRUVSEL:</b> 选择栅极驱动欠压锁定电平 1 = 栅极驱动 UVLO 设置为 5.4V 0 = 栅极驱动 UVLO 设置为 2.7V
bit 2	<b>未实现:</b> 读为 0
bit 1	<b>EADIS:</b> 误差放大器禁止位 1 = 禁止误差放大器 (输出停置在低电平, 钳位至 $1 * BG$ ) 0 = 使能误差放大器 (正常工作)
bit 0	<b>ANAOEN:</b> 模拟多路开关输出控制位 1 = 模拟多路开关输出连接到外部引脚 GPA0 0 = 模拟多路开关输出不连接到外部引脚 GPA0

## 6.12 模式和 RFB 多路开关控制

MODECON 寄存器控制主器件 / 从器件配置以及准谐振模式下电流检测放大器的内部电阻反馈多路开关。

在主器件 / 从器件模式下，它允许 MCP19115 主器件的  $V_{REF2}$  信号缓存并连接到 GPIO 引脚。该输出信号可连接到  $V_{REF}$  输入端的从器件 PWM 驱动器 (MCP1631) 以通过从器件 PWM 控制器调节电流。在独立模式下， $V_{REF2}$  单位增益缓冲器不连接到单独的 GPIO 引脚。

RFB 多路开关用于选择将电流检测放大器 A2 的输出连接到内部 5 k $\Omega$  反馈电阻 (准谐振) 或  $I_{SOUT}$  引脚。

**寄存器 6-15: MODECON: 主器件 / 从器件和 RFB 多路开关控制寄存器**

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
MSC1	MSC0	RFB	—	—	—	—	—
bit 7							bit 0

**图注:**

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

bit 7-6 **MSC<1:0>**: 主器件 / 从器件配置位

- 00 = 器件设置为独立单元
- 01 = 器件设置为主器件
- 10 = 器件设置为从器件
- 11 = 保留

bit 5 **RFB**: 电流检测放大器 (A2) 输出电阻反馈多路开关配置位

- 0 =  $R_{FB\_INT}$  5 k $\Omega$
- 1 =  $I_{SOUT}$

bit 4-0 **未实现**: 读为 0

注:

## 7.0 典型性能曲线

注：以下图表是基于有限样本数的统计结果，仅供参考。所列出的性能特性未经测试，不做任何保证。一些图表中列出的数据可能超出规定的工作范围（如：超出了规定的电源电压范围），因此不在担保范围内。

注：除非另外说明，否则  $V_{IN} = 12V$ ,  $F_{SW} = 150\text{ kHz}$ ,  $T_A = +25^\circ\text{C}$ 。

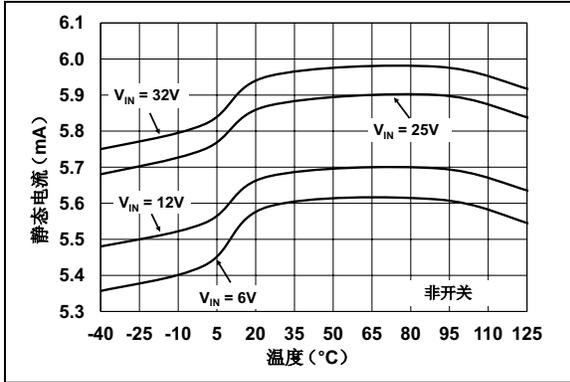


图 7-1:  $I_Q$ —温度曲线

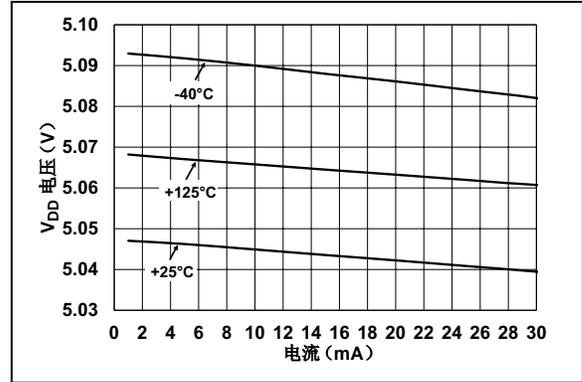


图 7-4: 负载调节

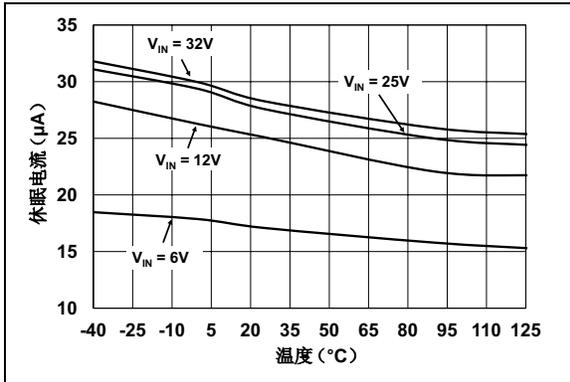


图 7-2:  $I_Q$ —温度曲线（休眠模式）

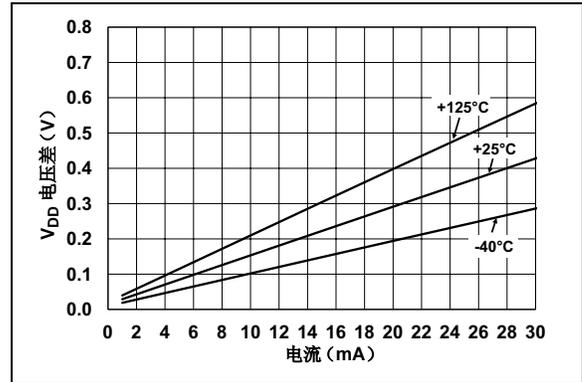


图 7-5:  $V_{DD}$  电压差—输出电流 (mA) 曲线

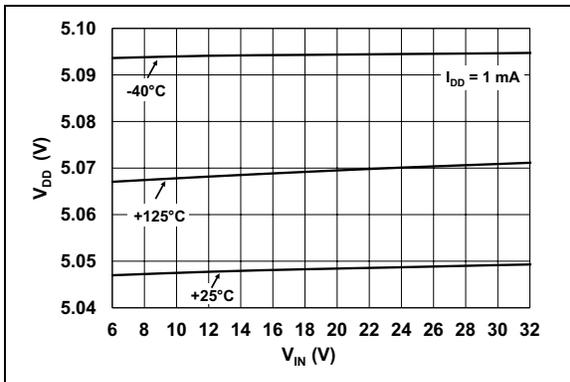


图 7-3: 线路稳定度

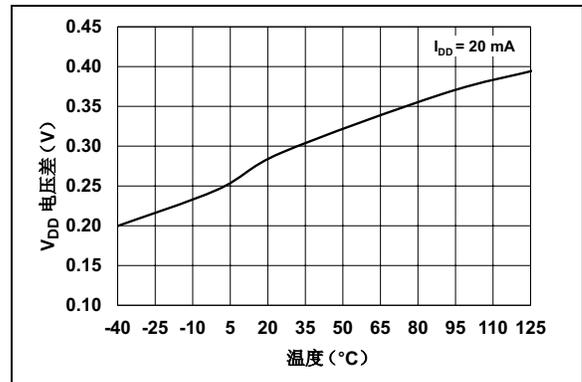


图 7-6:  $V_{DD}$  电压差—温度曲线

# MCP19114/5

注：除非另外说明，否则  $V_{IN} = 12V$ ， $F_{SW} = 150\text{ kHz}$ ， $T_A = +25^\circ\text{C}$ 。

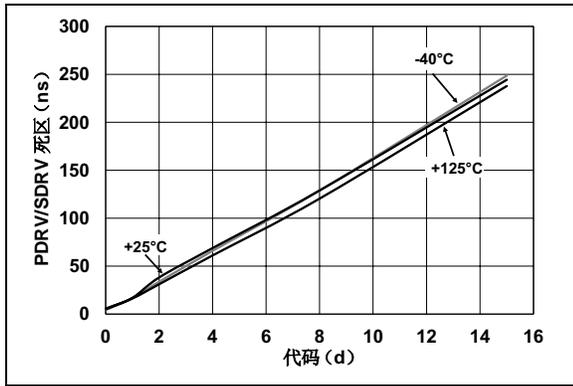


图 7-7: 输出驱动器死区——代码和温度曲线

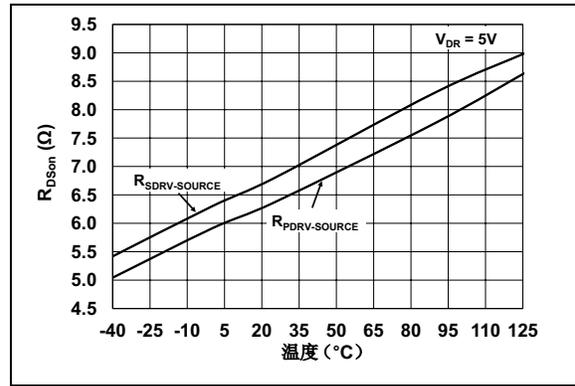


图 7-10: 源输出驱动器  $R_{DSon}$  —— 温度曲线

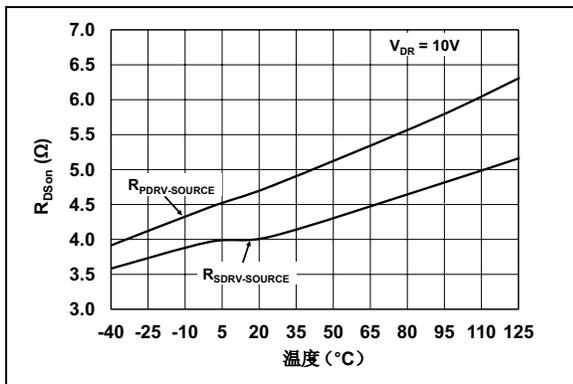


图 7-8: 源输出驱动器  $R_{DSon}$  —— 温度曲线

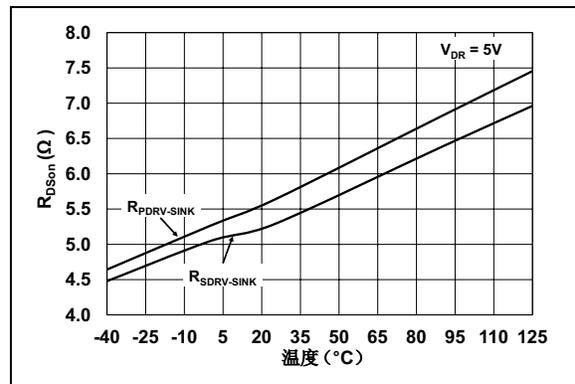


图 7-11: 灌输出驱动器  $R_{DSon}$  —— 温度曲线

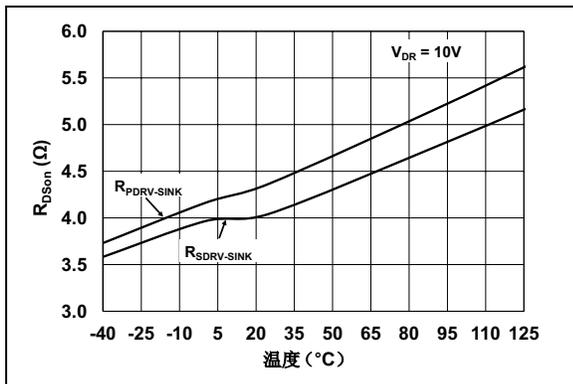


图 7-9: 灌输出驱动器  $R_{DSon}$  —— 温度曲线

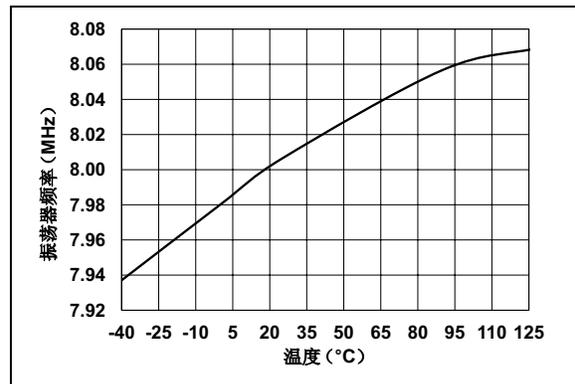


图 7-12: 振荡器频率 —— 温度曲线

注：除非另外说明，否则  $V_{IN} = 12V$ ， $F_{SW} = 150\text{ kHz}$ ， $T_A = +25^\circ\text{C}$ 。

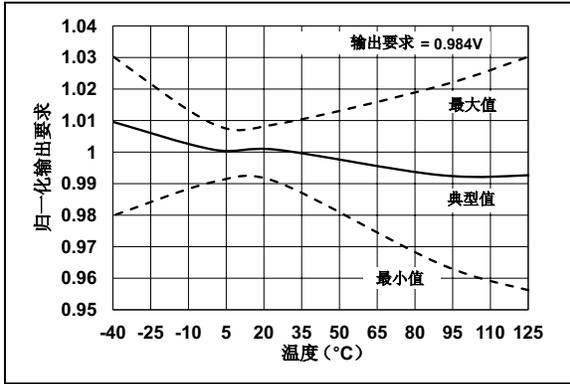


图 7-13: 归一化输出要求——温度曲线

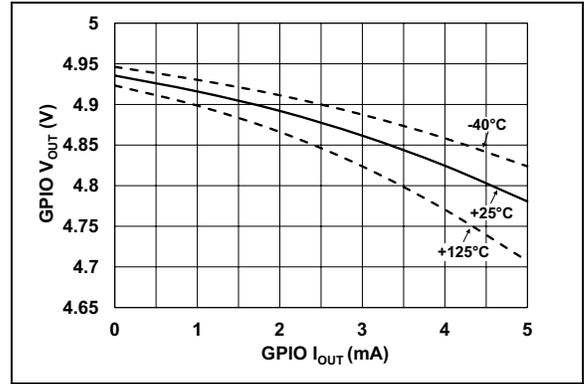


图 7-16: GPIO 平均输出电压——电流曲线

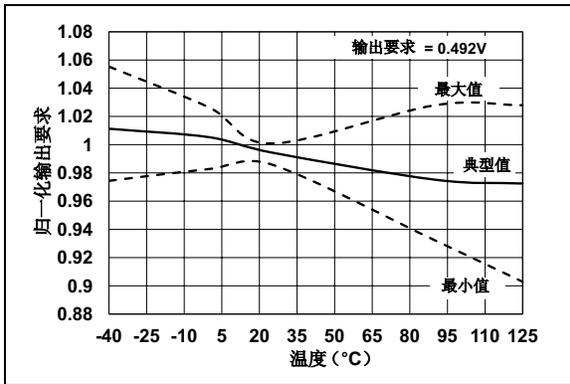


图 7-14: 归一化输出要求——温度曲线

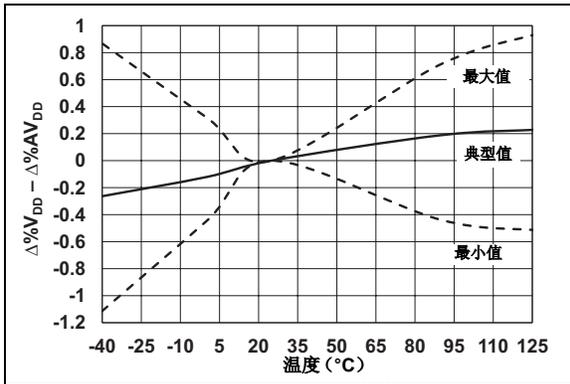


图 7-15:  $V_{DD}$ — $AV_{DD}$  温度漂移跟踪曲线

# MCP19114/5

---

注:

## 8.0 系统基准测试

为简化系统设计和基准测试，MCP19114/5 器件提供了一个多路开关，用于输出不同的内部模拟信号。这些信号可通过单位增益缓冲器在 GPA0 引脚上测量。GPA0 引脚的配置控制位于 ABECON 寄存器中。

单位增益模拟缓冲器输出上存在的信号控制位于 ADCON0 寄存器中。

**寄存器 8-1: ADCON0: 模数控制寄存器**

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	CHS4	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON
bit 7							bit 0

**图注:**

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

bit 7 **未实现:** 读为 0

bit 6-2 **CHS<4:0>:** 模拟通道选择位  
 00000 =  $V_{IN}/n$  模拟电压测量 ( $V_{IN}/15.5328$ )  
 00001 =  $V_{REF} + V_{ZC}$  (DAC 参考电压 +  $V_{ZC}$  基本电压 (设置电流调节级别))  
 00010 =  $OV\_REF$  (过压比较器的参考电压)  
 00011 =  $V_{BGR}$  (带隙参考电压)  
 00100 =  $V_S$  (与  $V_{OUT}$  成比例的电压)  
 00101 =  $EA\_SC$  (斜率补偿后的误差放大器输出)  
 00110 =  $A2$  (次级电流检测放大器输出, 位于  $R_{FB\_INT}$  连接处)  
 00111 =  $PEDESTAL$  (基本电压)  
 01000 = 保留  
 01001 = 保留  
 01010 =  $IP\_ADJ$  (在 PWM 比较器上进行基本和失调调节后的 IP 电压)  
 01011 =  $IP\_OFF\_REF$  (IP 失调参考电压)  
 01100 =  $V_{DR}/n$  ( $V_{DR}/n$  模拟驱动器电压测量 =  $0.229V/V * V_{DR}$ )  
 01101 =  $TEMP\_SNS$  (模拟电压, 表示内部温度)  
 01110 =  $DLL\_VCON$  (延迟锁定环参考电压——死区的控制电压)  
 01111 =  $SLPCMP\_REF$  (斜率补偿参考电压)  
 10000 = 未实现  
 10001 = 未实现  
 10010 = 未实现  
 10011 = 未实现  
 10100 = 未实现  
 10101 = 未实现  
 10110 = 未实现  
 10111 = 未实现  
 11000 = GPA0/AN0 (即 ADDR1)  
 11001 = GPA1/AN1 (即 ADDR0)  
 11010 = GPA2/AN2 (即温度传感器输入)  
 11011 = GPA3/AN3 (即 BIN)  
 11100 = GPB1/AN4  
 11101 = GPB4/AN5 (仅 MCP19115)  
 11110 = GPB5/AN6 (仅 MCP19115)  
 11111 = GPB6/AN7 (仅 MCP19115)

bit 1 **GO/DONE:** A/D 转换状态位  
 1 = A/D 转换周期正在进行。将该位置 1 可启动 A/D 转换周期。  
 A/D 转换完成后, 该位由硬件自动清零。  
 0 = A/D 转换已完成 / 未进行

bit 0 **ADON:** A/D 转换状态位  
 1 = A/D 转换器模块正在工作  
 0 = A/D 转换器模块已关闭且不消耗任何工作电流

# MCP19114/5

---

注:

## 9.0 器件校准

只读存储器的地址单元 2080h 至 208Fh 中包含出厂校准数据。有关如何从这些存储单元中读取数据的信息，请参见第 17.0 节“闪存程序存储器控制”。

### 9.1 校准字 1

校准字 1 位于存储单元 2080h 中。DCSRFB<6:0> 位用于设置电流检测差分放大器 (A2) 在配置为使用内部反馈电阻时的失调校准。提供  $\pm 30$  mV 的校准范围，20h 和 00h 为中间值，无失调。MSB 仅为极性。固件必须读取这些值并将其写入 DCSCAL 寄存器才能实现失调校准。

寄存器 9-1: CALWD1: 校准字 1 寄存器

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

U-0	R/P-1						
—	DCSRFB6	DCSRFB5	DCSRFB4	DCSRFB3	DCSRFB2	DCSRFB1	DCSRFB0
bit 7							bit 0

**图注:**

R = 可读位

P = 可编程位

U = 未用位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 13-7      **未实现:** 读为 0

bit 6-0      **DCSRFB<6:0>:** 配置为使用内部反馈电阻时的输入差分电流检测校准位

# MCP19114/5

## 9.2 校准字 2

校准字 2 位于存储单元 2081h 中。它包含用于退饱和和比较器电流测量输入失调的校准位。固件必须读取这些值并将其写入 DSTCAL 寄存器才能实现出厂失调校准。出厂失调校准将使失调电压最小化。退饱和和比较器为用户可能想要实现自己的失调电压值的少数示例之一。将用户定义的值写入 DSTCAL 寄存器可提供该灵活性。该寄存器还包含对内部 5k 反馈电阻进行 2% 以内微调所需的微调位，即使用 <RFBT5:0> 位。固件必须读取这些值并将其写入 RFBTCAL 寄存器才能实现出厂微调反馈电阻值。

### 寄存器 9-2: CALWD2: 校准字 2 寄存器

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	DST4	DST3	DST2	DST1	DST0
bit 13					bit 8

U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	RFBT5	RFBT4	RFBT3	RFBT2	RFBT1	RFBT0
bit 7							bit 0

#### 图注:

R = 可读位                      P = 可编程位                      U = 未用位, 读为 0  
 -n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

- bit 13            **未实现:** 读为 0
- bit 12-8        **DST<4:0>:** 退饱和和比较器电流测量失调校准位
  - 11111 = 最大负失调校准 (-30 mV)
  - 
  - 
  - 
  - 10000 = 中间值 (0 mV)
  - 00000 = 中间值 (0 mV)
  - 
  - 
  - 
  - 01111 = 最大正失调校准 (+30 mV)
- bit 7-6         **未实现:** 读为 0
- bit 5-0         **RFBT<5:0>:** 内部反馈电阻微调位

## 9.3 校准字 3

存储单元 2082h 的 VRO<5:0> 位用于校准输出稳压参考设定值 ( $V_{REF}$ ) 的缓冲放大器失调。固件必须读取这些值并将其写入 VROCAL 寄存器才能实现正确校准。

存储单元 2082h 的 BGR<3:0> 位用于校准带隙参考电压。固件必须读取这些值并将其写入 BGRCAL 寄存器才能实现正确校准。

### 寄存器 9-3: CALWD3: 校准字 3 寄存器

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
VRO5	VRO4	VRO3	VRO2	VRO1	VRO0
bit 13					bit 8

U-0	U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	—	BGR3	BGR2	BGR1	BGR0
bit 7							bit 0

#### 图注:

R = 可读位

P = 可编程位

U = 未用位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 13-8      **VRO<5:0>**: 参考电压 ( $V_{REF}$ ) 失调校准位

bit 7-4      **未使用**: 读为 0

bit 3-0      **BGR<3:0>**: 带隙参考电压校准位

## 9.4 校准字 4

存储单元 2083h 的 TTA<3:0> 位中包含用于出厂设置的过温阈值的校准位。固件必须读取这些值并将其写入 TTACAL 寄存器才能实现正确校准。

### 寄存器 9-4: CALWD4: 校准字 4 寄存器

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

U-0	U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	—	TTA3	TTA2	TTA1	TTA0
bit 7							bit 0

#### 图注:

R = 可读位

P = 可编程位

U = 未用位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 13-4      **未实现**: 读为 0

bit 3-0      **TTA<3:0>**: 过温阈值校准位

# MCP19114/5

## 9.5 校准字 5

存储单元 2084h 的 TANA<9:0> 位中包含来自内部温度传感器的 ADC 读数（硅片温度为 +30°C 时）。内部温度传感器的温度系数为 16 mV/°C。

### 寄存器 9-5: CALWD5: 校准字 5 寄存器

U-0	U-0	U-0	U-0	R/P-1	R/P-1
—	—	—	—	TANA9	TANA8
bit 13				bit 8	

R/P-1							
TANA7	TANA6	TANA5	TANA4	TANA3	TANA2	TANA1	TANA0
bit 7							bit 0

#### 图注:

R = 可读位  
-n = POR 时的值  
P = 可编程位  
1 = 置 1  
U = 未用位, 读为 0  
0 = 清零  
x = 未知

bit 13-10 未实现: 读为 0  
bit 9-0 TANA<9:0>: 内部硅片温度 (30°C 时) 的 ADC 读数校准位

## 9.6 校准字 6

存储单元 2085h 的 FCAL<6:0> 位用于设置内部振荡器校准。固件必须读取这些值并将其写入 OSCCAL 寄存器才能实现正确校准。

### 寄存器 9-6: CALWD6: 校准字 6 寄存器

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

U-0	R/P-1						
—	FCAL6	FCAL5	FCAL4	FCAL3	FCAL2	FCAL1	FCAL0
bit 7							bit 0

#### 图注:

R = 可读位  
-n = POR 时的值  
P = 可编程位  
1 = 置 1  
U = 未用位, 读为 0  
0 = 清零  
x = 未知

bit 13-7 未实现: 读为 0  
bit 6-0 FCAL<6:0>: 内部振荡器校准位

## 9.7 校准字 7

存储单元 2086h 的 DCS<6:0> 用于存储电流检测差分放大器 (A2) 在配置为使用  $I_{SOUT}$  时的出厂设置失调校准。提供 +/-30 mV 的配置范围, 20h 和 00h 为中间值, 无失调。MSB 仅为极性。固件必须将该值读取到 DCSCAL 寄存器才能实现失调校准。如果使用内部反馈电阻, 请参见寄存器 9-1。

**寄存器 9-7: CALWD7: 校准字 7 寄存器**

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

U-0	R/P-1						
—	DCS6	DCS5	DCS4	DCS3	DCS2	DCS1	DCS0
bit 7							bit 0

**图注:**

R = 可读位

P = 可编程位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 13-7      **未实现:** 读为 0

bit 6-0      **DCS<6:0>:** 使用  $I_{SOUT}$  时的差分电流检测放大器校准位

# MCP19114/5

## 9.8 校准字 8

存储单元 2089h 的 ADCCAL<13:0> 位中包含用于 A/D 转换器的校准位。校准字 8 (ADCCAL<13:0>) 包含满量程 ADC 参考电压的出厂测量值。该值表示每伏的 A/D 转换器计数次数。ADCC<4:0> 位表示 A/D 转换器计数的小数部分，可在过采样 ADC 以实现增强分辨率时提供更高的精度。该校准字可用于校准模数转换器读取的信号。

### 寄存器 9-8: CALWD8: 校准字 8 寄存器

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
ADCC13	ADCC12	ADCC11	ADCC10	ADCC9	ADCC8
bit 13					bit 8

R/P-1							
ADCC7	ADCC6	ADCC5	ADCC4	ADCC3	ADCC2	ADCC1	ADCC0
bit 7							bit 0

<b>图注:</b>							
R = 可读位	P = 可编程位	U = 未实现位，读为 0					
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知				

bit 13-5     **ADCC<13:5>**: A/D 转换器计数的整数部分  
 111111111 = 511  
 •  
 •  
 •  
 000000000 = 0

bit 4-0     **ADCC<4:0>**: A/D 转换器计数的小数部分  
 11111 = 0.96875  
 •  
 •  
 •  
 00001 = 0.03125  
 00000 = 0.00000

## 9.9 校准字 9

校准字 9 位于存储单元 208Ah 中。该存储单元中存储的值表示模拟测试缓冲器的失调电压（单位为 mV）。该值是一个 8 位二进制补码，用于补偿通过模拟测试多路开关发送的任何信号。请参见第 8.0 章了解测试信号详细信息。

**寄存器 9-9: CALWD9: 校准字 9 寄存器**

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

R/P-1							
BUFF7	BUFF6	BUFF5	BUFF4	BUFF3	BUFF2	BUFF1	BUFF0
bit 7							bit 0

**图注:**

R = 可读位	P = 可编程位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

bit 13-8      **未实现:** 读为 0

bit 7-0      **BUFF<7:0>:** 模拟缓冲器失调校准位

11111111 = 中间值 (-1 mV)

•

•

•

10000000 = 最大负失调 (-128 mV)

01111111 = 最大正失调 (127 mV)

•

•

•

00000000 = 中间值 (0 mV)

# MCP19114/5

## 9.10 校准字 10

存储单元 208Bh 的 A2CAL<7:0> 位中包含用于电流检测放大器 (A2) 增益误差的校准位。要在使用该放大器时获得最佳调节精度, 固件可读取该值并用其调节 VREF 命令。详细信息请参见第 6.8 节“输出稳压参考配置”。

寄存器 9-10: CALWD10: 校准字 10 寄存器

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

R/P-1							
A2CAL7	A2CAL6	A2CAL5	A2CAL4	A2CAL3	A2CAL2	A2CAL1	A2CAL0
bit 7							bit 0

**图注:**

R = 可读位                      P = 可编程位                      U = 未实现位, 读为 0  
-n = POR 时的值                1 = 置 1                              0 = 清零                              x = 未知

bit 13-8            未实现: 读为 0  
bit 7-0            A2CAL<7:0>: 电流检测放大器 (A2) 增益误差校准位

## 9.11 校准字 11

存储单元 2088h 的 HFVDD<9:0> 位用于存储  $V_{DD}/2$  的内部 ADC 读数（计数）。该值是 25°C 时的出厂存储值，可用于校准由  $V_{DD}$  供电的比例输出 ADC 测量值。

**寄存器 9-11: CALWD11: 校准字 11 寄存器**

U-0	U-0	U-0	U-0	R/P-1	R/P-1
—	—	—	—	HFVDD9	HFVDD8
bit 13				bit 8	

R/P-1							
HFVDD7	HFVDD6	HFVDD5	HFVDD4	HFVDD3	HFVDD2	HFVDD1	HFVDD0
bit 7							bit 0

**图注:**

R = 可读位

P = 可编程位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 13-10      **未实现:** 读为 0

bit 9-0        **HFVDD<9:0>:** 三分之一  $V_{DD}$  校准位

# MCP19114/5

---

注:

## 10.0 存储器构成

MCP19114/5 中有两种类型的存储器:

- 程序存储器
- 数据存储器
  - 特殊功能寄存器 (Special Function Register, SFR)
  - 通用 RAM

## 10.1 程序存储器构成

MCP19114/5 具有一个 13 位程序计数器, 可寻址 8000 x 14 的程序存储空间。但只有前 4000 x 14 (0000h-0FFFh) 是物理实现的。寻址超出地址边界的单元将导致操作返回到前 4000 x 14 的空间内。复位向量地址为 0000h, 中断向量地址为 0004h (见图 10-1)。程序存储总线的宽度 (指令字) 为 14 位: 由于所有指令均为单字指令, 因此 MCP19114/5 的空间可存储 4000 条指令。

图 10-1: MCP19114 的程序存储器映射和堆栈



# MCP19114/5

## 10.1.1 将程序存储器当作数据存储器读取

有两种方法可访问程序存储器中的常量。第一种方法是使用 RETLW 指令表。第二种方法是设置文件选择寄存器 (Files Select Register, FSR) 指向程序存储器。

### 10.1.1.1 RETLW 指令

RETLW 指令用于提供对常量表的访问。例 10-1 给出了创建这种表的推荐方法。

#### 例 10-1: RETLW 指令

```
constants
  BRW                ;Add Index in W to
                    ;program counter to
                    ;select data
  RETLW DATA0       ;Index0 data
  RETLW DATA1       ;Index1 data
  RETLW DATA2
  RETLW DATA3

my_function
  ;... LOTS OF CODE...
  MOVLW DATA_INDEX
  call constants
  ;... THE CONSTANT IS IN W
```

通过 BRW 指令，可以非常简单地实现这种表。如果代码必须保持对于前几代单片机的移植性，则 BRW 指令不可用，所以必须使用先前的表读方法。

## 10.2 数据存储器构成

数据存储器 (见图 10-1) 分为 4 个存储区，其中包含通用寄存器 (General Purpose Register, GPR) 和特殊功能寄存器 (SFR)。特殊功能寄存器位于各存储区的前 32 个单元。Bank 0 中的寄存器单元 20h-7Fh、Bank 1 中的寄存器单元 A0h-EFh 和 Bank 2 中的寄存器单元 120h-16Fh 是通用寄存器，可实现为静态 RAM。所有其他 RAM 均未实现，对其执行读操作时返回 0。STATUS 寄存器中的 RP<1:0> 位是存储区选择位。

#### 例 10-2: 存储区选择

RP1	RP0	
0	0	-> 选择 Bank 0
0	1	-> 选择 Bank 1
1	0	-> 选择 Bank 2
1	1	-> 选择 Bank 3

要将值从一个寄存器传送到另一个寄存器，该值必须通过 W 寄存器。这意味着对于所有寄存器至寄存器的传送，都需要两个指令周期。

可直接或间接访问整个数据存储器。直接寻址可能需要使用 RP<1:0> 位。间接寻址需要使用 FSR。间接寻址使用 STATUS 寄存器中的间接寄存器指针 (Indirect Register Pointer, IRP) 来访问数据存储器的 Bank0/Bank1 或 Bank2/Bank3 区域。

### 10.2.1 通用寄存器文件

MCP19114/5 的寄存器文件按 64 x 8 的形式构成。通过 FSR 可直接或间接访问每个寄存器 (见第 10.5 节“间接寻址, INDF 和 FSR 寄存器”)。

## 10.2.2 内核寄存器

内核寄存器包含会直接影响基本操作的寄存器。从任何存储区都可以寻址到内核寄存器。下表 10-1 列出了这些寄存器。详细信息，请参见表 10-2。

**表 10-1: 内核寄存器**

地址	BANKx
x00h、x80h、x100h 或 x180h	INDF
x02h、x82h、x102h 或 x182h	PCL
x03h、x83h、x103h 或 x183h	STATUS
x04h、x84h、x104h 或 x184h	FSR
x0Ah、x8Ah、x10Ah 或 x18Ah	PCLATH
x0Bh、x8Bh、x10Bh 或 x18Bh	INTCON

### 10.2.2.1 STATUS 寄存器

STATUS 寄存器包含：

- ALU 的算术运算状态
- 复位状态
- 数据存储器（RAM）的存储区选择位

和任何其他寄存器一样，STATUS 寄存器也可作为任何指令的目标寄存器。如果 STATUS 寄存器是影响 Z、DC 或 C 位的指令的目标寄存器，那么将禁止对这 3 位进行写操作。根据器件逻辑，这些位会被置 1 或清零。此外， $\overline{TO}$  和  $\overline{PD}$  位是不可写的。因此，当执行一条以 STATUS 寄存器作为目标寄存器的指令后，结果可能与预想的不同。

例如，CLRF STATUS 会清零高 3 位并将 Z 位置 1，从而使得 STATUS 寄存器的内容变为 000u u1uu（其中 u = 不变）。

因此，建议仅使用 BCF、BSF、SWAPF 和 MOVWF 指令来改变 STATUS 寄存器的内容，因为这些指令不影响任何状态位。

**注 1:** 在减法运算中，C 和 DC 位分别作为借位位和半借位位。

**寄存器 10-1: STATUS: STATUS 寄存器**

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC <sup>(1)</sup>	C <sup>(1)</sup>
bit 7						bit 0	

**图注:**

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

bit 7                      **IRP:** 寄存器存储区选择位（用于间接寻址）  
 1 = Bank 2 和 3（100h - 1FFh）  
 0 = Bank 0 和 1（00h - FFh）

bit 6-5                    **RP<1:0>:** 寄存器存储区选择位（用于直接寻址）  
 00 = Bank 0（00h - 7Fh）  
 01 = Bank 1（80h - FFh）  
 10 = Bank 2（100h - 17Fh）  
 11 = Bank 3（180h - 1FFh）

bit 4                       **$\overline{TO}$ :** 超时位  
 1 = 在上电复位、执行了 CLRWDT 指令或 SLEEP 指令后  
 0 = 发生了 WDT 超时

**注 1:** 对于借位，极性是相反的。减法指令通过加上第二个操作数的二进制补码实现。对于移位指令（RRF 和 RLF），此位值来自源寄存器的最高位或最低位。

# MCP19114/5

---

## 寄存器 10-1: STATUS: STATUS 寄存器 (续)

bit 3	<b><math>\overline{\text{PD}}</math></b> : 掉电标志位 1 = 上电复位或执行了 CLRWDT 指令后 0 = 执行了 SLEEP 指令
bit 2	<b>Z</b> : 零标志位 1 = 算术运算或逻辑运算结果为零 0 = 算术运算或逻辑运算结果不为零
bit 1	<b>DC</b> : 半进位 / 半借位标志位 <sup>(1)</sup> (ADDWF、ADDLW、SUBLW 和 SUBWF 指令) 1 = 结果的第 4 个低位向高位发生了进位 0 = 结果的第 4 个低位未向高位发生进位
bit 0	<b>C</b> : 进位 / 借位标志位 <sup>(1)</sup> (ADDWF、ADDLW、SUBLW 和 SUBWF 指令) <sup>(1)</sup> 1 = 结果的最高位发生了进位 0 = 结果的最高位未发生进位

**注 1:** 对于借位，极性是相反的。减法指令通过加上第二个操作数的二进制补码实现。对于移位指令 (RRF 和 RLF)，此位值来自源寄存器的最高位或最低位。

### 10.2.3 特殊功能寄存器

特殊功能寄存器是 CPU 和外设功能用来控制所需器件操作的寄存器 (见图 10-2)。这些寄存器为静态 RAM。

特殊寄存器可分为两类: 内核和外设。本节将介绍与单片机内核相关的特殊功能寄存器。与外设功能的操作相关的寄存器将在相应的外设功能章节中介绍。

## 10.3 数据存储

图 10-2: MCP19114/5 数据存储映射

寄存器地址		寄存器地址		寄存器地址		寄存器地址	
间接地址 <sup>(1)</sup>	00h	间接地址 <sup>(1)</sup>	80h	间接地址 <sup>(1)</sup>	100h	间接地址 <sup>(1)</sup>	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTGPA	05h	TRISGPA	85h	WPUGPA	105h	IOCA	185h
PORTGPB	06h	TRISGPB	86h	WPUGPB	106h	IOCB	186h
PIR1	07h	PIE1	87h	PE1	107h	ANSELA	187h
PIR2	08h	PIE2	88h	MODECON	108h	ANSELB	188h
PCON	09h		89h	ABECON	109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
TMR1L	0Ch		8Ch		10Ch	PORTICD <sup>(2)</sup>	18Ch
TMR1H	0Dh		8Dh		10Dh	TRISICD <sup>(2)</sup>	18Dh
T1CON	0Eh		8Eh		10Eh	ICKBUG <sup>(2)</sup>	18Eh
TMR2	0Fh		8Fh		10Fh	BIGBUG <sup>(2)</sup>	18Fh
T2CON	10h	VINUVLO	90h	SSPADD	110h	PMCON1	190h
PR2	11h	VINOVLO	91h	SSPBUF	111h	PMCON2	191h
	12h	VINCON	92h	SSPCON1	112h	PMADRL	192h
PWMPHL	13h	CC1RL	93h	SSPCON2	113h	PMADRH	193h
PWMPHH	14h	CC1RH	94h	SSPCON3	114h	PMDATL	194h
PWMRL	15h	CC2RL	95h	SSPMSK1	115h	PMDATH	195h
PWMRH	16h	CC2RH	96h	SSPSTAT	116h	DSTCAL	196h
	17h	CCDCON	97h	SSPADD2	117h	RFBTCAL	197h
	18h	DESATCON	98h	SSPMSK2	118h	OSCCAL	198h
VREFCON	19h	OVCON	99h		119h	DCSCAL	199h
VREF2CON	1Ah	OVREFCON	9Ah		11Ah	TTACAL	19Ah
OSCTUNE	1Bh	DEADCON	9Bh		11Bh	BGRCAL	19Bh
ADRESL	1Ch	SLPCRCON	9Ch		11Ch	VROCAL	19Ch
ADRESH	1Dh	ICOACON	9Dh		11Dh		19Dh
ADCON0	1Eh	ICLEBCON	9Eh		11Eh		19Eh
ADCON1	1Fh		9Fh		11Fh	保留	19Fh
	20h		A0h		120h		1A0h
通用寄存器		通用寄存器 80 字节		通用寄存器 80 字节			
96 字节			EFh		16Fh		1EFh
		访问 Bank 0	F0h	访问 Bank 0	170h	访问 Bank 0	1F0h
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank2		Bank3	

■ 未实现数据存储单元，读为 0。

注 1: 非物理寄存器。  
2: 仅当 DBGEN = 0 和 ICKBUG<INBUG> = 1 时可访问。

表 10-2: MCP19114/5 特殊寄存器汇总 BANK 0

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR 时的值	所有其他复位时的值 <sup>(1)</sup>
Bank 0											
00h	INDF	使用 FSR 的内容对数据存储器（非物理寄存器）进行寻址来寻址此单元								xxxx xxxx	xxxx xxxx
01h	TMR0	Timer0 模块寄存器								xxxx xxxx	uuuu uuuu
02h	PCL	程序计数器（PC）的低字节								0000 0000	0000 0000
03h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h	FSR	间接数据存储器地址指针								xxxx xxxx	uuuu uuuu
05h	PORTGPA	GPA7	GPA6	GPA5	—	GPA3	GPA2	GPA1	GPA0	xxx- xxxx	uuu- uuuu
06h	PORTGPB	GPB7	GPB6	GPB5	GPB4	—	—	GPB1	GPB0	xxxx --xx	uuuu --uu
07h	PIR1	—	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
08h	PIR2	CDSIF	—	—	OTIF	OVIF	DRUVIF	OVLOIF	UVLOIF	0--0 0000	0--0 0000
09h	PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$	---- --qq	---- --uu
0Ah	PCLATH	—	—	—	程序计数器高 5 位的写缓冲区					---0 0000	---0 0000
0Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF <sup>(2)</sup>	0000 000x	0000 000u
0Ch	TMR1L	16 位 TMR1 低字节的保持寄存器								xxxx xxxx	uuuu uuuu
0Dh	TMR1H	16 位 TMR1 高字节的保持寄存器								xxxx xxxx	uuuu uuuu
0Eh	T1CON	—	—	T1CKPS1	T1CKPS0	—	—	TMR1CS	TMR1ON	--00 --00	--uu --uu
0Fh	TMR2	Timer2 模块寄存器								0000 0000	uuuu uuuu
10h	T2CON	—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0	---- -000	---- -000
11h	PR2	Timer2 模块周期寄存器								1111 1111	1111 1111
12h	—	未实现								—	—
13h	PWMPHL	SLAVE 相移寄存器								xxxx xxxx	uuuu uuuu
14h	PWMPHH	SLAVE 相移寄存器								xxxx xxxx	uuuu uuuu
15h	PWMRL	PWM 寄存器低字节								xxxx xxxx	uuuu uuuu
16h	PWMRH	PWM 寄存器低字节								xxxx xxxx	uuuu uuuu
17h	—	未实现								—	—
18h	—	未实现								—	—
19h	VREFCON	VREF7	VREF6	VREF5	VREF4	VREF3	VREF2	VREF1	VREF0	0000 0000	0000 0000
1Ah	VREF2CON	VREF27	VREF26	VREF25	VREF24	VREF23	VREF22	VREF21	VREF20	0000 0000	0000 0000
1Bh	OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	---0 0000
1Ch	ADRESL	A/D 结果的低 8 位								xxxx xxxx	uuuu uuuu
1Dh	ADRESH	A/D 结果的高 2 位								0000 00xx	0000 00uu
1Eh	ADCON0	—	CHS4	CHS3	CHS2	CHS1	CHS0	$\overline{GO}/\overline{DONE}$	ADON	-000 0000	-000 0000
1Fh	ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	—	—	-000 ----	-000 ----

图注: — = 未实现单元, 读为 0, u = 不变, x = 未知, q = 取值视条件而定, 阴影 = 未实现

注 1: 其他（非上电）复位包括正常工作时的  $\overline{MCLR}$  复位和看门狗定时器复位。

注 2:  $\overline{MCLR}$  和 WDT 复位不会影响前一个值的数据锁存器。复位时, IOCF 位将清零, 但是, 如果存在不匹配情况, 将重新置 1。

表 10-3: MCP19114/5 特殊寄存器汇总 BANK 1

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR 时的值	所有其他复位时的值 <sup>(1)</sup>
Bank 1											
80h	INDF	使用 FSR 的内容对数据存储单元（非物理寄存器）进行寻址来寻址此单元								xxxx xxxx	uuuu uuuu
81h	OPTION_REG	RAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h	PCL	程序计数器（PC）的低字节								0000 0000	0000 0000
83h	STATUS	IRP	RP1	RP0	T0	PD	Z	DC	C	0001 1xxx	000q quuu
84h	FSR	间接数据存储单元地址指针								xxxx xxxx	uuuu uuuu
85h	TRISGPA	TRISA7	TRISA6	TRISA5	—	TRISA3	TRISA2	TRISA1	TRISA0	1110 1111	1110 1111
86h	TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	TRISB1	TRISB0	1111 0011	1111 0011
87h	PIE1	—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
88h	PIE2	CDSIE	—	—	OTIE	OVIE	DRUVIE	OVLOIE	UVLOIE	0--0 0000	0--0 0000
89h	—	未实现								—	—
8Ah	PCLATH	—	—	—	程序计数器高 5 位的写缓冲区					---0 0000	---0 0000
8Bh	INTCON	GIE	PEIE	TOIE	INTE	IOCE	T0IF	INTF	IOCF <sup>(2)</sup>	0000 000x	0000 000u
8Ch	—	未实现								—	—
8Dh	—	未实现								—	—
8Eh	—	未实现								—	—
8Fh	—	未实现								—	—
90h	VINUVLO	—	—	UVLO5	UVLO4	UVLO3	UVLO2	UVLO1	UVLO0	--xx xxxx	--uu uuuu
91h	VINOVLO	—	—	OVLO5	OVLO4	OVLO3	OVLO2	OVLO1	OVLO0	--xx xxxx	--uu uuuu
92h	VINCON	UVLOEN	UVLOOUT	UVLOINTP	UVLOINTN	OVLOEN	OVLOOUT	OVLOINTP	OVLOINTN	0x00 0x00	0u00 0u00
93h	CC1RL	捕捉 1/ 比较 1 寄存器 1 x 低字节（LSB）								xxxx xxxx	uuuu uuuu
94h	CC1RH	捕捉 1/ 比较 1 寄存器 2 x 高字节（MSB）								xxxx xxxx	uuuu uuuu
95h	CC2RL	捕捉 2/ 比较 2 寄存器 1 x 低字节（LSB）								xxxx xxxx	uuuu uuuu
96h	CC2RH	捕捉 2/ 比较 2 寄存器 2 x 高字节（MSB）								xxxx xxxx	uuuu uuuu
97h	CCDCON	CC2M<3:0>				CC1M<3:0>				xxxx xxxx	uuuu uuuu
98h	DESATCON	CDSMUX	CDSWDE	保留	CDSPOL	CDSOE	CDSOUT	CDSINTP	CDSINTN	0000 0x00	0000 0u00
99h	OVCON	—	—	—	—	OVEN	OVOUT	OVINTP	OVINTN	---- 0x00	---- 0u00
9Ah	OVREFCON	OOV7	OOV6	OOV5	OOV4	OOV3	OOV2	OOV1	OOV0	xxxx xxxx	uuuu uuuu
9Bh	DEADCON	PDRVDT3	PDRVDT2	PDRVDT1	PDRVDT0	SDRVDT3	SDRVDT2	SDRVDT1	SDRVDT0	xxxx xxxx	uuuu uuuu
9Ch	SLPCRCON	—	SLPBY	SLPS5	SLPS4	SLPS3	SLPS2	SLPS1	SLPS0	-xxx xxxx	-uuu uuuu
9Dh	ICOACON	—	—	—	—	ICOAC3	ICOAC2	ICOAC1	ICOAC0	---- xxxx	---- uuuu
9Eh	ICLEBCON	—	—	—	—	—	—	ICLEBC1	ICLEBC0	---- --xx	---- --uu
9Fh	—	未实现								—	—

图注: — = 未实现单元, 读为 0, u = 不变, x = 未知, q = 取值视条件而定, 阴影 = 未实现

注 1: 其他（非上电）复位包括正常工作时的 MCLR 复位和看门狗定时器复位。

注 2: MCLR 和 WDT 复位不会影响前一个值的数据锁存器。复位时, IOCF 位将清零, 但是, 如果存在不匹配情况, 将重新置 1。

表 10-4: MCP19114/5 特殊寄存器汇总 BANK 2

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR 时的值	所有其他复位时的值 <sup>(1)</sup>	
Bank 2												
100h	INDF	使用 FSR 的内容对数据存储寄存器（非物理寄存器）进行寻址来寻址此单元								xxxx xxxx	xxxx xxxx	
101h	TMR0	Timer0 模块寄存器								xxxx xxxx	uuuu uuuu	
102h	PCL	程序计数器（PC）的低字节								0000 0000	0000 0000	
103h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu	
104h	FSR	间接数据存储寄存器地址指针								xxxx xxxx	uuuu uuuu	
105h	WPUGPA	—	—	WPUA5	—	WPUA3	WPUA2	WPUA1	WPUA0	--1- 1111	--u- uuuu	
106h	WPUGPB	WPUB7	WPUB6	WPUB5	WPUB4	—	—	WPUB1	—	1111 --1-	uuuu --u-	
107h	PE1	PDRVEN	SDRVEN	PDRVBY	SDRVBY	—	ISPUEN	PWMSTR_PEN	PWMSTR_SEN	0000 -100	0000 -100	
108h	MODECON	MSC1	MSC0	RFB	—	—	—	—	—	001- ----	001- ----	
109h	ABECON	DIGOEN	DSEL2	DSEL1	DSEL0	DRUVSEL	—	EADIS	ANAOEN	0000 0-00	0000 0-00	
10Ah	PCLATH	—	—	—	程序计数器高 5 位的写缓冲区				---	0 0000	---	0 0000
10Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF <sup>(2)</sup>	0000 000x	0000 000u	
10Ch	—	未实现								—	—	
10Dh	—	未实现								—	—	
10Eh	—	未实现								—	—	
10Fh	—	未实现								—	—	
110h	SSPADD	ADD<7:0>								0000 0000	0000 0000	
111h	SSPBUF	同步串行端口接收缓冲区 / 发送寄存器								xxxx xxxx	uuuu uuuu	
112h	SSPCON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				0000 0000	0000 0000	
113h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000	
114h	SSPCON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000	
115h	SSPMSK1	MSK<7:0>								1111 1111	1111 1111	
116h	SSPSTAT	SMP	CKE	$D/\overline{A}$	P	S	$R/\overline{W}$	UA	BF	—	—	
117h	SSPADD2	ADD2<7:0>								0000 0000	0000 0000	
118h	SSPMSK2	MSK2<7:0>								1111 1111	1111 1111	
119h	—	未实现								—	—	
11Ah	—	未实现								—	—	
11Bh	—	未实现								—	—	
11Ch	—	未实现								—	—	
11Dh	—	未实现								—	—	
11Eh	—	未实现								—	—	
11Fh	—	未实现								—	—	

图注： — = 未实现单元，读为 0，u = 不变，x = 未知，q = 取值视条件而定，阴影 = 未实现

- 注 1: 其他（非上电）复位包括正常工作时的 MCLR 复位和看门狗定时器复位。  
 注 2: MCLR 和 WDT 复位不会影响前一个值的数据锁存器。复位时，IOCF 位将清零，但是，如果存在不匹配情况，将重新置 1。

表 10-5: MCP19114/5 特殊寄存器汇总 BANK 3

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR 时的值	所有其他复位时的值 <sup>(1)</sup>
Bank 3											
180h	INDF	使用 FSR 的内容对数据存储器（非物理寄存器）进行寻址来寻址此单元								xxxx xxxx	uuuu uuuu
181h	OPTION_REG	RAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
182h	PCL	程序计数器（PC）的低字节								0000 0000	0000 0000
183h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
184h	FSR	间接数据存储器地址指针								xxxx xxxx	uuuu uuuu
185h	IOCA	IOCA7	IOCA6	IOCA5	—	IOCA3	IOCA2	IOCA1	IOCA0	000- 0000	0000 0000
186h	IOCB	IOCB7	IOCB6	IOCB5	IOCB4	—	—	IOCB1	IOCB0	0000 -000	0000 -000
187h	ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	---- 1111	---- 1111
188h	ANSELB	—	ANSB6	ANSB5	ANSB4	—	—	ANSB1	—	-111 --1-	-111 --1-
189h	—	未实现								—	—
18Ah	PCLATH	—	—	—	程序计数器高 5 位的写缓冲区					---0 0000	---0 0000
18Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCE	TOIF	INTF	IOCF <sup>(2)</sup>	0000 000x	0000 000u
18Ch	PORTICD <sup>(3)</sup>	在线调试端口寄存器								xxx --xx	uuuu --uu
18Dh	TRISICD <sup>(3)</sup>	在线调试 TRIS 寄存器								1111 0011	1111 0011
18Eh	ICKBUG <sup>(3)</sup>	在线调试寄存器								0000 0000	000u uuuu
18Fh	BIGBUG <sup>(3)</sup>	在线调试断点寄存器								0000 0000	uuuu uuuu
190h	PMCON1	—	CALSEL	—	—	—	WREN	WR	RD	-0-- -000	-0-- -000
191h	PMCON2	程序存储器控制寄存器 2（非物理寄存器）								---- ----	---- ----
192h	PMADRL	PMADRL7	PMADRL6	PMADRL5	PMADRL4	PMADRL3	PMADRL2	PMADRL1	PMADRL0	0000 0000	0000 0000
193h	PMADRH	—	—	—	—	PMADRH3	PMADRH2	PMADRH1	PMADRH0	---- -000	---- -000
194h	PMDATL	PMDATL7	PMDATL6	PMDATL5	PMDATL4	PMDATL3	PMDATL2	PMDATL1	PMDATL0	0000 0000	0000 0000
195h	PMDATH	—	—	PMDATH5	PMDATH4	PMDATH3	PMDATH2	PMDATH1	PMDATH0	--00 0000	--00 0000
196h	DSTCAL	—	—	—	DSTCAL4	DSTCAL3	DSTCAL2	DSTCAL1	DSTCAL0	---x xxxx	---u uuuu
197h	RFBTCAL	—	—	RFBCAL5	RFBCAL4	RFBCAL3	RFBCAL2	RFBCAL1	RFBCAL0	--xx xxxx	--uu uuuu
198h	OSCCAL	—	FCALT6	FCALT5	FCALT4	FCALT3	FCALT2	FCALT1	FCALT1	-xxx xxxx	-uuu uuuu
199h	DCSCAL	—	DCSCAL6	DCSCAL5	DCSCAL4	DCSCAL3	DCSCAL2	DCSCAL1	DCSCAL0	-xxx xxxx	-uuu uuuu
19Ah	TTACAL	—	—	—	—	TTA3	TTA2	TTA1	TTA0	---- xxxx	---- uuuu
19Bh	BGRCAL	—	—	—	—	BGRT3	BGRT2	BGRT1	BGRT0	---- xxxx	---- uuuu
19Ch	VROCAL	—	—	—	VROT4	VROT3	VROT2	VROT1	VROT0	---x xxxx	---u uuuu
19Dh	—	未实现								—	—
19Eh	—	未实现								—	—
19Fh	—	保留								—	—

图注： — = 未实现单元，读为 0，u = 不变，x = 未知，q = 取值视条件而定，阴影 = 未实现

注 1： 其他（非上电）复位包括正常工作时的  $\overline{MCLR}$  复位和看门狗定时器复位。

注 2：  $\overline{MCLR}$  和 WDT 复位不会影响前一个值的数据锁存器。复位时，IOCF 位将清零，但是，如果存在不匹配情况，将重新置 1。

注 3： 仅当 DBGEN = 0 和 ICKBUG < INBUG > = 1 时可访问。

# MCP19114/5

## 10.3.1 OPTION\_REG 寄存器

OPTION\_REG 寄存器是可读写寄存器，其中包含各种控制位用于配置：

- Timer0/WDT 预分频器
- 外部 GPA2/INT 中断
- Timer0
- PORTGPA 和 PORTGPB 上的弱上拉

**注：** 要使 Timer0 实现 1:1 预分频器分配，可将 OPTION\_REG 寄存器的 PSA 位置 1，从而将预分频器分配给 WDT。请参见第 22.1.3 节“软件可编程预分频器”。

### 寄存器 10-2: OPTION\_REG: 选项寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
<u>RAPU</u>	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

**图注：**

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

- bit 7        **RAPU**: 端口 GPx 上拉使能位 <sup>(1)</sup>  
 1 = 禁止端口 GPx 上拉  
 0 = 使能端口 GPx 上拉
- bit 6        **INTEDG**: 中断边沿选择位  
 0 = INT 引脚信号的上升沿触发中断  
 1 = INT 引脚信号的下降沿触发中断
- bit 5        **T0CS**: TMR0 时钟源选择位  
 1 = T0CKI 引脚上信号的跳变  
 0 = 内部指令周期时钟
- bit 4        **T0SE**: TMR0 时钟源边沿选择位  
 1 = 在 T0CKI 引脚电平发生由高到低的跳变时递增  
 0 = 在 T0CKI 引脚电平发生由低到高的跳变时递增
- bit 3        **PSA**: 预分频器分配位  
 1 = 将预分频器分配给 WDT  
 0 = 将预分频器分配给 Timer0 模块
- bit 2-0     **PS<2:0>**: 预分频比选择位

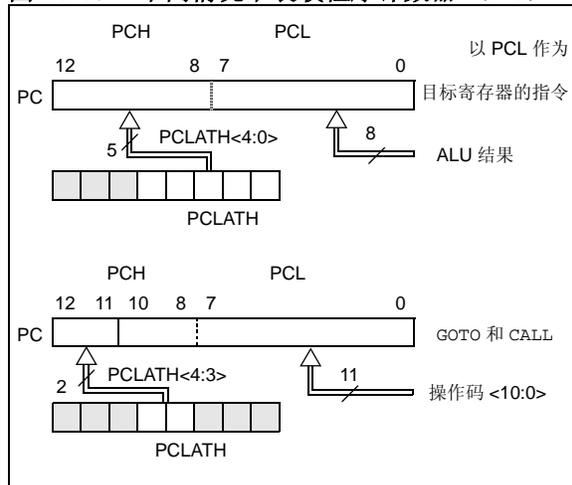
位值	TMR0 预分频比	WDT 预分频比
000	1: 2	1: 1
001	1: 4	1: 2
010	1: 8	1: 4
011	1: 16	1: 8
100	1: 32	1: 16
101	1: 64	1: 32
110	1: 128	1: 64
111	1: 256	1: 128

**注 1:** 还必须使能各个 WPUx 位。

## 10.4 PCL 和 PCLATH

程序计数器（Program Counter, PC）为 13 位宽。它的低字节来自可读写的 PCL 寄存器。高字节（PC<12:8>）来自 PCLATH，不可直接读写。任何复位都将清零 PC。图 10-3 显示了装载 PC 的两种情形：上方的示例显示了写入 PCL（PCLATH<4:0> → PCH）时如何装载 PC，图 10-3 下方的示例显示了执行 CALL 或 GOTO 指令（PCLATH<4:3> → PCH）时如何装载 PC。

图 10-3: 不同情况下装载程序计数器（PC）



### 10.4.1 修改 PCL 寄存器

执行任何将 PCL 寄存器作为目标寄存器的指令的同时会导致程序计数器 PC<12:8> 位（PCH）替换为 PCLATH 寄存器的内容。这使得程序计数器的整个内容可通过将所需值的高 5 位写入 PCLATH 寄存器来进行更改。在将低 8 位写入 PCL 寄存器后，程序计数器的所有 13 位的值将更改为 PCLATH 寄存器中的值以及那些写入到 PCL 寄存器的值。

### 10.4.2 计算 GOTO

计算 GOTO 指令是通过向程序计数器添加偏移量（ADDWF PCL）来实现的。通过修改 PCL 寄存器跳转到查找表或程序跳转表（计算 GOTO）时应十分小心。假设 PCLATH 设置为表起始地址，当表长大于 255 个指令或存储器地址的低 8 位在表的中间从 0xFFh 计满返回到 0X00h 时，对于表起始地址与表内单元之间发生的每次地址计满返回，PCLATH 都必须递增 1。

更多信息，请参见应用笔记 AN556——“Implementing a Table Read”（DS00000556）。

### 10.4.3 计算函数调用

计算函数 CALL 允许程序维护函数表并提供其他方法来执行状态机或查找表。当使用计算函数 CALL 执行表读操作时，应注意表地址是否会导致 PCL 的值超出存储边界（每个存储区为 256 个字节）。

使用 CALL 指令时，可使用 CALL 指令的操作数来装载 PCH<2:0> 和 PCL 寄存器。使用 PCLATH<4:3> 来装载 PCH<4:3>。

### 10.4.4 堆栈

MCP19114/5 具有一个 8 级深 x 13 位宽的硬件堆栈（见图 10-1）。堆栈空间既不属于程序空间，也不属于数据空间，并且堆栈指针不可读写。当执行 CALL 指令或由于中断导致程序跳转时，PC 的值会被压入堆栈。当执行 RETURN、RETLW 或 RETFIE 指令时，PC 值从堆栈弹出。PCLATH 的值不受压栈或出栈操作的影响。

该堆栈用作循环缓冲区。这就是说，在压栈 8 次后，第 9 次压栈操作会覆盖第 1 次压栈操作存储的值，而第 10 次压栈操作会覆盖第 2 次压栈操作存储的值（以此类推）。

- |  |
|--|
| <p><b>注 1:</b> 没有任何状态用于指示堆栈溢出或堆栈下溢情况。</p> <p><b>注 2:</b> 没有称为 PUSH 或 POP 的指令或助记符。这两个操作是在执行 CALL、RETURN、RETLW 和 RETFIE 指令或跳转到中断地址向量时发生。</p> |
|--|

## 10.5 间接寻址，INDF 和 FSR 寄存器

INDF 寄存器不是物理寄存器。对 INDF 寄存器寻址将导致间接寻址。

通过使用 INDF 寄存器可以实现间接寻址。使用 INDF 寄存器的任何指令实际将访问文件选择寄存器（FSR）所指向的数据。间接读取 INDF 将产生 00h。直接写入 INDF 寄存器将导致空操作（但状态位可能受到影响）。8 位 FSR 寄存器和 STATUS 寄存器的 IRP 位一起构成有效的 9 位地址，如图 10-4 所示。

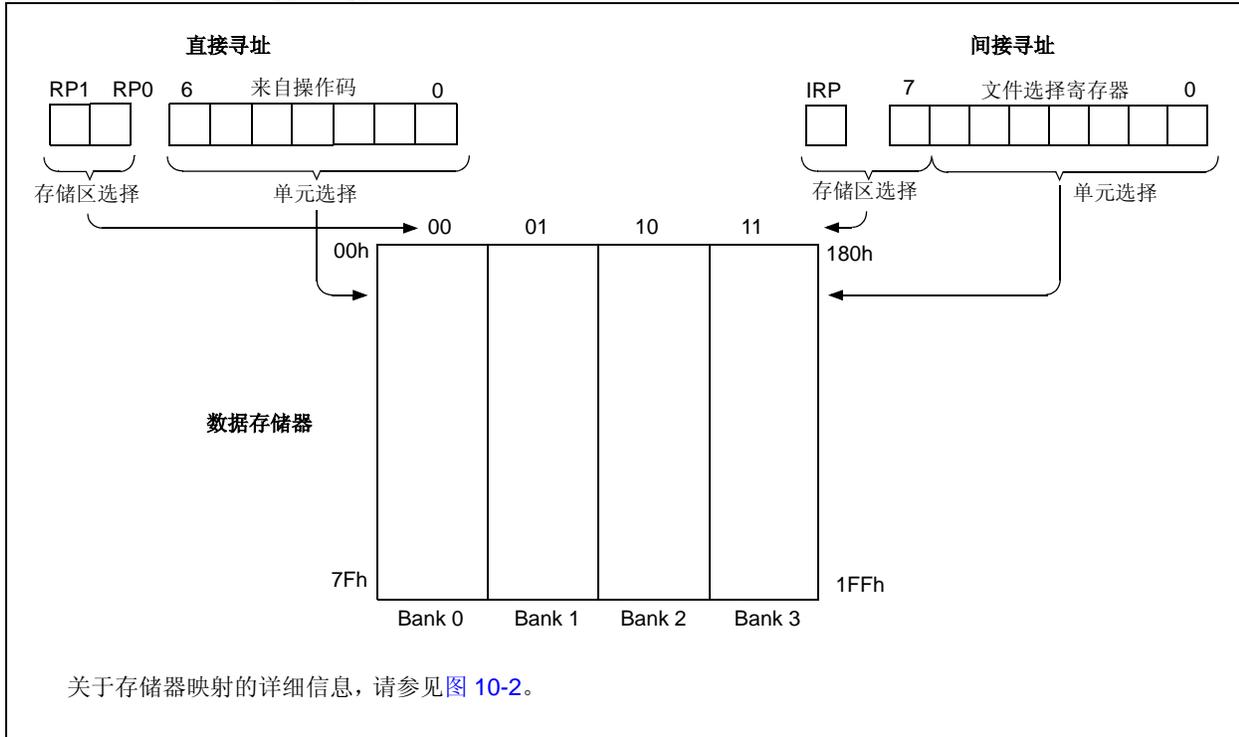
例 10-3 显示了一段可通过间接寻址来清零 RAM 单元 40h-7Fh 的简单程序。

# MCP19114/5

例 10-3: 间接寻址

```
MOVLW 0x40 ;initialize pointer
MOVWF FSR ;to RAM
NEXT CLR F INDF ;clear INDF register
      INCF FSR ;inc pointer
      BTFSS FSR,7 ;all done?
      GOTO NEXT ;no clear next
CONTINUE ;yes continue
```

图 10-4: 直接 / 间接寻址



## 11.0 器件配置

器件配置包括配置字寄存器、代码保护以及器件 ID。

**注：** 配置字寄存器中的  $\overline{\text{DBGEN}}$  位由器件开发工具（包括调试器和编程器）自动管理。对于正常的器件操作，此位应保持为 1。调试仅在 MCP19115 上可用。

### 11.1 配置字

有几个配置字位可用于使能不同的定时器和选择存储器保护选项。这些位实现为位于 2007h 处的配置字寄存器。

**寄存器 11-1: CONFIG: 配置字寄存器**

R/P-1	U-1	R/P-1	R/P-1	U-1	R/P-1
$\overline{\text{DBGEN}}$	—	WRT1	WRT0	—	BOREN
bit 13			bit 8		

U-1	R/P-1	R/P-1	R/P-1	R/P-1	U-1	U-1	U-1
—	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRT}}\overline{\text{E}}$	WDTE	—	—	—
bit 7							bit 0

**图注：**

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

- bit 13       **$\overline{\text{DBGEN}}$** : ICD 调试位  
1 = 禁止 ICD 调试模式  
0 = 使能 ICD 调试模式
- bit 12      **未实现**: 读为 0
- bit 11-10   **WRT<1:0>**: 闪存程序存储器自写使能位  
11 = 写保护关闭  
10 = 000h 至 3FFh 受写保护，400h 至 FFFh 可由 PMCON1 控制修改  
01 = 000h 至 7FFh 受写保护，800h 至 FFFh 可由 PMCON1 控制修改  
00 = 000h 至 FFFh 受写保护，整个程序存储器受写保护
- bit 9        **未实现**: 读为 0
- bit 8        **BOREN**: 欠压复位使能位  
1 = BOR 在休眠期间禁止而在工作期间使能  
0 = 禁止 BOR
- bit 7        **未实现**: 读为 0
- bit 6         **$\overline{\text{CP}}$** : 代码保护  
1 = 程序存储器不受代码保护  
0 = 程序存储器受外部读写保护
- bit 5        **MCLRE**:  $\overline{\text{MCLR}}$  引脚功能选择  
1 =  $\overline{\text{MCLR}}$  引脚为  $\overline{\text{MCLR}}$  功能引脚，并使能内部弱上拉  
0 =  $\overline{\text{MCLR}}$  引脚为备用功能引脚，内部禁止  $\overline{\text{MCLR}}$  功能
- bit 4         **$\overline{\text{PWRT}}\overline{\text{E}}$** : 上电延时定时器使能位 <sup>(1)</sup>  
1 = 禁止 PWRT  
0 = 使能 PWRT
- bit 3        **WDTE**: 看门狗定时器使能位  
1 = 使能 WDT  
0 = 禁止 WDT
- bit 2-0      **未实现**: 读为 0

**注 1:** 该位保留，而不是由用户控制。

## 11.2 代码保护

代码保护用于保护器件不受未经授权的访问。任何代码保护设置都不会影响对程序存储器的内部访问。

### 11.2.1 程序存储器保护

整个程序存储空间都通过配置字寄存器中的  $\overline{\text{CP}}$  位来防止外部读写操作。当  $\overline{\text{CP}} = 0$  时，将禁止对程序存储器的外部读写操作，读取时将返回全 0。无论保护位的设置如何，CPU 都可以继续读取程序存储器。对程序存储器的写操作取决于写保护设置。更多信息，请参见第 11.3 节“写保护”。

## 11.3 写保护

写保护用于保护器件不受意外的自写访问。在允许修改程序存储器其他区域时，引导加载程序软件之类的应用程序可受到保护。

配置字寄存器中的  $\text{WRT}\langle 1:0 \rangle$  位定义受保护的程序存储块的大小。

## 11.4 ID 地址单元

有 4 个存储单元（2000h-2003h）被指定为 ID 地址单元，供用户存储校验和或其他代码标识号。这些单元在正常执行过程中不能访问，但在编程 / 校验模式下可读写。使用 MPLAB 集成开发环境（IDE）时，只报告 ID 地址单元的低 7 位。

## 12.0 振荡器模式

MCP19114/5 配置了一个 8 MHz 内部振荡器。

### 12.1 内部振荡器 (INTOSC)

内部振荡器模块提供 8 MHz 的系统时钟源。可使用 OSCTUNE 寄存器中的校准值微调内部振荡器的频率。

## 12.2 振荡器校准

8 MHz 的内部振荡器经过出厂校准。出厂校准值位于 CALWD6 只读寄存器中。这些值必须从 CALWD6 寄存器读取并存储在 OSCCAL 寄存器中。有关读取程序存储器的步骤，请参见第17.0节“闪存程序存储器控制”。

**注：** 必须将 CALWD6 寄存器中的 FCAL<6:0> 位写入 OSCCAL 寄存器以校准内部振荡器。

## 12.3 用户模式下的频率调节

除了出厂校准外，基本频率也可以在用户的应用中调节。该频率调节功能允许用户偏离出厂校准的频率。用户可以通过写入 OSCTUNE 寄存器调节频率（见寄存器 12-1）。

**寄存器 12-1: OSCTUNE: 振荡器调节寄存器**

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0
bit 7							bit 0

**图注：**

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

bit 7-5	<b>未实现：</b> 读为 0
bit 4-0	<b>TUN&lt;4:0&gt;：</b> 频率调节位
	01111 = 最高频率
	01110 =
	•
	•
	•
	00001 =
	00000 = 中心频率。振荡器模块以校准后的频率运行。
	11111 =
	•
	•
	•
	10000 = 最低频率

# MCP19114/5

## 12.3.1 上电、唤醒和改变基本频率时振荡器的延时

在 OSCTUNE 寄存器用于偏移内部振荡器频率的应用中，不应要求内部振荡器的频率立即稳定下来。这种情况下，频率可朝新值逐渐偏移。该频移时间不足基本频率的 8 个周期。

上电时，如果使能上电延时定时器，则器件在上电期间保持复位状态。

从休眠模式唤醒或 POR 后，将调用一个约 10  $\mu$ s 的内部延时使存储器偏置稳定下来，之后才开始执行程序。

表 12-1: 与时钟源相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	83

图注： — = 未实现单元，读为 0。时钟源不使用阴影单元。

表 12-2: 与时钟源相关的配置字寄存器汇总

名称	Bit	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	寄存器所在页
CONFIG6	13:8	—	—	—	—	—	—	—	—	62
	7:0	—	FCAL6	FCAL5	FCAL4	FCAL3	FCAL2	FCAL1	FCAL0	

图注： — = 未实现单元，读为 0。时钟源不使用阴影单元。

## 13.0 复位

复位逻辑用于使 MCP19114/5 进入已知状态。复位的来源可通过器件状态位决定。

有多种方式可以复位这两个器件：

- 上电复位（Power-On Reset, POR）
- 过温复位（Overtemperature Reset, OT）
- $\overline{\text{MCLR}}$  复位
- WDT 复位
- 欠压复位（Brown-Out Reset, BOR）

要使  $V_{DD}$  稳定，可以使能可选的上电延时定时器来延长 POR 事件之后的复位时间。

一些寄存器不受任何复位条件的影响；在 POR 时它们的状态未知，而在任何其他复位时不变。以下情况下，大部分其他寄存器复位为“复位状态”：

- 上电复位
- $\overline{\text{MCLR}}$  复位
- 休眠期间的  $\overline{\text{MCLR}}$  复位
- WDT 复位
- 欠压复位

WDT（看门狗定时器）唤醒不会导致寄存器的复位方式与 WDT 复位相同，因为唤醒被视为恢复正常工作。TO 和 PD 位在不同的复位情形下会置 1 或者清零，如表 13-1 所示。软件可使用这些位确定复位的性质。关于所有寄存器的复位状态的完整说明，请参见表 13-2。

图 13-1 给出了片上复位电路的简化框图。

$\overline{\text{MCLR}}$  复位路径上有一个噪声滤波器，可用于检测并滤除小脉冲。关于脉宽规范，请参见第 5.0 节“数字电气特性”。

图 13-1: 片上复位电路的简化框图

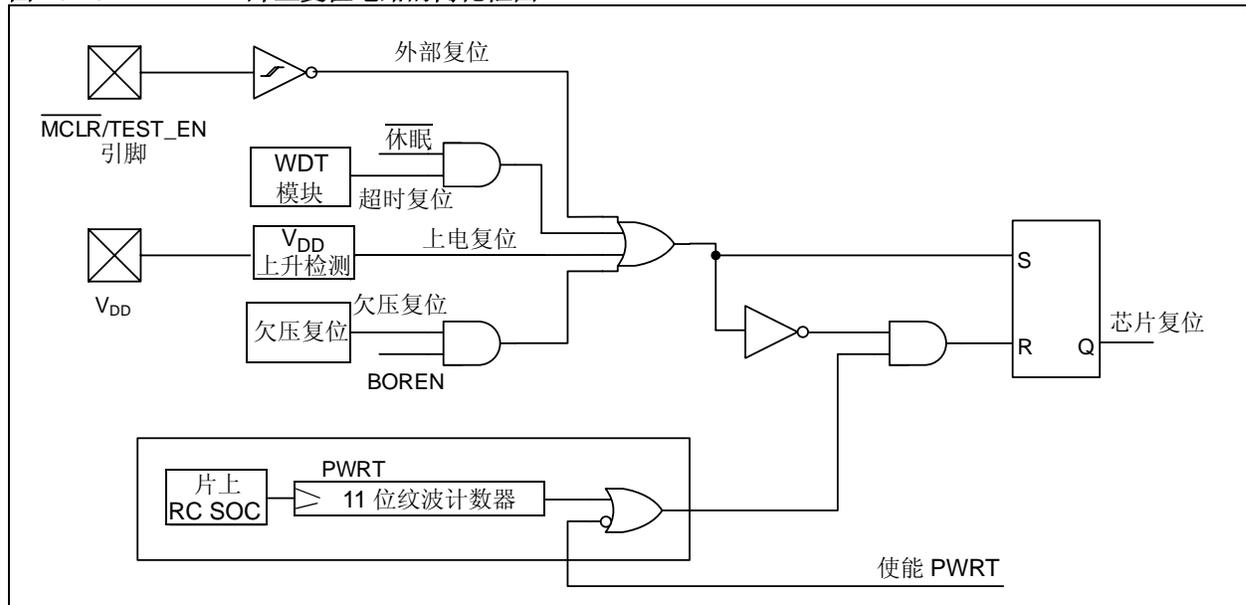


表 13-1: 各种情形下的超时

上电		从休眠模式 唤醒
$\overline{\text{PWRT}} = 0$	$\overline{\text{PWRT}} = 1$	
$T_{\text{PWRT}}$	—	—

# MCP19114/5

表 13-2: 状态 /PCON 位及其含义

POR	BOR	TO	PD	条件
0	x	1	1	上电复位
u	0	1	1	欠压复位
u	u	0	u	WDT 复位
u	u	0	0	WDT 唤醒
u	u	u	u	正常工作期间的 MCLR 复位
u	u	1	0	休眠期间的 MCLR 复位

图注: u = 不变, x = 未知

## 13.1 上电复位 (POR)

在  $V_{DD}$  上升到足以正常工作的电压之前, 片上 POR 电路将使芯片保持在复位状态。要使用 POR 功能, 只需将 MCLR 引脚通过一个电阻连接到  $V_{DD}$ 。这样可以省去产生上电复位通常所需的外部 RC 元件。

当器件开始正常工作 (退出复位状态) 时, 器件的工作参数 (即, 电压、频率和温度等) 必须得到满足, 以确保其正常工作。如果不满足这些条件, 那么器件必须保持在复位状态, 直到满足工作条件为止。

## 13.2 MCLR

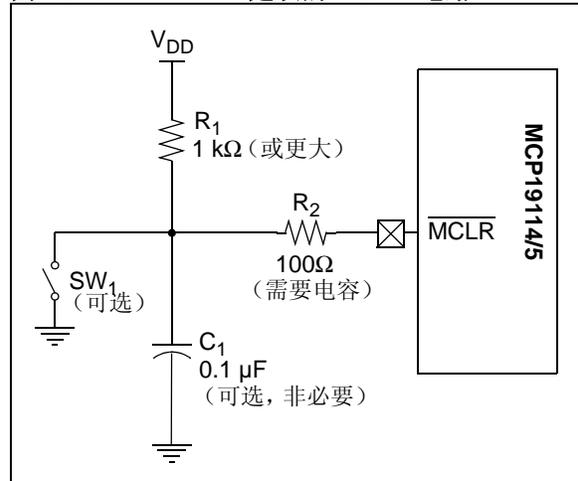
MCP19114/5 在 MCLR 复位路径上有一个噪声滤波器。该滤波器将检测并滤除小脉冲。

应当注意, WDT 复位未将 MCLR 引脚驱动为低电平。

如果施加到 MCLR 引脚上的电压超过其规范, 可导致在 ESD 事件期间产生 MCLR 复位以及超出器件规范的过大电流。因此, Microchip 建议不再将 MCLR 引脚直接连接到  $V_{DD}$ 。建议使用图 13-2 中所示的电阻电容 (Resistor-Capacitor, RC) 网络。

通过清零 CONFIG 寄存器中的 MCLRE 位使能内部 MCLR 选项。当 MCLRE = 0 时, 在内部会生成芯片的复位信号。当 MCLRE = 1 时, MCLR 引脚作为外部复位输入引脚。在此模式下, MCLR 引脚具有到  $V_{DD}$  的弱上拉。

图 13-2: 建议的 MCLR 电路

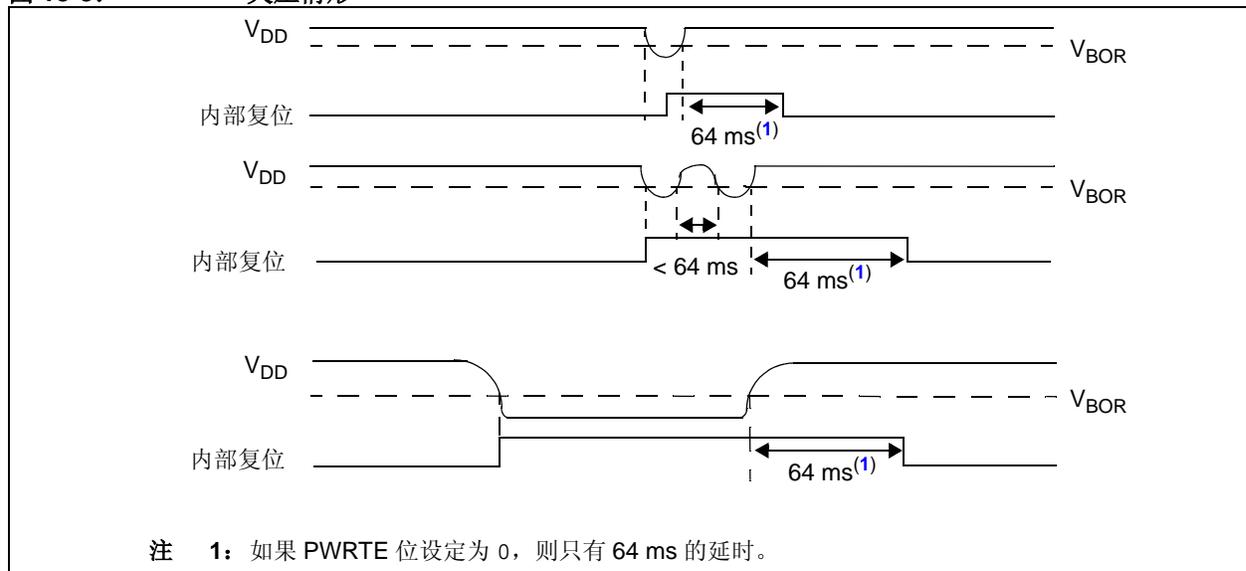


### 13.3 欠压复位 (BOR)

CONFIG 寄存器的 BOREN<8> 位用于使能或禁止 BOR 模式，如 CONFIG 寄存器中所定义。当  $V_{DD}$  降到  $V_{BOR}$  以下的时间大于最小值  $100\ \mu\text{s}$  时，将发生欠压复位。发生任何复位（上电复位、欠压复位、看门狗定时器复位等）时，器件都将保持复位状态，直到  $V_{DD}$  升高到  $V_{BOR}$  以上为止（见图 13-3）。如果使能了上电延时定时器，则它将在复位操作后开始工作，并使芯片在额外的  $64\text{ms}$  内保持复位。上电期间，建议使能 BOR 配置位来保持 MCU 处于复位状态（OSC 关闭，不执行代码），直到  $V_{DD}$  超过  $V_{BOR}$  阈值。用户可以通过清零 PWRTE 位，选择添加一个额外的  $64\text{ms}$  延时。此时， $V_{DD}$  电压电平高到足以只运行 MCU 功能；所有其他器件功能不能运行。这与  $V_{IN}$  的值无关，其通常为  $V_{DD} + V_{DROPOUT}$ 。掉电期间使能 BOR 时，当  $V_{DD}$  低于  $V_{BOR}$  阈值时，MCU 操作保持在复位状态。当禁止 BOR 或工作在休眠模式下时，在  $V_{DD}$  低于  $V_{POR}$  阈值时，POR 将保持器件处于复位状态。由于该器件的工作频率为  $F_{OSC} = 8\ \text{MHz}$  且处理器完全运行的电压为  $V_{DD} = 2\text{V}$ ，因此建议在上电和掉电期间始终使能 BOR。

**注：** 上电延时定时器由 CONFIG 寄存器中的 PWRTE 位使能。如果在上电延时定时器运行过程中， $V_{DD}$  电压降到  $V_{BOR}$  以下，芯片将重新回到欠压复位状态并且重新初始化上电延时定时器。一旦  $V_{DD}$  上升到  $V_{BOR}$  以上，上电延时定时器将执行一个  $64\ \text{ms}$  的复位。

图 13-3: 欠压情形



## 13.4 上电延时定时器 (PWRT)

上电延时定时器仅在上电时从POR复位开始提供64 ms (标称值)的固定延时。上电延时定时器随内部 RC 振荡器工作。只要PWRT处于活动状态,芯片就保持在复位状态。PWRT延时允许 $V_{DD}$ 上升到可接受的电平。CONFIG寄存器中的PWRTE位可禁止(如果已置1)或使能(如果已清零或设定)上电延时定时器。

由于存在以下变化,不同芯片的上电定时器延时也将各不相同:

- $V_{DD}$  变化
- 温度变化
- 工艺变化

**注:** 如果  $\overline{MCLR}$  引脚上的尖峰电压低于  $V_{SS}$ , 感应电流大于 80 mA, 可能会导致器件锁死。因此,当  $\overline{MCLR}$  引脚驱动为低电平时,应串连一个 50-100 $\Omega$  的电阻,而不是将该引脚直接连接到  $V_{SS}$ 。

上电延时定时器可用于在POR事件之后延迟器件执行。该定时器通常用于使 $V_{DD}$ 在允许器件开始运行之前先稳定下来。

上电延时定时器由CONFIG寄存器中的PWRTE位控制。

## 13.5 看门狗定时器 (WDT) 复位

如果固件未在超时周期内发出CLRWDT指令,看门狗定时器将产生复位。STATUS寄存器中的TO和PD位发生变化即指示发生了WDT复位。更多信息,请参见第16.0节“看门狗定时器(WDT)”。

## 13.6 启动序列

在POR恢复时,在器件开始执行程序之前必须发生以下事件:

- 上电延时定时器运行结束(如果使能)
- 振荡器起振定时器运行结束
- 必须释放 $\overline{MCLR}$ (如果使能)

总超时时间将根据PWRTE位的状态而变化。例如,在PWRTE位擦除(禁止PWRT)时,将完全没有超时。图13-4、13-5和13-6描述了超时序列。

由于超时产生自POR脉冲,因此如果 $\overline{MCLR}$ 保持足够长时间的低电平,超时将结束。随后将 $\overline{MCLR}$ 电平拉高后,器件将立即开始执行程序(见图13-5)。这对于测试或同步多个并行工作的MCP19114/5器件来说非常有用。

### 13.6.1 电源控制(PCON)寄存器

电源控制(PCON)寄存器(地址8Eh)有2个状态位,用来指示最后发生的复位类型。

图 13-4: 上电时的超时序列 (延时的  $\overline{MCLR}$ ): 情形 1

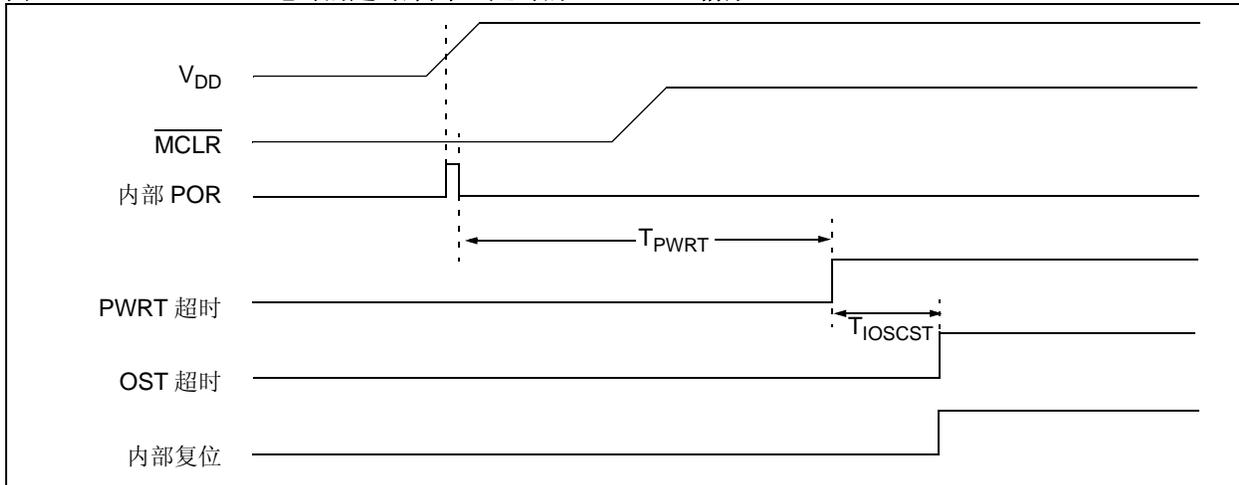


图 13-5: 上电时的超时序列 (延时的  $\overline{\text{MCLR}}$ ): 情形 2

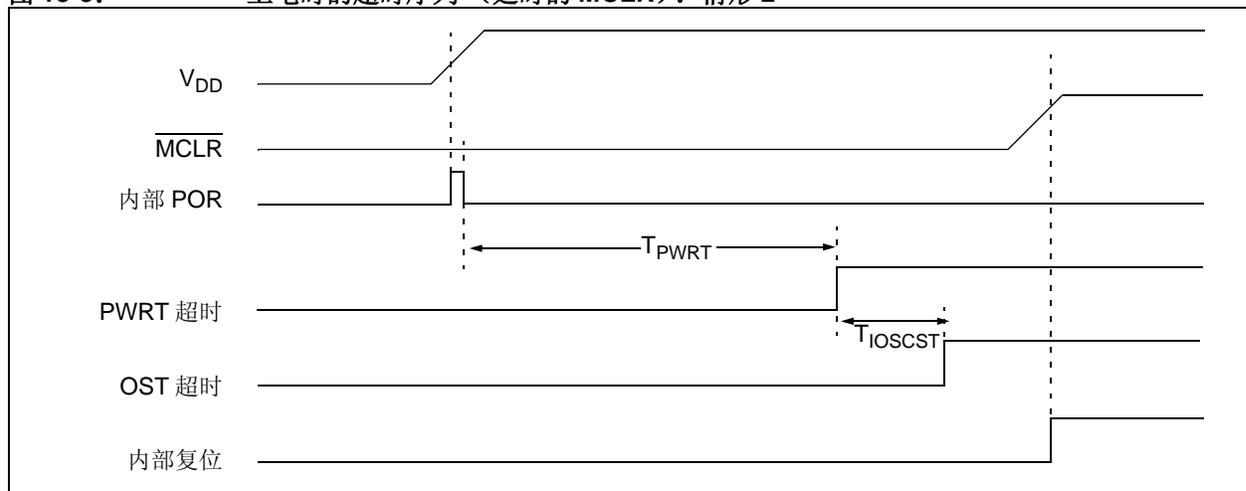
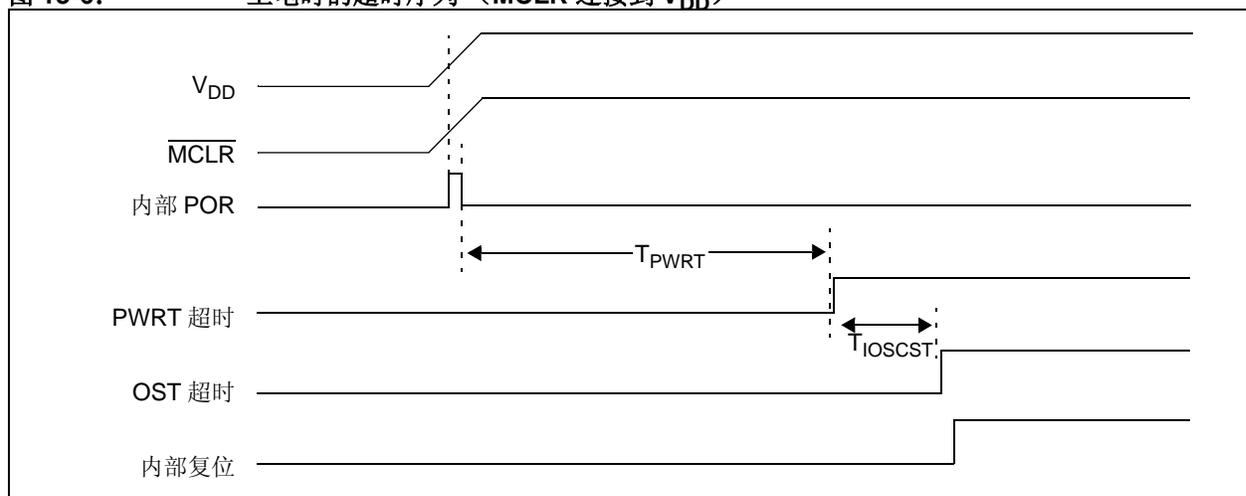


图 13-6: 上电时的超时序列 ( $\overline{\text{MCLR}}$  连接到  $V_{DD}$ )



## 13.7 确定复位原因

发生任何复位时, STATUS 和 PCON 寄存器中的多个位都会更新以指示复位原因。表 13-3 和 13-4 给出了这些寄存器的复位条件。

表 13-3: 复位状态位及其含义

$\overline{\text{POR}}$	$\overline{\text{BOR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	条件
0	x	1	1	上电复位
u	0	1	1	欠压复位
u	u	0	u	WDT 复位
u	u	0	0	通过 WDT 从休眠模式唤醒
u	u	1	0	通过中断从休眠模式唤醒
u	u	u	u	正常工作期间的 $\overline{\text{MCLR}}$ 复位
u	u	1	0	休眠期间的 $\overline{\text{MCLR}}$ 复位
0	u	0	x	不允许。 $\overline{\text{TO}}$ 在 POR 时置 1。
0	u	x	0	不允许。 $\overline{\text{PD}}$ 在 POR 时置 1。

# MCP19114/5

表 13-4: 特殊寄存器的复位条件 (注 2)

条件	程序计数器	STATUS 寄存器	PCON 寄存器
上电复位	0000h	0001 1xxx	---- --0u
欠压复位	0000	0001 1xxx	---- --u0
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0000h	000u uuuu	---- --uu
休眠期间的 $\overline{\text{MCLR}}$ 复位	0000h	0001 0uuu	---- --uu
WDT 复位	0000h	0000 uuuu	---- --uu
通过 WDT 从休眠模式唤醒	PC + 1	uuu0 0uuu	---- --uu
通过中断从休眠模式唤醒	PC + 1 <sup>(1)</sup>	uuu1 0uuu	---- --uu

图注: u = 不变, x = 未知, - = 未实现位, 读为 0。

注 1: 如果器件被中断唤醒且全局中断允许位 (GIE) 置 1, 则执行 PC + 1 后, 返回地址被压入堆栈且 PC 装入中断向量 (0004h)。

2: 如果状态位未实现, 则该位读为 0。

## 13.8 电源控制（PCON）寄存器

电源控制（PCON）寄存器包含用于区分以下复位的标志位：

- 上电复位（ $\overline{\text{POR}}$ ）
- 欠压复位（ $\overline{\text{BOR}}$ ）

PCON 寄存器位如寄存器 13-1 所示。

**寄存器 13-1: PCON: 电源控制寄存器**

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOR}}$
bit 7						bit 0	

图注：

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

bit 7-2            **未实现：** 读为 0  
 bit 1             **$\overline{\text{POR}}$ ：** 上电复位状态位  
                   1 = 未发生上电复位  
                   0 = 发生了上电复位（在上电复位发生后必须用软件置 1）  
 bit 0             **$\overline{\text{BOR}}$ ：** 欠压复位  
                   1 = 未发生欠压复位  
                   0 = 发生了欠压复位（在欠压复位发生后必须用软件置 1）

**表 13-5: 与复位相关的寄存器汇总**

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
PCON	—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	91
STATUS	IPR	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	71

图注： — = 未实现位，读为 0。复位未使用阴影单元。

注 1： 其他（非上电）复位包括正常工作时的  $\overline{\text{MCLR}}$  复位和看门狗定时器复位。

# MCP19114/5

---

注:

## 14.0 中断

MCP19114/5 具有多个中断源：

- 外部中断（INT 引脚）
- 电平变化中断（IOC）中断
- Timer0 溢出中断
- Timer1 溢出中断
- Timer2 匹配中断
- ADC 中断
- 系统输入欠压错误
- 系统输入过压错误
- SSP
- BCL
- 退饱和检测
- 栅极驱动 UVLO
- 捕捉 / 比较 1
- 捕捉 / 比较 2
- 过温

中断控制寄存器（INTCON）和外设中断请求寄存器（PIRx）用于记录标志位中的各个中断请求。INTCON 寄存器还具有独立和全局中断允许位。

INTCON 寄存器的全局中断允许位 GIE 用于允许（如果置 1）所有未屏蔽的中断或禁止（如果清零）所有中断。各个中断可以通过 INTCON 寄存器和 PIEx 寄存器中的相应允许位来禁止。GIE 在复位时清零。

当中断得到处理时，将自动进行以下操作：

- 将 GIE 清零以禁止任何其他中断。
- 将返回地址压入堆栈。
- 将 PC 装入 0004h。

中断服务程序（Interrupt Service Routine, ISR）中的固件应通过查询中断标志位来确定中断源。在退出 ISR 之前必须将中断标志位清零，以避免重复的中断。由于 GIE 位被清零，所以执行 ISR 期间发生的任何中断将会通过其中断标志位进行记录，但是不会使处理器重新定向到中断向量。

- 注 1:** 各个中断标志位的置 1 与其相应屏蔽位或 GIE 位的状态无关。
- 注 2:** 执行清零 GIE 位的指令时，将忽略下一个周期内待处理的任何中断。被忽略的中断在 GIE 位再次置 1 时仍处于待处理状态。

RETFIE 指令通过将前一个地址从堆栈中弹出、从影子寄存器恢复保存的内容并将 GIE 位置 1 来退出 ISR。

关于特定中断操作的更多信息，请参见其相应的外设章节。

### 14.1 中断响应延时

对于外部中断事件（例如，INT 引脚或 PORTGPx 变化中断），中断响应延时将是 3 个或 4 个指令周期。精确的响应延时取决于中断事件发生的时间（见图 14-2）。对于单周期或双周期指令，响应延时相同。

### 14.2 GPA2/INT 中断

GPA2/INT 引脚上的外部中断为边沿触发：如果 OPTION\_REG 寄存器的 INTEDG 位置 1，则为上升沿触发；如果 INTEDG 位清零，则为下降沿触发。当 GPA2/INT 引脚上出现一个有效边沿时，INTCON 寄存器的 INTF 位置 1。可以通过清零 INTCON 寄存器的 INTE 控制位来禁止该中断。在重新允许该中断前，必须在中断服务程序中先用软件将 INTF 位清零。如果 INTE 位在进入休眠模式之前置 1，则 GPA2/INT 中断可将处理器从休眠状态唤醒。关于休眠的详细信息，请参见第 15.0 节“掉电模式（休眠）”；有关通过 GPA2/INT 中断从休眠状态唤醒的时序，请参见第 15.1 节“从休眠模式唤醒”。

**注：** ANSEL 寄存器必须经过初始化，以将模拟通道配置为数字输入。配置为模拟输入的引脚将读为 0，且无法产生中断。

# MCP19114/5

图 14-1: 中断逻辑

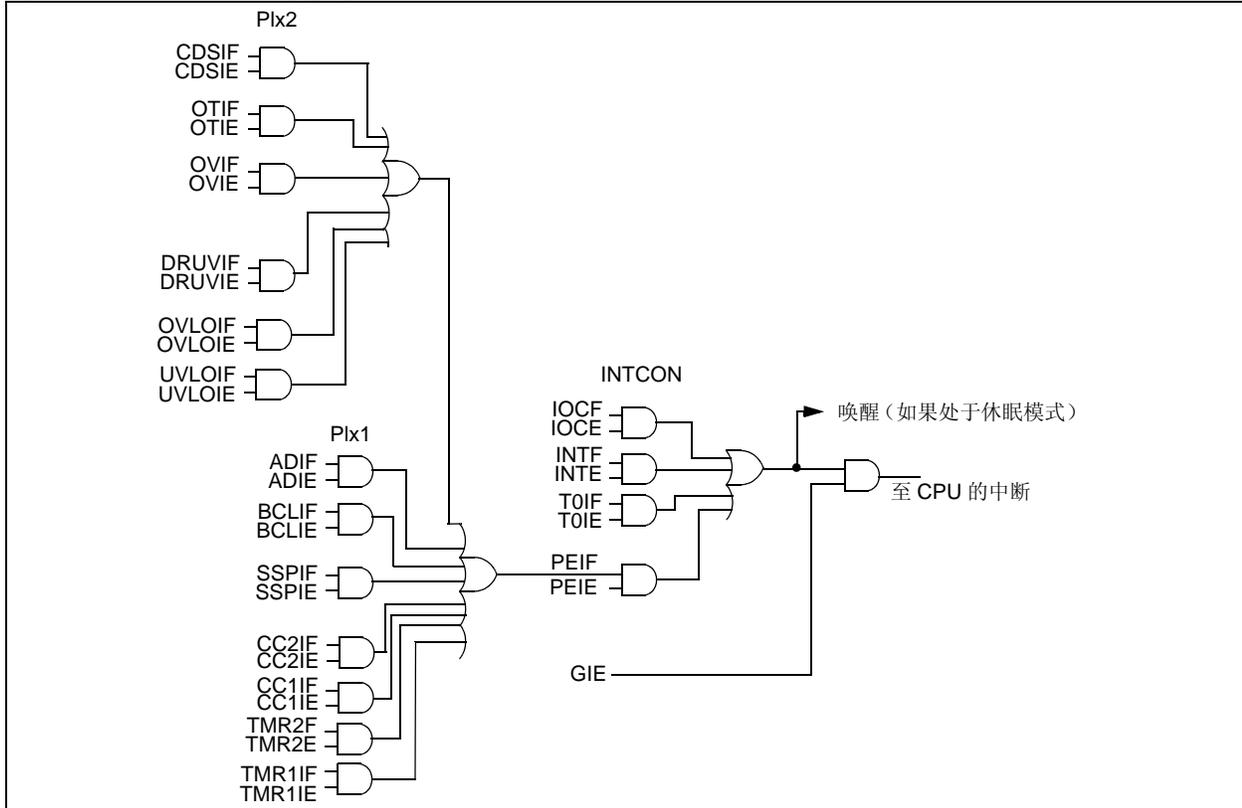
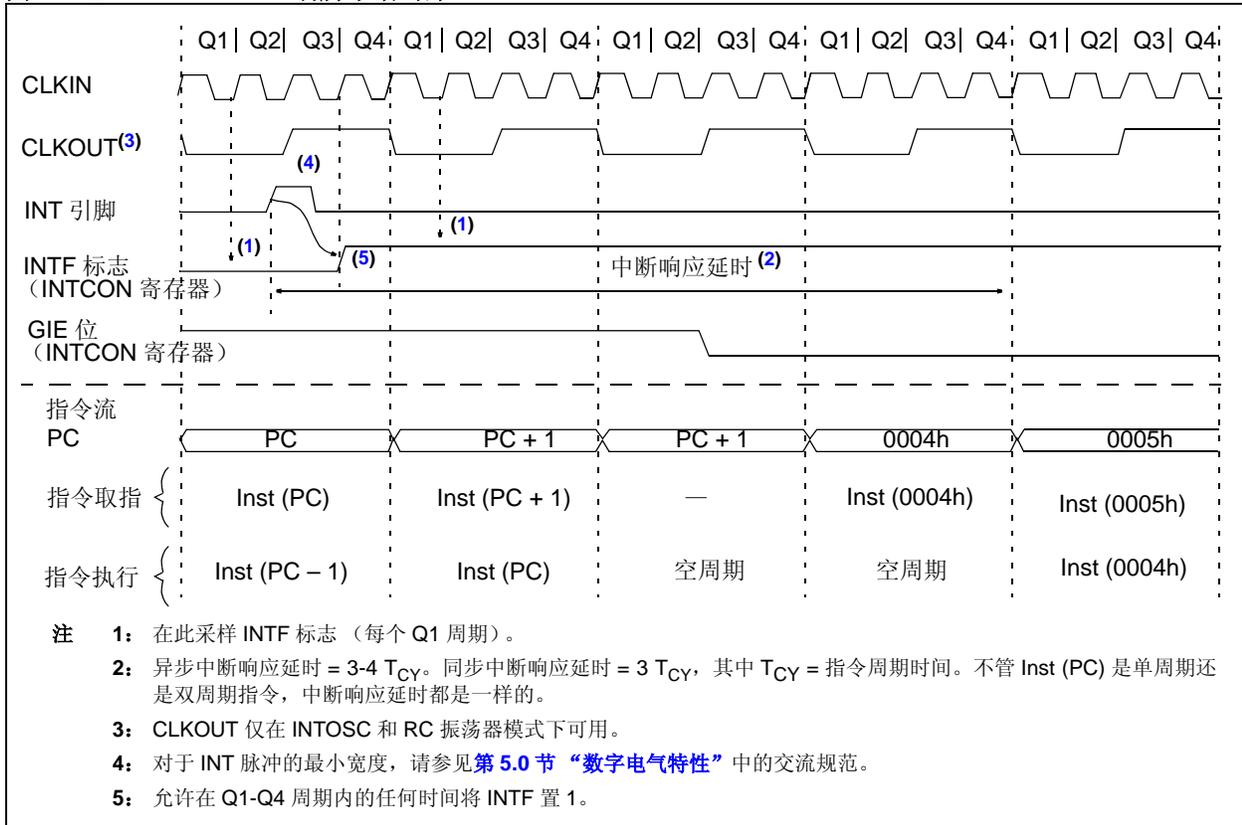


图 14-2: INT 引脚中断时序



### 14.3 中断控制寄存器

#### 14.3.1 INTCON 寄存器

INTCON 寄存器是可读写寄存器，其中包含 TMR0 寄存器溢出、电平变化中断和外部 INT 引脚中断的各个允许位和标志位。

**注：** 当中断条件产生时，无论 INTCON 寄存器的相应允许位或全局允许位 GIE 的状态如何，中断标志位都将置 1。用户软件应在允许一个中断前，先将相应的中断标志位清零。

#### 寄存器 14-1: INTCON: 中断控制寄存器

R/W-0	R/W-x						
GIE	PEIE	TOIE	INTE	IOCE	TOIF	INTF	IOCF
bit 7							bit 0

#### 图注:

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

bit 7	<b>GIE:</b> 全局中断允许位 1 = 允许所有未屏蔽的中断 0 = 禁止所有中断
bit 6	<b>PEIE:</b> 外设中断允许位 1 = 允许所有未屏蔽的外设中断 0 = 禁止所有外设中断
bit 5	<b>TOIE:</b> TMR0 溢出中断允许位 1 = 允许 TMR0 中断 0 = 禁止 TMR0 中断
bit 4	<b>INTE:</b> INT 外部中断允许位 1 = 允许 INT 外部中断 0 = 禁止 INT 外部中断
bit 3	<b>IOCE:</b> 电平变化中断允许位 <sup>(1)</sup> 1 = 允许电平变化中断 0 = 禁止电平变化中断
bit 2	<b>TOIF:</b> TMR0 溢出中断标志位 <sup>(2)</sup> 1 = TMR0 寄存器已溢出 (必须用软件清零) 0 = TMR0 寄存器未溢出
bit 1	<b>INTF:</b> 外部中断标志位 1 = 发生了外部中断 (必须用软件清零) 0 = 未发生外部中断
bit 0	<b>IOCF:</b> 电平变化中断标志位 1 = 至少一个电平变化中断引脚的状态发生改变 0 = 没有一个电平变化中断引脚的状态发生改变

**注 1:** 还必须使能 IOCx 寄存器。

**2:** 当 TMR0 计满返回时，TOIF 位置 1。TMR0 在复位时不变，而且应在清零 TOIF 位之前初始化。

# MCP19114/5

## 14.3.1.1 PIE1 寄存器

PIE1 寄存器包含外设中断允许位，如寄存器 14-2 所示。

**注 1:** 要允许任何外设中断，必须将 INTCON 寄存器的 PEIE 位置 1。

**寄存器 14-2: PIE1: 外设中断允许寄存器 1**

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 7      **未实现:** 读为 0
- bit 6      **ADIE:** ADC 中断允许位  
1 = 允许 ADC 中断  
0 = 禁止 ADC 中断
- bit 5      **BCLIE:** MSSP 总线冲突中断允许位  
1 = 允许 MSSP 总线冲突中断  
0 = 禁止 MSSP 总线冲突中断
- bit 4      **SSPIE:** 同步串行端口 (MSSP) 中断允许位  
1 = 允许 MSSP 中断  
0 = 禁止 MSSP 中断
- bit 3      **CC2IE:** 捕捉 2/ 比较 2 中断允许位  
1 = 允许捕捉 2/ 比较 2 中断  
0 = 禁止捕捉 2/ 比较 2 中断
- bit 2      **CC1IE:** 捕捉 1/ 比较 1 中断允许位  
1 = 允许捕捉 1/ 比较 1 中断  
0 = 禁止捕捉 1/ 比较 1 中断
- bit 1      **TMR2IE:** Timer2 中断允许位  
1 = 允许 Timer2 中断  
0 = 禁止 Timer2 中断
- bit 0      **TMR1IE:** Timer1 中断允许位  
1 = 允许 Timer1 中断  
0 = 禁止 Timer1 中断

## 14.3.1.2 PIE2 寄存器

PIE2 寄存器包含外设中断允许位，如寄存器 14-3 所示。

**注 1:** 要允许任何外设中断，必须将 INTCON 寄存器的 PEIE 位置 1。

**寄存器 14-3: PIE2: 外设中断允许寄存器 2**

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CDSIE	—	—	OTIE	OVIE	DRUVIE	OVLOIE	UVLOIE
bit 7							bit 0

**图注:**

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 7      **CDSIE:** 退饱和和检测中断允许位  
1 = 允许 DESAT 检测中断  
0 = 禁止 DESAT 检测中断
- bit 6-5    **未实现:** 读为 0
- bit 4      **OTIE:** 过温中断允许位  
1 = 允许过温中断  
0 = 禁止过温中断
- bit 3      **OVIE:**  $V_{OUT}$  过压中断允许位  
1 = 允许 OV 中断  
0 = 禁止 OV 中断
- bit 2      **DRUVIE:** 栅极驱动欠压锁定中断允许位  
1 = 允许栅极驱动 UVLO 中断  
0 = 禁止栅极驱动 UVLO 中断
- bit 1      **OVLOIE:**  $V_{IN}$  过压锁定中断允许位  
1 = 允许 OVLO 中断  
0 = 禁止 OVLO 中断
- bit 0      **UVLOIE:**  $V_{IN}$  欠压锁定中断允许位  
1 = 允许 UVLO 中断  
0 = 禁止 UVLO 中断

# MCP19114/5

## 14.3.1.3 PIR1 寄存器

PIR1 寄存器包含外设中断标志位，如寄存器 14-4 所示。

**注 1:** 当中断条件产生时，无论 INTCON 寄存器的相应允许位或全局允许位 GIE 的状态如何，中断标志位都将置 1。用户软件应在允许一个中断前，先将相应的中断标志位清零。

寄存器 14-4: PIR1: 外设中断标志寄存器 1

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF
bit 7							bit 0

**图注:**

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

- bit 7            **未实现:** 读为 0
- bit 6            **ADIF:** ADC 中断标志位  
                  1 = ADC 转换完成  
                  0 = ADC 转换未完成或尚未开始
- bit 5            **BCLIF:** MSSP 总线冲突中断标志位  
                  1 = 中断等待响应  
                  0 = 没有等待响应的中断
- bit 4            **SSPIF:** 同步串行端口 (MSSP) 中断标志位  
                  1 = 中断等待响应  
                  0 = 没有等待响应的中断
- bit 3            **CC2IF:** 捕捉 2/ 比较 2 中断标志位  
                  1 = 发生捕捉或比较  
                  0 = 未发生捕捉或比较
- bit 2            **CC1IF:** 捕捉 1/ 比较 1 中断标志位  
                  1 = 发生捕捉或比较  
                  0 = 未发生捕捉或比较
- bit 1            **TMR2IF:** Timer2 与 PR2 匹配中断标志位  
                  1 = Timer2 与 PR2 发生匹配 (必须用软件清零)  
                  0 = Timer2 与 PR2 未发生匹配
- bit 0            **TMR1IF:** Timer1 中断标志位  
                  1 = Timer1 已计满返回 (必须用软件清零)  
                  0 = Timer1 未计满返回

## 14.3.1.4 PIR2 寄存器

PIR2寄存器包含外设中断标志位，如寄存器14-3所示。

**注 1:** 当中断条件产生时，无论INTCON寄存器的相应允许位或全局允许位GIE的状态如何，中断标志位都将置1。用户软件应在允许一个中断前，先将相应的中断标志位清零。

**寄存器 14-5: PIR2: 外设中断标志寄存器 2**

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CDSIF	—	—	OTIF	OVIF	DRUVIF	OVLOIF	UVLOIF
bit 7							bit 0

**图注:**

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

- bit 7            **CDSIF:** DESAT 检测比较器模块中断标志位  
 1 = 中断等待响应  
 0 = 没有等待响应的中断
- bit 6-5        **未实现:** 读为 0
- bit 4            **OTIF:** 过温中断标志位  
 1 = 发生了过温事件  
 0 = 未发生过温事件
- bit 3            **OVIF:** 过压中断标志位  
OVINTP 位置 1 时:  
 1 = 检测到  $V_{OUT}$  未过压至过压边沿  
 0 = 未检测到  $V_{OUT}$  未过压至过压边沿  
OVINTN 位置 1 时:  
 1 = 检测到  $V_{OUT}$  过压至未过压边沿  
 0 = 未检测到  $V_{OUT}$  过压至未过压边沿
- bit 2            **DRUVIF:** 栅极驱动欠压锁定中断标志位  
 1 = 发生了栅极驱动欠压锁定  
 0 = 未发生栅极驱动欠压锁定
- bit 1            **OVLOIF:**  $V_{IN}$  过压锁定中断标志位  
OVLOINTP 位置 1 时:  
 1 = 检测到  $V_{IN}$  未过压至  $V_{IN}$  过压边沿  
 0 = 未检测到  $V_{IN}$  未过压至  $V_{IN}$  过压边沿  
OVLOINTN 位置 1 时:  
 1 = 检测到  $V_{IN}$  过压至  $V_{IN}$  未过压边沿  
 0 = 未检测到  $V_{IN}$  过压至  $V_{IN}$  未过压边沿
- bit 0            **UVLOIF:**  $V_{IN}$  欠压锁定中断标志位  
UVLOINTP 位置 1 时:  
 1 = 检测到  $V_{IN}$  未欠压至  $V_{IN}$  欠压边沿  
 0 = 未检测到  $V_{IN}$  未欠压至  $V_{IN}$  欠压边沿  
UVLOINTN 位置 1 时:  
 1 = 检测到  $V_{IN}$  欠压至  $V_{IN}$  未欠压边沿  
 0 = 未检测到  $V_{IN}$  欠压至  $V_{IN}$  未欠压边沿

# MCP19114/5

表 14-1: 与中断相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	95
OPTION_REG	RAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	78
PIE1	—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	96
PIE2	CDSIE	—	—	OTIE	OVIE	DRUVIE	OVLOIE	UVLOIE	97
PIR1	—	ADIF	BCLIF	SSPIF	—	—	TMR2IF	TMR1IF	98
PIR2	CDSIF	—	—	OTIF	OVIF	DRUVIF	OVLOIF	UVLOIF	99

图注: — = 未实现单元, 读为 0。中断不使用阴影单元。

## 14.4 中断期间的现场保护

中断期间, 仅将返回的 PC 值保存在堆栈中。通常, 用户可能想要在中断期间保存主要寄存器 (例如, W 寄存器和 STATUS 寄存器)。这必须由软件实现。

临时保持寄存器 W\_TEMP 和 STATUS\_TEMP 应位于 GPR 的后 16 字节 (见图 10-3)。这 16 个单元是所有存储区共有的, 不需要分区。这使得现场保护和恢复操作更简单。例 14-1 中显示的代码可用于:

- 存储 W 寄存器
- 存储 STATUS 寄存器
- 执行 ISR 代码
- 恢复 STATUS 寄存器 (以及存储区选择位寄存器)
- 恢复 W 寄存器

**注:** MCP19114/5 不需要保存 PCLATH。但是, 如果在 ISR 和主代码中使用经计算的 GOTO, 则必须在 ISR 中保存和恢复 PCLATCH。

例 14-1: 将 STATUS 和 W 寄存器的值保存在 RAM 中

```
MOVWF  W_TEMP           ;Copy W to TEMP register
SWAPF  STATUS,W         ;Swap status to be saved into W
                        ;Swaps are used because they do not affect the status bits
MOVWF  STATUS_TEMP      ;Save status to bank zero STATUS_TEMP register
:
:(ISR)                  ;Insert user code here
:
SWAPF  STATUS_TEMP,W    ;Swap STATUS_TEMP register into W
                        ;(sets bank to original state)
MOVWF  STATUS           ;Move W into STATUS register
SWAPF  W_TEMP,F         ;Swap W_TEMP
SWAPF  W_TEMP,W         ;Swap W_TEMP into W
```

## 15.0 掉电模式（休眠）

器件通过执行 SLEEP 指令进入掉电模式。

进入休眠模式时，存在以下条件：

1. WDT 将清零但是保持运行（如果使能了在休眠期间工作）。
2. STATUS 寄存器的  $\overline{PD}$  位清零。
3. STATUS 寄存器的  $\overline{TO}$  位置 1。
4. 禁止 CPU 时钟。
5. ADC 无法运行，因为缺少 4V LDO 电源 ( $AV_{DD}$ )。
6. I/O 端口保持执行 SLEEP 指令之前的状态（驱动为高电平、低电平或高阻态）。
7. WDT 之外的复位不受休眠模式影响。
8. 模拟电路电源 ( $AV_{DD}$ ) 在休眠模式下被移除。
9. POR 电平更改为 1.2V 典型值。

关于外设 在休眠期间工作的更多详细信息，请参见各个章节。

要最大程度地降低电流消耗，应考虑以下条件：

- I/O 引脚不应悬空。
- I/O 引脚的外部电路灌电流。
- I/O 引脚的内部电路拉电流。
- 从带内部弱上拉的引脚汲取的电流。
- 使用 Timer1 振荡器的模块。

为了避免输入引脚悬空而引入开关电流，应在外部将高阻输入的 I/O 引脚拉为  $V_{DD}$  或 GND。

SLEEP 指令移除模拟电路电源。在休眠模式下  $AV_{DD}$  被关闭，以将电流消耗降至最低，并保持典型值为 30  $\mu A$  的关断电流。5V LDO ( $V_{DD}$ ) 电压在休眠模式下降至 2.5V-3V。外部从 5V LDO ( $V_{DD}$ ) 汲取的电流应不超过 200  $\mu A$ 。在休眠模式下，外部负载汲取的电流超过 200  $\mu A$  时，存在加重  $V_{DD}$  电压负担并触发 POR 的风险。休眠模式期间发生 POR 事件会将器件从休眠模式唤醒。执行 SLEEP 指令不会改变模拟电路的使能状态。

## 15.1 从休眠模式唤醒

可以通过以下任一事件将器件从休眠模式唤醒：

1.  $\overline{MCLR}$  引脚上的外部复位输入（如果使能）
2. POR 复位
3. 看门狗定时器（如果使能）
4. 任何外部中断
5. 能够在休眠期间运行的外设产生的中断（更多信息，请参见各个外设）

前 2 个事件会使器件复位。后 3 个事件被认为是程序执行的延续。要确定是发生器件复位还是唤醒事件，请参见第 13.7 节“确定复位原因”。

以下外设中断可将器件从休眠模式唤醒：

1. 电平变化中断
2. INT 引脚的外部中断

当执行 SLEEP 指令时，下一条指令 ( $PC + 1$ ) 被预先取出。如果希望通过中断事件唤醒器件，则必须置 1 相应的中断允许位。唤醒与 GIE 位的状态无关。如果 GIE 位清零，器件将继续执行 SLEEP 指令后的指令。如果 GIE 位置 1，器件先执行 SLEEP 指令后的指令，然后调用中断服务程序。如果不想执行 SLEEP 指令后的指令，用户应该在 SLEEP 指令后面放置一条 NOP 指令。

器件从休眠模式唤醒时，WDT 清零，与唤醒的原因无关。

# MCP19114/5

## 15.1.1 使用中斷喚醒

当禁止全局中断（GIE 清零），并且任一中断源的中断允许位和中断标志位均置 1，将发生下列某一事件：

- 如果在执行 SLEEP 指令之前发生中断
  - SLEEP 指令将作为 NOP 执行
  - WDT 和 WDT 预分频器不会清零
  - STATUS 寄存器的  $\overline{TO}$  位不会置 1
  - STATUS 寄存器的  $\overline{PD}$  位不会清零

- 如果在执行 SLEEP 指令期间或之后发生中断
  - SLEEP 指令将完全执行
  - 器件将立即从休眠模式唤醒
  - WDT 和 WDT 预分频器将清零
  - STATUS 寄存器的  $\overline{TO}$  位将置 1
  - STATUS 寄存器的  $\overline{PD}$  位将清零

即使在执行 SLEEP 指令之前，检查过标志位，这些标志位也可能在 SLEEP 指令执行完毕之前被置 1。要确定是否执行了 SLEEP 指令，可以测试  $\overline{PD}$  位。如果  $\overline{PD}$  位置 1，则说明 SLEEP 指令被作为一条 NOP 指令执行了。

图 15-1: 通过中断从休眠模式唤醒

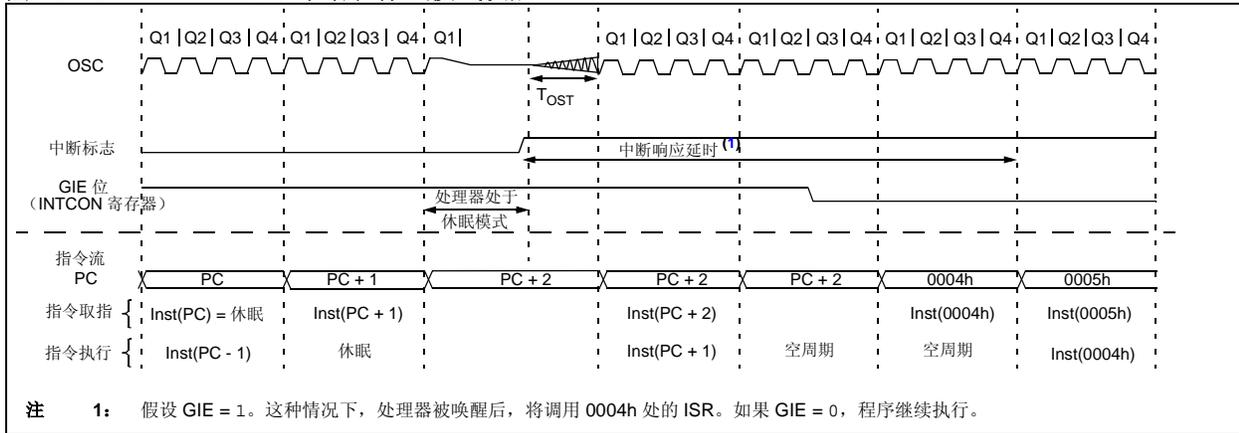


表 15-1: 与掉电模式相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	95
IOCA	IOCA7	IOCA6	IOCA5	—	IOCA3	IOCA2	IOCA1	IOCA0	122
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	—	—	IOCB1	IOCB0	122
PIE1	—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	96
PIE2	CDSIE	—	—	OTIE	OVIE	DRUVIE	OVLOIE	UVLOIE	97
PIR1	—	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	98
PIR2	CDSIF	—	—	OTIF	OVIF	DRUVIF	OVLOIF	UVLOIF	99
STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	71

图注: — = 未实现，读为 0。掉电模式不使用阴影单元。

## 16.0 看门狗定时器 (WDT)

看门狗定时器是自由运行的定时器。通过将 CONFIG 寄存器 (默认设置) 的 WDTE 位置 1 来使能 WDT。

正常工作期间, WDT 超时会产生器件复位。如果器件处于休眠模式, WDT 超时会使器件唤醒并继续正常工作。

清零 CONFIG 寄存器的 WDTE 位可永久禁止 WDT。更多信息, 请参见第 11.1 节“配置字”。

### 16.1 看门狗定时器 (WDT) 操作

正常工作期间, WDT 超时会产生器件复位。如果器件处于休眠模式, WDT 超时会使器件唤醒并继续正常工作; 这也称为 WDT 唤醒。清零 WDTE 配置位可永久禁止 WDT。

后分频器的分配完全受软件控制, 并且可在程序执行期间进行更改。

### 16.2 WDT 周期

WDT 的标称超时周期为 18 ms (无预分频器)。各器件的超时周期随温度、 $V_{DD}$  和工艺的不同而有所不同 (见表 5-3)。如果需要较长的超时周期, 则可在软件控制下通过写入 OPTION\_REG 寄存器来向 WDT 分配一个分频比最高为 1:128 的预分频器。因此, 可实现最长 2.3 秒的超时周期。

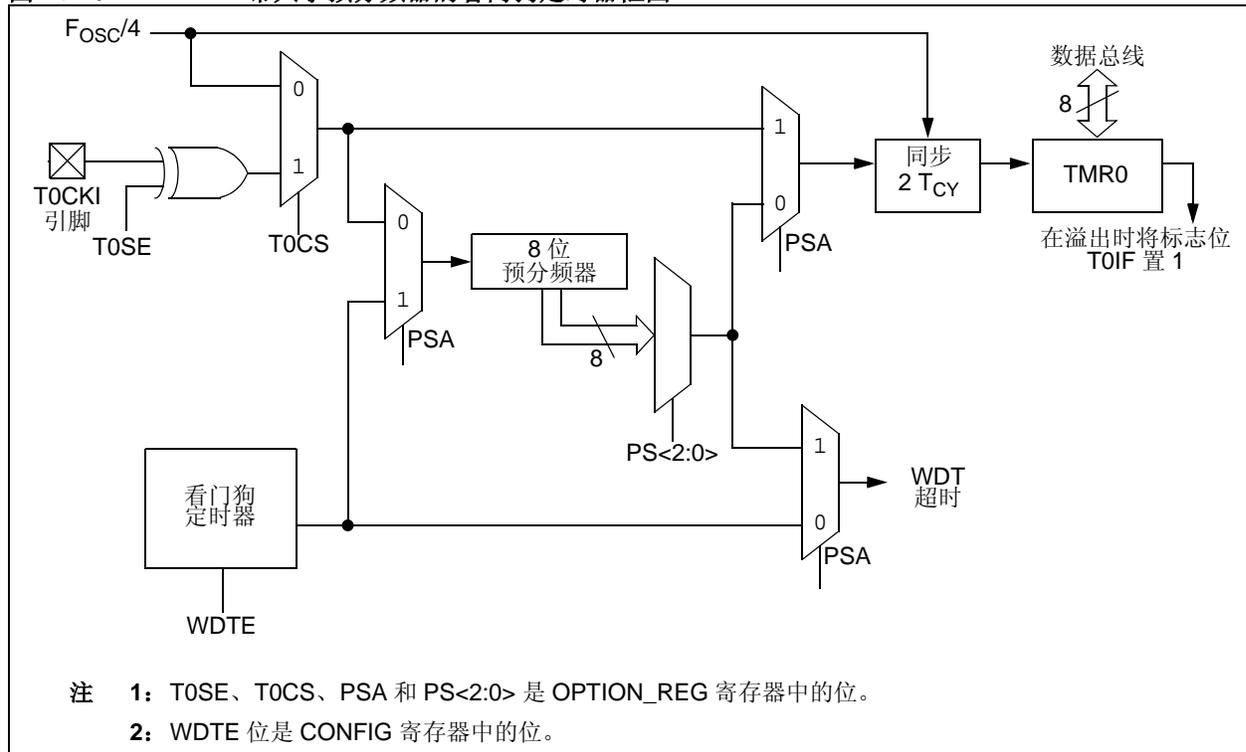
如果将 CLRWDT 和 SLEEP 指令分配给 WDT, 这些指令会清零 WDT 和预分频器, 并防止 WDT 超时和产生器件复位。

如果看门狗定时器超时, STATUS 寄存器中的  $\overline{TO}$  位将被清零。

### 16.3 WDT 编程注意事项

在最坏的情况下 (即,  $V_{DD}$  = 最小值, 温度 = 最大值, 最大 WDT 预分频比), 发生 WDT 超时之前可能需要几秒钟。

图 16-1: 带共享预分频器的看门狗定时器框图



# MCP19114/5

表 16-1: WDT 状态

条件	WDT
WDTE = 0	已清零
CLRWDT 命令	
退出休眠模式	

表 16-2: 与看门狗定时器相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
OPTION_REG	$\overline{\text{RAPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	78

图注: 看门狗定时器不使用阴影单元。

注 1: 关于 CONFIG 寄存器中所有位的操作, 请参见寄存器 11-1。

表 16-3: 与看门狗定时器相关的配置字寄存器汇总

名称	Bit	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	寄存器所在页
CONFIG	13:8	—	—	DBGEN	—	WRT1	WRT0	—	BOREN	81
	7:0	—	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE	—	—	—	

图注: — = 未实现单元, 读为 1。看门狗定时器不使用阴影单元。

## 17.0 闪存程序存储器控制

闪存程序存储器在正常工作期间（整个  $V_{IN}$  范围内）是可读写的。该存储器并未直接映射到寄存器文件空间中，而是通过特殊功能寄存器来间接寻址（见寄存器 17-1 至 17-5）。共有 6 个 SFR 用于读写此存储器：

- PMCON1
- PMCON2
- PMDATL
- PMDATH
- PMADRL
- PMADRH

当与程序存储器模块连接时，PMDATL 和 PMDATH 寄存器形成双字节字，存放要读 / 写的 14 位数据，而 PMADRL 和 PMADRH 寄存器形成双字节字，存放被访问的闪存单元的 13 位地址。这些器件具有 4K 字的程序闪存，地址范围为 0000h 到 0FFFh。

该程序存储器允许单字读操作和四字写操作。四字写操作会自动擦除目标存储单元的行并写入新数据（在写入前擦除）。

写入时间由片上定时器控制。写入 / 擦除电压由片上电荷泵产生，此电荷泵能在器件的电压范围内正常工作（针对字节或字操作）。

当器件受代码保护时，CPU 仍可继续读写闪存程序存储器。

根据闪存程序存储器使能（WRT<1:0>）位的设置，器件不一定能写入程序存储器的某些块，但总能读取程序存储器。

使能闪存程序存储器代码保护位（ $\overline{CP}$ ）后，程序存储器将受代码保护，器件编程器（ICSP）将无法访问数据或程序存储器。

## 17.1 PMADRH 和 PMADRL 寄存器

PMADRH 和 PMADRL 寄存器可寻址最高 4K 字的程序存储器。

当选择程序地址值时，地址的高字节（Most Significant Byte, MSB）写入 PMADRH 寄存器，而低字节（Least Significant Byte, LSB）写入 PMADRL 寄存器。

## 17.2 PMCON1 和 PMCON2 寄存器

PMCON1 是用于数据程序存储器访问的控制寄存器。

控制位 RD 和 WR 分别启动读操作和写操作。用软件只能将这些位置 1 而无法清零。在读操作或写操作完成后，由硬件将它们清零。由于无法用软件将 WR 位清零，可以避免因意外而过早地终止写操作。

当 WREN 位置 1 时，允许进行写操作。上电时，WREN 位被清零。

当校准字单元中存储的校准位需要传送到 SFR 微调寄存器时，CALSEL 位允许用户读取测试存储器中的单元。CALSEL 位仅用于读操作，如果在 CALSEL = 1 时尝试写操作，则不会执行写入。

PMCON2 不是物理寄存器。读 PMCON2 时得到的将都是 0。PMCON2 寄存器专用于闪存存储器写序列。

# MCP19114/5

## 17.3 闪存程序存储器控制寄存器

寄存器 17-1: **PMDATL: 程序存储器数据低字节寄存器**

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

图注:

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

bit 7-0                **PMDATL<7:0>**: 用于程序存储器读 / 写操作的数据低 8 位

寄存器 17-2: **PMADRL: 程序存储器地址低字节寄存器**

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

图注:

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

bit 7-0                **PMADRL<7:0>**: 用于程序存储器读 / 写操作的地址低 8 位

寄存器 17-3: **PMDATH: 程序存储器数据高字节寄存器**

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

图注:

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

bit 7-6                **未实现**: 读为 0

bit 5-0                **PMDATH<5:0>**: 程序存储器的数据高 6 位

## 寄存器 17-4: PMADRH: 程序存储器地址高字节寄存器

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	PMADRH<3:0>			
bit 7							bit 0

### 图注:

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

bit 7-4                      **未实现:** 读为 0  
 bit 3-0                      **PMADRH<3:0>:** 用于程序存储器读操作的高 4 位地址或高位。

## 寄存器 17-5: PMCON1: 程序存储器控制寄存器 1

U-1	R/W-0	U-0	U-0	U-0	R/W-0	R/S-0	R/S-0
—	CALSEL	—	—	—	WREN	WR	RD
bit 7							bit 0

### 图注:

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

bit 7                      **未实现:** 读为 1  
 bit 6                      **CALSEL:** 程序存储器校准空间选择位  
                             1 = 选择仅用于读操作的测试存储器区域 (用于装载校准微调寄存器)  
                             0 = 选择用于读操作的用户区域  
 bit 5-3                      **未实现:** 读为 0  
 bit 2                      **WREN:** 程序存储器写使能位  
                             1 = 允许执行写操作  
                             0 = 禁止对 EEPROM 执行写操作  
 bit 1                      **WR:** 写控制位  
                             1 = 启动对程序存储器的写周期。(此位在写操作完成后由硬件清零。用软件只能将 WR 位置 1, 但不能清零。)  
                             0 = 对闪存存储器的写周期完成  
 bit 0                      **RD:** 读控制位  
                             1 = 启动程序存储器的读操作。(读操作占用一个周期。RD 由硬件清零; 用软件只能将 RD 位置 1, 但不能清零。)  
                             0 = 不启动闪存存储器的读操作

# MCP19114/5

## 17.3.1 读闪存程序存储器

要读取程序存储器单元，用户必须将两字节的地址写入 PMADRL 和 PMADRH 寄存器，然后将控制位 RD (PMCON1 寄存器中的 bit 0) 置 1。一旦将读控制位置 1，闪存程序存储器控制器将使用之后的第二个指令周期读取数据。这会导致紧跟 BSF PMCON1,RD 指令后的第二条指令被忽略。在紧接着的下一个周期，PMDATL 和 PMDATH 寄存器中就有数据了，可在随后的指令中将该数据作为两个字节读取。PMDATL 和 PMDATH 寄存器将把此值保存至下一次读操作或用户向该单元写入数据时（写操作期间）为止。

### 例 17-1: 闪存程序读操作

```

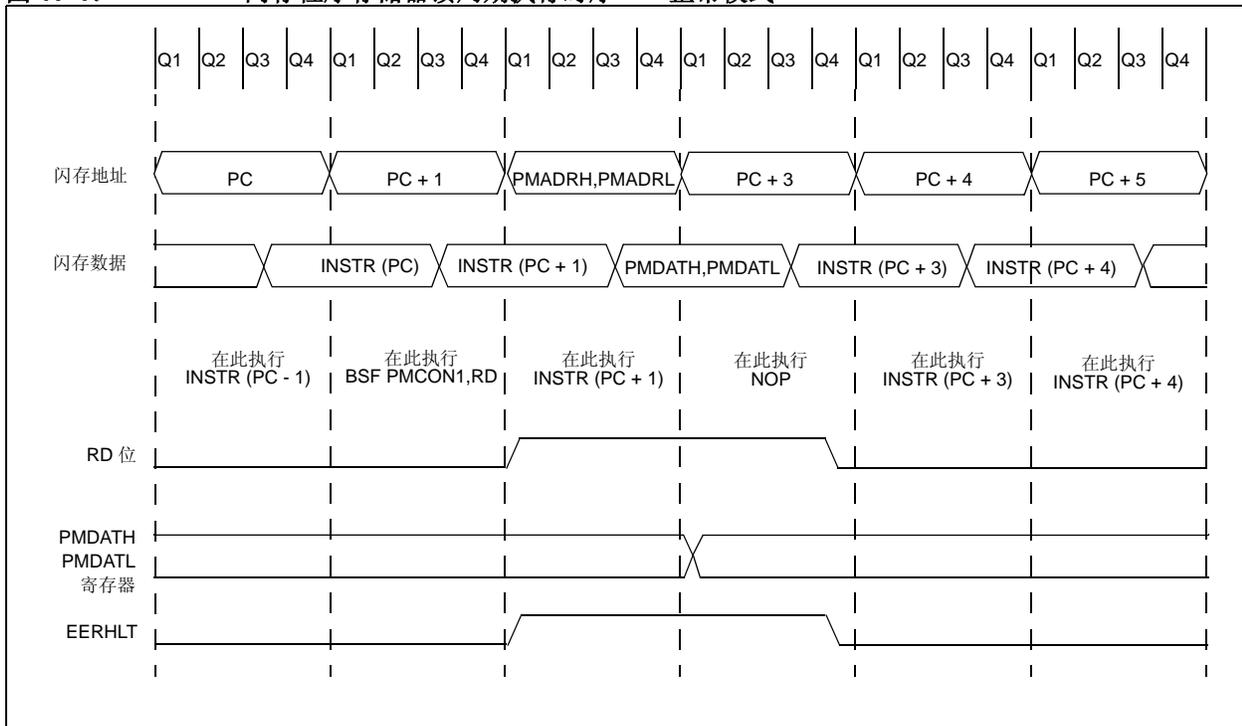
BANKSELPM_ADR; Change STATUS bits RP1:0 to select bank with PMADR
MOVLWMS_PROG_PM_ADDR;
MOVWFPADRH; MS Byte of Program Address to read
MOVLWLS_PROG_PM_ADDR;
MOVWFPADRL; LS Byte of Program Address to read
BANKSELPMCON1; Bank to containing PMCON1
BSF PMCON1, RD; EE Read

NOP      ; First instruction after BSF PMCON1,RD executes normally

NOP      ; Any instructions here are ignored as program
          ; memory is read in second cycle after BSF PMCON1,RD
          ;

BANKSELPMDATL; Bank to containing PMADRL
MOVFPMDATL, W; W = LS Byte of Program PMDATL
MOVFPMDATH, W; W = MS Byte of Program PMDATH
    
```

图 17-1: 闪存程序存储器读周期执行时序——正常模式



### 17.3.2 写入闪存程序存储器

闪存程序存储器的字只有在该字位于存储器的未受保护段中时才能写入，如第 11.1 节“配置字”（位 <WRT1:0>）中定义。

**注：** 写保护位用于保护用户程序不被用户代码修改。由 ICSP 执行编程时，这些位不起任何作用。当代码保护位编程为代码保护时，可防止通过 ICSP 接口对程序存储器执行写操作。

闪存程序存储器必须以四字块的方式写入。更多详细信息，请参见图 17-2 和 17-3。块由地址连续的四个字组成，其下边界由  $PMADRL<1:0> = 00$  的地址定义。程序存储器的所有块写入都通过四字写操作并以擦除十六字的形式完成。写操作是边沿对齐的，不会跨越这些边界进行。

要写入程序数据，必须先将 **WREN** 位装入缓冲区寄存器（见图 17-2）。这可通过先将目标地址写入 **PMADRL** 和 **PMADRH**，再将数据写入 **PMDATL** 和 **PMDATH** 来完成。设置好地址和数据后，必须执行以下事件序列：

1. 依次将 55h 和 AAh 写入 **PMCON2**（闪存编程序列）。
2. 将 **PMCON1** 寄存器的 **WR** 控制位置 1。

所有四个缓冲区寄存器单元都应写入正确的数据。如果四字块中写入的数据少于 4 个字，则必须对未写入的程序存储器单元执行读操作。这将从未写入的程序存储器单元获取数据，并将数据装入 **PMDATL** 和 **PMDATH** 寄存器。然后，必须执行将数据传输到缓冲区寄存器的事件序列。

要将数据从缓冲区寄存器传输到程序存储器，**PMADRL** 和 **PMADRH** 必须指向四字块的最后一个单元（ $PMADRL<1:0> = 11$ ）。然后，必须执行以下事件序列：

1. 依次将 55h 和 AAh 写入 **PMCON2**（闪存编程序列）。
2. 将 **PMCON1** 寄存器的控制位 **WR** 置 1 以开始写操作。

用户必须遵循同样的特定序列来启动程序块中各个字的写入，并按顺序（000、001、010 和 011）写入各程序字。对最后一个字（ $PMADRL<1:0> = 11$ ）执行写操作后，将自动擦除十六字的块，并将四字缓冲区寄存器的内容写入程序存储器中。

执行 **BSF PMCON1, WR** 指令后，处理器需要两个周期来设置擦除 / 写操作。将 **WR** 位置 1 后，用户必须放置两条 **NOP** 指令。由于数据将被写入缓冲区寄存器，因此块前三个字的写入将立即发生。处理器将仅在擦除发生的周期内（即擦除十六字块的最后一个字）暂停内部操作，通常为 4 ms。这不是休眠模式，因为时钟和外设将继续运行。在四字写周期之后，处理器将恢复操作并执行 **PMCON1** 写指令后的第三条指令。必须对高 12 字重复上述序列。

**注：** 仅在行边界后或者 **PMCON1<WR>** 置 1 且 **PMADRL<3:0> = xxxxx0011** 时，才能为四字写操作启动擦除操作。

有关测试模式下缓冲区寄存器和控制信号的框图，请参见图 17-2。

### 17.3.3 防止误写操作的保护措施

有些情况下，不应写入器件程序存储器。为防止误写操作，已内置了多种机制。上电时，**WREN** 将被清零。此外，上电定时器（72 ms 持续时间）还将阻止程序存储器写操作。

在电源故障或软件故障期间，写操作的启动序列以及 **WREN** 位有助于防止意外写操作的发生。

### 17.3.4 代码保护期间的操作

当器件有代码保护时，CPU 能够从程序存储器读取解读数据，也能够将解读数据写入程序存储器。此时会禁止测试模式访问。

### 17.3.5 写保护期间的操作

当程序存储器有写保护时，CPU 可以从程序存储器读取数据并执行。受写保护的程序存储器部分不可由 CPU 通过 **PMCON** 寄存器进行修改。写保护在 **ICSP** 模式下不起作用。

# MCP19114/5

图 17-2: 对 4K 闪存程序存储器的块写操作

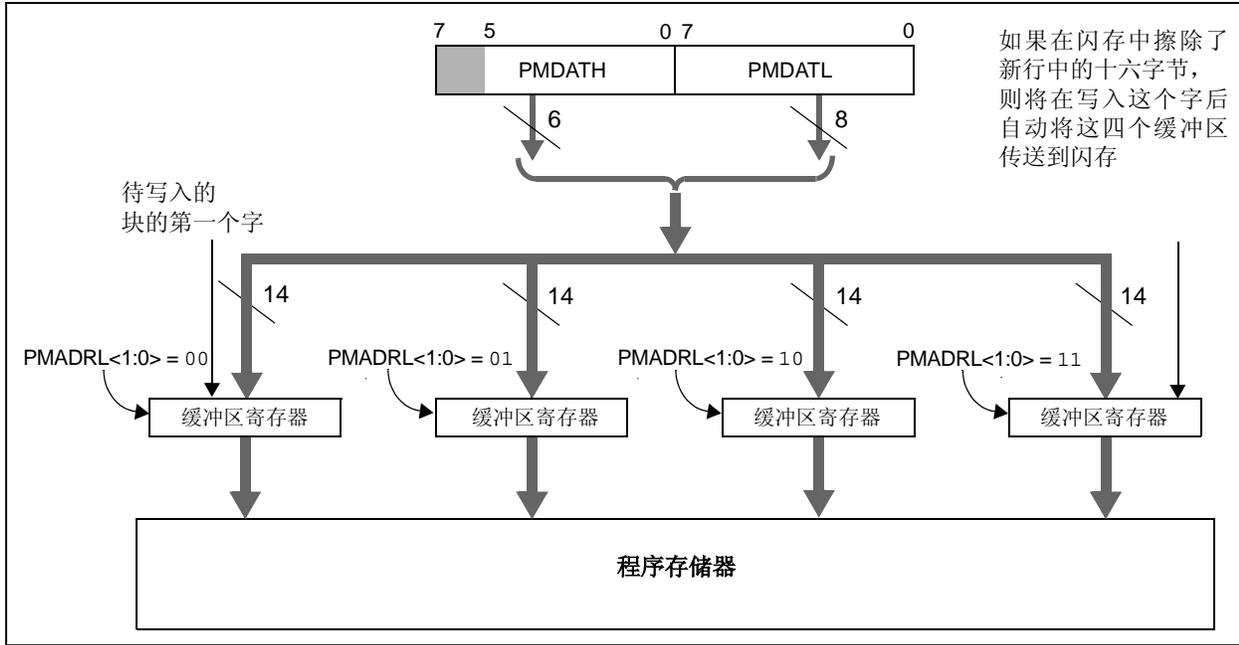
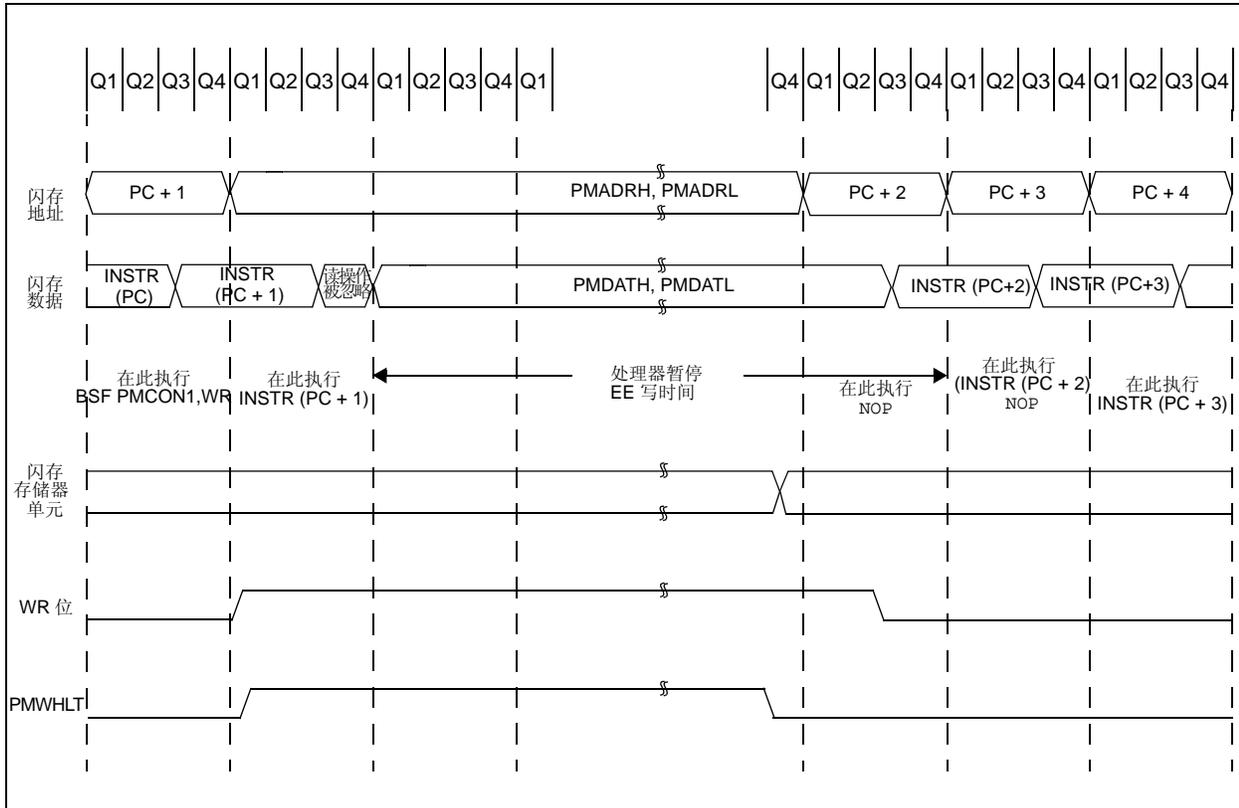


图 17-3: 闪存程序存储器长写周期执行



## 18.0 I/O 端口

通常，当使能某个外设时，该引脚将不能用作通用 I/O 引脚。

每个端口都有 2 个寄存器与其操作相关。这 2 个寄存器是：

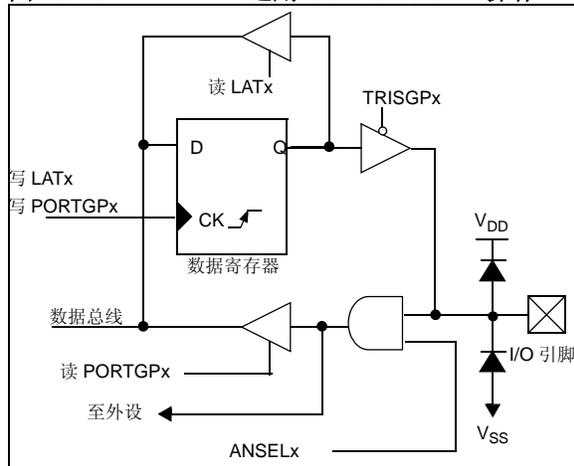
- TRISGPx 寄存器（数据方向寄存器）
- PORTGPx 寄存器（用于读器件引脚上的电平）

有些端口可能具有下列一个或多个额外的寄存器。这些寄存器是：

- ANSELx 寄存器（模拟选择）
- WPUGPx 寄存器（弱上拉）

支持模拟功能的端口还具有 ANSELx 寄存器，该寄存器可禁止数字输入并节省功耗。图 18-1 给出了通用 I/O 端口的简化模型，未给出与其他外设的接口。

图 18-1: 通用 I/O PORTGPx 操作



例 18-1: 初始化 PORTGPA

```

; This code example illustrates
; initializing the PORTGPA register. The
; other ports are initialized in the same
; manner.

BANKSEL PORTGPA;
CLRF   PORTGPA;Init PORTA
BANKSEL ANSELA;
CLRF   ANSELA;digital I/O
BANKSEL TRISGPA;
MOVLW  B'00011111';Set GPA<3:0> as
        ;inputs
MOVWF  TRISGPA;and set GPA<7:5> as
        ;outputs

```

## 18.1 PORTGPA 和 TRISGPA 寄存器

PORTGPA 是 8 位宽的双向端口，由五个 CMOS I/O、一个漏极开路 I/O 和一个仅用作输入的漏极开路引脚组成（GPA4 不可用）。其对应的数据方向寄存器是 TRISGPA。将 TRISGPA 某位设置为 1，可以将相应的 PORTGPA 引脚作为输入引脚（即，禁止输出驱动器）。将 TRISGPA 某位设置为 0，可以将相应的 PORTGPA 引脚作为输出引脚（即，使能输出驱动器）。GPA5 例外，它仅可作为输入引脚，并且其 TRISGPA 位始终读为 1。例 18-1 显示了如何初始化 I/O 端口。

读 PORTGPA 寄存器读的是引脚的状态，而写该寄存器将会写入端口锁存器。所有写操作都是读 - 修改 - 写操作。

即使在 PORTGPA 引脚用作模拟输入引脚的时候，TRISGPA 寄存器仍然控制 PORTGPA 引脚的输出驱动器。当用作模拟输入引脚时，用户必须确保 TRISGPA 寄存器中的位保持置 1。配置为模拟输入的 I/O 引脚始终读为 0。如果该引脚配置为数字输出（端口功能或备用功能），则 TRISGPA 位必须清零以使该引脚驱动信号，读操作可反映引脚状态。

### 18.1.1 电平变化中断

每个 PORTGPA 引脚都可单独配置为电平变化中断引脚。控制位 IOCB<7:4> 和 IOCB<2:0> 可使能或禁止每个引脚的中断功能。上电复位时会禁止电平变化中断功能。更多信息，请参见第 19.0 节“电平变化中断”。

### 18.1.2 弱上拉

PORTGPA<3:0> 和 PORTGPA5 具有内部弱上拉。PORTGPA<7:6> 没有内部弱上拉。各个控制位可以使能或禁止内部弱上拉（见寄存器 18-3）。如果将端口引脚配置为输出（备用功能）或者在上电复位时将 OPTION\_REG 寄存器的 RAPU 位置 1，弱上拉会自动关闭。通过将 CONFIG 寄存器的 bit 5 置 1 将 GPA5 配置为 MCLR 引脚时，将使能 GPA5 上的弱上拉；当 GPA5 为 I/O 时，将禁止该功能。MCLR 上拉不能通过软件控制。

## 18.1.3 ANSELA 寄存器

ANSELA 寄存器用于将 I/O 引脚的输入模式配置为模拟模式。将相应的 ANSELA 位设置为高电平将使引脚上的所有数字读操作都读为 0，并允许引脚上的模拟功能正常工作。

ANSELA 位的状态不会影响数字输出功能。TRISGPA 清零且 ANSELA 置 1 的引脚将仍作为数字输出引脚工作，但输入模式将变为模拟输入模式。当在受影响的端口上执行读 - 修改 - 写指令时，这会引发意外操作。

**注：** 复位后，ANSELA 位默认为模拟模式。将任何引脚作为数字通用或者外设输入时，相应的 ANSELA 位必须由用户软件初始化为 0。

## 18.1.4 PORTGPA 功能和输出优先级

每个 PORTGPA 引脚均复用多个功能。表 18-1 显示了引脚及其组合功能和输出优先级。更多信息，请参见本数据手册中的相应章节。

PORTGPA 寄存器中的 GPA7 引脚是真正的漏极开路引脚，未连接回 V<sub>DD</sub>。

当使能多个输出时，实际引脚控制权将属于优先级最高的外设。

优先级列表中未列出模拟输入功能，例如 ADC。使用 ANSELA 寄存器将 I/O 引脚设置为模拟模式时，这些输入是有效的。当引脚处于模拟模式时，数字输出功能可以按照表 18-1 中列出的优先级控制引脚。

表 18-1: PORTGPA 输出优先级

引脚名称	功能优先级 <sup>(1)</sup>
GPA0	GPA0
	TEST_OUT
GPA1	GPA1
	CLKPIN
GPA2	GPA2
	T0CKI
	INT
GPA3	GPA3
GPA5	GPA5 (漏极开路, 仅输入)
	MCLR
	TEST_EN
GPA6	GPA6
	CCD
	ICSPDAT
GPA7	GPA7 (漏极开路输出, ST 输入)
	SCL

**注 1:** 输出功能优先级按从最低到最高排列。

## 寄存器 18-1: PORTGPA: PORTGPA 寄存器

R/W-x	R/W-x	R-x	U-0	R/W-x	R/W-x	R/W-x	R/W-x
GPA7	GPA6	GPA5	—	GPA3	GPA2	GPA1	GPA0
bit 7							bit 0

### 图注:

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

- bit 7      **GPA7:** 通用漏极开路 I/O 引脚位  
 1 = 端口引脚电平 >  $V_{IH}$   
 0 = 端口引脚电平 <  $V_{IL}$
- bit 6      **GPA6:** 通用 I/O 引脚位  
 1 = 端口引脚电平 >  $V_{IH}$   
 0 = 端口引脚电平 <  $V_{IL}$
- bit 5      **GPA5/MCLR/TEST\_EN5:** 通用漏极开路输入引脚
- bit 4      **未实现:** 读为 0
- bit 3-0    **GPA<3:0>:** 通用 I/O 引脚位  
 1 = 端口引脚电平 >  $V_{IH}$   
 0 = 端口引脚电平 <  $V_{IL}$

## 寄存器 18-2: TRISGPA: PORTGPA 三态寄存器

R/W-1	R/W-1	R-1	U-0	R/W-1	R/W-1	R/W-1	R/W-1
TRISA7	TRISA6	TRISA5	—	TRISA3	TRISA2	TRISA1	TRISA0
bit 7							bit 0

### 图注:

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

- bit 7-6    **TRISA<7:6>:** PORTGPA 三态控制位  
 1 = PORTGPA 引脚配置为输入 (三态)  
 0 = PORTGPA 引脚配置为输出
- bit 5      **TRISA5:** GPA5 端口三态控制位  
 由于 GPA5 仅可作为输入引脚, 所以该位始终为 1
- bit 4      **未实现:** 读为 0
- bit 3-0    **TRISA<3:0>:** PORTGPA 三态控制位  
 1 = PORTGPA 引脚配置为输入 (三态)  
 0 = PORTGPA 引脚配置为输出

# MCP19114/5

## 寄存器 18-3: WPUGPA: 弱上拉 PORTGPA 寄存器

U-0	U-0	R/W-1	U-0	R/W-1	R/W-1	R/W-1	R/W-1
—	—	WPUA5	—	WPUA3	WPUA2	WPUA1	WPUA0
bit 7							bit 0

### 图注:

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

- bit 7-6 **未实现:** 读为 0
- bit 5 **WPUA5:** 弱上拉寄存器位  
1 = 使能上拉  
0 = 禁止上拉
- bit 4 **未实现:** 读为 0
- bit 3-0 **WPUA<3:0>:** 弱上拉寄存器位  
1 = 使能上拉  
0 = 禁止上拉

- 注 1: 仅当使能全局  $\overline{\text{RAPU}}$  位、该引脚处于输入模式 ( $\text{TRISGPA} = 1$ )、使能各个 WPUA 位 ( $\text{WPUA} = 1$ ) 并且该引脚未配置为模拟输入时, 才会使能弱上拉器件。
- 2: 在 CONFIG 寄存器中将 GPA5 配置为  $\overline{\text{MCLR}}$  时, 也会使能 GPA5 上的弱上拉。

## 寄存器 18-4: ANSELA: 模拟选择 GPA 寄存器

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
u = 不变	x = 未知	-n = POR 时的值
1 = 置 1	0 = 清零	

- bit 7-4 **未实现:** 读为 0
- bit 3-0 **ANSA<3:0>:** 模拟选择 GPA 寄存器位  
1 = 模拟输入。引脚被分配为模拟输入引脚。(1)  
0 = 数字 I/O。引脚被分配为端口或是特殊功能引脚。

- 注 1: 将引脚设置为模拟输入时会自动禁止数字输入电路、弱上拉和电平变化中断功能 (如果可用)。必须将相应的 TRISA 位设置为输入模式, 以便允许外部控制引脚电压。

表 18-2: 与 PORTGPA 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	114
OPTION_REG	$\overline{\text{RAPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	78
PORTGPA	GPA7	GPA6	GPA5	—	GPA3	GPA2	GPA1	GPA0	113
TRISGPA	TRISA7	TRISA6	TRISA5	—	TRISA3	TRISA2	TRISA1	TRISA0	113
WPUGPA	—	—	WPUA5	—	WPUA3	WPUA2	WPUA1	WPUA0	114

图注: — = 未实现单元, 读为 0。PORTGPA 不使用阴影单元。

## 18.2 PORTGPB 和 TRISGPB 寄存器

由于特殊功能引脚需求，使用了有限数量的 PORTGPB I/O。在 24 引脚 QFN MCP19114 上，实现了 GPB0 和 GPB1。GPB0 为漏极开路通用 I/O 和 SDA 引脚。GPB1 为通用 I/O、模拟输入和 VREF2 DAC 输出。28 引脚 QFN MCP19114 有四个额外的通用 PORTGPB I/O 引脚。其对应的数据方向寄存器是 TRISGPB。将 TRISGPB 某位设置为 1，可以让相应的 PORTGPB 引脚作为输入引脚（即，禁止输出驱动器）。将 TRISGPB 某位设置为 0，可以让相应的 PORTGPB 引脚作为输出引脚（即，使能输出驱动器）。例 18-1 说明了如何初始化 I/O 端口。

PORTGPB 的某些引脚会针对外设复用备用功能或时钟功能。通常，当使能某个外设或时钟功能时，该引脚将不能用作通用 I/O 引脚。

读 PORTGPB 寄存器读的是引脚的状态，而写该寄存器将会写入端口锁存器。所有写操作都是读 - 修改 - 写操作。

即使在 PORTGPB 引脚用作模拟输入引脚时，TRISGPB 寄存器仍然控制 PORTGPB 引脚的输出驱动器。当用作模拟输入引脚时，用户必须确保 TRISGPB 寄存器中的位保持置 1。配置为模拟输入的 I/O 引脚始终读为 0。如果该引脚配置为数字输出（端口功能或备用功能），则 TRISGPB 位必须清零以使该引脚驱动信号，读操作可反映引脚状态。

### 18.2.1 电平变化中断

每个 PORTGPB 引脚都可单独配置为电平变化中断引脚。控制位 IOCB<7:4> 和 IOCB<2:0> 可启用或禁止每个引脚的中断功能。上电复位时会禁止电平变化中断功能。更多信息，请参见第 19.0 节“电平变化中断”。

### 18.2.2 弱上拉

每个 PORTGPB 引脚均具有可单独配置的内部弱上拉。控制位 WPUB<7:4> 和 WPUB<1> 可启用或禁止每个上拉（请参见寄存器 18-7）。当端口引脚配置为输出时，各弱上拉会自动关闭。上电复位时，OPTION\_REG 寄存器的 RAPU 位会禁止所有上拉。

### 18.2.3 ANSELB 寄存器

ANSELB 寄存器用于将 I/O 引脚的输入模式配置为模拟模式。将相应的 ANSELB 位设置为高电平将使引脚上的所有数字读操作都读为 0，并允许引脚上的模拟功能正常工作。

ANSELB 位的状态不会影响数字输出功能。TRISGPB 清零且 ANSELB 置 1 的引脚仍将仍作为数字输出工作，但输入模式将变为模拟输入模式。当在受影响的端口上执行读 - 修改 - 写指令时，这会引发意外操作。

即使在 PORTGPB 引脚用作模拟输入引脚时，TRISGPB 寄存器仍然控制 PORTGPB 引脚的输出驱动器。当引脚用作模拟输入引脚时，用户应确保 TRISGPB 寄存器中的位保持置 1。配置为模拟输入的 I/O 引脚始终读为 0。

**注：** 在发生复位之后，ANSELB 位默认为模拟模式。将任何引脚作为数字通用或者外设输入时，相应的 ANSELB 位必须由用户软件初始化为 0。

## 18.2.4 PORTGPB 功能和输出优先级

每个 PORTGPB 引脚均复用多个功能。表 18-3 显示了引脚及其组合功能和输出优先级。更多信息，请参见本数据手册中的相应章节。

PORTGPB 寄存器中的 GPB0 引脚是真正的漏极开路引脚，未连接回  $V_{DD}$ 。

当使能多个输出时，实际引脚控制权将属于优先级最高的外设。

下表中未包含模拟输入功能（例如 ADC）和一些数字输入功能。使用 ANSELB 寄存器将 I/O 引脚设置为模拟模式时，这些输入是有效的。当引脚处于模拟模式时，数字输出功能可以按照表 18-3 中列出的优先级控制引脚。

表 18-3: PORTGPB 输出优先级

引脚名称	功能优先级 <sup>(1)</sup>
GPB0	GPB0（漏极开路输入 / 输出）
	SDA
GPB1	GPB1
	VREF2
GPB4	GPB4（MCP19114）
	ICSPDAT
GPB5	GPB5（MCP19114）
GPB6	GPB6（MCP19114）
GPB7	GPB7（MCP19114）
	CCD2

注 1: 输出功能优先级按从最低到最高排列。

寄存器 18-5: PORTGPB: PORTGPB 寄存器

R/W-x	R/W-x	R/W-x	R/W-x	U-0	U-0	R/W-x	R/W-x
GPB7 <sup>(1)</sup>	GPB6 <sup>(1)</sup>	GPB5 <sup>(1)</sup>	GPB4 <sup>(1)</sup>	—	—	GPB1	GPB0
bit 7						bit 0	

**图注:**

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n = POR 时的值

1 = 置 1

0 = 清零

bit 7-4 **GPB<7:4>**: 通用 I/O 引脚位

1 = 端口引脚电平 >  $V_{IH}$

0 = 端口引脚电平 <  $V_{IL}$

bit 3-2 **未实现**: 读为 0

bit 1-0 **GPB<1:0>**: 通用 I/O 引脚位

1 = 端口引脚电平 >  $V_{IH}$

0 = 端口引脚电平 <  $V_{IL}$

注 1: 仅 MCP19115。

# MCP19114/5

## 寄存器 18-6: TRISGPB: PORTGPB 三态寄存器

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	R/W-1	R/W-1
TRISB7 <sup>(1)</sup>	TRISB6 <sup>(1)</sup>	TRISB5 <sup>(1)</sup>	TRISB4 <sup>(1)</sup>	—	—	TRISB1	TRISB0
bit 7							bit 0

### 图注:

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

- bit 7-4      **TRISB<7:4>**: PORTGPB 三态控制位  
1 = PORTGPB 引脚配置为输入 (三态)  
0 = PORTGPB 引脚配置为输出
- bit 3-2      **未实现**: 读为 0
- bit 1-0      **TRISB<1:0>**: PORTGPB 三态控制位  
1 = PORTGPB 引脚配置为输入 (三态)  
0 = PORTGPB 引脚配置为输出

注 1: 仅 MCP19115。

## 寄存器 18-7: WPUGPB: 弱上拉 PORTGPB 寄存器

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	R/W-1	U-0
WPUB7 <sup>(2)</sup>	WPUB6 <sup>(2)</sup>	WPUB5 <sup>(2)</sup>	WPUB4 <sup>(2)</sup>	—	—	WPUB1	—
bit 7							bit 0

### 图注:

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

- bit 7-4      **WPUB<7:4>**: 弱上拉寄存器位  
1 = 使能上拉  
0 = 禁止上拉
- bit 3-2      **未实现**: 读为 0
- bit 1        **WPUB<1>**: 弱上拉寄存器位  
1 = 使能上拉  
0 = 禁止上拉
- bit 0        **未实现**: 读为 0

- 注 1: 仅当使能全局  $\overline{\text{RAPU}}$  位、该引脚处于输入模式 ( $\text{TRISGPA} = 1$ )、使能各个 WPUB 位 ( $\text{WPUB} = 1$ ) 并且该引脚未配置为模拟输入时, 才会使能弱上拉器件。
- 2: 仅 MCP19115。

## 寄存器 18-8: ANSELB: 模拟选择 GPB 寄存器

U-0	R/W-1	R/W-1	R/W-1	U-0	U-0	R/W-1	U-0
—	ANSB6 <sup>(1)</sup>	ANSB5 <sup>(1)</sup>	ANSB4 <sup>(1)</sup>	—	—	ANSB1	—
bit 7							bit 0

### 图注:

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

- bit 7      **未实现:** 读为 0
- bit 6-4    **ANSB<6:4>:** 模拟选择 GPB 寄存器位  
 1 = 模拟输入。引脚被分配为模拟输入引脚<sup>(1)</sup>。  
 0 = 数字 I/O。引脚被分配为端口或是特殊功能。
- bit 3-2    **未实现:** 读为 0
- bit 1      **ANSB1:** 模拟选择 GPB 寄存器位  
 1 = 模拟输入。引脚被分配为模拟输入引脚<sup>(1)</sup>。  
 0 = 数字 I/O。引脚被分配为端口或是特殊功能。
- bit 0      **未实现:** 读为 0

- 注 1: 仅 MCP19115。
- 注 2: 将引脚设置为模拟输入时会自动禁止数字输入电路、弱上拉以及电平变化中断（如果可用）。必须将相应的 TRIS 位设置为输入模式，以便允许外部控制引脚电压。

表 18-4: 与 PORTGPB 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
ANSELB	—	ANSB6 <sup>(1)</sup>	ANSB5 <sup>(1)</sup>	ANSB4 <sup>(1)</sup>	—	—	ANSB1	—	119
OPTION_REG	$\overline{\text{RAPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	78
PORTGPB	GPB7 <sup>(1)</sup>	GPB6 <sup>(1)</sup>	GPB5 <sup>(1)</sup>	GPB4 <sup>(1)</sup>	—	—	GPB1	GPB0	117
TRISGPB	TRISB7 <sup>(1)</sup>	TRISB6 <sup>(1)</sup>	TRISB5 <sup>(1)</sup>	TRISB4 <sup>(1)</sup>	—	—	TRISB1	TRISB0	118
WPUGPB	WPUB7 <sup>(1)</sup>	WPUB6 <sup>(1)</sup>	WPUB5 <sup>(1)</sup>	WPUB4 <sup>(1)</sup>	—	—	WPUB1	—	118

图注: — = 未实现单元, 读为 0。PORTGPB 寄存器不使用阴影单元。

- 注 1: 仅 MCP19115。

# MCP19114/5

---

注:

## 19.0 电平变化中断

每个 PORTGPA 和 PORTGPB 引脚均可单独配置为电平变化中断引脚。控制位 IOCA 和 IOCB 可使能或禁止每个引脚的中断功能。请参见寄存器 19-1 和 19-2。上电复位时会禁止电平变化中断功能。

在 CONFIG 寄存器中将 GPA5 配置为  $\overline{\text{MCLR}}$  引脚时，将禁止 GPA5 上的电平变化中断功能。

对于已使能电平变化中断的引脚，这些值会与上次读取 PORTGPA 或 PORTGPB 时锁存的旧值进行比较。上次读取的所有 PORTGPA 和 PORTGPB 引脚的不匹配输出将进行逻辑或运算，以将 INTCON 寄存器中的电平变化中断标志位 (IOCF) 置 1。

### 19.1 使能模块

要允许独立的端口引脚产生中断，必须将 INTCON 寄存器的 IOCE 位置 1。如果 IOCE 位被禁止，则仍然会对引脚上的信号进行边沿检测，但是不产生中断。

### 19.2 独立的引脚配置

要允许引脚检测电平变化中断，需要将 IOCA 或 IOCB 寄存器的相关 IOCAx 或 IOCBx 位置 1。

### 19.3 清零中断标志

用户可在中断服务程序中通过以下操作清除中断：

- a) 读 PORTGPA 或 PORTGPB 并清零标志位 IOCF。  
这会结束不匹配条件。
- 或
- b) 写 PORTGPA 或 PORTGPB 并清零标志位 IOCF。  
这会结束不匹配条件。

不匹配条件会继续将标志位 IOCF 置 1。读 PORTGPA 或 PORTGPB 将结束不匹配条件，并允许将标志位 IOCF 清零。保存上次读取值的锁存器不受  $\overline{\text{MCLR}}$  复位的影响。此次复位后，如果存在不匹配条件，IOCF 标志位将继续置 1。

<b>注：</b> 如果在执行任何 PORTGPA 或 PORTGPB 操作时 I/O 引脚上发生电平变化，则 IOCF 中断标志位可能不会置 1。
--

### 19.4 休眠期间的操作

如果 IOCE 位置 1，电平变化中断的中断序列会将器件从休眠模式唤醒。

# MCP19114/5

## 19.5 电平变化中断寄存器

寄存器 19-1: IOCA: 电平变化中断 PORTGPA 寄存器

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
IOCA7	IOCA6	IOCA5	—	IOCA3	IOCA2	IOCA1	IOCA0
bit 7							bit 0

**图注:**

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

bit 7-6        **IOCA<7:6>:** 电平变化中断 PORTGPA 寄存器位  
1 = 使能引脚的电平变化中断。  
0 = 禁止引脚的电平变化中断。

bit 5        **IOCA<5>:** 电平变化中断 PORTGPA 寄存器位 <sup>(1)</sup>  
1 = 使能引脚的电平变化中断。  
0 = 禁止引脚的电平变化中断。

bit 4        **未实现:** 读为 0

bit 3-0      **IOCA<3:0>:** 电平变化中断 PORTGPA 寄存器位  
1 = 使能引脚的电平变化中断。  
0 = 禁止引脚的电平变化中断。

注 1: 当 GPA5 配置为  $\overline{\text{MCLR}}$  时, 将禁止 GPA5 上的电平变化中断功能。

寄存器 19-2: IOCB: 电平变化中断 PORTGPB 寄存器

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
IOCB7 <sup>(1)</sup>	IOCB6 <sup>(1)</sup>	IOCB5 <sup>(1)</sup>	IOCB4 <sup>(1)</sup>	—	—	IOCB1	IOCB0
bit 7							bit 0

**图注:**

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

bit 7-4      **IOCB<7:4>:** 电平变化中断 PORTGPB 寄存器位  
1 = 使能引脚的电平变化中断。  
0 = 禁止引脚的电平变化中断。

bit 3-2      **未实现:** 读为 0

bit 1-0      **IOCB<1:0>:** 电平变化中断 PORTGPB 寄存器位  
1 = 使能引脚的电平变化中断。  
0 = 禁止引脚的电平变化中断。

注 1: 仅 MCP19115。

表 19-1: 与电平变化中断有关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	114
ANSELB	—	ANSB6 <sup>(1)</sup>	ANSB5 <sup>(1)</sup>	ANSB4 <sup>(1)</sup>	—	—	ANSB1	—	119
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	95
IOCA	IOCA7	IOCA6	IOCA5	—	IOCA3	IOCA2	IOCA1	IOCA0	122
IOCB	IOCB7 <sup>(1)</sup>	IOCB6 <sup>(1)</sup>	IOCB5 <sup>(1)</sup>	IOCB4 <sup>(1)</sup>	—	—	IOCB1	IOCB0	122
TRISGPA	TRISA7	TRISA6	TRISA5	—	TRISA3	TRISA2	TRISA1	TRISA0	113
TRISGPB	TRISB7 <sup>(1)</sup>	TRISB6 <sup>(1)</sup>	TRISB5 <sup>(1)</sup>	TRISB4 <sup>(1)</sup>	—	—	TRISB1	TRISB0	118

图注: — = 未实现单元, 读为 0。电平变化中断不使用阴影单元。

注 1: 仅 MCP19115。

# MCP19114/5

---

注:

## 20.0 内部温度指示器模块

MCP19114/5 配有一个温度电路，可用于测量硅芯片的工作温度。电路的工作温度范围在  $-40^{\circ}\text{C}$  至  $+125^{\circ}\text{C}$  之间。输出电压与器件温度成比例。温度指示器的输出内部连接到器件 ADC。

### 20.1 电路工作原理

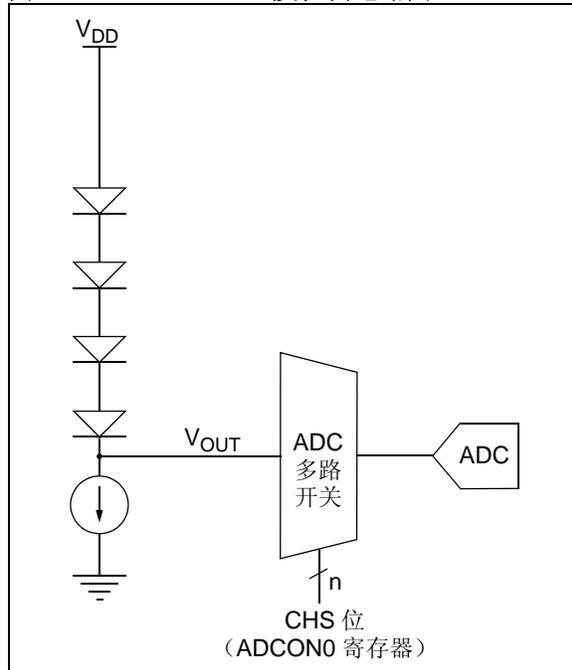
该内部温度测量电路始终使能。

### 20.2 温度输出

电路的输出使用内部模数转换器测量。保留通道 13 用于温度电路输出。详细信息，请参见第 21.0 节“模数转换器 (ADC) 模块”。

利用公式 20-1 进行 ADC 测量可计算出硅芯片的温度。出厂存储的  $30^{\circ}\text{C}$  时的 10 位 ADC 值位于地址 2084h 处。该电路的温度系数是  $15.7\text{ mV}/^{\circ}\text{C}$  ( $\pm 0.8\text{ mV}/^{\circ}\text{C}$ )。其他温度读数可根据  $30^{\circ}\text{C}$  标记值进行计算。

图 20-1: 温度检测电路图



公式 20-1: 硅芯片的温度

$$\text{硅芯片温度 } (^{\circ}\text{C}) = \frac{(\text{ADC 读数 (计数)} - 30^{\circ}\text{C 时 ADC 读数 (计数)})}{4.0 (\text{计数}/^{\circ}\text{C})} + 30^{\circ}\text{C}$$

# MCP19114/5

---

注:

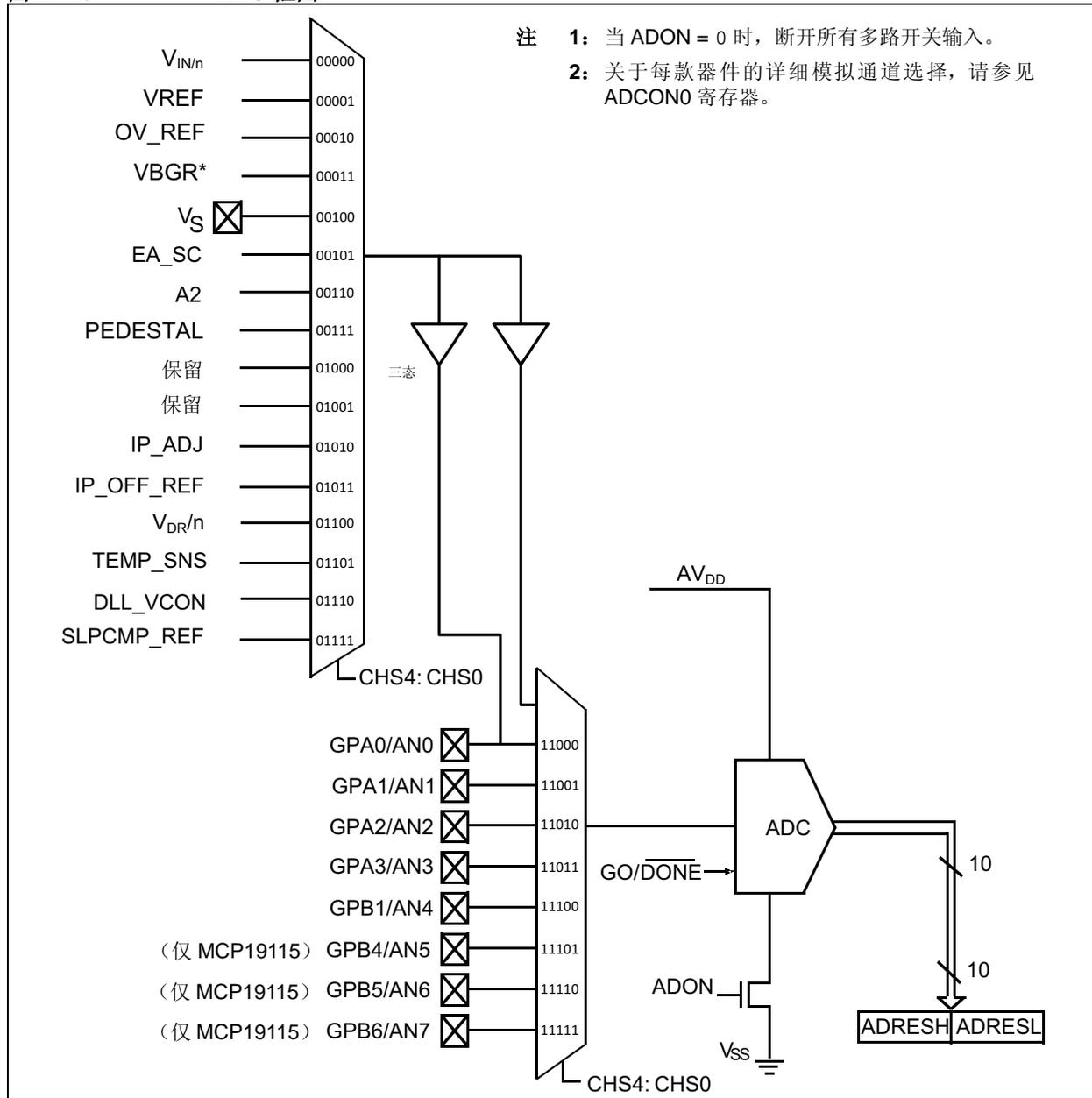
## 21.0 模数转换器 (ADC) 模块

模数转换器 (ADC) 可将模拟输入信号转换为该信号的 10 位二进制表示。该器件使用模拟输入, 这些输入通过多路开关连接到一个采样保持电路。采样保持电路的输出与转换器的输入相连。转换器通过逐次逼近法产生 10 位二进制结果, 并将转换结果以右对齐的方式存储在 ADC 结果寄存器 (ADRESH:ADRESL 寄存器对) 中。图 21-1 给出了 ADC 的框图。

内部带隙为 ADC 提供参考电压。

**注:** 当  $V_{IN}$  大于  $AV_{DD} + V_{DROPOUT}$  时,  $AV_{DD}$  进行调节, 确保 A/D 读数准确。当  $V_{IN}$  大于  $V_{DD} + V_{DROPOUT}$  时,  $V_{DD}$  进行调节, 实现准确的比率测量。

图 21-1: ADC 框图



# MCP19114/5

## 21.1 ADC 配置

配置和使用 ADC 时必须考虑以下功能：

- 端口配置
- 通道选择
- ADC 转换时钟源
- 中断控制
- 结果格式

### 21.1.1 端口配置

ADC 可用于将模拟信号转换为对应的数字信号。转换模拟信号时，应通过设置相关 TRIS 和 ANSEL 位来将 I/O 引脚配置为模拟引脚。更多信息，请参见第 18.0 节“[I/O 端口](#)”。

**注：** 在任何定义为数字输入的引脚上施加模拟电压可能导致输入缓冲器消耗的电流过大。

### 21.1.2 通道选择

在 MCP19114 上最多有 21 个通道可供使用，在 MCP19115 上最多有 24 个通道可供使用：

- AN<4:0> 引脚
- AN<7:5> 引脚（仅 MCP19115）
- $V_{IN}$ ：输入电压（ $V_{IN}$ ）的 1/15.53
- $V_{REF}$ ：调节设定值的参考电压
- OV\_REF：OV 比较器的参考电压
- $V_{BGR}$ ：带隙参考电压
- $V_S$ ：与  $V_{OUT}$  成比例的电压
- EA\_SC：斜率补偿后的误差放大器输出
- A2：次级电流检测放大器输出
- PEDESTAL
- 保留
- 保留
- IP\_ADJ：基本和失调调节后的  $I_P$
- IP\_OFF\_REF： $I_P$  失调参考电压
- $V_{DR}$ ： $V_{DR} * 0.229V/V$
- TEMP\_SNS：表示内部温度的模拟电压（见公式 20-1）
- DLL\_VCON：延迟锁定环参考电压
- SLPCMP\_REF：斜率补偿参考电压

ADCON0 寄存器的 CHS<4:0> 位控制哪一路通道与采样保持电路相连。

更换通道时需要一段时间的延时才能启动下一次转换。更多信息，请参见第 21.2 节“[ADC 工作原理](#)”。

### 21.1.3 ADC 转换时钟

可通过软件设置 ADCON1 寄存器的 ADCS 位来选择转换时钟源。有以下 5 种时钟频率可供选择：

- $F_{OSC}/8$
- $F_{OSC}/16$
- $F_{OSC}/32$
- $F_{OSC}/64$
- $F_{RC}$ （由内部振荡器除以 16 得到的时钟）

完成 1 位转换所需的时间定义为  $T_{AD}$ 。一次完整的 10 位转换需要 11 个  $T_{AD}$  周期，如图 21-2 所示。

要实现正确的转换，必须满足适当的  $T_{AD}$  规范。更多信息，请参见第 4.0 节“[电气特性](#)”中的 A/D 转换要求。表 21-1 给出了适当的 ADC 时钟选择的示例。

**注：** 除非使用  $F_{RC}$ ，否则系统时钟频率的任何变化都会改变 ADC 时钟频率，从而对 ADC 结果产生不利影响。

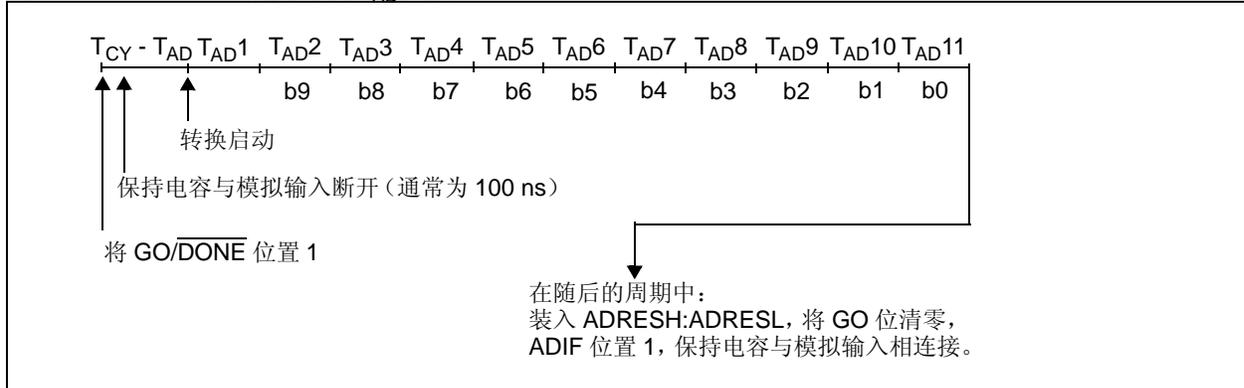
表 21-1: ADC 时钟周期 ( $T_{AD}$ ) 与器件工作频率的关系

ADC 时钟周期 ( $T_{AD}$ )		器件频率 ( $F_{OSC}$ )
ADC 时钟源	ADCS<2:0>	8 MHz
$F_{OSC}/8$	001	1.0 $\mu s$ <sup>(2)</sup>
$F_{OSC}/16$	101	2.0 $\mu s$
$F_{OSC}/32$	010	4.0 $\mu s$
$F_{OSC}/64$	110	8.0 $\mu s$ <sup>(3)</sup>
$F_{RC}$	x11	2.0-6.0 $\mu s$ <sup>(1, 4)</sup>

**图注：** 阴影单元表示超出了建议范围。

- 注 1：** 当  $V_{DD} > 3.0V$  时， $F_{RC}$  源提供 4  $\mu s$  的典型  $T_{AD}$  时间。
- 注 2：** 这些值均违反了最小  $T_{AD}$  时间要求。
- 注 3：** 为了加快转换速度，建议选用其他时钟源。
- 注 4：** 仅在休眠期间执行转换时才建议使用  $F_{RC}$  时钟源。

图 21-2: 模数转换  $T_{AD}$  周期



### 21.1.4 中断

ADC 模块具有在模数转换完成时产生中断的功能。ADC 中断标志位是 PIR1 寄存器中的 ADIF 位。ADC 中断允许位是 PIE1 寄存器中的 ADIE 位。ADIF 位必须用软件清零。

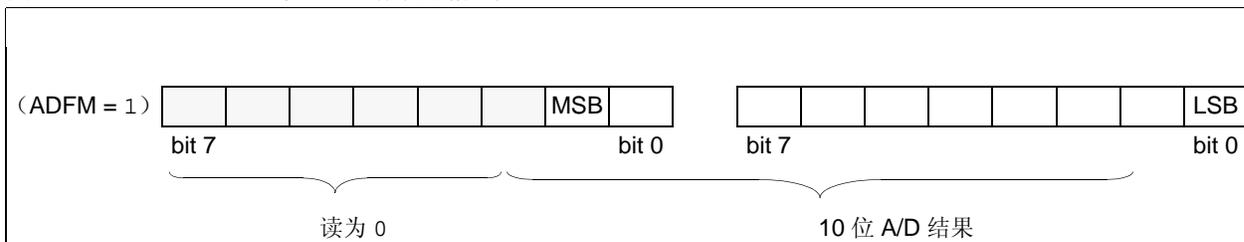
- 注 1:** 无论是否允许 ADC 中断，每次转换完成时都会将 ADIF 位置 1。
- 注 2:** 仅当在选择了  $F_{RC}$  振荡器时，ADC 才能在休眠期间工作。

器件在工作期间或休眠模式下都可产生该中断。如果器件处于休眠模式，该中断将唤醒器件。从休眠状态唤醒时，总是执行紧跟 SLEEP 指令后的下一条指令。如果用户试图从休眠状态唤醒器件并恢复主代码执行，必须禁止 INTCON 寄存器的 GIE 和 PEIE 位。如果 INTCON 寄存器中的 GIE 和 PEIE 位被使能，则转为执行中断服务程序。

### 21.1.5 结果格式

10 位 A/D 转换结果只能采用右对齐格式。

图 21-3: 10 位 A/D 结果的格式



# MCP19114/5

## 21.2 ADC 工作原理

### 21.2.1 启动转换

要使能 ADC 模块，必须将 ADCON0 寄存器的 ADON 位设置为 1。将 ADCON0 寄存器的 GO/DONE 位设置为 1 可启动模数转换。

**注：** 不应在启动 ADC 的同一条指令中将 GO/DONE 位置 1。请参见第 21.2.5 节“A/D 转换步骤”。

### 21.2.2 完成转换

转换完成时，ADC 模块将：

- 清零 GO/DONE 位
- 将 ADIF 中断标志位置 1
- 使用新的转换结果更新 ADRESH:ADRESL 寄存器

### 21.2.3 终止转换

如果必须要在转换完成前终止转换，则可用软件将 GO/DONE 位清零。ADRESH:ADRESL 寄存器将不会使用部分完成的模数转换结果进行更新。相反，ADRESH:ADRESL 寄存器对仍然保留先前转换的值。此外，启动另一次采集之前，需要两个 ADC 时钟周期。延时之后，将在选定通道上自动开始输入采集。

**注：** 器件复位强制所有寄存器进入复位状态。因此，会关闭 ADC 模块并终止任何待处理的转换。

### 21.2.4 休眠期间的 ADC 工作

ADC 不可工作在休眠模式下。AV<sub>DD</sub> 4V 参考电压已除去以将休眠电流降至最低。

### 21.2.5 A/D 转换步骤

以下是使用 ADC 执行模数转换的示例步骤：

1. 配置端口：
  - 禁止引脚输出驱动器（见 TRISGPx 寄存器）
  - 将引脚配置为模拟引脚（见 ANSELx 寄存器）
2. 配置 ADC 模块：
  - 选择 ADC 转换时钟
  - 选择 ADC 输入通道
  - 启动 ADC 模块
3. 配置 ADC 中断（可选）：
  - 清零 ADC 中断标志
  - 允许 ADC 中断
  - 允许外设中断
  - 允许全局中断<sup>(1)</sup>
4. 等待所需的采集时间<sup>(2)</sup>。
5. 通过将 GO/DONE 位置 1 启动转换。
6. 通过以下任一方式等待 ADC 转换完成：
  - 查询 GO/DONE 位
  - 等待 ADC 中断（已允许中断）
7. 读取 ADC 结果。
8. 清零 ADC 中断标志（如果已允许中断，则需要进行此操作）。

**注 1：** 如果用户尝试将器件从休眠模式下唤醒并继续执行主程序代码，则可以禁止全局中断。

**2：** 请参见第 21.4 节“A/D 采集要求”。

### 例 21-1: A/D 转换

```
;This code block configures the ADC
;for polling, Frc clock and AN0 input.
;
;Conversion start & polling for completion ;
are included.
;
BANKSELADCON1;
MOVLWB'01110000';Frc clock
MOVWFADCON1;
BANKSELTRISGPA;
BSF TRISGPA,0;Set GPA0 to input
BANKSELANSELA;
BSF ANSELA,0;Set GPA0 to analog
BANKSELADCON0;
MOVLWB'01100001';Select channel AN0
MOVWFADCON0;Turn ADC On
CALLSampleTime;Acquisiton delay
BSF ADCON0,1;Start conversion
BTFSCADCON0,1;Is conversion done?
GOTO$-1 ;No, test again
BANKSELADRESH;
MOVFADRESH,W;Read upper 2 bits
MOVWFRESULTHI;store in GPR space
BANKSELADRESL;
MOVFADRESL,W;Read lower 8 bits
MOVWFRESULTLO;Store in GPR space
```

## 21.3 ADC 寄存器定义

以下寄存器用于控制 ADC 的工作：

### 寄存器 21-1: ADCON0: A/D 控制寄存器 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	CHS4	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON
bit 7							bit 0

#### 图注:

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

bit 7 **未实现:** 读为 0

bit 6-2 **CHS<4:0>:** 模拟通道选择位

- 00000 =  $V_{IN/n}$  模拟电压测量 ( $V_{IN/n} = V_{IN}/15.5328$ )
- 00001 = VREF (DAC 参考电压, 设置电流调节级别)
- 00010 = OV\_REF (过压比较器的参考电压)
- 00011 = VBGR (带隙参考电压)
- 00100 =  $V_S$  (与  $V_{OUT}$  成比例的电压)
- 00101 = EA\_SC (斜率补偿后的误差放大器输出)
- 00110 = A2 (次级电流检测放大器输出)
- 00111 = PEDESTAL (基本电压)
- 01000 = 保留
- 01001 = 保留
- 01010 = IP\_ADJ (在 PWM 比较器上进行基本和失调调节后的 IP 电压)
- 01011 = IP\_OFF\_REF (IP 失调参考电压)
- 01100 =  $V_{DR}/n$  ( $V_{DR}/n$  模拟驱动器电压测量 =  $0.229V/V * V_{DR}$ )
- 01101 = TEMP\_SNS (模拟电压, 表示内部温度)
- 01110 = DLL\_VCON (延迟锁定环参考电压——死区的控制电压)
- 01111 = SLPCMP\_REF (斜率补偿参考电压)
- 10000 = 未实现
- 10001 = 未实现
- 10010 = 未实现
- 10011 = 未实现
- 10100 = 未实现
- 10101 = 未实现
- 10110 = 未实现
- 10111 = 未实现
- 11000 = GPA0/AN0 (即 ADDR1)
- 11001 = GPA1/AN1 (即 ADDR0)
- 11010 = GPA2/AN2 (即温度传感器输入)
- 11011 = GPA3/AN3 (即 BIN)
- 11100 = GPB1/AN4
- 11101 = GPB4/AN5 (仅 MCP19115)
- 11110 = GPB5/AN6 (仅 MCP19115)
- 11111 = GPB6/AN7 (仅 MCP19115)

bit 1 **GO/DONE:** A/D 转换状态位

- 1 = A/D 转换周期正在进行。将该位置 1 可启动 A/D 转换周期。  
A/D 转换完成后, 该位由硬件自动清零。
- 0 = A/D 转换已完成 / 未进行

bit 0 **ADON:** ADC 使能位

- 1 = 使能 ADC
- 0 = 禁止 ADC 且不消耗工作电流

# MCP19114/5

## 寄存器 21-2:        ADCON1: A/D 控制寄存器 1

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	ADCS2	ADCS1	ADCS0	—	—	—	—
bit 7							bit 0

### 图注:

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

bit 7        未实现: 读为 0

bit 6-4     **ADCS<2:0>**: A/D 转换时钟选择位  
 000 = 保留  
 001 =  $F_{OSC}/8$   
 010 =  $F_{OSC}/32$   
 x11 =  $F_{RC}$  (由内部振荡器除以 16 得到的时钟)  
 100 = 保留  
 101 =  $F_{OSC}/16$   
 110 =  $F_{OSC}/64$

bit 3-0     未实现: 读为 0

## 寄存器 21-3:        ADRESH: ADC 结果寄存器的高字节

U-0	U-0	U-0	U-0	U-0	U-0	R-x	R-x
—	—	—	—	—	—	ADRES9	ADRES8
bit 7							bit 0

### 图注:

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

bit 7-2     未实现: 读为 0

bit 1-0     **ADRES<9:8>**: A/D 结果的高 2 位

## 寄存器 21-4:        ADRESL: ADC 结果寄存器的低字节

R-x							
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit 7							bit 0

### 图注:

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

bit 7-0     **ADRES<7:0>**: A/D 结果的低 8 位

## 21.4 A/D 采集要求

为了使 ADC 达到规定精度，必须使充电保持电容 ( $C_{HOLD}$ ) 充满至输入通道的电压。图 21-4 显示了模拟输入模型。信号源阻抗 ( $R_S$ ) 和内部采样开关阻抗 ( $R_{SS}$ ) 直接影响给电容  $C_{HOLD}$  充电所需的时间。采样开关阻抗 ( $R_{SS}$ ) 随器件电压 ( $V_{DD}$ ) 不同而有所不同，请参见图 21-4。模拟信号源的最大阻抗推荐值为 10 k $\Omega$ 。

采集时间随着源阻抗的降低而缩短。选择（或改变）模拟输入通道后，A/D 采集必须在转换开始前完成。可以使用公式 21-1 计算最小采集时间。该公式假定误差为 1/2 LSB（即 ADC 的 1,024 步）。1/2 LSB 的误差是 ADC 达到规定分辨率所能允许的最大误差。

### 公式 21-1: 采集时间示例

**假设:** 温度 = +50°C, 外部阻抗为 10 k $\Omega$ ,  $V_{DD}$  为 5.0V

$$\begin{aligned} T_{ACQ} &= \text{放大器稳定时间} + \text{保持电容充电时间} + \text{温度系数} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 2 \mu\text{s} + T_C + [( \text{温度} - 25^\circ\text{C} ) (0.05 \mu\text{s}/^\circ\text{C})] \end{aligned}$$

用以下公式近似求得  $T_C$  的值:

$$V_{APPLIED} \left( 1 - \frac{1}{(2^n + 1) - 1} \right) = V_{CHOLD} \quad ; [1] \text{ 对 } V_{CHOLD} \text{ 充电 (1/2 lsb 误差范围)}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{T_C}{RC}} \right) = V_{CHOLD} \quad ; [2] \text{ 根据 } V_{APPLIED} \text{ 对 } V_{CHOLD} \text{ 充电}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{T_C}{RC}} \right) = V_{APPLIED} \left( 1 - \frac{1}{(2^n + 1) - 1} \right) \quad ; \text{合并 [1] 和 [2]}$$

**注:** 其中  $n$  = ADC 的位数。

求解  $T_C$ :

$$\begin{aligned} T_C &= -C_{HOLD} (R_{IC} + R_{SS} + R_S) \ln(1/2047) \\ &= -10 \text{ pF} (1 \text{ k}\Omega + 7 \text{ k}\Omega + 10 \text{ k}\Omega) \ln(0.0004885) \\ &= 1.37 \mu\text{s} \end{aligned}$$

因此:

$$\begin{aligned} T_{ACQ} &= 2 \mu\text{s} + 1.37 \mu\text{s} + [(50^\circ\text{C} - 25^\circ\text{C}) (0.05 \mu\text{s}/^\circ\text{C})] \\ &= 4.67 \mu\text{s} \end{aligned}$$

- 注 1:** 在每次转换后，充电保持电容 ( $C_{HOLD}$ ) 并不放电。  
**注 2:** 模拟信号源的最大阻抗推荐值为 10 k $\Omega$ 。此要求是为了符合引脚泄漏电流规范。

# MCP19114/5

图 21-4: 模拟输入模型

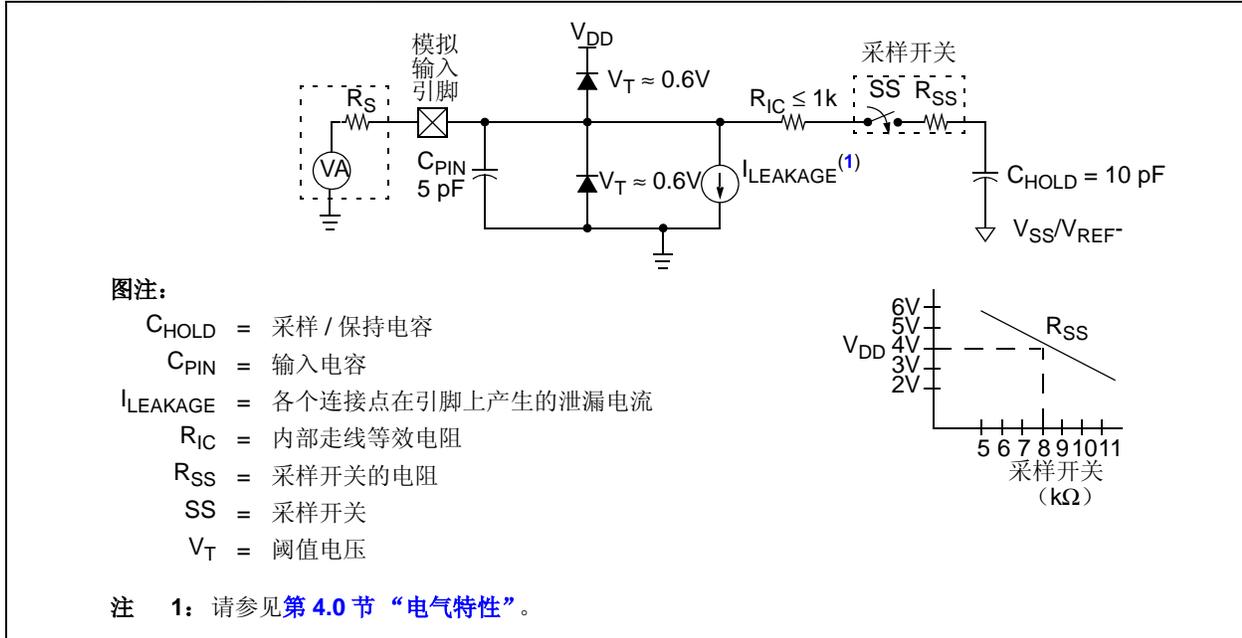


图 21-5: ADC 传递函数

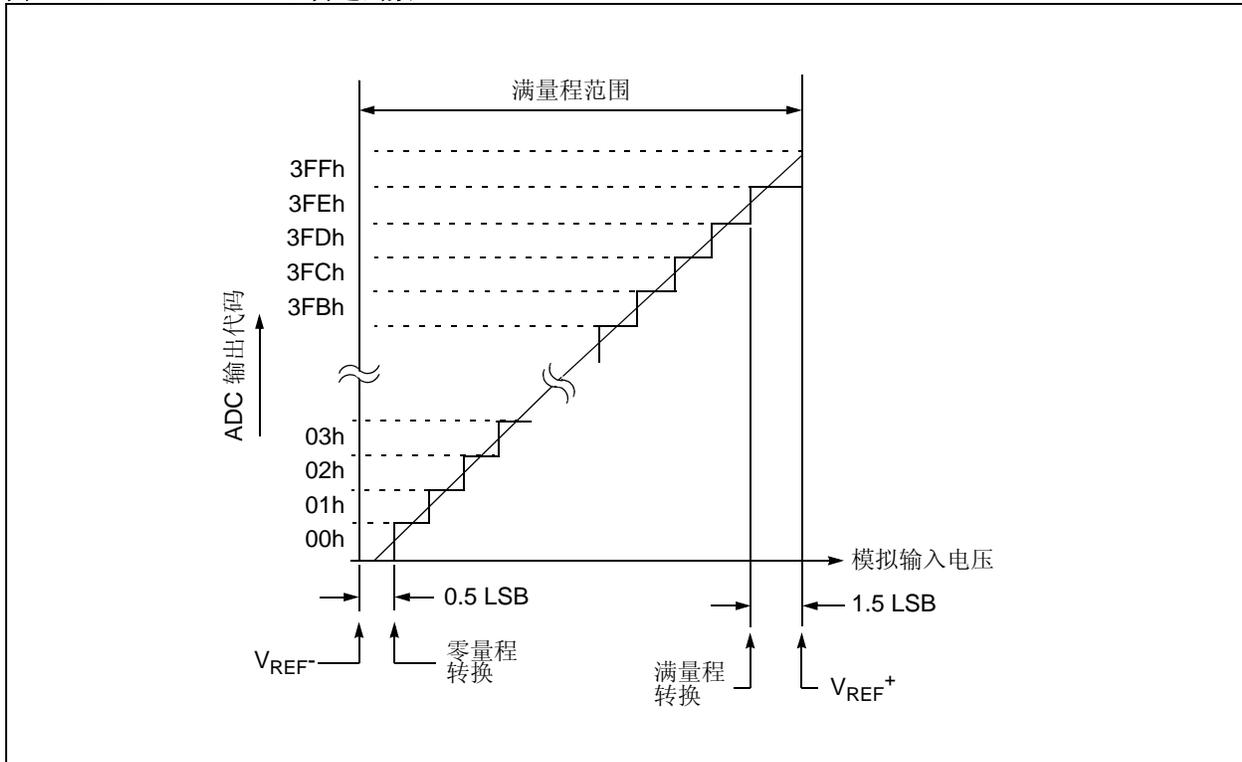


表 21-2: 与 ADC 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
ADCON0	—	CHS4	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	131
ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	—	—	132
ADRESH	—	—	—	—	—	—	ADRES9	ADRES8	132
ADRESL	ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0	132
ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	114
ANSELB	—	ANSB6	ANSB5	ANSB4	—	—	ANSB1	—	119
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	95
PIE1	—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	96
PIR1	—	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	98
TRISGPA	TRISA7	TRISA6	TRISA5	—	TRISA3	TRISA2	TRISA1	TRISA0	113
TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	TRISB1	TRISB0	118

图注: — = 未实现, 读为 0。ADC 模块不使用阴影单元。

# MCP19114/5

---

注:

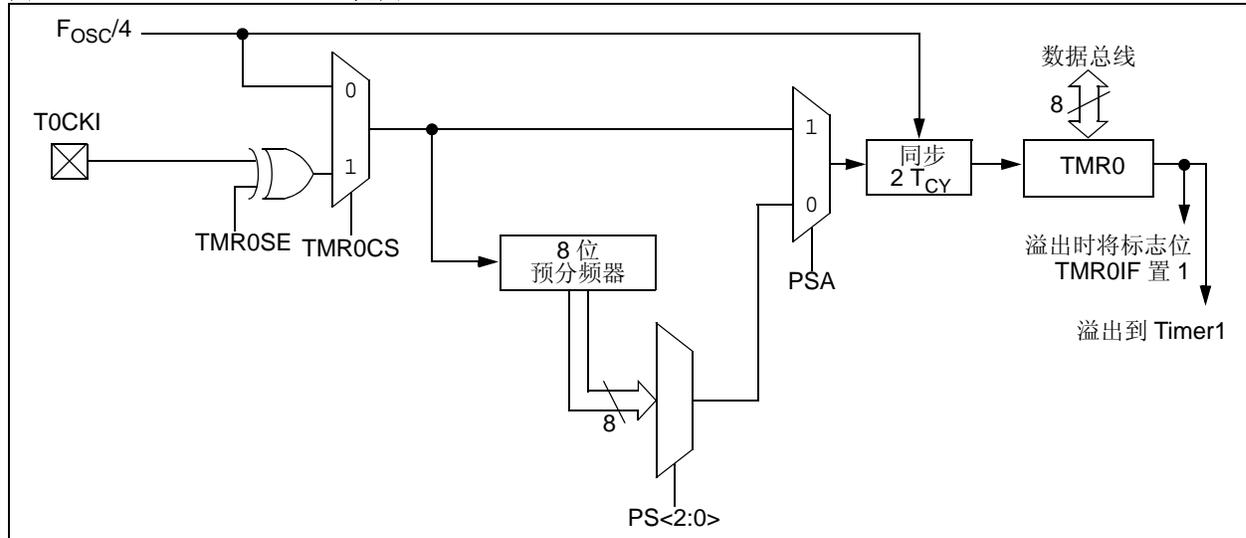
## 22.0 TIMER0 模块

Timer0 模块是具有以下功能的 8 位定时器 / 计数器：

- 8 位定时器 / 计数器寄存器 (TMR0)
- 8 位预分频器
- 可编程的内部或外部时钟源
- 可编程的外部时钟边沿选择
- 溢出时中断

图 22-1 给出了 Timer0 模块的框图。

图 22-1: TIMER0 框图



### 22.1 Timer0 工作原理

Timer0 模块可用作 8 位定时器或 8 位计数器。

#### 22.1.1 8 位定时器模式

如果不使用预分频器，则 Timer0 模块在每个指令周期递增 1。通过将 OPTION\_REG 寄存器的 T0CS 位清零可选择 8 位定时器模式。

如果对 TMR0 执行写操作，则在接下来的两个指令周期，它都不会递增 1。

**注：** 考虑到写入 TMR0 后两个指令周期的延时，可调整写入 TMR0 寄存器的值。

#### 22.1.2 8 位计数器模式

在 8 位计数器模式下，Timer0 模块将在 T0CKI 引脚的每个上升沿或下降沿递增。递增沿由 OPTION\_REG 寄存器的 T0SE 位决定。

通过将 OPTION\_REG 寄存器中的 T0CS 位设置为 1 来选择使用 T0CKI 引脚的 8 位计数器模式。

#### 22.1.3 软件可编程预分频器

单个软件可编程预分频器可与 Timer0 或看门狗定时器 (WDT) 配合使用，但不能同时与两者配合使用。预分频器的分配由 OPTION\_REG 寄存器的 PSA 位控制。要将预分频器分配给 Timer0，PSA 位必须清零。

Timer0 模块具有 8 个预分频比选项，范围为 1:2 至 1:256。可通过 OPTION\_REG 寄存器中的 PS<2:0> 位选择预分频值。为了让 Timer0 模块使用 1:1 预分频值，必须通过将 OPTION\_REG 寄存器的 PSA 位置 1 来禁止预分频器。

该预分频器不可读写。分配给 Timer0 模块时，写入 TMR0 寄存器的所有指令都将清零预分频器。

## 22.1.4 在 TIMER0 与 WDT 模块间切换预分频器

将预分频器分配给 Timer0 或 WDT 时，切换预分频值可能会产生意外的器件复位。将预分频器的分配从 Timer0 切换到 WDT 模块时，必须执行如例 22-1 中所示的指令序列。

### 例 22-1: 更改预分频器 (TIMER0 → WDT)

```
BANKSELTMR0;
CLRWDWT ;Clear WDT
CLRFTMR0;Clear TMR0 and
;prescaler
BANKSELOPTION_REG;
BSF OPTION_REG,PSA;Select WDT
CLRWDWT ;
;
MOVLWb'11111000';Mask prescaler
ANDWFOPTION_REG,W;bits
IORLWb'00000101';Set WDT prescaler
MOVWFOPTION_REG;to 1:32
```

将预分频器的分配从 WDT 切换到 Timer0 模块时，必须执行以下指令序列（见例 22-2）。

### 例 22-2: 更改预分频器 (WDT → TIMER0)

```
CLRWDWT ;Clear WDT and
;prescaler
BANKSELOPTION_REG;
MOVLWb'11110000';Mask TMR0 select and
ANDWFOPTION_REG,W;prescaler bits
IORLWb'00000011';Set prescale to 1:16
MOVWFOPTION_REG;
```

表 22-1: 与 TIMER0 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
INTCON	GIE	PEIE	TOIE	INTE	IOCIE	TOIF	INTF	IOCIF	95
OPTION_REG	RAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	78
TMR0	Timer0 模块寄存器								137*
TRISGPA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	113

图注: — = 未实现单元，读为 0。Timer0 模块不使用阴影单元。

\* 此页提供寄存器信息。

## 22.1.5 TIMER0 中断

当 TMR0 寄存器从 FFh 溢出到 00h 时，Timer0 会产生中断。每次 TMR0 寄存器溢出时，不管是否允许 Timer0 中断，INTCON 寄存器的 TOIF 中断标志位都会置 1。只能用软件将 TOIF 位清零。Timer0 中断允许位是 INTCON 寄存器的 TOIE 位。

**注:** 由于在休眠状态下定时器是关闭的，所以 Timer0 中断无法唤醒处理器。

## 22.1.6 TIMER0 与外部时钟配合使用

当 Timer0 处于计数器模式时，通过在内部相位时钟的 Q2 和 Q4 周期对预分频器输出进行采样来实现 T0CKI 输入与 Timer0 寄存器的同步。因此，外部时钟源的高低电平周期必须满足第 4.0 节“电气特性”中所示的时序要求。

## 22.1.7 休眠期间的工作

处理器处于休眠模式时，Timer0 无法工作。在处理器处于休眠模式时，TMR0 寄存器的内容将保持不变。

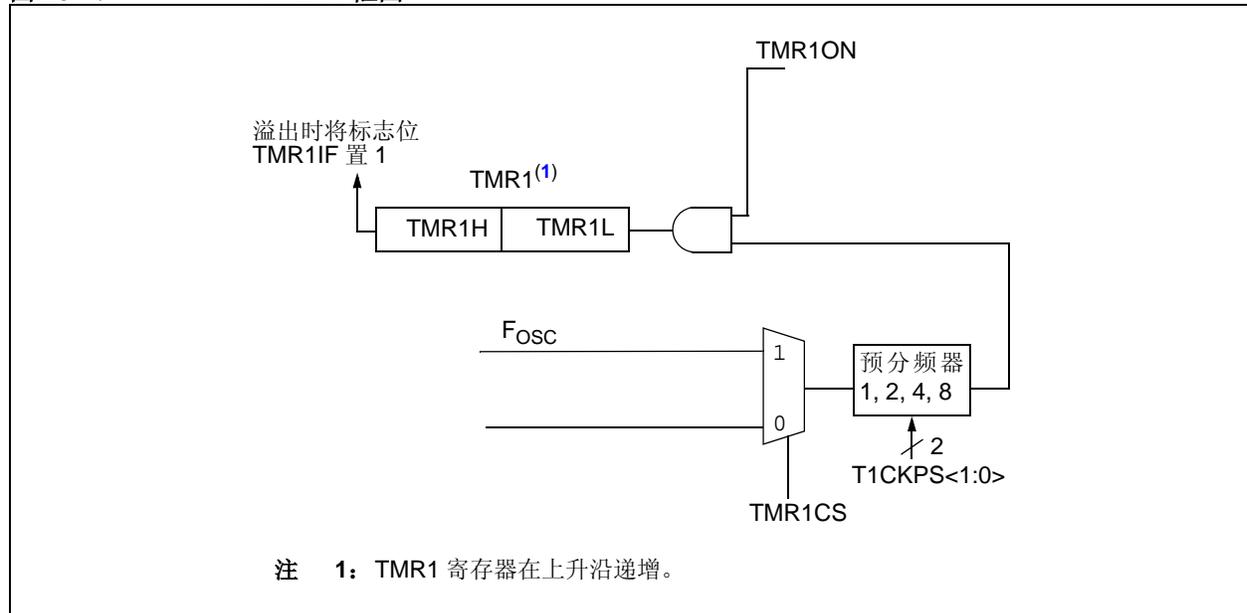
## 23.0 带门控的 TIMER1 模块

Timer1 模块是具有以下功能的 16 位定时器:

- 16 位定时器寄存器对 (TMR1H:TMR1L)
- 可读写 (以上两个寄存器)
- 可选的内部时钟源
- 2 位预分频器
- 溢出时中断

图 23-1 给出了 Timer1 模块的框图。

图 23-1: TIMER1 框图



### 23.1 Timer1 工作原理

Timer1 模块是 16 位递增定时器, 通过 TMR1H:TMR1L 寄存器对可对该模块进行访问。通过写 TMR1H 或 TMR1L 可直接更新计数器。该定时器在每个指令周期递增 1。

配置 T1CON 寄存器中的 TMR1ON 位可启用 Timer1。  
表 23-1 显示了 Timer1 的使能选择。

### 23.2 时钟源选择

T1CON 寄存器的 TMR1CS 位用于选择 Timer1 的时钟源。表 23-1 显示了时钟源选择。

#### 23.2.1 内部时钟源

TMR1H:TMR1L 寄存器对将在  $F_{OSC}$  或  $F_{OSC}/4$  的整数倍 (由 Time1 预分频器决定) 处递增。

例如, 选择  $F_{OSC}$  内部时钟源时, Timer1 寄存器的值在每个指令时钟周期递增 4 个计数。

表 23-1: 时钟源选择

TMR1CS	时钟源
1	8 MHz 系统时钟 ( $F_{OSC}$ )
0	2 MHz 指令时钟 ( $F_{OSC}/4$ )

# MCP19114/5

## 23.3 Timer1 预分频器

Timer1 具有四种预分频比选择，允许对时钟输入进行 1、2、4 或 8 分频。T1CON 寄存器的 T1CKPS 位控制预分频器计数器。不能直接对预分频计数器进行读写操作；但是，通过写 TMR1H 或 TMR1L 可清零预分频计数器。

## 23.4 Timer1 中断

Timer1 寄存器对 (TMR1H:TMR1L) 递增计数到 FFFFh 并计满返回 0000h。当 Timer1 计满返回时，PIR1 寄存器的 Timer1 中断标志位置 1。要允许计满返回中断，应将以下位置 1：

- T1CON 寄存器的 TMR1ON 位
- PIE1 寄存器的 TMR1IE 位
- INTCON 寄存器的 PEIE 位
- INTCON 寄存器的 GIE 位

在中断服务程序中清零 TMR1IF 位可以清除中断。

**注：** 允许中断前，应将 TMR1H:TMR1L 寄存器对以及 TMR1IF 位清零。

寄存器 23-1: T1CON: TIMER1 控制寄存器

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0/	R/W-0
—	—	T1CKPS1	T1CKPS0	—	—	TMR1CS	TMR1ON
bit 7							bit 0

### 图注：

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

- bit 7-6            **未实现：** 读为 0
- bit 5-4            **T1CKPS<1:0>:** Timer1 输入时钟预分频比选择位  
11 = 1:8 预分频比  
10 = 1:4 预分频比  
01 = 1:2 预分频比  
00 = 1:1 预分频比
- bit 3-2            **未实现：** 读为 0
- bit 1              **TMR1CS:** Timer1 时钟源控制位  
1 = 8 MHz 系统时钟 (F<sub>OSC</sub>)  
0 = 2 MHz 指令时钟 (F<sub>OSC</sub>/4)
- bit 0              **TMR1ON:** Timer1 使能位  
1 = 使能 Timer1  
0 = 停止 Timer1，清零 Timer1 门控单稳态触发器

## 23.5 休眠模式下的 Timer1

与其他标准中档 Timer1 模块不同，MCP19114/5 Timer1 模块的时钟仅来自内部系统时钟，因此既不能在休眠模式下运行，也不能用于将器件从休眠模式唤醒。

## 23.6 Timer1 控制寄存器

Timer1 控制寄存器 (T1CON) (如寄存器 23-1 所示) 用于控制 Timer1 以及选择 Timer1 模块的各种功能。

表 23-2: 与 TIMER1 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	95
PIE1	—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	96
PIR1	—	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	98
TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								139*
TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								139*
T1CON	—	—	T1CKPS1	T1CKPS0	—	—	TMR1CS	TMR1ON	140

图注: — = 未实现, 读为 0。Timer1 模块不使用阴影单元。

\* 此页提供寄存器信息。

# MCP19114/5

---

注:

## 24.0 TIMER2 模块

Timer2 模块是具有以下功能的 8 位定时器：

- 8 位定时器寄存器（TMR2）
- 8 位周期寄存器（PR2）
- TMR2 与 PR2 匹配时产生中断
- 可软件编程的预分频比（1:1、1:4 和 1:16）

Timer2 框图请参见图 24-1。

### 24.1 Timer2 工作原理

Timer2 模块的时钟输入是系统时钟（ $F_{osc}$ ）。时钟馈入预分频比选项为 1:1、1:4 或 1:16 的 Timer2 预分频器。然后，预分频器的输出用于使 TMR2 寄存器递增计数。

TMR2 的值和 PR2 的值不断进行比较以确定何时匹配。TMR2 将从 00h 开始递增，直到其与 PR2 中的值匹配。发生匹配时，TMR2 将在下一个递增周期复位为 00h。

Timer2/PR2 比较器的匹配输出用于将 PIR1 寄存器中的 TMR2IF 中断标志位置 1。

TMR2 和 PR2 寄存器均完全可读写。发生任何复位时，TMR2 寄存器均会设置为 00h，PR2 寄存器设置为 FFh。

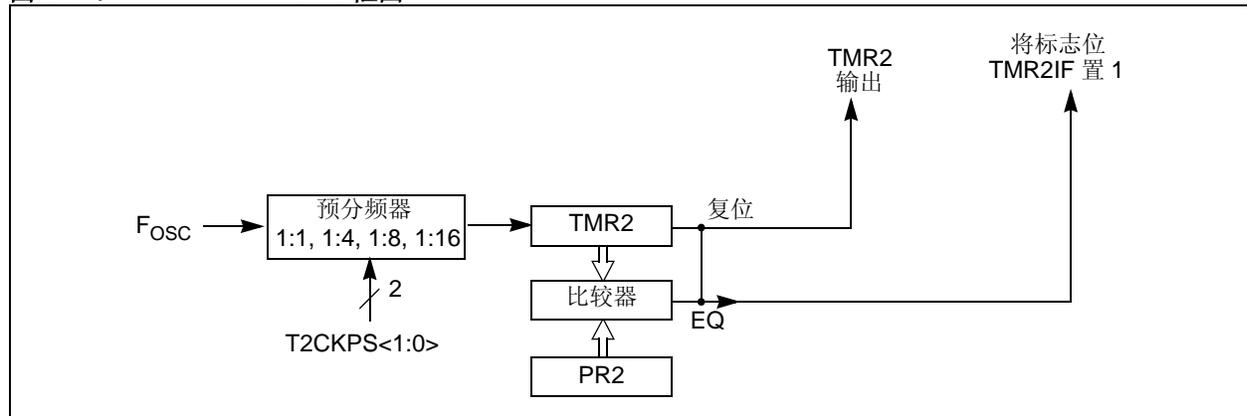
通过将 T2CON 寄存器中的 TMR2ON 位置 1 来使能 Timer2。通过将 TMR2ON 位清零来关闭 Timer2。

Timer2 预分频器由 T2CON 寄存器中的 T2CKPS 位控制。发生以下情况时，预分频器计数器清零：

- 对 TMR2 进行写操作
- 对 T2CON 进行写操作
- 发生任何器件复位（上电复位、 $\overline{MCLR}$  复位、看门狗定时器复位或欠压复位）。

**注：** 写 T2CON 时 TMR2 不会清零。

图 24-1: TIMER2 框图



# MCP19114/5

## 24.2 Timer2 控制寄存器

寄存器 24-1: T2CON: TIMER2 控制寄存器

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0
bit 7						bit 0	

**图注:**

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

bit 7-3            **未实现:** 读为 0  
bit 2              **TMR2ON:** Timer2 使能位  
                    1 = 使能 Timer2  
                    0 = 关闭 Timer2  
bit 1-0            **T2CKPS<1:0>:** Timer2 时钟预分频比选择位  
                    00 = 预分频值为 1  
                    01 = 预分频值为 4  
                    10 = 预分频值为 8  
                    11 = 预分频值为 16

表 24-1: 与 TIMER2 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
INTCON	GIE	PEIE	TOIE	INTE	IOCE	TOIF	INTF	IOCF	95
PIE1	—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	96
PIR1	—	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	98
PR2	Timer2 模块周期寄存器								143*
T2CON	—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0	144
TMR2	8 位 TMR2 时基的保持寄存器								143*

图注: — = 未实现, 读为 0。Timer2 模块不使用阴影单元。

\* 此页提供寄存器信息。

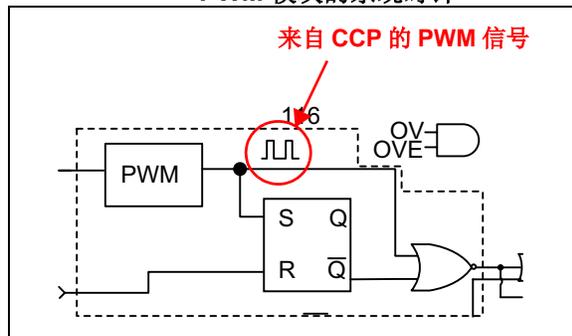
## 25.0 增强型 PWM 模块

MCP19114/5 上实现的 PWM 模块是标准中档单片机中捕捉 / 比较 / PWM (Capture/Compare/PWM, CCP) 模块的缩小版本。该模块仅具有 PWM 模块, 对标准中档单片机中的 PWM 模块稍作修改。在 MCP19114/5 中, PWM 模块用于生成系统时钟或系统振荡器。该系统时钟用于控制 MCP19114/5 开关频率以及设置允许的最大占空比。PWM 模块不会持续调节占空比来控制输出电压。这一过程由模拟控制环路和相关电路完成。

### 25.1 标准脉宽调制模式

CCP 将仅工作在 PWM 模式下。PWM 信号用于设置 MCP19114/5 的工作频率和允许的最大占空比。图 25-1 为 MCP19114/5 框图片断, 显示了来自 CCP 模块的 PWM 信号。

图 25-1: MCP19114/5 框图片断显示来自 PWM 模块的系统时钟



涉及系统时钟 PWM 信号的工作模式有两种。它们分别为独立 (非频率同步) 模式和频率同步模式。

#### 25.1.1 独立 (非频率同步) 模式

MCP19114/5 在独立模式下运行时, PWM 信号用作系统时钟。它将以编程的开关频率和最大占空比 ( $D_{CLOCK}$ ) 工作。编程的最大占空比不会逐周期进行调整来控制 MCP19114/5 系统输出。控制输出所需的占空比 ( $D_{PDRVON}$ ) 由 MCP19114/5 模拟环路和相关电路调整。但是,  $D_{CLOCK}$  可用于设置允许的最大  $D_{PDRVON}$  值。

#### 公式 25-1:

$$D_{BUCK} < 1 - D_{CLOCK}$$

#### 25.1.2 开关频率同步模式

MCP19114/5 可以编程为开关频率主器件或从器件。除系统时钟将同时应用于 GPA1 外, 主器件的工作方式与第 25.1.1 节“独立 (非频率同步) 模式”中所述相同。

从器件将接收 GPA1 上的主系统时钟。该主系统时钟将与 TIMER2 模块的输出进行逻辑或运算。此逻辑或运算信号会将 PWMRL 锁存到 PWMRH, 将 PWMPHL 锁存到 PWMPHH。

图 25-1 显示了 PWM 模式下 CCP 模块的简化框图。

PWMPHL 寄存器允许将相移添加到从器件系统时钟。

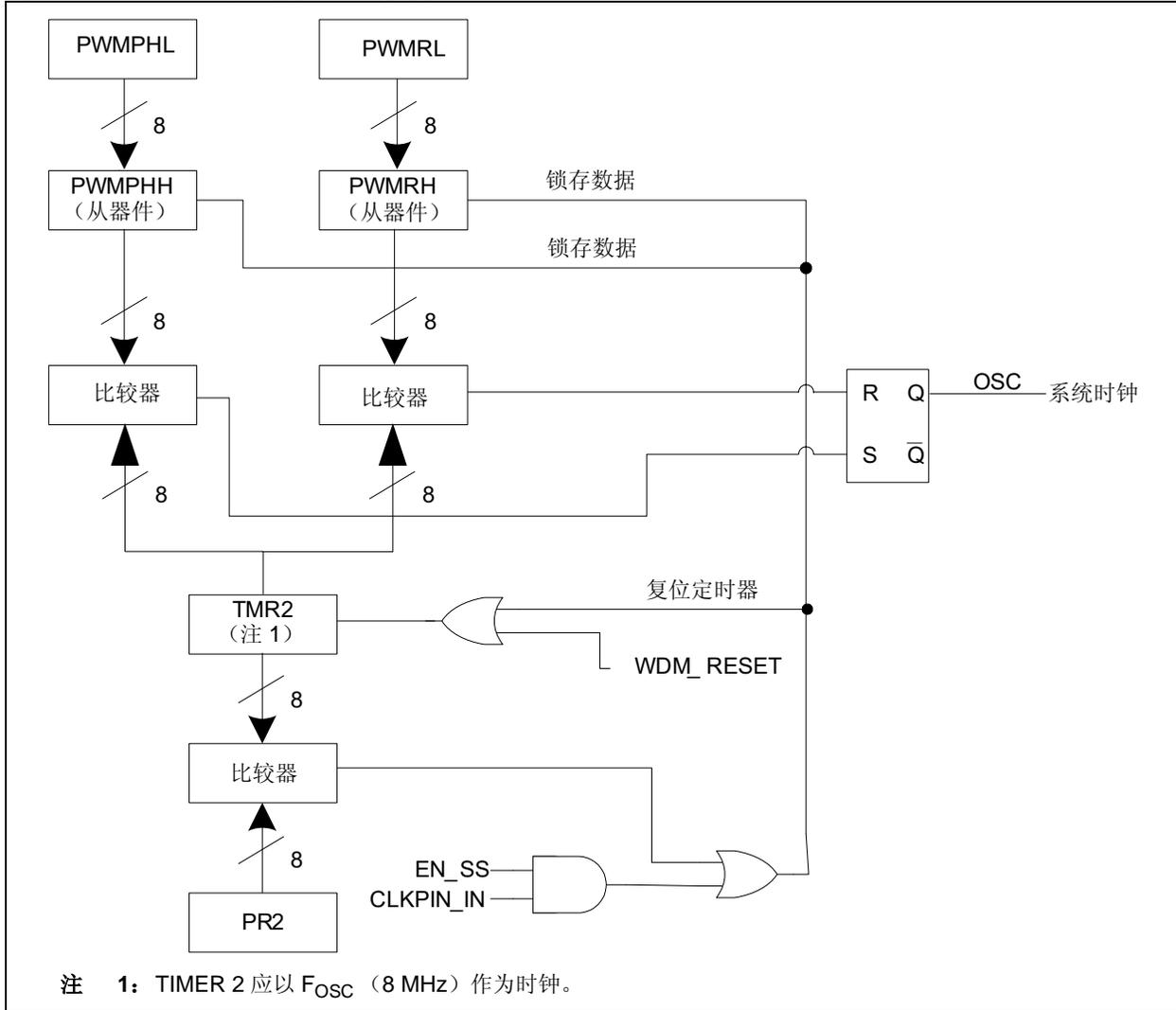
这就要求 MCP19114/5 从器件系统时钟起点与主器件系统时钟之间存在一定偏移, 偏移量由编程设定。从器件相移通过写入 PWMPHL 寄存器来指定。从器件相移可利用下面的公式计算。

#### 公式 25-2:

$$\text{从器件相移} = PWMPHL \times T_{OSC} \times (T2 \text{ 预分频值})$$

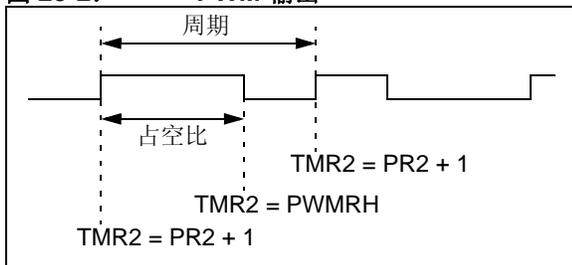
# MCP1914/5

图 25-1: 简化的 PWM 框图



PWM 输出 (图 25-2) 具有时基 (周期) 和输出保持高电平 (占空比) 的时间。PWM 的频率是周期的倒数 (1/周期)。

图 25-2: PWM 输出



## 25.1.3 PWM 周期

PWM 周期通过写入 PR2 寄存器来指定。PWM 周期可利用下面的公式计算。

公式 25-3:

$$PWM \text{ 周期} = [(PR2) + 1] \times T_{OSC} \times (T2 \text{ 预分频值})$$

当 TMR2 中的值与 PR2 中的值相等时, 在下一个递增周期中将发生以下 2 个事件:

- TMR2 被清零
- PWM 占空比从 PWMRL 锁存到 PWMRH

## 25.1.4 PWM 占空比 (D<sub>CLOCK</sub>)

PWM 占空比 (D<sub>CLOCK</sub>) 通过写入 PWMRL 寄存器来指定。可用的分辨率最多为 8 位。下面的公式是用来计算 PWM 占空比 (D<sub>CLOCK</sub>)。

### 公式 25-4:

$$PWM \text{ 占空比} = PWMRL \times T_{OSC} \times (T2 \text{ 预分频值})$$

PWMRL 位可以随时写入，但在 PR2 与 TMR2 之间发生匹配后，占空比值才会锁存到 PWMRH 中。

## 25.2 休眠期间的工作

当器件处于休眠模式下时，分配的定时器将不会递增，模块的状态也不会改变。如果 CLKPIN 引脚正在驱动一个值，则会继续驱动该值。当器件被唤醒时，它将从该状态继续。

表 25-1: 与 PWM 模块相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
MODECON	MSC1	MSC0	RFB	—	—	—	—	—	51
T2CON	—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0	144
PR2	Timer2 模块周期寄存器								143
PWMRL	PWM 寄存器低字节								145*
PWMPHL	相移低字节								145*

图注: — = 未实现单元，读为 0。PWM 模式不使用阴影单元。

\* 此页提供寄存器信息。

# MCP19114/5

---

注:

## 26.0 双捕捉 / 比较 (CCD) 模块

MCP19114/5 器件实现了 CCD 模块。该模块是基于标准 CCP 模块的新模块。它具有两个仅捕捉和比较寄存器集，不具有 PWM 功能。

### 26.1 捕捉模式

在捕捉模式下，当 DIMI 引脚上发生事件时，CCxRH:CCxRL 寄存器集将捕捉 TMR1 寄存器的 16 位值。事件定义为下列情况之一：

- 每个下降沿
- 每个上升沿
- 每 4 个上升沿
- 每 16 个上升沿

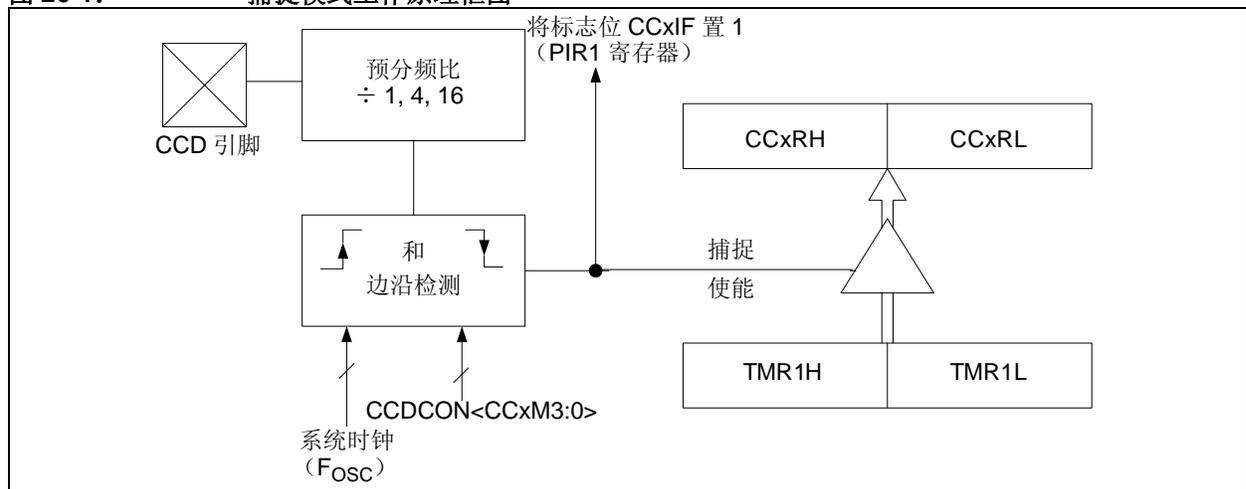
事件类型通过控制位 CCxM3:CCxM0（寄存器集 1 的 CCDCON<3:0> 或寄存器集 2 的 CCDCON<7:4>）进行配置。进行捕捉时，中断请求标志位 CCxIF（寄存器集 1 的 PIR1<2> 或寄存器集 2 的 PIR1<3>）置 1。该中断标志必须用软件清零。如果在读取寄存器集中的值之前发生了另一次捕捉，那么之前捕捉的值将会被新值覆盖。

#### 26.1.1 CCX 引脚配置

在捕捉模式下，应通过将 DIMI 引脚的 TRIS 位置 1，将该引脚配置为输入引脚。

**注：** 如果将 DIMI 引脚配置为输出引脚，则对该端口的写操作可能引发一次捕捉事件。

图 26-1: 捕捉模式工作原理框图



#### 26.1.2 TIMER1 模式选择

欲使 CCD 模块使用捕捉功能，Timer1 必须工作在指令时钟下。如果 Timer1 工作在 8 MHz 时钟下，捕捉功能可能无法正常工作。

#### 26.1.3 软件中断

当捕捉模式改变时，可能会产生错误的捕捉中断。用户应保持捕捉中断允许位清零，以避免错误中断，还应在工作模式发生任何改变之后清零中断标志位 CCxIF。

#### 26.1.4 CCD 预分频器

通过 CCxM3:CCxM0 位，可以指定 4 种预分频比设置。每当 CCD 寄存器集被禁止或未设置为捕捉模式时，预分频器计数器就会被清零。任何复位都将清零预分频器计数器。

从一个捕捉预分频比切换到另一个捕捉预分频比可能会产生中断。另外，预分频器计数器将不会清零，因此，第一个捕捉可能从非零预分频比开始。建议在改变预分频比之前禁止寄存器集（CCxM3:0 = 00xx）。

# MCP19114/5

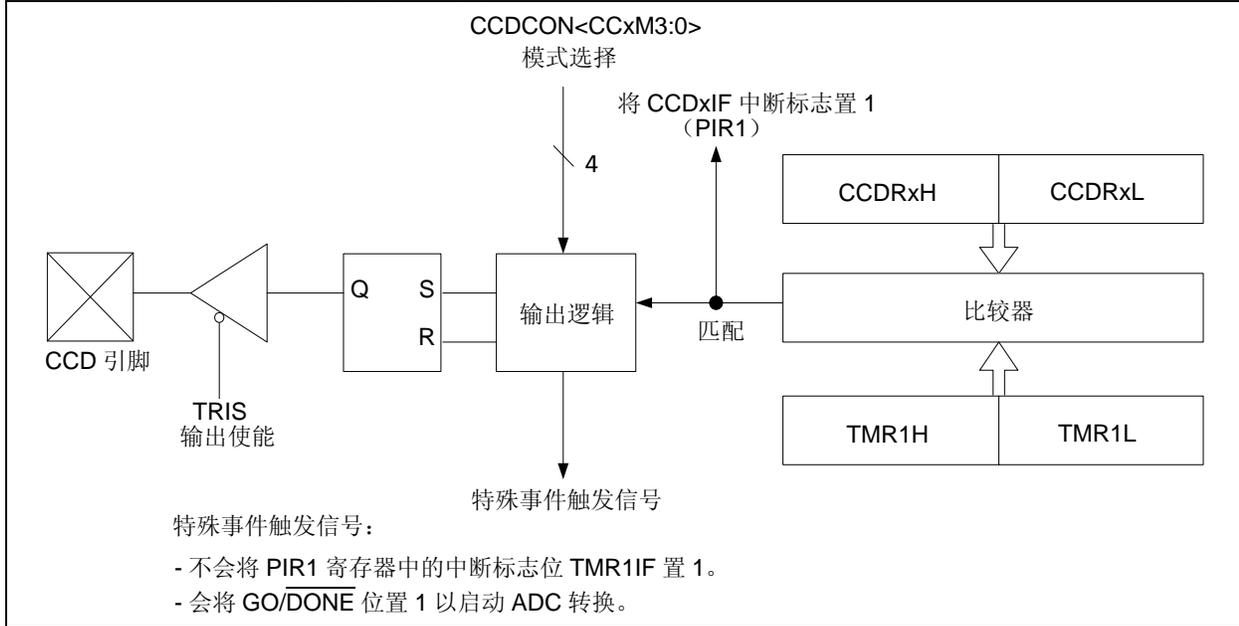
## 26.2 比较模式

在比较模式下，始终用 16 位 CCDRx 寄存器值与 TMR1 寄存器对的值进行比较。当两者匹配时，CMPx 引脚可能会出现以下几种情况：

- 输出高电平
- 输出低电平
- 翻转
- 保持不变

引脚的动作由 CCxM3:CCxM0 控制位的值决定。同时，中断标志位 CCP1IF 置 1。

图 26-2: 比较模式工作原理框图



### 26.2.1 CMPx 引脚配置

用户必须通过将 CMPx 引脚的 TRIS 位清零，将该引脚配置为输出。

**注：** 清零 CCxM<3:0> 位会将 CMPx 比较输出锁存器设置为默认状态。这不是 GPIO 引脚数据锁存器。匹配时置 1 或匹配时翻转的默认状态是 0，而匹配时清零的默认状态是 1。

### 26.2.2 TIMER1 模式选择

欲使 CCD 模块使用比较功能，Timer1 必须工作在指令时钟下。如果 Timer1 工作在 8 MHz 时钟下，比较功能可能无法正常工作。

### 26.2.3 软件中断模式

当选择产生软件中断模式时，CCP1 引脚不受影响。CCP1IF 位置 1 时，将导致 CCx 中断（如果允许）。

### 26.2.4 特殊事件触发器

在此模式下，产生内部硬件触发信号，可用于启动操作。CCD 的特殊事件触发信号输出不会复位 TMR1 寄存器对，如果使能了 A/D 模块，将启动 A/D 转换。

**注：** CCD 模块的特殊事件触发信号不会将中断标志位 TMR1IF（PIR1 寄存器的 bit 0）置 1。

## 26.3 双捕捉 / 比较寄存器

双捕捉 / 比较模块是基于标准 CCP 的新模块。它不具有 PWM 功能。

### 寄存器 26-1: CCDCON: 双捕捉 / 比较控制模块

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| CC2M3 | CC2M2 | CC2M1 | CC2M0 | CC1M3 | CC1M2 | CC1M1 | CC1M0 |
| bit 7 |       |       |       |       |       |       | bit 0 |

#### 图注:

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

bit 7-4

#### CC2M<3:0>: CC 寄存器集 2 模式选择位

- 00xx = 捕捉 / 比较关闭 (复位该模块)
- 0100 = 捕捉模式, 每个下降沿捕捉
- 0101 = 捕捉模式, 每个上升沿捕捉
- 0110 = 捕捉模式, 每 4 个上升沿捕捉
- 0111 = 捕捉模式, 每 16 个上升沿捕捉
- 1000 = 比较模式, 匹配时输出置 1 (CC2IF 位置 1)
- 1001 = 比较模式, 匹配时输出清零 (CC2IF 位置 1)
- 1010 = 比较模式, 匹配时输出电平翻转 (CC2IF 位置 1)
- 1011 = 保留
- 11xx = 比较模式, 匹配时产生软件中断 (CC2IF 位置 1, CMP2 引脚不受影响且配置为 I/O)
- 1111 = 比较模式, 触发特殊事件 (CC2IF 位置 1; **CC2 不会复位 TMR1<sup>(1)</sup>, 如果已使能 A/D 模块, 会启动 A/D 转换。CMP2 引脚不受影响且配置为 I/O 端口。**)

bit 3-0

#### CC1M<3:0>: CC 寄存器集 1 模式选择位

- 00xx = 捕捉 / 比较关闭 (复位该模块)
- 0100 = 捕捉模式, 每个下降沿捕捉
- 0101 = 捕捉模式, 每个上升沿捕捉
- 0110 = 捕捉模式, 每 4 个上升沿捕捉
- 0111 = 捕捉模式, 每 16 个上升沿捕捉
- 1000 = 比较模式, 匹配时输出置 1 (CC1IF 位置 1)
- 1001 = 比较模式, 匹配时输出清零 (CC1IF 位置 1)
- 1010 = 比较模式, 匹配时输出电平翻转 (CC1IF 位置 1)
- 1011 = 保留
- 11xx = 比较模式, 匹配时产生软件中断 (CC1IF 位置 1, CMP1 引脚不受影响且配置为 I/O)
- 1111 = 比较模式, 触发特殊事件 (CC1IF 位置 1; **CC1 复位 TMR1, 并启动 A/D 转换 (如果已使能 A/D 模块)。CMP1 引脚不受影响且配置为 I/O 端口。**)

注 1: 当设置比较中断时, PIC 通常会复位 TMR1。该模块不会复位 TMR1。

# MCP19114/5

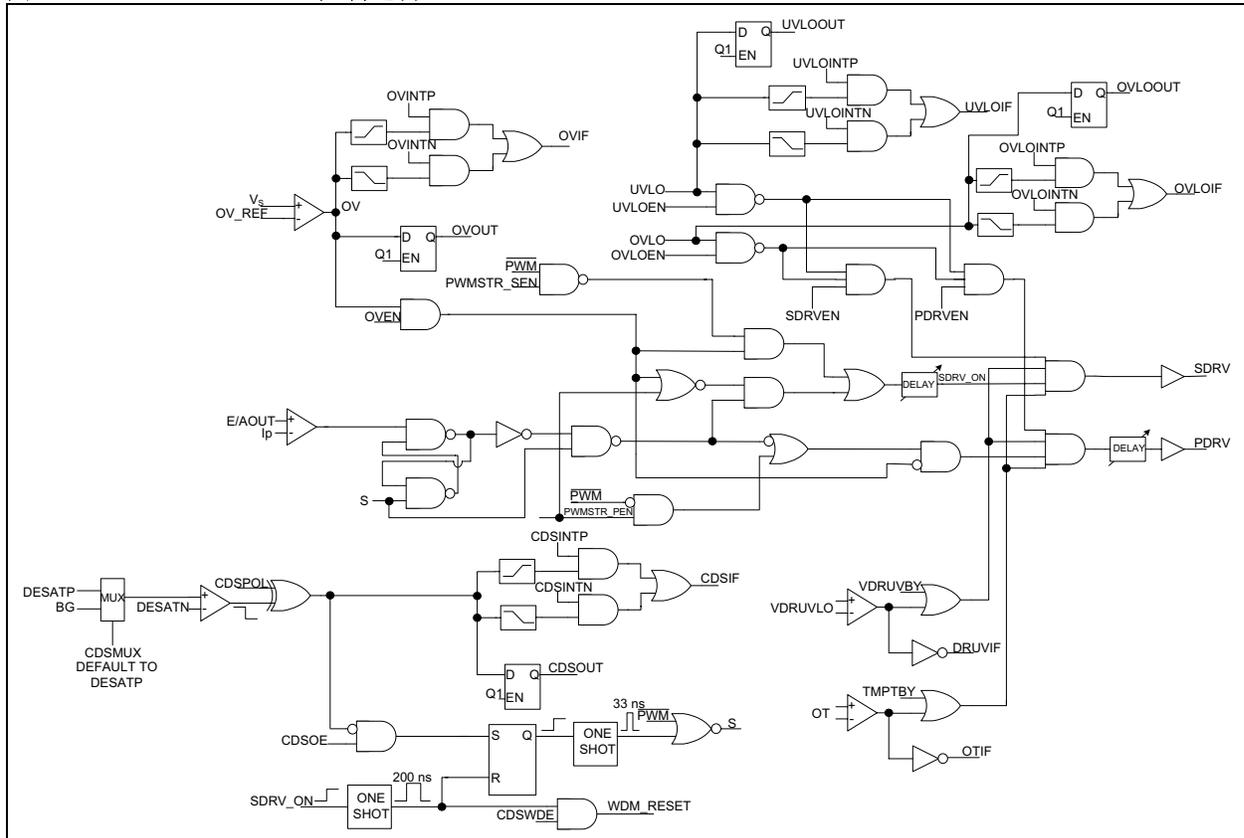
---

注:

## 27.0 PWM 控制逻辑

PWM控制逻辑实现了标准的比较器模块，可识别输入欠压、输入过压和退饱和检测之类的事件。控制逻辑在硬件上执行操作来适当使能 / 禁止输出驱动（PDRV/SDRV），以及设置由软件读取的相应中断标志位。该控制逻辑还定义了正常 PWM 操作。关于控制逻辑中各个位的定义，请参见特殊功能寄存器（SFR）章节。

图 27-1: PWM 控制逻辑



# MCP19114/5

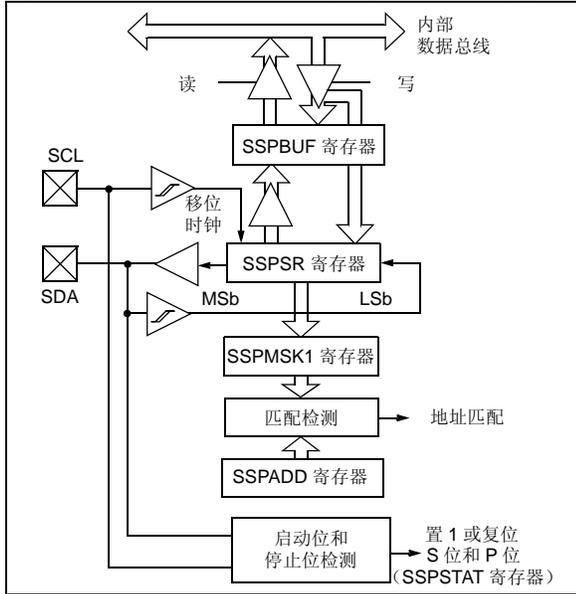
---

注:



# MCP19114/5

图 28-2: MSSP 框图 (I<sup>2</sup>C™ 从模式)



## 28.2 I<sup>2</sup>C 模式概述

I<sup>2</sup>C 总线是一种多主器件串行数据通信总线。器件在由主器件启动通信的主 / 从器件环境中进行通信。从器件通过寻址进行控制。

I<sup>2</sup>C 总线规定了两种信号连接：

- 串行时钟 (SCL)
- 串行数据 (SDA)

SCL 和 SDA 连接都是双向的漏极开路线路，它们都需要使用用于电源电压的上拉电阻。线路下拉为地电压时，信号视为逻辑 0；线路保持悬空时，信号视为逻辑 1。

图 28-3 显示了配置为主器件和从器件的两个器件之间的典型连接。

I<sup>2</sup>C 总线工作时可以有一个或多个主器件，以及一个或多个从器件。

对于给定器件，有 4 种可能的工作模式：

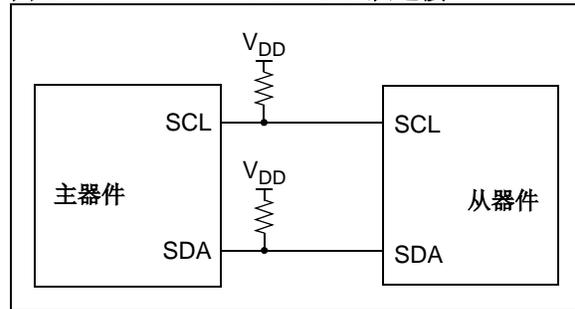
- 主发送模式  
(主器件向从器件发送数据)
- 主接收模式  
(主器件从从器件接收数据)
- 从发送模式  
(从器件向主器件发送数据)
- 从接收模式  
(从器件从主器件接收数据)

要开始进行通信，主器件需要以主发送模式启动。主器件送出启动位，后面跟随它希望进行通信的从器件的地址字节。后面再跟随单个读 / 写位，该位决定主器件是向从器件发送数据还是从从器件接收数据。

如果总线上存在所请求的从器件，从器件会使用应答位 (也称为 ACK) 进行响应。然后，主器件会以发送模式或接收模式继续通信，从器件则以互补模式 (分别为接收模式或发送模式) 继续通信。

启动位由 SCL 线保持为高电平时 SDA 线的由高至低跳变来指示。地址和数据字节随后送出，先发送最高有效位 (MSb)。在主器件希望从从器件读取数据时，送出的读 / 写位为逻辑 1，在主器件希望向从器件写入数据时，该位为逻辑 0。

图 28-3: I<sup>2</sup>C™ 主 / 从连接



应答位 ( $\overline{\text{ACK}}$ ) 是低电平有效信号，它会将 SDA 线保持为低电平，用于指示发送器，从器件已接收到发送数据，并已准备好接收更多数据。

数据位的跳变总是在 SCL 线保持低电平时执行。在 SCL 线保持高电平时发生的跳变用于指示启动位和停止位。

如果主器件希望向从器件写入数据，则它会重复发送一个字节的 数据，而从器件则在接收每个字节之后使用 ACK 位进行响应。在该示例中，主器件处于主发送模式，从器件处于从接收模式。

如果主器件希望从从器件读取数据，则它会从从器件重复接收一个字节的 数据，并在接收每个字节之后使用 ACK 位进行响应。在该示例中，主器件处于主接收模式，从器件处于从发送模式。

在传输最后一个数据字节之后，主器件可以通过发送停止位来结束数据发送。如果主器件处于接收模式，它会发送停止位来代替最后一个 ACK 位。停止位由 SCL 线保持为高电平时 SDA 线的由低至高跳变来指示。

在某些情况下，主器件可能希望维持对总线的控制，并重新启动另一次数据发送。如果是这样，主器件可以在它处于接收模式时，发送另一个启动位来代替停止位或最后一个 ACK 位。

I<sup>2</sup>C 总线规定了三种报文协议：

- 主器件向从器件写数据的单一报文。
- 主器件从从器件读数据的单一报文。
- 主器件对一个或多个从器件启动至少两次写操作或读操作，或者读写操作组合的组合报文。

在一个器件发送逻辑 1（或将线路保留悬空），第二个器件发送逻辑 0（或将线路保持为低电平）时，第一个器件会检测到线路不为逻辑 1。这种检测在用于 SCL 线时，称为时钟延长。时钟延长为从器件提供了一种控制数据流的机制。这种检测在用于 SDA 线时，称为仲裁。仲裁可以确保任意时刻只有一个主器件在进行通信。

## 28.2.1 时钟延长

在从器件尚未完成数据处理时，它可以通过时钟延长这一过程来延迟更多数据的传输。寻址到的从器件可以在接收或发送一位数据之后将 SCL 时钟线保持为低电平，指示它尚未准备好继续。与从器件进行通信的主器件将会尝试将 SCL 线拉高，以传输下一位数据，但它会检测到时钟线尚未被释放。由于 SCL 连接是漏极开路，所以从器件可以一直将线路保持为低电平，直到它准备好继续通信为止。

通过时钟延长，无法与发送器保持同速的接收器可以控制传入数据流。

## 28.2.2 仲裁

每个主器件都必须监视总线上是否出现启动位和停止位。如果器件检测到总线正忙，则在总线恢复为空闲状态之前，它无法开始新的报文。

但是，可能会有两个主器件尝试同时或近乎同时启动数据发送。发生这种情况时，将会开始仲裁过程。每个发送器会检查 SDA 数据线的电平，并将它与自己期望的电平进行比较。第一个发现两个电平不匹配的发送器会在仲裁中失败，必须停止在 SDA 线上发送数据。

例如，如果一个发送器将 SDA 线保持为逻辑 1（保留悬空），而第二个发送器将它保持为逻辑 0（下拉为低电平），则结果是 SDA 线将为低电平。那么，第一个发送器会发现线路电平与期望电平不同，并断定有另一个发送器正在进行通信。

发现电平不同的第一个发送器将是仲裁失败的发送器，必须停止驱动 SDA 线。如果该发送器同时也是主器件，则它还必须停止驱动 SCL 线。然后，它可以在尝试重新启动数据发送之前监视线路上是否出现停止条件。与此同时，另一个未发现 SDA 线的期望电平与实际电平不同的器件将继续原来的数据发送。它可以无需进行任何复杂处理，因为到目前为止，发送条件与所期望的完全相同，没有其他发送器对报文产生干扰。

当主器件对多个从器件进行寻址时，也会对从发送模式进行仲裁，但这种情况较少见。

如果有两个主器件在地址阶段向两个不同的从器件发送报文，则发送较小从器件地址的主器件总是会在仲裁中获胜。当两个主器件向同一从器件地址发送报文时，地址有时会指向多个从器件，仲裁过程必须继续进入到数据阶段。

仲裁通常极少发生，但它是正确支持多主器件所必需的过程。

# MCP19114/5

## 28.3 I<sup>2</sup>C 模式操作

所有 MSSP I<sup>2</sup>C 通信都是针对字节的，并且会先移出 MSb。有 6 个 SFR 寄存器和 2 个中断标志用作模块与 PIC 单片机和用户软件的接口。模块通过两个引脚 SDA 和 SCL 来与其他外部 I<sup>2</sup>C 器件进行通信。

### 28.3.1 字节格式

I<sup>2</sup>C 中的所有通信都采用 9 位形式。从主器件向从器件（或者反之）发送一个字节之后，将会送回一个应答位。在 SCL 线第 8 个下降沿之后，在 SDA 上输出数据的器件会将该引脚改为输入，并在下一个时钟脉冲读入应答值。

时钟信号 SCL 由主器件提供。在 SCL 信号为低电平时，数据可以有效地更改，并且在时钟上升沿进行采样。在 SCL 线为高电平时，SDA 线上的电平变化定义总线上的一些特殊条件，以下章节会对此进行说明。

### 28.3.2 I<sup>2</sup>C 术语的定义

在 I<sup>2</sup>C 通信的描述中存在一些用语和术语，它们具有特定于 I<sup>2</sup>C 的定义。表 28-1 定义了此类词语的用法，在本文档其他部分中，将不加说明地使用它们。该表根据 Philips I<sup>2</sup>C 规范改写。

### 28.3.3 SDA 和 SCL 引脚

在 SSPEN 位置 1 的情况下选择任意 I<sup>2</sup>C 模式时，SCL 和 SDA 引脚将会强制设为漏极开路。用户应通过将相应的 TRIS 位置 1 来将这些引脚设置为输入。

**注：** 在使能 I<sup>2</sup>C 模式时，数据将设为输出 0。

### 28.3.4 SDA 保持时间

SDA 引脚的保持时间通过 SSPCON3 寄存器的 SDAHT 位进行选择。保持时间是 SDA 在 SCL 的下降沿之后保持有效的时间。将 SDAHT 位置 1 可以选择最低 300 ns 的较长保持时间，这对于电容较大的总线会有帮助。

表 28-1: I<sup>2</sup>C™ 总线术语

术语	说明
发送器	将数据移送到总线上的器件。
接收器	从总线上移入数据的器件。
主器件	启动数据传输、产生时钟信号和终止数据传输的器件。
从器件	主器件寻址到的器件。
多主器件	有多个器件可以启动数据传输的总线。
仲裁	用于确保每次只有一个主器件控制总线的过程。仲裁获胜可以确保报文不会被损坏。
同步	用于将总线上两个或更多器件的时钟进行同步的过程。
空闲	没有任何主器件在控制总线，并且 SDA 和 SCL 线均为高电平。
有效	每当有一个或多个主器件在控制总线时。
可寻址的从器件	已接收到匹配地址、并且正在由主器件提供时钟的从器件。
匹配地址	送入从器件中、并与 SSPADDx 中的存储值匹配的地址字节。
写请求	从器件接收到 $R/\overline{W}$ 位清零的匹配地址，并已准备好移入数据。
读请求	主器件发送 $R/\overline{W}$ 位置 1 的地址字节，表示要求从器件在时钟控制下将数据移出。从器件在接收到该地址字节后会立即移出所有数据字节，直到发生重复启动或停止条件。
时钟延长	总线上的器件通过将 SCL 保持为低电平来暂停通信的时间。
总线冲突	每当模块进行输出并期望 SDA 线为高电平，却采样到 SDA 线为低电平时。

### 28.3.5 启动条件

I<sup>2</sup>C 规范将启动条件定义为在 SCL 线为高电平时，SDA 从高电平变为低电平状态。启动条件总是由主器件产生，指示总线从空闲状态变为有效状态。图 28-4 给出了启动和停止条件的波形图。

如果模块在将 SDA 线置为低电平之前采样到 SDA 线为低电平，则会在产生启动条件时发生总线冲突。这一点不符合 I<sup>2</sup>C 规范，该规范规定启动时不能发生总线冲突。

### 28.3.6 停止条件

停止条件定义为在 SCL 线为高电平时，SDA 线从低电平变为高电平状态。

**注：** 在停止条件生效之前，必须至少出现一个 SCL 低电平时间，因此，如果 SDA 线变为低电平然后再次变为高电平，而 SCL 线保持高电平，则只会检测到启动条件。

### 28.3.7 重复启动条件

重复启动条件在每次停止条件有效的时候有效。如果主器件希望在终止当前传输之后仍占用总线，主器件可以发出重复启动条件。重复启动对从器件产生的影响与启动条件相同，即复位所有从器件逻辑并使之准备随时钟移入地址。主器件可以寻址同一个或另一个从器件。

在 10 位寻址从模式下，要从寻址到的从器件中随时钟移出数据，主器件需要产生重复启动条件。从器件完全寻址（高地址字节和低地址字节均匹配）之后，主器件可以发出重复启动条件和 R/W 位置 1 的高地址字节。然后，从器件逻辑会采用该时钟，并准备送出数据。

在 10 位模式下，与 R/W 清零的地址字节完全匹配后，前一次匹配标志会置 1 并保持置 1。在产生停止条件之前，R/W 清零的高地址或高地址匹配都会失败。

### 28.3.8 启动 / 停止条件中断屏蔽

SSPCON3 寄存器的 SCIE 和 PCIE 位可以用于允许在通常不支持中断功能的从模式下产生中断。对于已允许启动和停止检测中断的从模式，这两位没有任何作用。

图 28-4: I<sup>2</sup>C™ 启动和停止条件

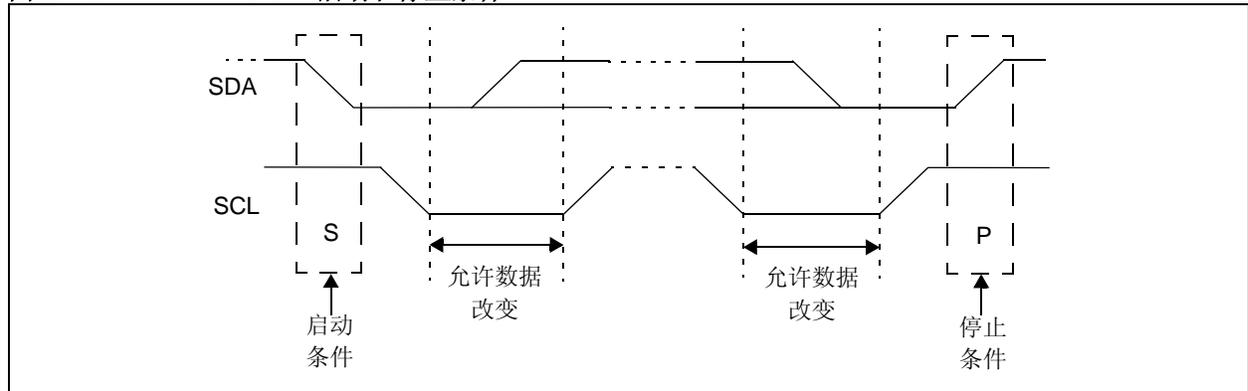
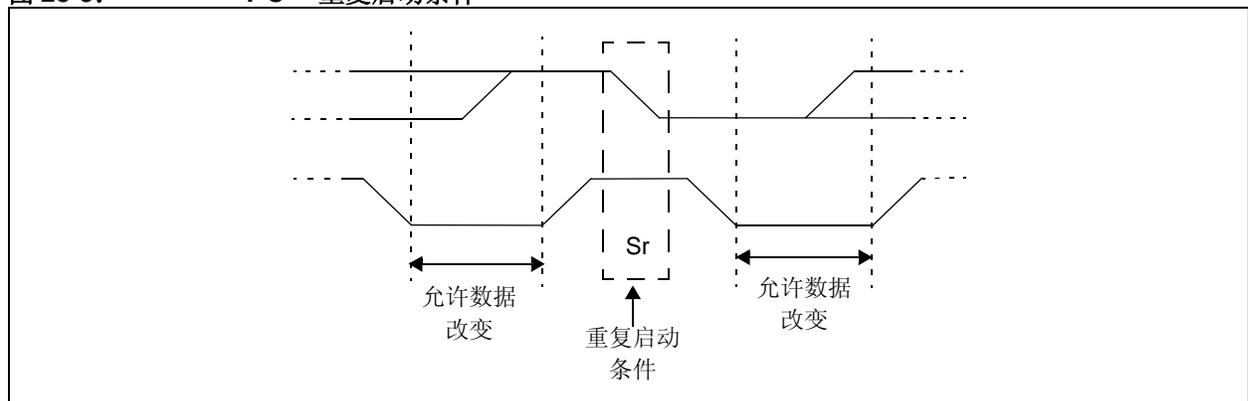


图 28-5: I<sup>2</sup>C™ 重复启动条件



## 28.3.9 应答序列

I<sup>2</sup>C 中，所有传输字节的第 9 个 SCL 脉冲都专门用作应答信号。它使接收器件可以通过将 SDA 下拉为低电平来响应发送器。发送器在该时间内必须释放对线路的控制，以移入响应信号。应答 (ACK) 是低电平有效信号，它会将 SDA 线下拉为低电平，用于指示发送器器件已接收到发送数据并已准备好接收更多数据。

ACK 的结果会被放入 SSPCON2 寄存器的 ACKSTAT 位中。

当 AHEN 和 DHEN 位置 1 时，从器件软件允许用户设置要回送到发送器的 ACK 值。用户可以通过置 1/ 清零 SSPCON2 寄存器的 ACKDT 位来决定响应。

如果 SSPCON3 寄存器的 AHEN 和 DHEN 位清零，从器件硬件会产生 ACK 响应。

有一些条件下，从器件不会发送 ACK。如果在接收到数据字节时，SSPSTAT 寄存器的 BF 位或 SSPCON1 寄存器的 SSPOV 位置 1，从器件将不发送 ACK 信号。

对模块进行寻址时，在总线上的第 8 个 SCL 下降沿之后，SSPCON3 寄存器的 ACKTIM 位会置 1。ACKTIM 位指示有效总线的应答时间。ACKTIM 状态位仅在 AHEN 位或 DHEN 位使能时有效。

## 28.4 I<sup>2</sup>C 从模式操作

MSSP 从模式可以在 4 种模式下工作，这些模式通过 SSPCON1 寄存器的 SSPM 位进行选择。这些模式可以分为 7 位和 10 位寻址模式。10 位寻址模式的工作方式与 7 位寻址模式相同，只是在处理较大地址时需要一些额外的开销。

带启动位和停止位中断的模式的工作方式与其他模式相同，只是在检测到启动、重复启动或停止条件时，另外会将 SSPIF 置 1。

### 28.4.1 从模式地址

SSPADD 寄存器包含从模式地址。在启动或重复启动条件之后接收到的第一个字节将与该寄存器中的存储值进行比较。如果字节匹配，则值会被装入 SSPBUF 寄存器，并产生中断。如果值不匹配，则模块会进入空闲状态，并且不会向软件指示是否发生了什么情况。

SSPMSK1 寄存器会影响地址匹配过程。更多信息，请参见第 28.4.10 节“SSPMSK1 寄存器”。

### 28.4.2 SECOND 从模式地址

SSPADD2 寄存器包含另一个 7 位从模式地址。在启动或重复启动条件之后接收到的第一个字节将与该寄存器中的存储值进行比较。如果字节匹配，则值会被装入 SSPBUF 寄存器，并产生中断。如果值不匹配，则模块会进入空闲状态，并且不会向软件指示是否发生了什么情况。

SSPMSK2 寄存器会影响地址匹配过程。更多信息，请参见第 28.4.10 节“SSPMSK1 寄存器”。

#### 28.4.2.1 I<sup>2</sup>C 从器件 7 位寻址模式

在 7 位寻址模式下，在确定地址是否匹配时，所接收数据字节的 LSb 会被忽略。

#### 28.4.2.2 I<sup>2</sup>C 从器件 10 位寻址模式

在 10 位寻址模式下，接收到的第一个字节将与二进制值“1 1 1 1 0 A9 A8 0”进行比较。A9 和 A8 是 10 位地址的两个 MSb，存储在 SSPADD 寄存器的 bit 2 和 bit 1 中。

在应答高字节之后，UA 位会置 1，SCL 会保持低电平，直到用户使用低地址更新 SSPADD 为止。在低地址字节随时钟移入之后，全部 8 位将与 SSPADD 中的低地址值进行比较。即使地址不匹配，SSPIF 和 UA 也会置 1，SCL 会保持低电平，直到 SSPADD 发生更新可再次接收高字节为止。当 SSPADD 发生更新时，UA 位会被清零。这可以确保模块准备好在下一通信时接收高地址字节。

在所有 10 位寻址通信开始时，都需要以写请求方式进行高地址和低地址匹配。在寻址到从器件后，通过发出重复启动条件并随时钟移入 R/W 位置 1 的高地址字节来启动数据发送。然后，从器件硬件将会应答读请求，并准备好随着时钟移出数据。这只有在从器件接收到完全匹配的高地址和低地址字节之后才有效。

### 28.4.3 从接收

当接收到的匹配地址字节的 R/W 位清零时，SSPSTAT 寄存器的 R/W 位会清零。接收到的地址被装入 SSPBUF 寄存器并产生应答。

当接收到的地址存在溢出条件时，将会产生无应答信号。溢出条件定义为 SSPSTAT 寄存器的 BF 位置 1 或 SSPCON1 寄存器的 SSPOV 位置 1。SSPCON3 寄存器的 BOEN 位会修改该操作。更多信息，请参见寄存器 28-4。

每个传输的数据字节都会产生 MSSP 中断。标志位 SSPIF 必须用软件清零。

当 SSPCON2 寄存器的 SEN 位被置 1 时，SCL 将在接收到每个字节后保持低电平（时钟延长）。必须通过将 SSPCON1 寄存器的 CKP 位置 1 来释放时钟，10 位模式下的特殊情况除外。

### 28.4.3.1 7 位寻址接收

本节描述了将 MSSP 模块配置为工作在 7 位寻址模式下的 I<sup>2</sup>C 从器件时的标准事件序列，还描述了由硬件和软件所作的所有决定及其对接收的影响。图 28-5 和 28-6 用直观的方式对此作了说明。

以下列出了实现 I<sup>2</sup>C 通信时通常必须完成的详细步骤。

1. 检测到启动位。
2. SSPSTAT 寄存器的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPIF 会置 1。
3. 接收到 R/W 位清零的匹配地址。
4. 从器件通过将 SDA 下拉为低电平而向主器件发送 ACK，并将 SSPIF 位置 1。
5. 用软件清零 SSPIF 位。
6. 软件从 SSPBUF 中读取接收的地址，使 BF 标志清零。
7. 如果 SEN = 1，从器件软件会通过将 CKP 位置 1 来释放 SCL 线。
8. 主器件随时钟移出数据字节。
9. 从器件通过将 SDA 驱动为低电平而向主器件发送 ACK，并将 SSPIF 位置 1。
10. 用软件清零 SSPIF。
11. 软件从 SSPBUF 中读取接收的字节，使 BF 清零。
12. 对于从主器件接收到的所有字节，重复步骤 8-12。
13. 主器件发送停止条件，将 SSPSTAT 的 P 位置 1，总线变为空闲状态。

### 28.4.3.2 使用 AHEN 和 DHEN 时的 7 位接收

在 AHEN 和 DHEN 置 1 时，从器件接收的工作方式与不使用这些选项时的工作方式相同，只是在 SCL 的第 8 个下降沿之后添加了额外的中断和时钟延长。这些额外中断允许从器件软件决定是否应答（ACK）接收的地址或数据字节，而不是由硬件决定。

以下列表介绍了要对 I<sup>2</sup>C 通信使用这些选项时，从器件软件需要执行的步骤。图 28-7 显示了同时使用地址和数据保持功能的模块。图 28-8 包含了 SSPCON2 寄存器集的 SEN 位置 1 时的操作。

1. SSPSTAT 寄存器的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPIF 会置 1。
2. R/W 位清零的匹配地址随时钟移入。在 SCL 的第 8 个下降沿之后，SSPIF 置 1，CKP 清零。
3. 从器件清零 SSPIF。
4. 从器件可以查看 SSPCON3 寄存器的 ACKTIM 位，以确定 SSPIF 是在 ACK 之前还是之后置 1。
5. 从器件从 SSPBUF 中读取地址值，使 BF 标志清零。
6. 从器件通过设置 ACKDT 来设置随时钟移出到主器件的 ACK 值。
7. 从器件通过将 CKP 置 1 来释放时钟。
8. SSPxIF 会在 ACK 之后置 1，不会在 NACK 之后置 1。
9. 如果 SEN = 1，从器件硬件会在 ACK 之后延长时钟。
10. 从器件清零 SSPIF。

**注：** 即使不进行时钟延长，且 BF 已清零，SSPIF 仍然会在 SCL 的第 9 个下降沿之后置 1。只有向主器件发送了 NACK 信号后，SSPIF 才不会置 1。

11. 在所接收数据字节的第 8 个 SCL 下降沿之后，SSPIF 置 1，CKP 清零。
12. 从器件通过查看 SSPCON3 寄存器的 ACKTIM 位来确定中断源。
13. 从器件从 SSPBUF 中读取接收的数据，使 BF 清零。
14. 对于接收的每个数据字节，重复步骤 7-14。
15. 从器件发送 ACK = 1 或主器件发送停止条件可结束通信。如果发送了停止条件且禁止了在检测到停止条件时中断，则从器件只能通过查询 SSPSTAT 寄存器的 P 位才能知道停止条件。

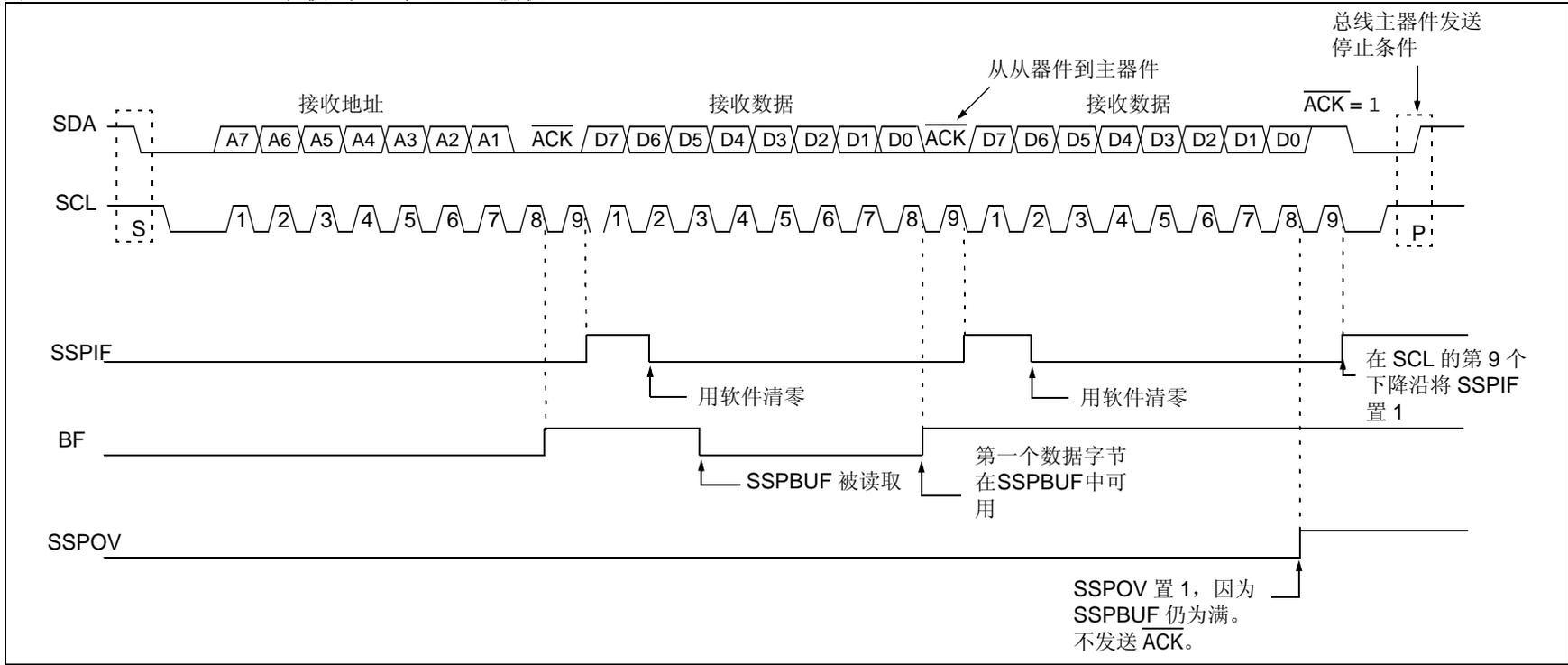
图 28-6: I<sup>2</sup>C™ 从模式, 7 位地址, 接收 (SEN = 0, AHEN = 0, DHEN = 0)

图 28-7: I<sup>2</sup>C™ 从模式, 7 位地址, 接收 (SEN = 1, AHEN = 0, DHEN = 0)

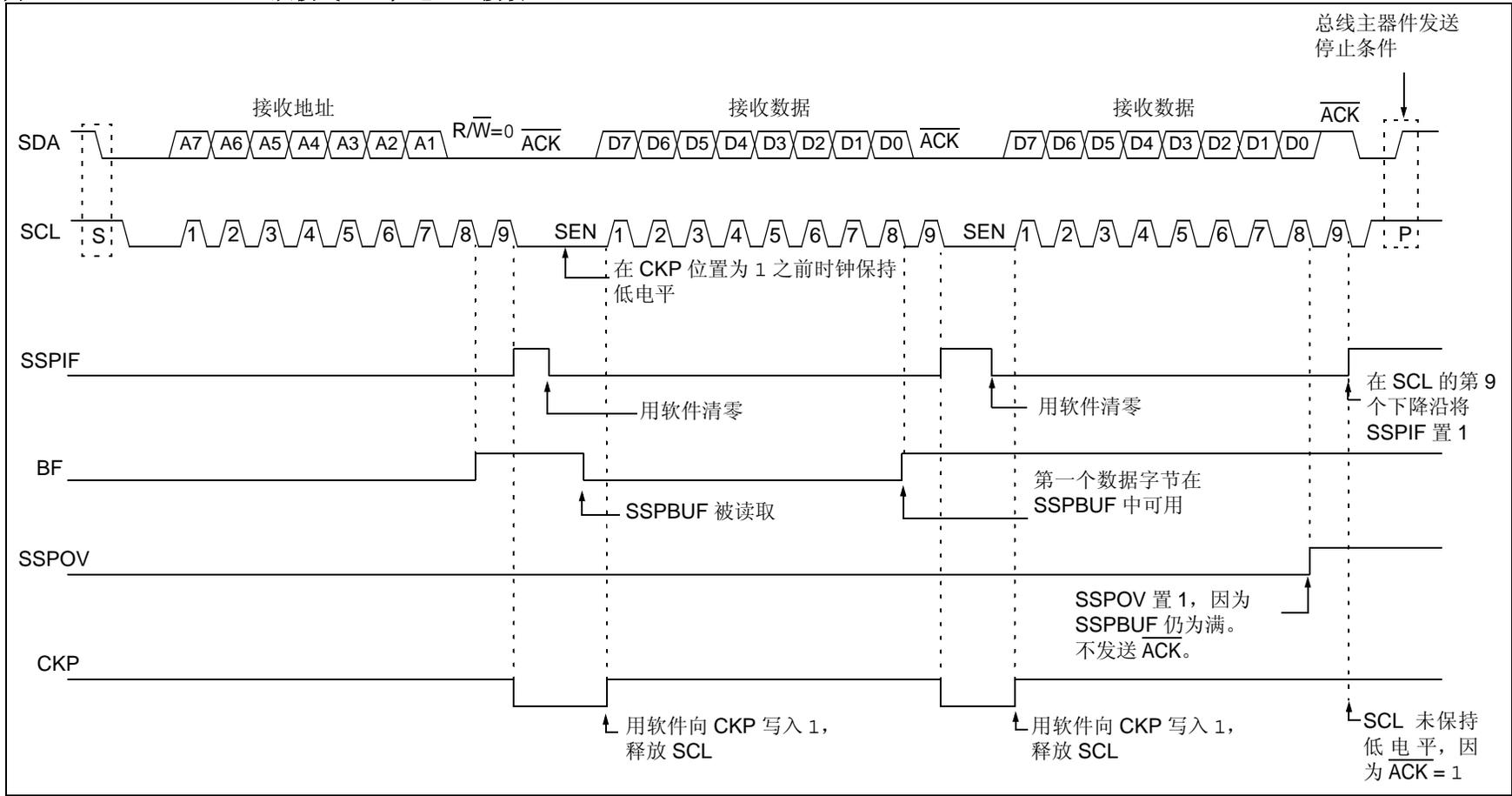


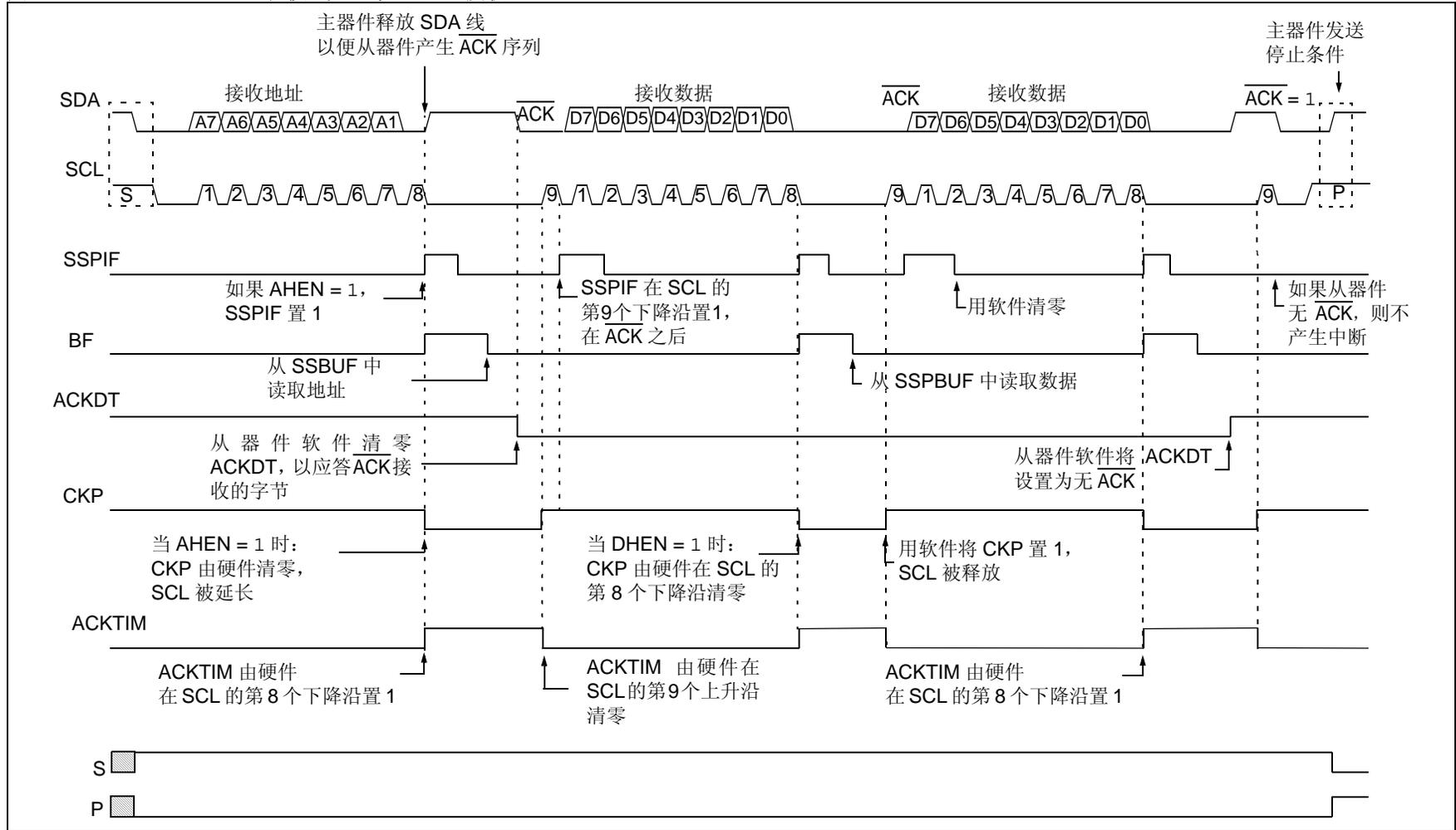
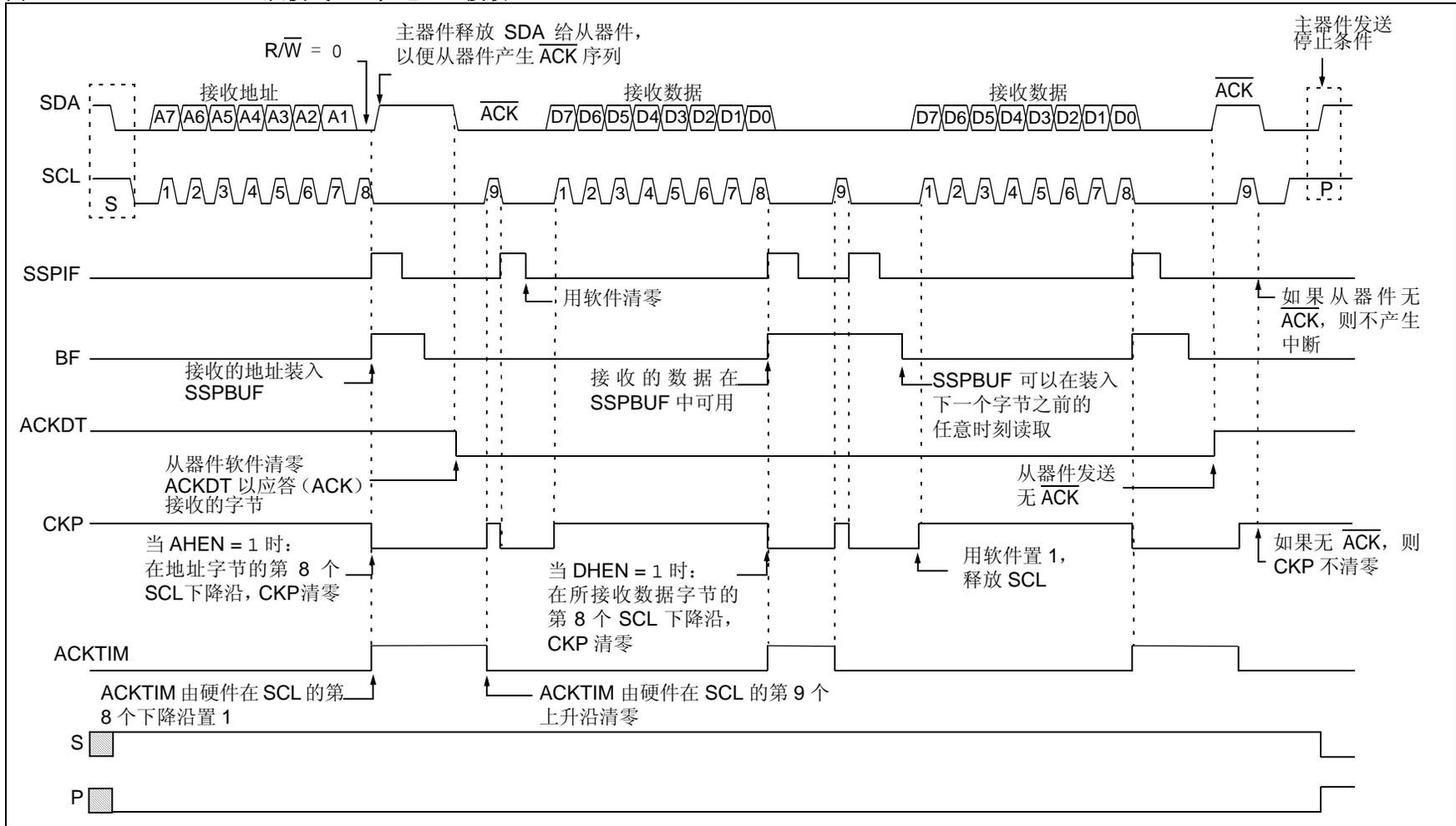
图 28-8: I<sup>2</sup>C™ 从模式, 7 位地址, 接收 (SEN = 0, AHEN = 1, DHEN = 1)

图 28-9: I<sup>2</sup>C™ 从模式, 7 位地址, 接收 (SEN = 1, AHEN = 1, DHEN = 1)



## 28.4.4 从发送

当输入地址字节的  $\overline{R/\overline{W}}$  位置 1 并发生地址匹配时，SSPSTAT 寄存器的  $\overline{R/\overline{W}}$  位被置 1。接收到的地址会被装入 SSPBUF 寄存器，并且从器件会在第 9 个位发送 ACK 脉冲。

在  $\overline{ACK}$  之后，从器件硬件会清零 CKP 位，并且 SCL 引脚保持低电平。更多详细信息，请参见第 28.4.7 节“时钟延长”。通过延长时钟，主控器件只有在从器件准备发送数据时才能发送另一个时钟脉冲。

发送的数据必须被装入 SSPBUF 寄存器，同时也被装入 SSPSR 寄存器。然后，应通过将 SSPCON1 寄存器的 CKP 位置 1 来释放 SCL 引脚。发送的数据必须被装入 SSPBUF 寄存器，同时也被装入了 SSPSR 寄存器。这可确保在 SCL 为高电平期间 SDA 信号是有效的。

来自主接收器的  $\overline{ACK}$  脉冲将在第 9 个 SCL 输入脉冲的上升沿被锁存。该  $\overline{ACK}$  值会被复制到 SSPCON2 寄存器的 ACKSTAT 位中。如果 ACKSTAT 置 1（无 ACK 应答信号），那么表示数据传输已完成。这种情况下，在从器件锁存无  $\overline{ACK}$  时，从器件会进入空闲状态，并等待出现另一个启动位。如果 SDA 线为低电平（ $\overline{ACK}$ ），则必须将下一个要发送的数据装入 SSPBUF 寄存器。同样，必须通过将 CKP 位置 1 来释放 SCL 引脚。

每个数据传输字节都会产生一个 MSSP 中断。SSPIF 位必须用软件清零，SSPSTAT 寄存器用于确定字节的状态。SSPIF 位在第 9 个时钟脉冲的下降沿被置 1。

### 28.4.4.1 从模式总线冲突

从器件接收到读请求，开始在 SDA 线上移出数据。如果检测到总线冲突，并且 SSPCON3 寄存器的 SBCDE 位置 1，PIR 寄存器的 BCLIF 位会置 1。在检测到总线冲突时，从器件会变为空闲状态，等待再次被寻址。用户软件可以通过使用 BCLIF 位来处理从器件总线冲突。

### 28.4.4.2 7 位发送

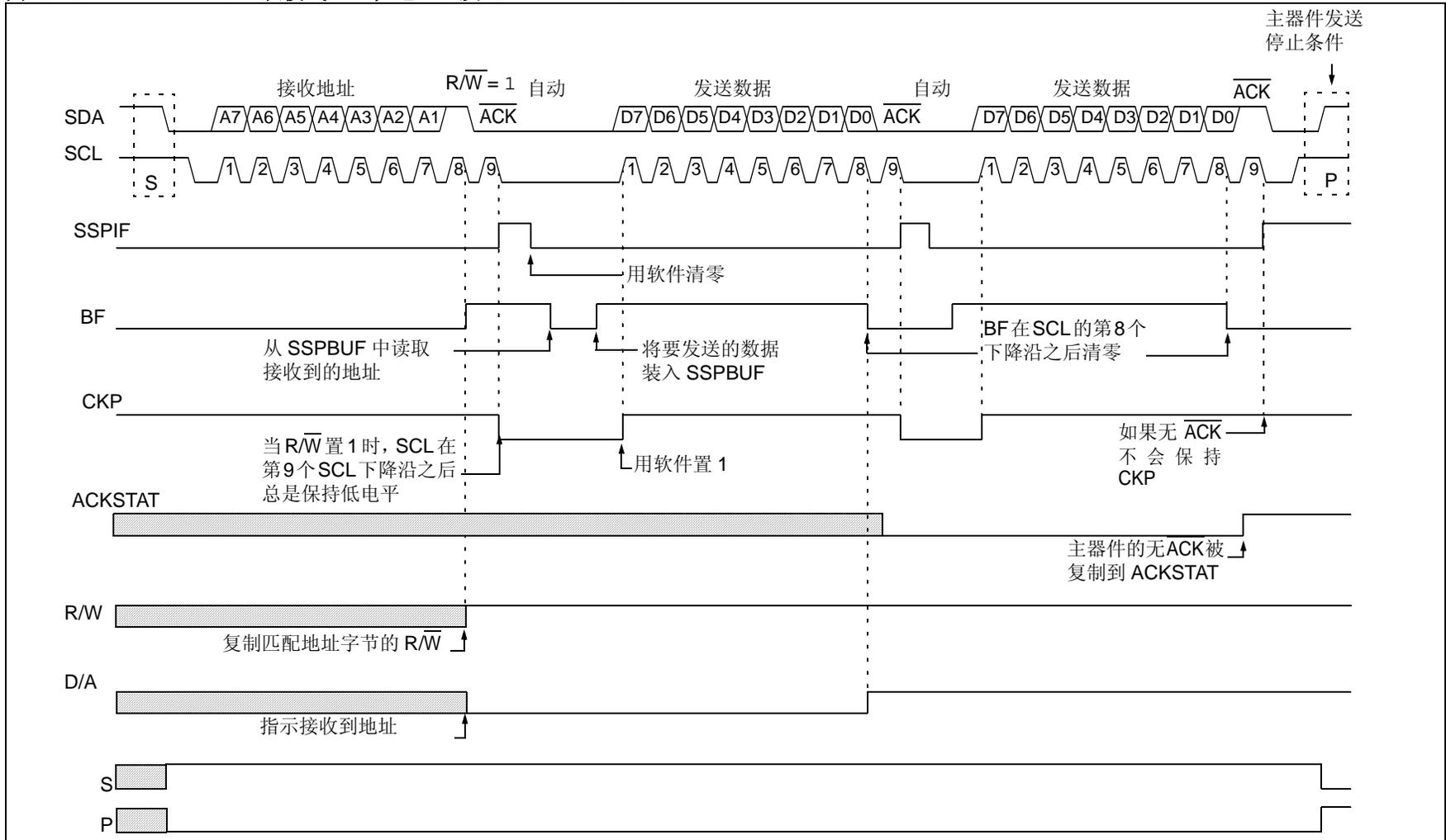
主器件可以向从器件发送读请求，然后从从器件中随时钟移出数据。以下列表列出了在实现标准数据发送时，从器件软件需要执行的操作。图 28-10 可用作该列表的参考。

1. 主器件在 SDA 和 SCL 上发送一个启动条件。
2. SSPSTAT 寄存器的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPIF 会置 1。
3. 从器件接收到  $\overline{R/\overline{W}}$  位置 1 的匹配地址，并将 SSPIF 位置 1。
4. 从器件硬件产生  $\overline{ACK}$ ，并将 SSPIF 置 1。
5. 用户将 SSPIF 位清零。
6. 软件从 SSPBUF 中读取接收的地址，使 BF 清零。
7.  $\overline{R/\overline{W}}$  置 1，所以 CKP 会在  $\overline{ACK}$  之后自动清零。
8. 从器件软件将发送数据装入 SSPBUF。
9. CKP 位置 1，释放 SCL，使主器件可以从从器件中随时钟移出数据。
10. 来自主器件的  $\overline{ACK}$  响应装入 ACKSTAT 寄存器之后，SSPIF 置 1。
11. SSPIF 位清零。
12. 从器件软件通过检查 ACKSTAT 位来确定主器件是否要随时钟移出更多数据。

**注 1:** 如果主器件应答 ( $\overline{ACK}$ )，时钟将被延长。  
**注 2:** ACKSTAT 是惟一一个在 SCL 的上升沿（第 9 个）而不是下降沿发生更新的位。

13. 对于每个发送字节，重复步骤 9-13。
14. 如果主器件发送无  $\overline{ACK}$ ，则不会保持时钟，但 SSPIF 仍然会置 1。
15. 主器件发送重复启动条件或停止条件。
16. 从器件不再被寻址。

图 28-10: I<sup>2</sup>C™ 从模式, 7 位地址, 发送 (AHEN = 0)



## 28.4.4.3 使能地址保持时的 7 位发送

将 SSPCON3 寄存器的 AHEN 位置 1 时，器件会在所接收匹配地址的第 8 个下降沿之后延长时钟和产生中断。在匹配地址随时钟移入之后，CKP 会清零，SSPIF 中断标志会置 1。

图 28-11 给出了在使能 AHEN 时 7 位地址从发送的标准波形图。

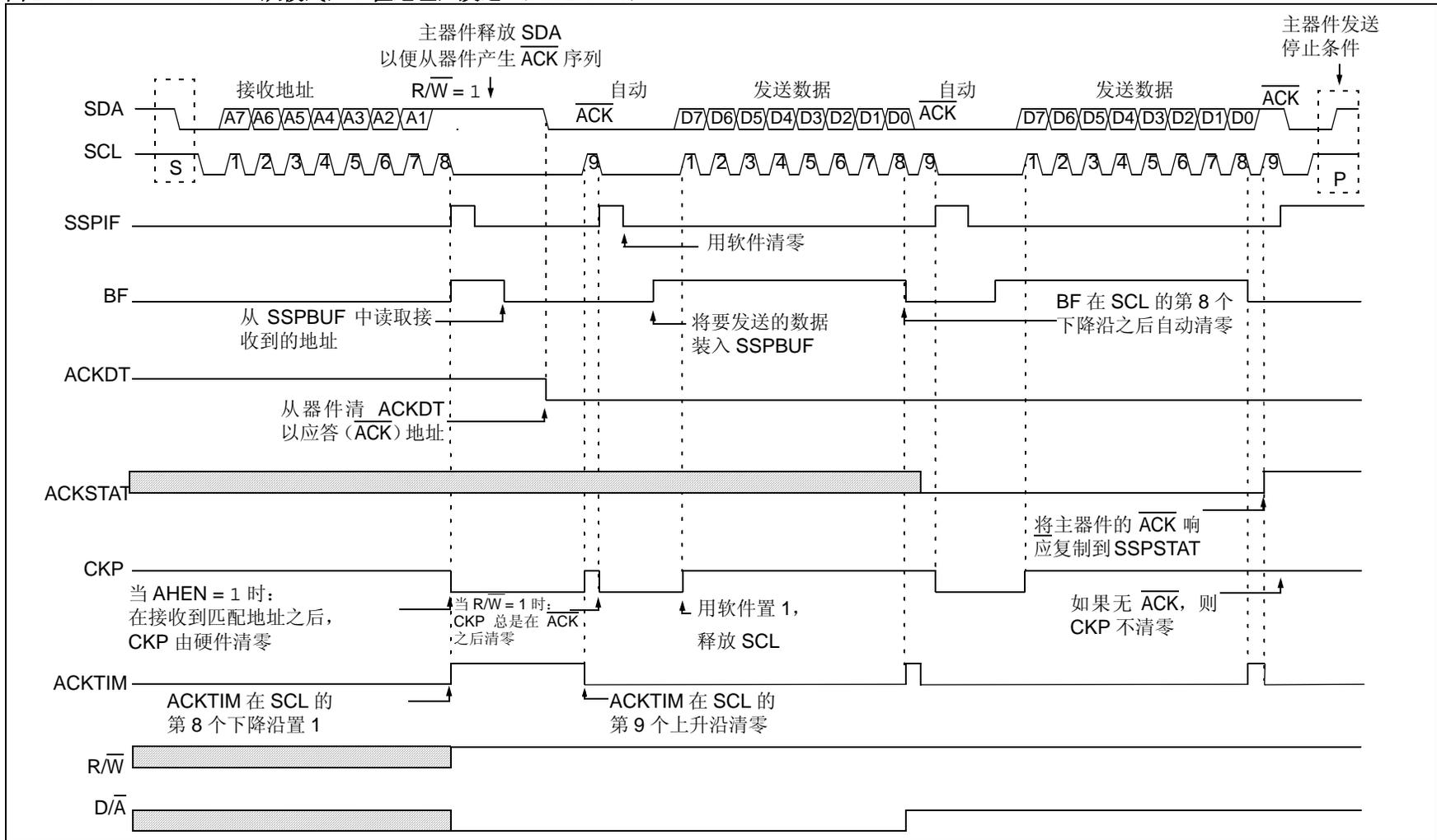
1. 总线启动时为空闲模式。
2. 主器件发送启动条件；SSPSTAT 寄存器的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPIF 会置 1。
3. 主器件发送  $\overline{R/W}$  位置 1 的匹配地址。在 SCL 线的第 8 个下降沿之后，CKP 位清零，并产生 SSPIF 中断。
4. 从器件软件清零 SSPIF。
5. 从器件软件读取 SSPCON3 寄存器的 ACKTIM 位，以及 SSPSTAT 寄存器的  $\overline{R/W}$  和  $\overline{D/A}$  位，以确定中断源。
6. 从器件从 SSPBUF 寄存器中读取地址值，使 BF 位清零。
7. 从器件软件根据该信息确定它是产生  $\overline{ACK}$  还是产生无 ACK，并相应地设置 SSPCON2 寄存器的 ACKDT 位。
8. 从器件将 CKP 位置 1，释放 SCL。
9. 主器件随时钟移入来自从器件的  $\overline{ACK}$  值。
10. 如果  $\overline{R/W}$  位置 1，则在  $\overline{ACK}$  之后，从器件硬件会自动将 CKP 位清零，将 SSPIF 置 1。
11. 从器件软件清零 SSPIF。
12. 从器件将要发送给主器件的值装入 SSPBUF，使 BF 位置 1。

**注：** 只有在  $\overline{ACK}$  之后，才能装入 SSPBUF。

13. 从器件将 CKP 位置 1，释放时钟。
14. 主器件从从器件中移出数据，并在第 9 个 SCL 脉冲发送 ACK 值。
15. 从器件硬件将  $\overline{ACK}$  值复制到 SSPCON2 寄存器的 ACKSTAT 位中。
16. 对于从从器件发送到主器件的每个字节重复步骤 10-15。
17. 如果主器件发送非  $\overline{ACK}$ ，从器件将释放总线以允许主器件发送停止条件并结束通信。

**注：** 主器件必须对于最后一个字节发送无  $\overline{ACK}$ ，以确保从器件释放 SCL 线来接收停止条件。

图 28-11: I<sup>2</sup>C™ 从模式, 7 位地址, 发送 (AHEN = 1)



# MCP19114/5

## 28.4.5 从模式 10 位地址接收

本节描述了 MSSP 模块配置为工作在 10 位寻址模式下的 I<sup>2</sup>C 从器件时的标准事件序列。

图 28-12 用直观的方式对此作了说明。

以下列出了实现 I<sup>2</sup>C 通信时从器件软件必须完成的步骤。

1. 总线启动时为空闲模式。
2. 主器件发送启动条件；SSPSTAT 寄存器的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPIF 会置 1。
3. 主器件发送 R $\overline{W}$  位清零的匹配高地址；SSPSTAT 寄存器的 UA 位置 1。
4. 从器件发送  $\overline{ACK}$ ，SSPIF 置 1。
5. 用软件清零 SSPIF 位。
6. 软件从 SSPBUF 中读取接收的地址，使 BF 标志清零。
7. 从器件将低地址装入 SSPADD，释放 SCL。
8. 主器件向从器件发送匹配的低地址字节；UA 位置 1。

**注：** 只有在  $\overline{ACK}$  序列之后，才允许更新 SSPADD 寄存器。

9. 从器件发送  $\overline{ACK}$ ，SSPIF 置 1。

**注：** 如果低地址不匹配，SSPIF 和 UA 仍然会置 1，从而让从器件软件可以将 SSPADD 恢复为高地址。由于不匹配，BF 不会置 1。CKP 不受影响。

10. 从器件清零 SSPIF。
11. 从器件从 SSPBUF 中读取接收的匹配地址，使 BF 清零。
12. 从器件将高地址装入 SSPADD。
13. 主器件随时钟将数据字节移入从器件，并在第 9 个 SCL 脉冲随时钟将  $\overline{ACK}$  移出从器件；SSPIF 置 1。
14. 如果 SSPCON2 寄存器的 SEN 位置 1，CKP 会被硬件清零，时钟会被延长。
15. 从器件清零 SSPIF。
16. 从器件从 SSPBUF 中读取接收的字节，使 BF 清零。
17. 如果 SEN 置 1，从器件会将 CKP 置 1，以释放 SCL。
18. 对于接收的每个字节重复步骤 13-17。
19. 主器件发送停止条件来结束数据发送。

## 28.4.6 带地址或数据保持的 10 位寻址

在 AHEN 或 DHEN 置 1 时，使用 10 位寻址的接收方式与 7 位模式相同。惟一的区别是需要使用 UA 位来更新 SSPADD 寄存器。所有功能（特别是在 CKP 位清零，SCL 线保持低电平时）都是相同的。图 28-13 可以用作 AHEN 置 1 时 10 位寻址模式下从器件的参考图示。

图 28-14 给出了 10 位寻址模式下从发送器的标准波形图。

图 28-12: I<sup>2</sup>C™ 从模式, 10 位地址, 接收 (SEN = 1, AHEN = 0, DHEN = 0)

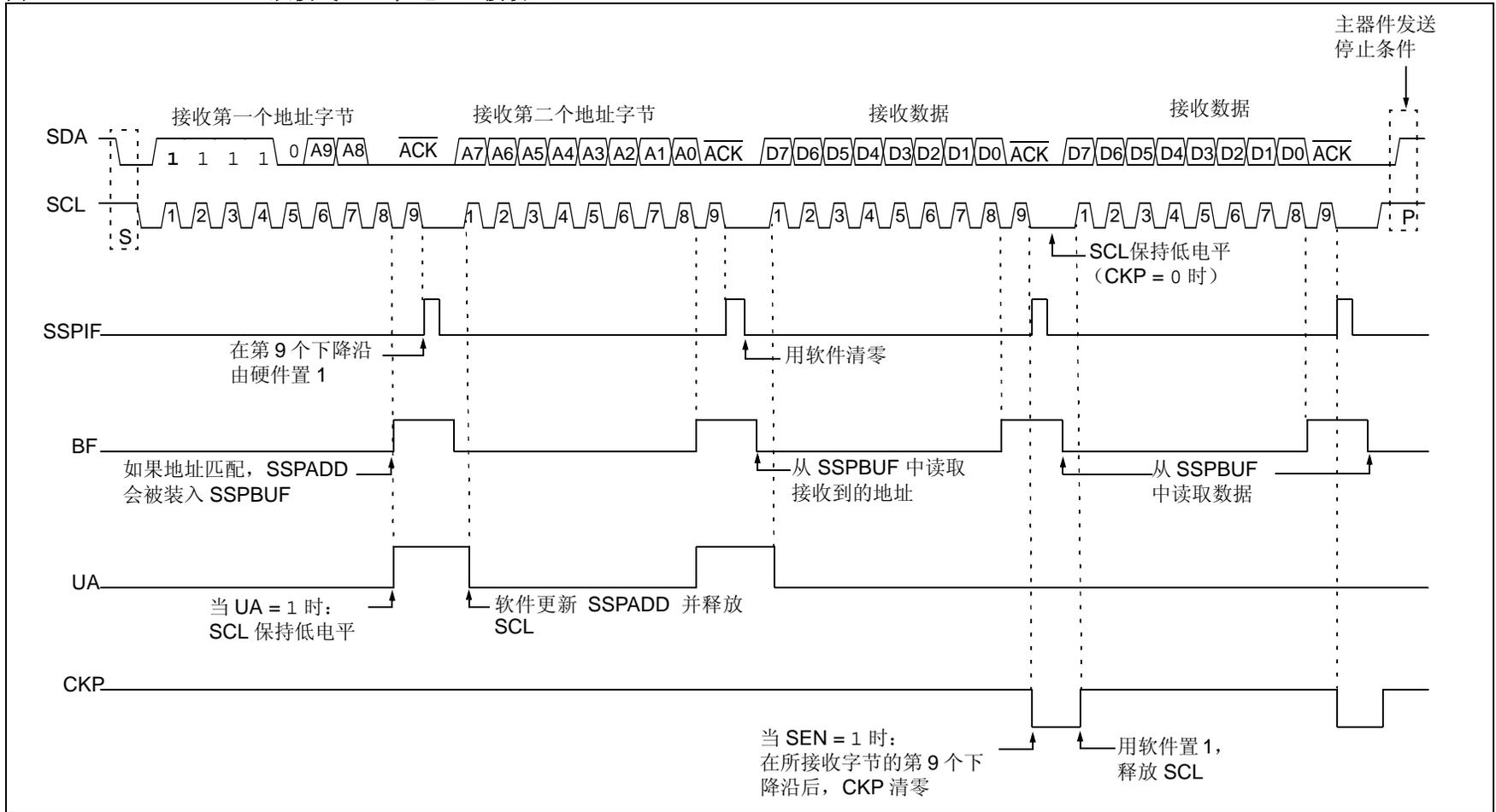


图 28-13: I<sup>2</sup>C™ 从模式, 10 位地址, 接收 (SEN = 0, AHEN = 1, DHEN = 0)

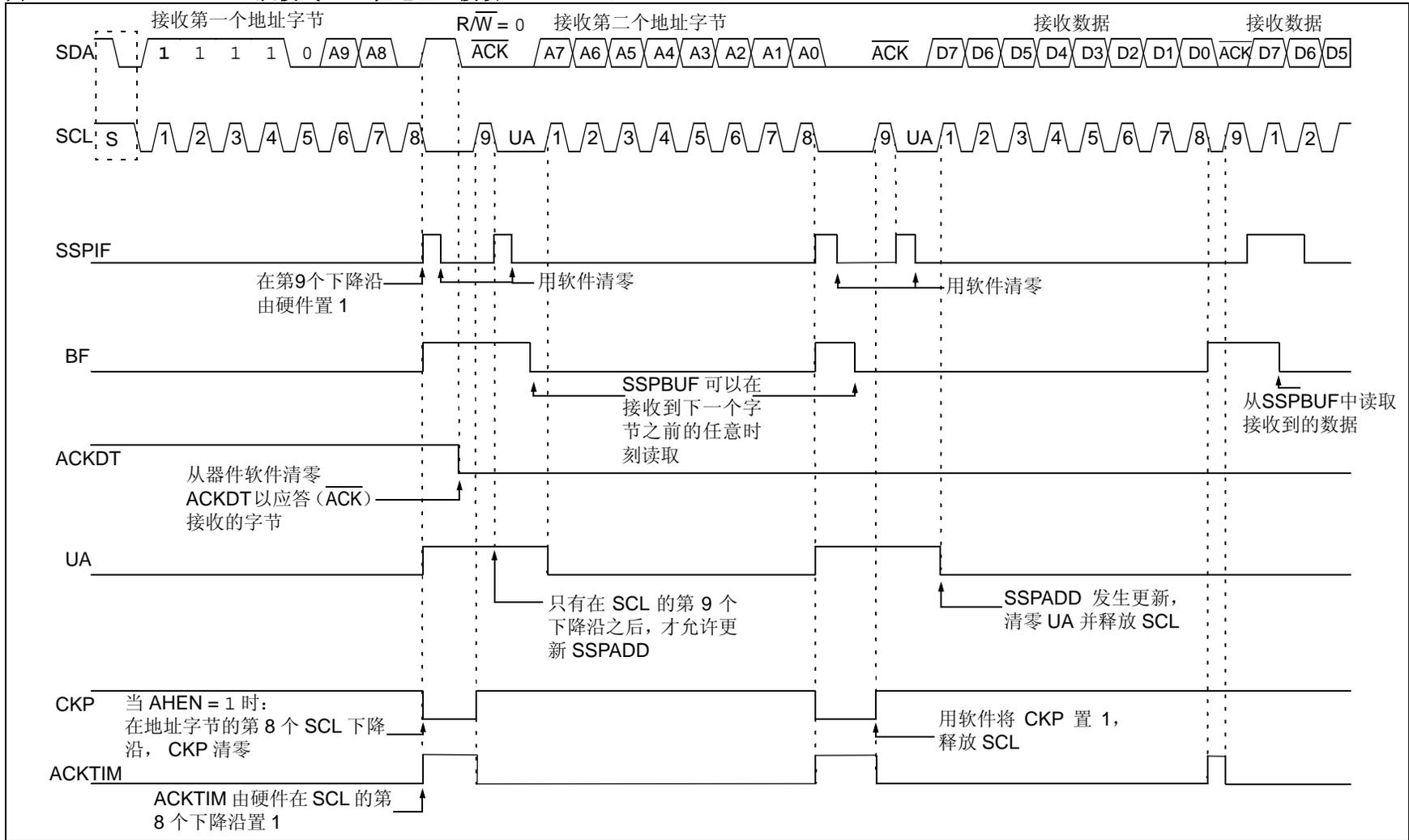
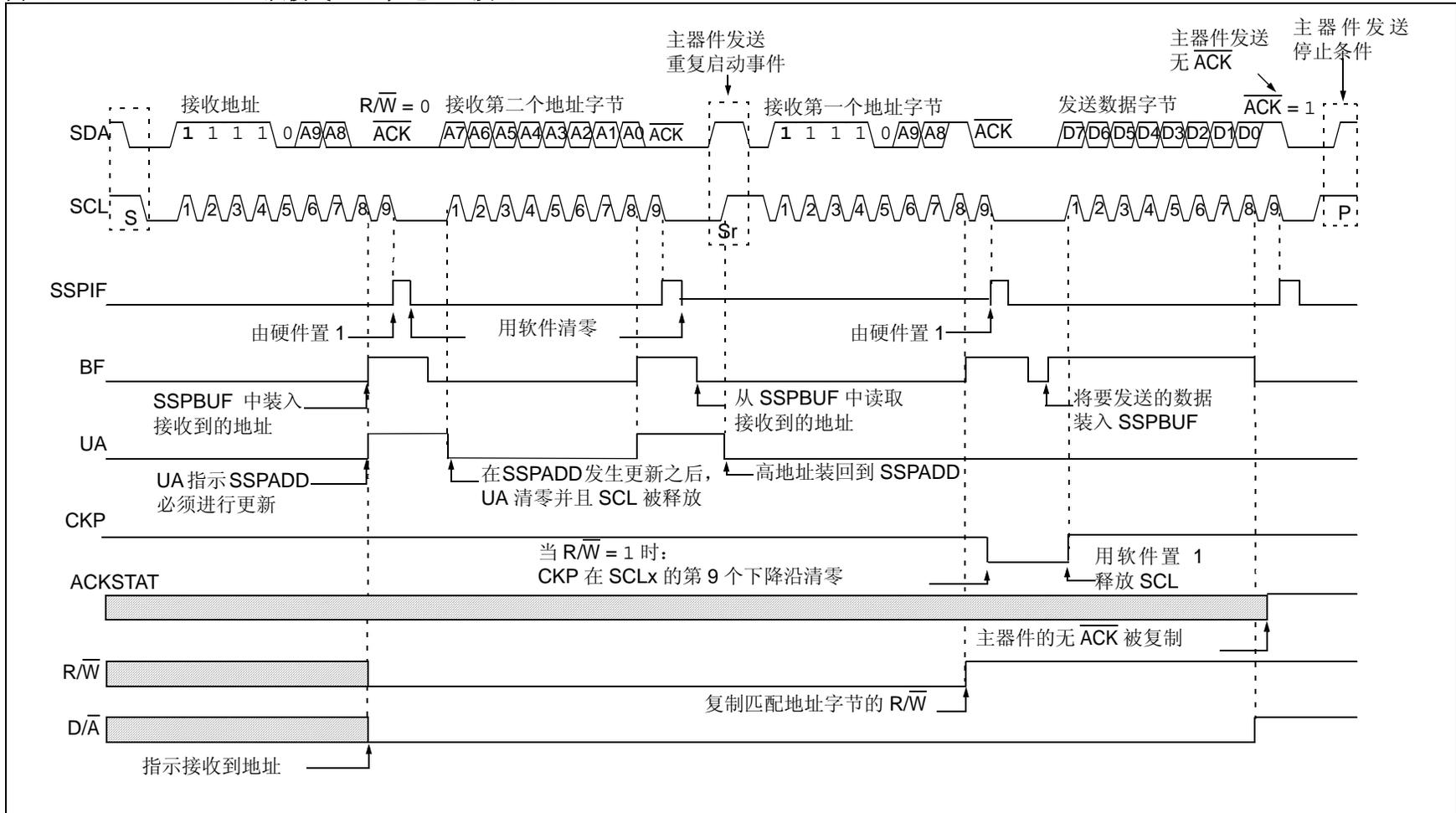


图 28-14: I<sup>2</sup>C™ 从模式, 10 位地址, 发送 (SEN = 0, AHEN = 0, DHEN = 0)



## 28.4.7 时钟延长

当总线上的某个器件将 SCL 线保持为低电平而有效暂停通信时，就发生了时钟延长现象。从器件可以延长时钟，以便可以有更多时间来处理数据或准备响应主器件。时钟延长时并不关心主器件的工作，因为任何时候只需总线上主器件处于活动状态但是不传输数据就可以被认为是时钟延长。由从器件进行的任何时钟延长对于主器件软件都是不可见的，都由产生 SCL 的硬件进行处理。

SSPCON1 寄存器的 CKP 位用于在软件中控制时钟延长。每当 CKP 位清零时，模块就会等待 SCL 线变为低电平，然后保持低电平状态不变。将 CKP 置 1 将会释放 SCL，允许继续进行通信。

### 28.4.7.1 正常时钟延长

如果 SSPSTAT 的  $\overline{R/W}$  位置 1（读请求），则在  $\overline{ACK}$  之后，从器件硬件会清零 CKP。这让从器件可以有时间使用要传输给主器件的数据更新 SSPBUF。如果 SSPCON2 寄存器的 SEN 位置 1，则在  $\overline{ACK}$  序列之后，从器件硬件将总是延长时钟。在从器件就绪之后，软件会将 CKP 置 1，并继续进行通信。

**注 1:** BF 位对于时钟是否延长没有影响。这一点与模块的先前版本不同：如果在 SCL 的第 9 个下降沿之前读取了 SSPBUF，先前版本将不会延长时钟，清零 CKP。

**2:** 如果在 SCL 的第 9 个下降沿之前装入 SSPBUF，则模块的先前版本不会为数据发送延长时钟。现在，对于读请求，总是会将会该位清零。

### 28.4.7.2 10 位寻址模式

在 10 位寻址模式下，UA 位置 1 时，时钟始终延长。只有在这时才会 CKP 未清零时延长 SCL。在写入 SSPADD 之后，SCL 会立即被释放。

**注:** 如果第二个地址字节不匹配，先前版本的模块不会延长时钟。

### 28.4.7.3 字节无应答

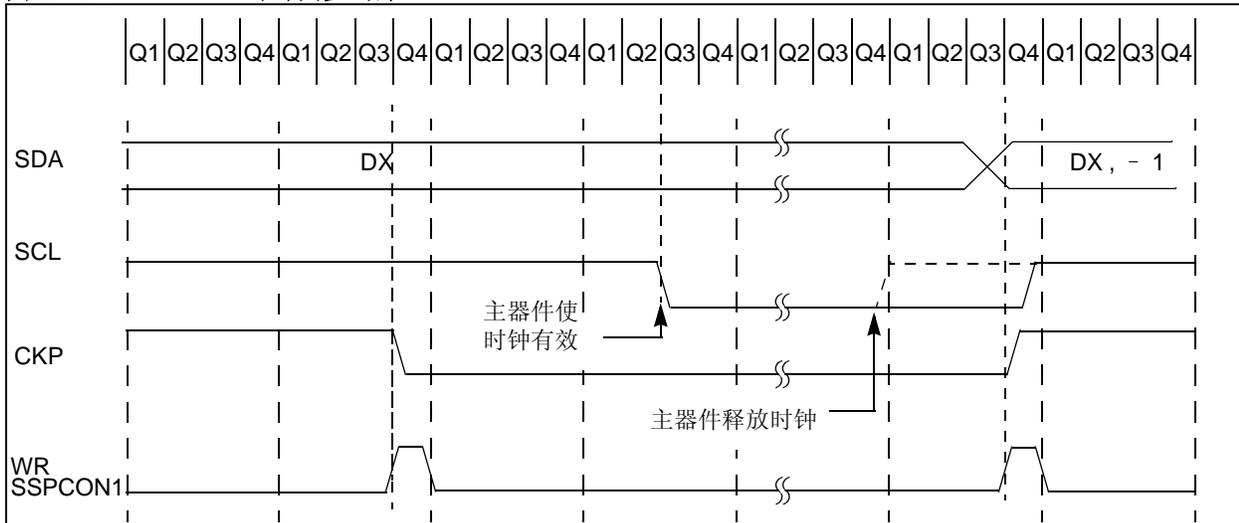
当 SSPCON3 寄存器的 AHEN 位置 1 时，在所接收匹配地址字节的第 8 个 SCL 下降沿之后，硬件会将 CKP 清零。当 SSPCON3 寄存器的 DHEN 位置 1 时，在所接收数据的第 8 个 SCL 下降沿之后，CKP 会被清零。

通过在 SCL 的第 8 个下降沿之后延长时钟，从器件可以检查接收到的地址或数据，并确定是否要应答接收到的数据。

### 28.4.8 时钟同步时序和 CKP 位

每当 CKP 位清零时，模块就会等待 SCL 线变为低电平，然后保持低电平状态不变。但是，清零 CKP 位并不会将 SCL 输出置为低电平，只有在已经采样到 SCL 输出为低电平之后才会。然而，将 CKP 位清零并未表明 SCL 输出一定为低电平，除非已经采样到 SCL 输出为低电平，实际上，需要外部 I2C 主器件将 SCL 线拉低。SCL 输出将保持低电平，直到 CKP 位置 1 且 I<sup>2</sup>C 总线上的所有其他器件已释放 SCL 为止。这可以确保对 CKP 位的写操作不会违反 SCL 的最小高电平时间要求（见图 28-16）。

图 28-15: 时钟同步时序



### 28.4.9 广播呼叫地址支持

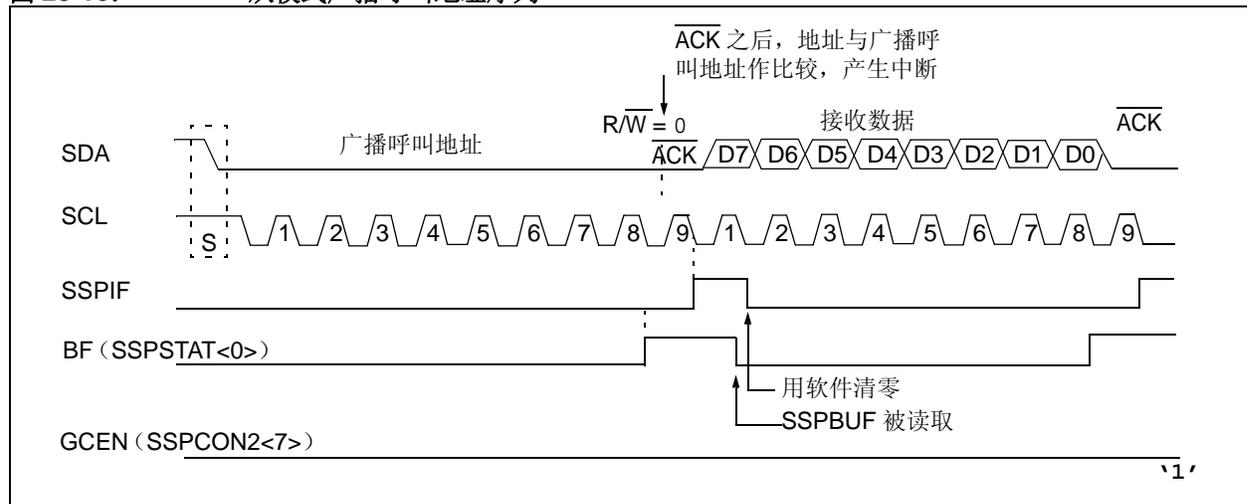
在 I<sup>2</sup>C 总线的寻址过程中，通常由启动条件后的第一个字节决定主器件将寻址哪个从器件。但广播呼叫地址例外，它能寻址所有器件。当使用这个地址时，理论上所有器件都应该发送一个应答信号来响应。

广播呼叫地址是 I<sup>2</sup>C 协议中的保留地址，定义为地址 0x00。如果 SSPCON2 寄存器的 GCEN 位置 1，则无论 SSPADD 中存储的值如何，在接收到该地址时，从模块都会自动发送 ACK。在从器件随时钟移入 R/W 位清零的全零地址之后，将会产生中断，从器件软件可以读取 SSPBUF 并进行响应。图 28-17 显示了广播呼叫接收序列。

在 10 位地址模式下，UA 位不会在接收到广播呼叫地址时置 1。从器件会准备接收作为数据的第二个字节，这与在 7 位模式下相同。

如果 SSPCON3 寄存器的 AHEN 位置 1，则与接收到任意其他地址时相同，从器件硬件会在 SCL 的第 8 个下降沿之后延长时钟。然后，从器件必须与正常情况下一样，设置它的 ACKDT 值，并释放时钟来继续进行通信。

图 28-16: 从模式广播呼叫地址序列



### 28.4.10 SSPMSK1 寄存器

SSP 掩码 (SSPMSK1) 寄存器在 I<sup>2</sup>C 从模式下可用，用作地址比较操作期间 SSPSR 寄存器中保存的值的掩码。SSPMSK1 寄存器中的零 (0) 位可使接收地址中相应位变为“无关位”。

发生任何复位条件时，该寄存器都会复位到全 1 状态，因此，在写入掩码值之前对标准 SSP 操作没有影响。

SSPMSK1 寄存器在以下期间保持有效：

- 7 位地址模式：A<7:1> 的地址比较。
- 10 位地址模式：仅 A<7:0> 的地址比较。在接收地址的第一个 (高) 字节期间，SSP 掩码没有影响。

## 28.5 I<sup>2</sup>C 主模式

通过将 SSPCON1 中的相应 SSPM 位置 1 和清零，同时将 SSPEN 位置 1，可以使能主模式。在主模式下，SDA 和 SCK 引脚必须被配置为输入。当需要将引脚驱动为低电平时，MSSP 外设硬件将改写输出驱动器的 TRIS 控制。

通过在检测到启动和停止条件时产生中断来支持主模式操作。停止 (P) 位和启动 (S) 位在复位时或禁止 MSSP 模块时清零。当 P 位置 1 或总线空闲时，可取得 I<sup>2</sup>C 总线的控制权。

在固件控制的主模式下，用户代码根据启动位和停止位条件检测执行所有的 I<sup>2</sup>C 总线操作。在该模式下，启动和停止条件检测是唯一有效的电路。所有其他通信都通过用户软件直接操作 SDA 和 SCL 线来完成。

下列事件会使 SSP 中断标志位 SSPIF 置 1（如果允许 SSP 中断，则产生该中断）：

- 检测到启动条件
- 检测到停止条件
- 数据传输字节发送 / 接收
- 应答发送 / 接收
- 产生重复启动条件

**注 1:** 当被配置为 I<sup>2</sup>C 主模式时，MSSP 模块不允许事件排队。例如，在启动条件结束前，不允许用户发出启动条件并立即写 SSPBUF 寄存器以发起传输。这种情况下，将不会写入 SSPxBUF，WCOL 位将被置 1，这表明没有发生对 SSPxBUF 的写操作。

**2:** 处于主模式时，如果 SEN/PEN 位清零，并且启动 / 停止条件完成，则会屏蔽启动 / 停止检测和产生中断。

### 28.5.1 I<sup>2</sup>C 主模式操作

主器件产生所有串行时钟脉冲和启动 / 停止条件。以停止条件或重复启动条件结束传输。因为重复启动条件也是下一次串行传输的开始，因此 I<sup>2</sup>C 总线不会被释放。

在主发送器模式下，串行数据通过 SDA 输出，而串行时钟由 SCL 输出。发送的第一个字节包括接收器件的从器件地址（7 位）和读 / 写 (R/W) 位。在这种情况下，R/W 位将是逻辑 0。每次发送 8 位串行数据。每发送一个字节，会收到一个应答位。输出启动和停止条件，表明串行传输的开始和结束。

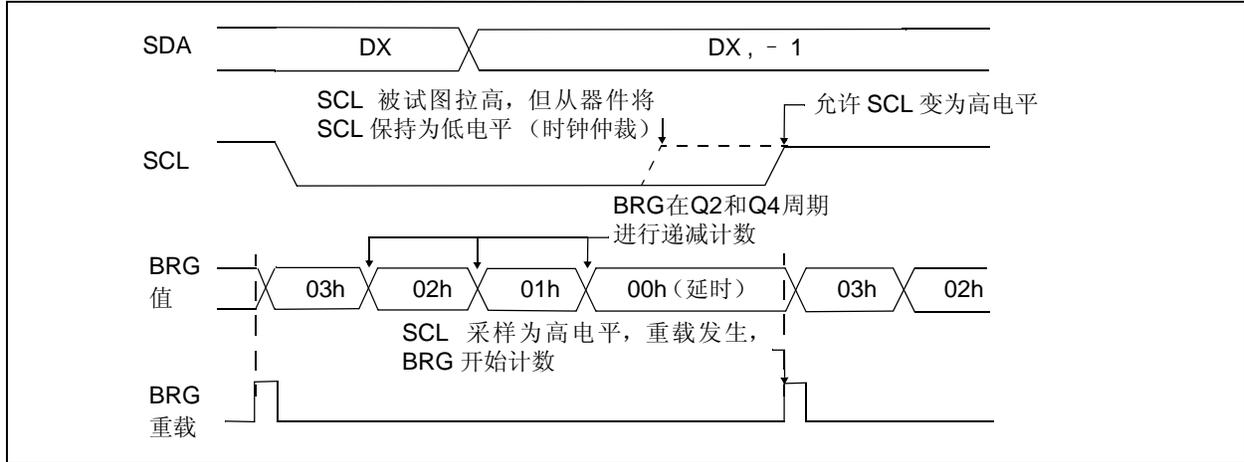
在主接收模式下，发送的第一个字节包括发送器件的从器件地址（7 位）和 R/W 位。在这种情况下，R/W 位将是逻辑 1。因此，发送的第一个字节是一个 7 位从器件地址，后跟 1 指示接收位。串行数据通过 SDA 接收，而串行时钟由 SCL 输出。每次接收 8 位串行数据。每接收到一个字节，都会发送一个应答位。启动和停止条件分别代表发送的开始和结束。

波特率发生器用于设置从 SCL 输出的时钟的频率。更多详细信息，请参见第 28.6 节“波特率发生器”。

### 28.5.2 时钟仲裁

如果在任何接收、发送或重复启动 / 停止条件期间，主器件释放了 SCL 引脚（允许 SCL 悬空为高电平），就会发生时钟仲裁。如果允许 SCL 引脚悬空为高电平，波特率发生器 (BRG) 将暂停计数直到实际采样到 SCL 引脚为高电平为止。当 SCL 引脚被采样为高电平时，会将 SSPADD<7:0> 的内容重新装入波特率发生器并使之开始计数。这可以保证当外部器件将时钟拉低时，SCL 始终在至少一个 BRG 计满返回周期内保持高电平（图 28-17）。

**图 28-17: 带有时钟仲裁的波特率发生器时序**



### 28.5.3 WCOL 状态标志

如果在启动、重复启动、停止、接收或发送序列过程中用户写 SSPBUF, 则 WCOL 被置 1, 同时缓冲区内容不变 (未发生写操作)。在任何时候 WCOL 位置 1 都表示在模块不处于空闲状态时对 SSPBUF 尝试了某种操作。

**注:** 由于不允许事件排队, 在启动条件结束之前, 不能写 SSPCON2 寄存器的低 5 位。

### 28.5.4 I<sup>2</sup>C 主模式启动条件时序

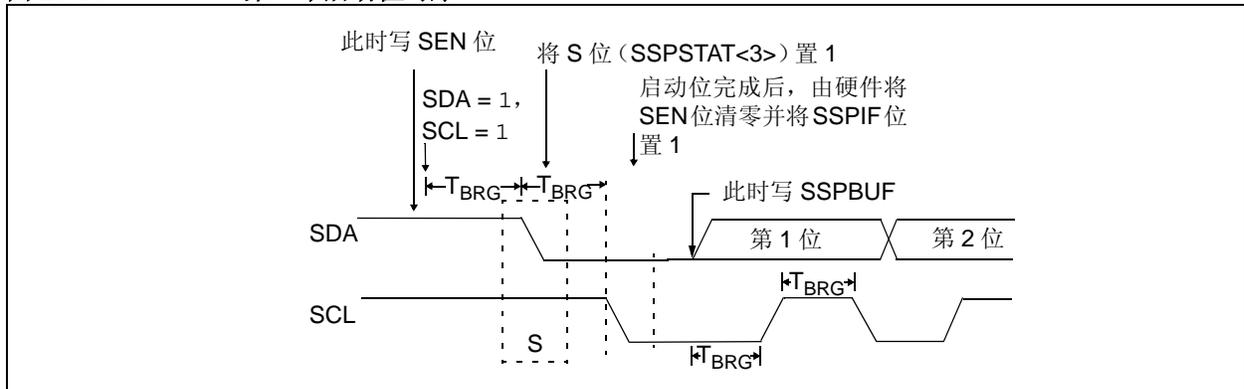
要发出启动条件, 用户应将 SSPCON2 寄存器的启动使能位 SEN 置 1。当 SDA 和 SCL 引脚采样为高电平时, 波特率发生器重新装入 SSPADD<7:0> 的值并开始计数。当波特率发生器发生超时 ( $T_{BRG}$ ) 时, 如果 SCL

和 SDA 都被采样为高电平, 则 SDA 引脚被驱动为低电平。当 SCL 为高电平时, 将 SDA 驱动为低电平将产生启动条件, 并使 SSPSTAT1 寄存器的 S 位置 1。随后波特率发生器重新装入 SSPADD<7:0> 的值并恢复计数。当波特率发生器再次超时 ( $T_{BRG}$ ) 时, SSPCON2 寄存器的 SEN 位将自动被硬件清零, 波特率发生器暂停工作, SDA 线保持低电平, 启动条件结束。

**注 1:** 如果在启动条件开始时, SDA 和 SCL 引脚已经采样为低电平, 或者在启动条件期间, SCL 在 SDA 线被驱动为低电平之前已经采样为低电平, 则会发生总线冲突总线冲突中断标志位 BCLIF 置 1, 启动条件中止, I<sup>2</sup>C 模块复位到空闲状态。

**注 2:** Philips I<sup>2</sup>C 规范规定启动时不能发生总线冲突。

**图 28-18: 第一个启动位时序**

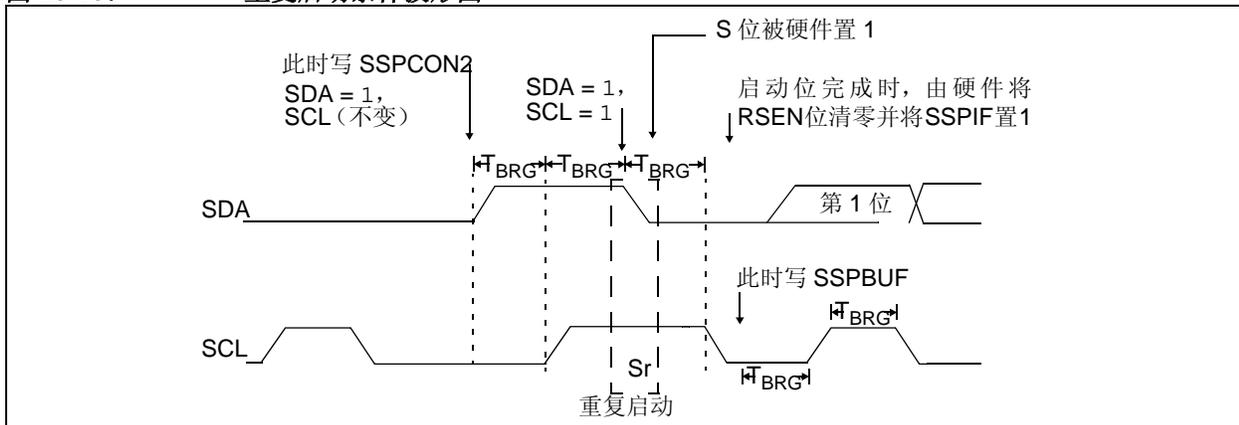


## 28.5.5 I<sup>2</sup>C 主模式重复启动条件时序

当 SSPCON2 寄存器的 RSEN 位设定为高电平，并且主器件状态机不再有效时，会产生重复启动条件。当 RSEN 位置 1 时，SCL 引脚被拉为低电平。当 SCL 引脚被采样为低电平时，波特率发生器会装入值并开始计数。在一个波特率发生器计数周期 ( $T_{BRG}$ ) 后，SDA 引脚被释放（其引脚电平被拉高）。当波特率发生器超时，如果 SDA 被采样为高电平，SCL 引脚将被置为无效（拉为高电平）。当 SCL 被采样为高电平时，波特率发生器被重载并开始计数。SDA 和 SCL 必须在一个  $T_{BRG}$  内始终被采样为高电平。接下来，在一个  $T_{BRG}$  中，将 SDA 引脚置为有效（SDA = 0），同时 SCL 保持高电平。SCL 引脚被拉为低电平。之后 SSPCON2 寄存器的 RSEN 位将自动清零，这次波特率发生器不会重载，SDA 引脚保持低电平。一旦在 SDA 和 SCL 引脚上检测到启动条件，SSPSTAT 寄存器的 S 位就会被置 1。直到波特率发生器发生超时后，SSPIF 位才会置 1。

- 注 1:** 有其他事件在进行时，编程设置对 RSEN 无效。
- 注 2:** 在重复启动条件发生期间，下列事件将会导致总线冲突：
- 当 SCL 由低电平变为高电平时，采样到 SDA 为低电平。
  - 在 SDA 被拉低之前，SCL 变为低电平。这表示另一个主器件正尝试发送数据 1。

**图 28-19:** 重复启动条件波形图



### 28.5.6 I<sup>2</sup>C 主模式发送

发送一个数据字节、一个 7 位地址或一个 10 位地址的另一半，都是通过写一个值到 SSPBUF 寄存器来实现的。该操作将使缓冲区满标志位（Buffer Full, BF）置 1，波特率发生器开始计数，同时启动下一次发送。地址/数据的每一位将在 SCL 的下降沿置为有效之后移出到 SDA 引脚。在一个波特率发生器计满返回周期（ $T_{BRG}$ ）内，SCL 保持低电平。在 SCL 被释放为高电平之前，数据应保持有效。当 SCL 引脚被释放为高电平时，它将在一个  $T_{BRG}$  内保持为高电平。在此期间以及 SCL 的下一个下降沿之后的一段时间内，SDA 引脚上的数据必须保持稳定。在第 8 位数据被移出（第 8 个时钟周期的下降沿）之后，BF 标志位被清零，同时主器件释放 SDA。此时如果发生地址匹配或是数据被正确接收，被寻址的从器件将在第 9 个位时间发出一个 ACK 位作为响应。ACK 的状态在第 9 个时钟的上升沿被写入 ACKSTAT 位。主器件接收到应答之后，应答状态位 ACKSTAT 会被清零。如果未收到应答，则该位保持置 1。第 9 个时钟周期之后，SSPIF 位会置 1，主控时钟（波特率发生器）暂停，直到下一个数据字节装入 SSPBUF。SCL 引脚保持低电平并且 SDA 保持不变（图 28-20）。

在写 SSPBUF 之后，地址的每一位在 SCL 的下降沿被移出，直至所有 7 位地址和 R/W 位都被移出。在第 8 个时钟的下降沿，主器件将释放 SDA 引脚，以允许从器件发出一个应答响应。在第 9 个时钟的下降沿，主器件通过采样 SDA 引脚来判断地址是否被从器件识别。ACK 位的状态被装入 SSPCON2 寄存器的 ACKSTAT 状态位。在发送地址的第 9 个时钟下降沿之后，SSPIF 置 1，BF 标志位清零，波特率发生器关闭直到下一次写 SSPBUF，同时 SCL 引脚保持低电平并且允许 SDA 引脚悬空。

#### 28.5.6.1 BF 状态标志

在发送模式下，SSPSTAT 寄存器的 BF 位在 CPU 写 SSPBUF 时置 1，在所有 8 位数据移出后清零。

#### 28.5.6.2 WCOL 状态标志

如果在发送过程中（即，SSPSR 仍在移出数据字节时）用户写 SSPBUF，则 WCOL 被置 1，同时缓冲区内容不变（未发生写操作）。

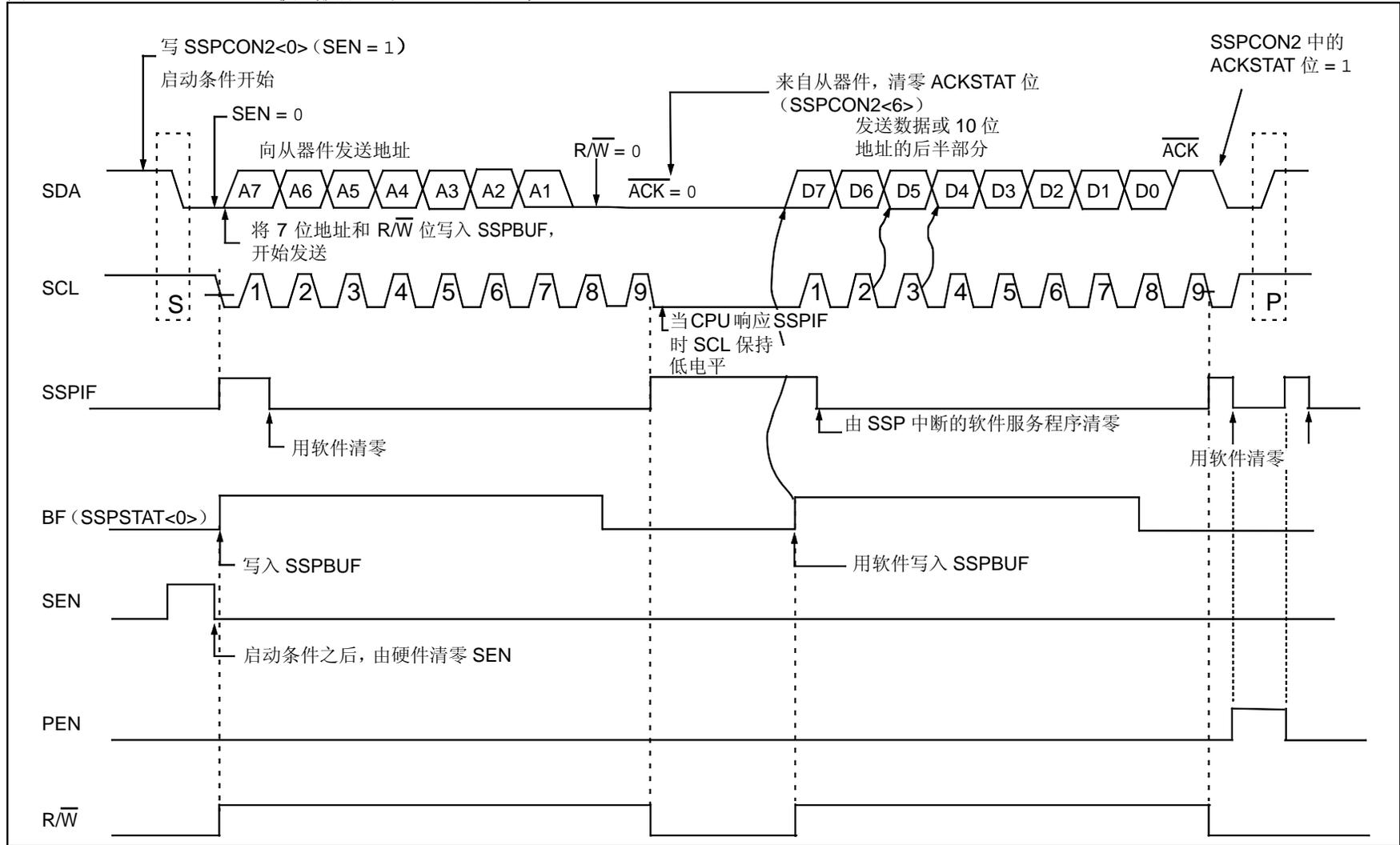
在下次发送前，WCOL 必须用软件清零。

#### 28.5.6.3 ACKSTAT 状态标志

在发送模式下，当从器件发送应答（ $\overline{ACK} = 0$ ）时，SSPCON2 寄存器的 ACKSTAT 位被清零；当从器件没有应答（ $\overline{ACK} = 1$ ）时，该位被置 1。从器件在识别出其地址（包括广播呼叫地址）或正确接收数据后，会发出一个应答。

#### 28.5.6.4 典型的发送序列：

1. 用户通过将 SSPCON2 寄存器的 SEN 位置 1，产生启动条件。
2. 在启动条件结束时，硬件将 SSPIF 置 1。
3. SSPIF 用软件清零。
4. 在进行下一步操作前，MSSP 模块将等待所需的启动时间。
5. 用户将从器件地址装入 SSPBUF 进行发送。
6. 器件地址从 SDA 引脚移出，直到发送完所有 8 位地址数据。数据发送会在写入 SSPBUF 后立刻开始。
7. MSSP 模块移入来自从器件的  $\overline{ACK}$  位，并将它的值写入 SSPCON2 寄存器的 ACKSTAT 位。
8. MSSP 模块在第 9 个时钟周期的末尾将 SSPIF 置 1，产生一个中断。
9. 用户将 8 位数据装入 SSPBUF。
10. 数据从 SDA 引脚移出，直到发送完所有 8 位数据。
11. MSSP 模块移入来自从器件的  $\overline{ACK}$  位，并将它的值写入 SSPCON2 寄存器的 ACKSTAT 位。
12. 对于发送的所有数据字节，重复步骤 8-11。
13. 用户通过将 SSPCON2 寄存器的 PEN 或 RSEN 位置 1，产生停止或重复启动条件。停止 / 重复启动条件完成时产生中断。

图 28-20: I<sup>2</sup>C™ 主模式波形 (发送, 7 或 10 位地址)

### 28.5.7 I<sup>2</sup>C 主模式接收

通过编程 SSPCON2 寄存器的接收使能位 (RCEN) 使能主模式接收。

**注：** RCEN 位置 1 前，MSSP 模块必须处于空闲状态，否则将忽略 RCEN 位。

波特率发生器开始计数，每次计满返回时，SCL 引脚的状态就会发生改变（由高变低或由低变高），同时数据被移入 SSPSR。在第 8 个时钟的下降沿之后，接收使能标志自动清零，SSPSR 的内容装入 SSPBUF，BF 标志位置 1，SSPIF 标志位置 1，波特率发生器暂停计数，且 SCL 保持为低电平。此时 MSSP 处于空闲状态，等待下一条命令。当 CPU 读缓冲区时，BF 标志位将自动清零。通过将 SSPCON2 寄存器的应答序列使能位 ACKEN 置 1，用户可以在接收结束后发送应答位。

#### 28.5.7.1 BF 状态标志

在接收过程中，当将地址或数据字节从 SSPSR 装入 SSPBUF 时，BF 位置 1。在读 SSPBUF 寄存器时清零。

#### 28.5.7.2 SSPOV 状态标志

在接收过程中，当 SSPSR 接收到 8 位数据且 BF 标志位已经在上一次接收中置 1 时，SSPOV 位置 1。

#### 28.5.7.3 WCOL 状态标志

如果在接收过程中（即，SSPSR 仍在移入数据字节时）用户写 SSPBUF，则 WCOL 位被置 1，同时缓冲区内内容不变（未发生写操作）。

#### 28.5.7.4 典型的接收序列：

1. 用户通过将 SSPCON2 寄存器的 SEN 位置 1，产生启动条件。
2. 在启动条件结束时，硬件将 SSPIF 置 1。
3. SSPIF 用软件清零。
4. 用户将要发送的从器件地址写入 SSPBUF 且 R/W 位置 1。
5. 器件地址从 SDA 引脚移出，直到发送完所有 8 位地址数据。数据发送会在写入 SSPBUF 后立刻开始。
6. MSSP 模块移入来自从器件的  $\overline{\text{ACK}}$  位，并将它的值写入 SSPCON2 寄存器的 ACKSTAT 位。
7. MSSP 模块在第 9 个时钟周期的末尾将 SSPIF 置 1，产生一个中断。
8. 用户将 SSPCON2 寄存器的 RCEN 位置 1，主器件随着时钟移入来自从器件的字节。
9. 在 SCL 的第 8 个下降沿之后，SSPIF 和 BF 置 1。
10. 主器件清零 SSPIF，并从 SSPBUF 中读取接收到的字节，使 BF 清零。
11. 主器件在 SSPCON2 寄存器的 ACKDT 位中设置要发送给从器件的  $\overline{\text{ACK}}$  值，并通过将 ACKEN 位置 1 来发出  $\overline{\text{ACK}}$ 。
12. 主器件随着时钟将  $\overline{\text{ACK}}$  移出到从器件，SSPIF 置 1。
13. 用户清零 SSPIF。
14. 对于从从器件接收的每个字节，重复步骤 8-13。
15. 主器件通过发送无  $\overline{\text{ACK}}$  或停止条件来结束通信。





## 28.5.10 休眠模式下的操作

在休眠模式下， $I^2C$  从模块能够接收地址或数据，并且在地址匹配或字节传输完成时，将处理器从休眠状态唤醒（如果允许 MSSP 中断）。

## 28.5.11 复位的影响

复位会禁止 MSSP 模块并终止当前的传输。

## 28.5.12 多主器件模式

在多主器件模式下，在检测到启动和停止条件时将产生中断，这可用于判断总线何时空闲。停止（P）位和启动（S）位在复位或禁止 MSSP 模块时清零。当 SSPSTAT 寄存器的 P 位置 1 时，或者总线空闲且 S 和 P 位都清零时，可以获得  $I^2C$  总线的控制权。当总线忙时，一旦出现停止条件，使能 SSP 中断的情况下将产生中断。

在多主器件模式下，必须一直监视 SDA 线来进行仲裁，以查看信号电平是否为期望的输出电平。此操作由硬件实现，其结果保存在 BCLIF 位中。

可能导致仲裁失败的情况是：

- 地址传输
- 数据传输
- 启动条件
- 重复启动条件
- 应答条件

## 28.5.13 多主器件通信、总线冲突和总线仲裁

多主器件模式是通过总线仲裁来支持的。当主器件将地址/数据位输出到 SDA 引脚时，如果一个主器件在 SDA 上输出 1（SDA 悬空为高电平），而另一个主器件输出 0，就会发生总线仲裁。当 SCL 引脚悬空为高电平时，数据应稳定。如果 SDA 引脚上期望的数据是 1，而实际采样到的数据是 0，则发生了总线冲突。主器件将把总线冲突中断标志位 BCLIF 置 1，并将  $I^2C$  端口复位到空闲状态（图 28-24）。

如果在发送过程中发生总线冲突，则会暂停发送，并且 BF 标志位被清零，SDA 和 SCL 线被置为无效，并且可写入 SSPBUF。当执行完总线冲突中断服务程序后，如果  $I^2C$  总线空闲，用户可通过发出启动条件恢复通信。

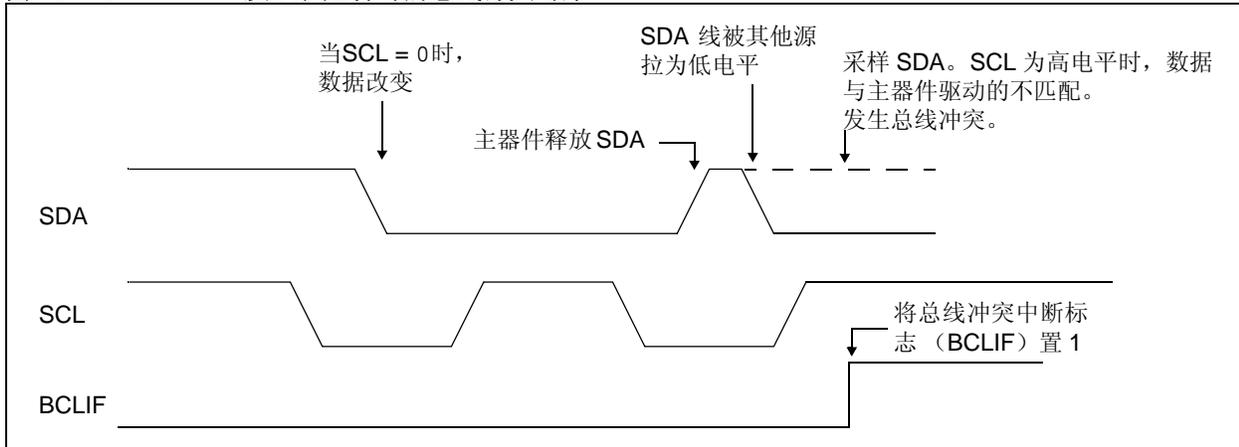
如果在启动、重复启动、停止或应答条件过程中发生总线冲突，则条件被中止，SDA 和 SCL 线被置为无效，SSPCON2 寄存器中的相应控制位清零。当执行完总线冲突中断服务程序后，如果  $I^2C$  总线空闲，用户可通过发出启动条件恢复通信。

主器件将继续监视 SDA 和 SCL 引脚。一旦产生停止条件，SSPIF 位将被置 1。

发生总线冲突时无论发送的进度如何，写入 SSPBUF 都会重新从第一个数据位开始发送数据。

在多主器件模式下，通过在检测到启动条件和停止条件时产生中断可以确定总线何时空闲。SSPSTAT 寄存器中的 P 位置 1 时，可以取得  $I^2C$  总线的控制权；否则总线处于空闲状态，且 P 位和 S 位均清零。

图 28-24: 发送和应答时的总线冲突时序



## 28.5.13.1 启动条件期间的总线冲突

启动条件期间，以下事件将导致总线冲突：

- 在启动条件开始时，SDA 或 SCL 被采样为低电平（图 28-25）。
- SDA 被置为低电平之前，SCL 采样为低电平（图 28-26）。

在启动条件期间，SDA 和 SCL 引脚都会被监视。

如果 SDA 引脚或 SCL 引脚已经是低电平，则发生以下所有事件：

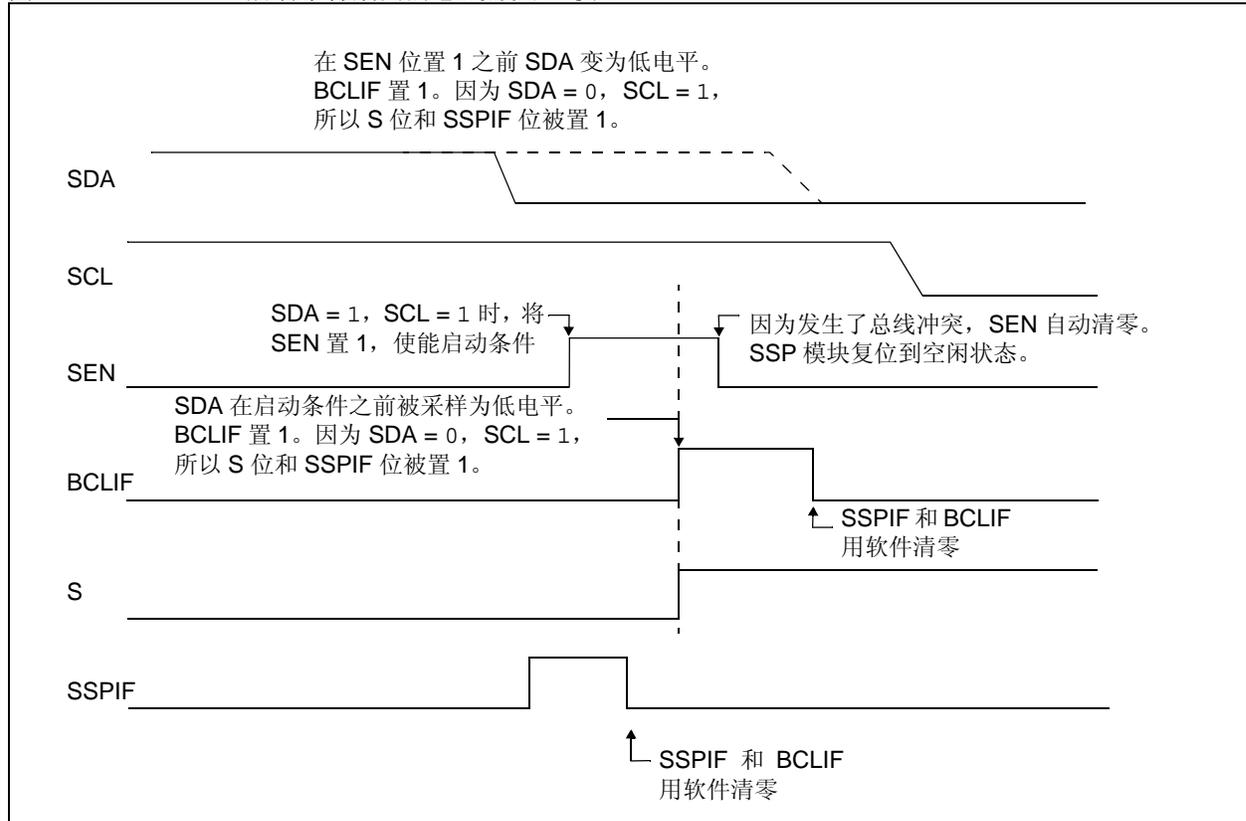
- 中止启动条件
- BCLIF 标志位置 1
- MSSP 模块复位为空闲状态（图 28-25）

启动条件从 SDA 和 SCL 引脚被置为无效开始。当 SDA 引脚采样为高电平时，波特率发生器装入值并递减计数。如果在 SDA 为高电平时，采样到 SCL 引脚为低电平，则发生了总线冲突，因为这表示另一台主器件正在启动条件期间试图发送一个数据 1。

如果 SDA 引脚在该计数周期内采样为低电平，则 BRG 复位，同时 SDA 线提前置为高电平（图 28-27）。但是，如果 SDA 引脚被采样为 1，SDA 引脚将在 BRG 计数结束时被置为低电平。接着，波特率发生器被重载并递减计数至 0；在此期间，如果 SCL 引脚采样到 0，则不会发生总线冲突。在 BRG 计数结束时，SCL 引脚被拉为低电平。

**注：** 在启动条件期间不太可能发生总线冲突，因为两个总线主器件不可能精确地在同一时刻发出启动条件。因此一个主器件将总是先于另一个主器件将 SDA 置为有效。但是上述情况不会引起总线冲突，因为两个主器件一定会对启动条件后的第一个地址进行仲裁。如果地址是相同的，必须继续对数据部分、重复启动条件或停止条件进行仲裁。

**图 28-25: 启动条件期间的总线冲突（仅 SDA）**



# MCP19114/5

图 28-26: 启动条件期间的总线冲突 (SCL = 0)

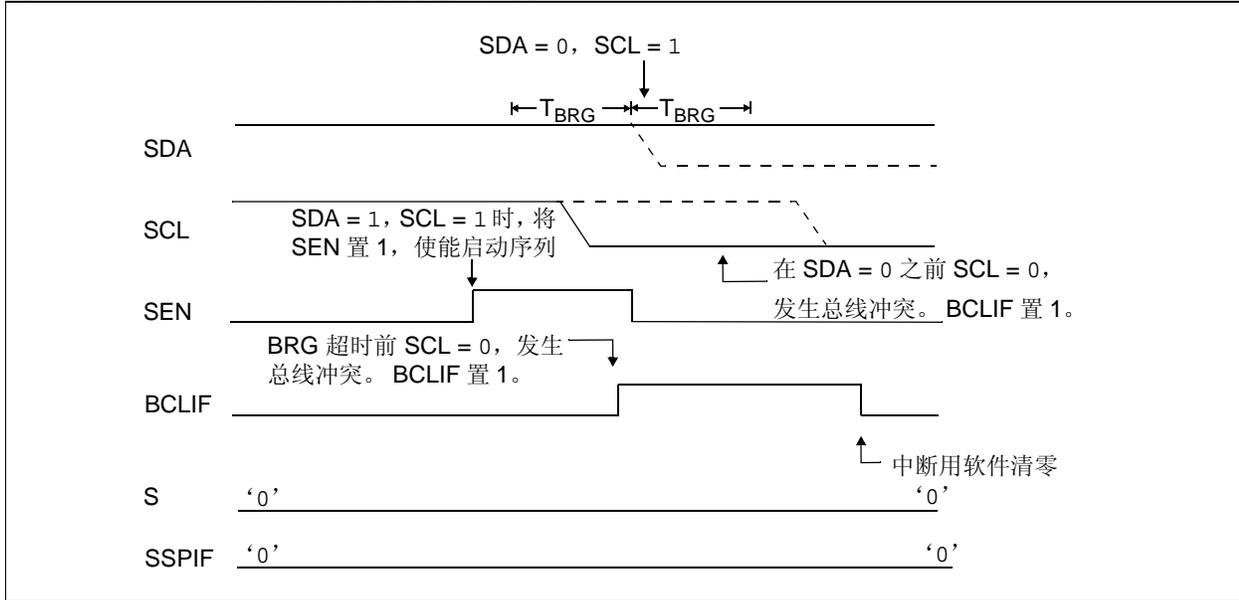
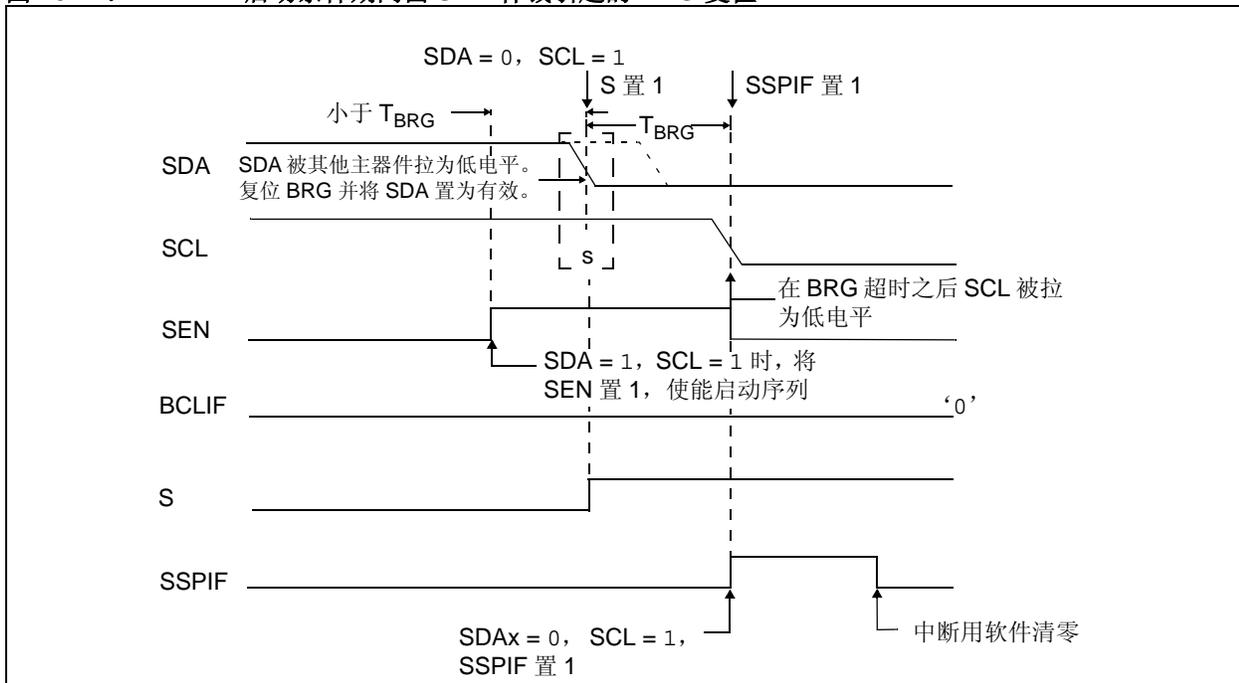


图 28-27: 启动条件期间由 SDA 仲裁引起的 BRG 复位



## 28.5.13.2 重复启动条件期间的总线冲突

在重复启动条件期间，以下事件会导致发生总线冲突：

- 在 SCL 由低电平变为高电平期间，在 SDA 上采样到低电平。
- 在 SDA 被置为低电平之前，SCL 变为低电平，表示另一个主器件正试图发送一个数据 1。

当用户释放 SDA 并允许该引脚悬空为高电平时，BRG 装入 SSPADD 的值并递减计数至 0。接着 SCL 引脚被置为无效，当 SCL 引脚被采样为高电平时，对 SDA 引脚进行采样。

如果 SDA 为低电平，则表示已发生了总线冲突（即，另一个主器件正试图发送一个数据 0，见图 28-28）。如果 SDA 被采样为高电平，则 BRG 被重载并开始计数。如果 SDA 在 BRG 超时之前从高电平变为低电平，则不会发生总线冲突，因为两个主器件不可能精确地在同一时刻将 SDA 置为有效。

如果 SCL 在 BRG 超时之前从高电平变为低电平，且 SDA 尚未被置为有效，那么将发生总线冲突。在此情况下，另一个主器件在重复启动条件期间正试图发送一个数据 1（见图 28-29）。

如果在 BRG 超时结束时 SCL 和 SDA 都仍然是高电平，则 SDA 引脚被拉低，BRG 被重载并开始计数。在计数结束时，不管 SCL 引脚的状态如何，SCL 引脚都被驱动为低电平，重复启动条件结束。

图 28-28: 重复启动条件期间的总线冲突（情形 1）

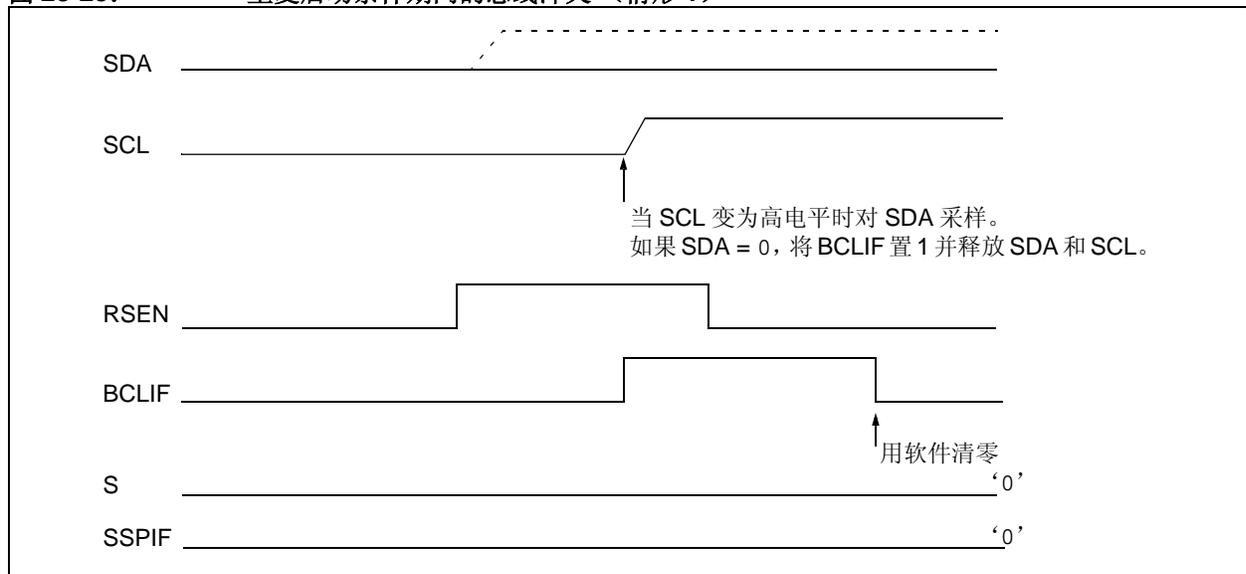
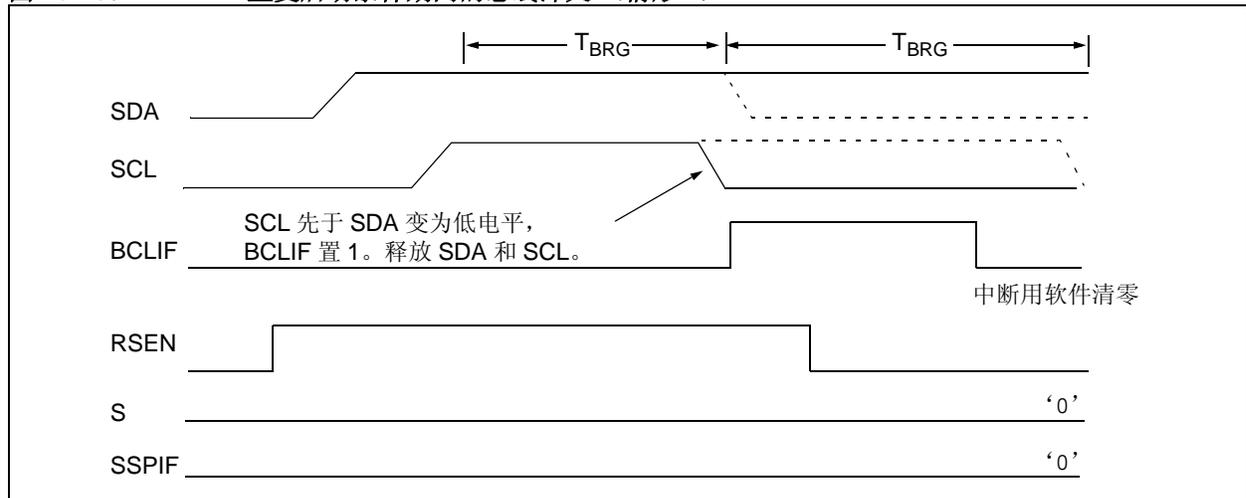


图 28-29: 重复启动条件期间的总线冲突（情形 2）



# MCP19114/5

## 28.5.13.3 停止条件期间的总线冲突

以下事件会导致停止条件期间发生总线冲突：

- a) SDA 已被置为无效并允许悬空为高电平之后，SDA 在 BRG 超时后采样为低电平。
- b) SCL 引脚被置为无效之后，SCL 在 SDA 变成高电平之前采样为低电平。

停止条件从 SDA 被置为低电平开始。当采样到 SDA 为低电平时，允许 SCL 引脚悬空。当引脚上采样到高电平（时钟仲裁）时，波特率发生器装入 SSPADD 的值并递减计数到 0。在 BRG 超时后，对 SDA 采样。如果采样到 SDA 为低电平，则表示已发生总线冲突。这是因为另一个主器件正试图发送数据 0（图 28-30）。如果 SCL 引脚在允许 SDA 悬空为高电平前被采样到低电平，也会发生总线冲突。这是另一个主器件正试图发送一个数据 0 的另外一种情况（图 28-31）。

图 28-30: 停止条件期间的总线冲突（情形 1）

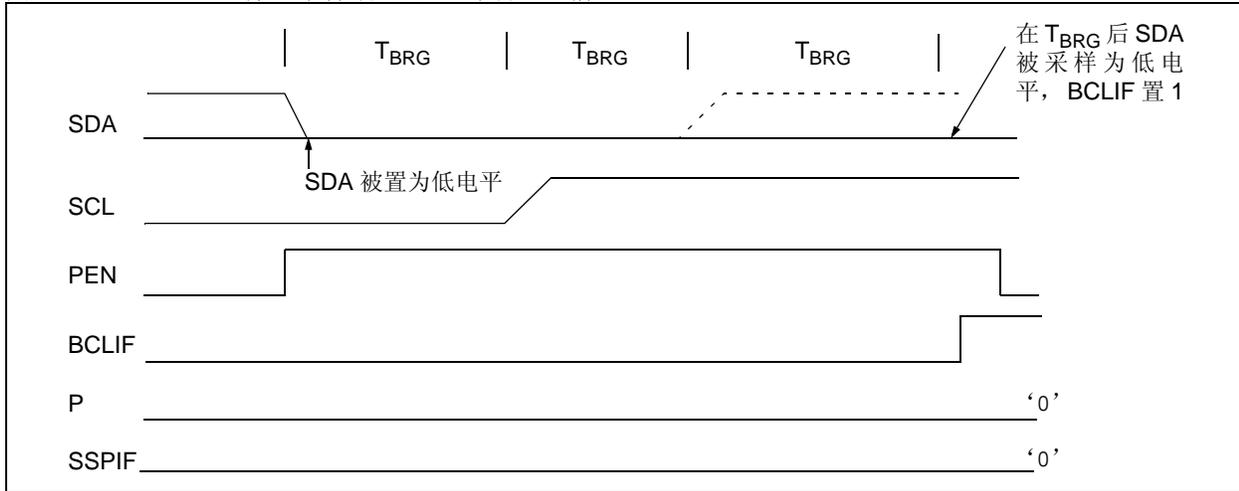


图 28-31: 停止条件期间的总线冲突（情形 2）

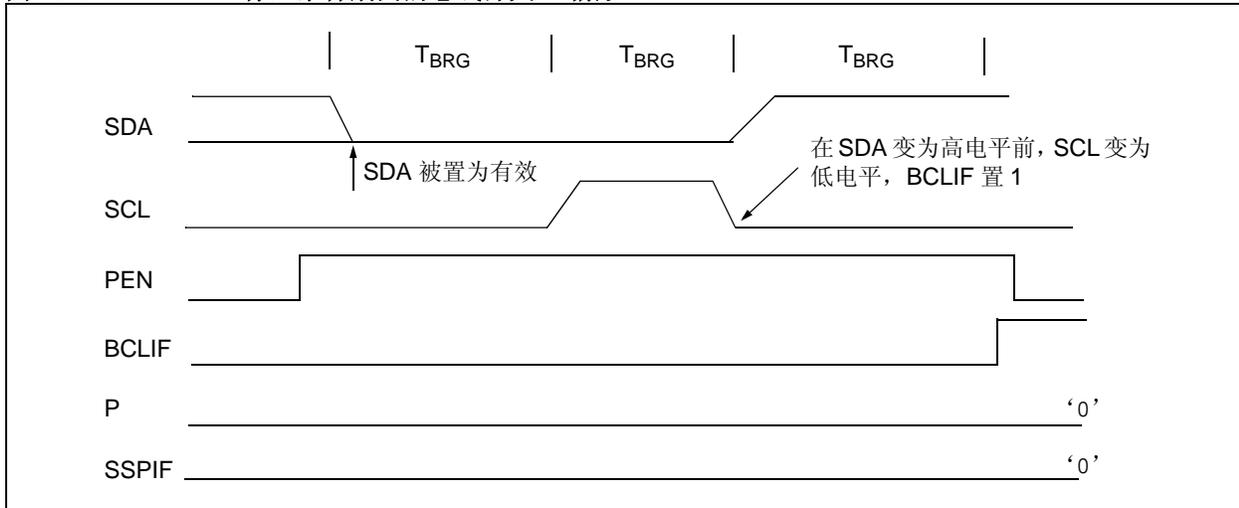


表 28-2: 与 I<sup>2</sup>C™ 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 所在页
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	95
PIE1	—	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	96
PIR1	—	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	98
TRISGPA	TRISA7	TRISA6	TRISA5	—	TRISA3	TRISA2	TRISA1	TRISA0	113
TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	TRISB1	TRISB0	118
SSPADD	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	196
SSPBUF	同步串口接收缓冲 / 发送寄存器								155*
SSPCON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	192
SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	194
SSPCON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	195
SSPMSK1	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0	196
SSPSTAT	SMP	CKE	D/A	P	S	R/W	UA	BF	191
SSPMSK2	MSK27	MSK26	MSK25	MSK24	MSK23	MSK22	MSK21	MSK20	197
SSPADD2	ADD27	ADD26	ADD25	ADD24	ADD23	ADD22	ADD21	ADD20	197

图注: — = 未实现, 读为 0。I<sup>2</sup>C 模式下的 MSSP 模块不使用阴影单元。

\* 此页提供寄存器信息。

# MCP19114/5

## 28.6 波特率发生器

MSSP 模块具有波特率发生器，用于在 I<sup>2</sup>C 主模式下生成时钟。波特率发生器 (BRG) 的重载值保存在 SSPADD 寄存器中。写 SSPBUF 时，波特率发生器将自动开始递减计数。

在给定操作完成时，内部时钟会自动停止计数，并且时钟引脚将保持它的最后状态。

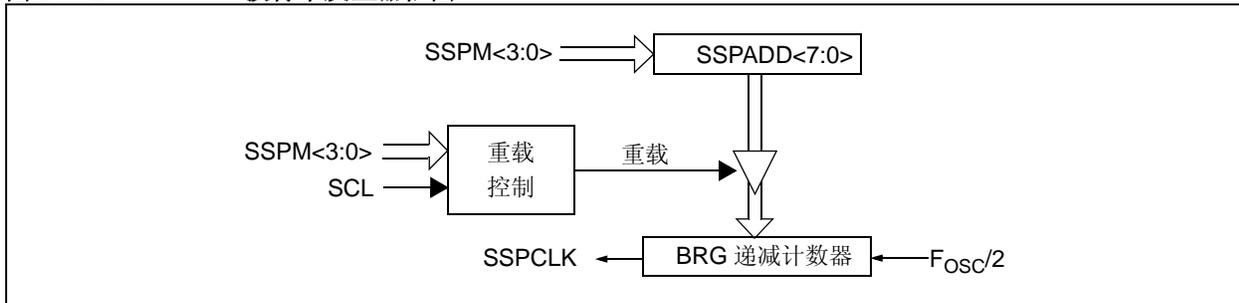
图 28-32 中的内部信号“重载”会使 SSPADD 的值装入 BRG 计数器。此操作在模块时钟信号的每个振荡周期内发生两次。指定重载信号何时置为有效的逻辑依赖于 MSSP 当前的工作模式。

表 28-3 列出了不同的指令周期下的时钟速率以及装入 SSPADD 的 BRG 值。

公式 28-1:

$$F_{CLOCK} = \frac{F_{OSC}}{(SSPADD + 1)(4)}$$

图 28-32: 波特率发生器框图



**注:** 在用作 I<sup>2</sup>C 的波特率发生器时，值 0x00、0x01 和 0x02 对于 SSPADD 是无效的。这是实现限制。

表 28-3: 不同 BRG 下的 MSSP 时钟速率

F <sub>OSC</sub>	F <sub>CY</sub>	BRG 值	F <sub>CLOCK</sub> (两次 BRG 计满返回)
8 MHz	2 MHz	04h	400 kHz <sup>(1)</sup>
8 MHz	2 MHz	0Bh	166 kHz
8 MHz	2 MHz	13h	100 kHz

**注 1:** I<sup>2</sup>C 接口并非在各个方面都符合 400 kHz I<sup>2</sup>C 规范 (该规范适用于速率大于 100 kHz 的情况)，在需要较高速率的应用场合应慎重使用。

## 寄存器 28-1: SSPSTAT: SSP 状态寄存器

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/A	P	S	R/W	UA	BF
bit 7							bit 0

### 图注:

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

bit 7	<b>SMP:</b> 数据输入采样位 1 = 禁止标准速度模式下的压摆率控制 (100 kHz 和 1 MHz) 0 = 使能高速模式下的压摆率控制 (400 kHz)
bit 6	<b>CKE:</b> 时钟边沿选择位 1 = 使能输入逻辑, 以使阈值符合 SMBus 规范 0 = 禁止 SMBus 特定输入
bit 5	<b>D/A:</b> 数据 / 地址位 1 = 指示上一个接收或发送的字节是数据 0 = 指示上一个接收或发送的字节是地址
bit 4	<b>P:</b> 停止位 (在 MSSP 模块被禁止且 SSPEN 被清零时, 该位会被清零。) 1 = 指示上次检测到停止位 (该位在复位时为 0) 0 = 上次未检测到停止位
bit 3	<b>S:</b> 启动位 (在 MSSP 模块被禁止且 SSPEN 被清零时, 该位会被清零。) 1 = 指示上次检测到启动位 (该位在复位时为 0) 0 = 上次未检测到启动位
bit 2	<b>R/W:</b> 读 / 写位信息 该位用来保存在最近一次地址匹配后的 R/W 位信息。该位仅在从地址匹配到出现下一个启动位、停止位或非 ACK 位之间有效。 <u>在 I<sup>2</sup>C 从模式下:</u> 1 = 读 0 = 写 <u>在 I<sup>2</sup>C 主模式下:</u> 1 = 正在进行发送 0 = 未进行发送 将该位与 SEN、RSEN、PEN、RCEN 或 ACKEN 进行逻辑或运算将指示 MSSP 是否处于空闲模式。
bit 1	<b>UA:</b> 更新地址位 (仅 10 位 I <sup>2</sup> C 模式) 1 = 指示用户需要更新 SSPADD 寄存器中的地址 0 = 不需要更新地址
bit 0	<b>BF:</b> 缓冲区满状态位 接收: 1 = 接收完成, SSPBUF 已满 0 = 接收未完成, SSPBUF 为空 发送: 1 = 数据发送正在进行 (不包括 ACK 和停止位), SSPBUF 已满 0 = 数据发送完成 (不包括 ACK 和停止位), SSPBUF 为空

# MCP19114/5

## 寄存器 28-2: SSPCON1: SSP 控制寄存器 1

R/C/HS-0	R/C/HS-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				
bit 7								bit 0

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n = POR 时的值
1 = 置 1	0 = 清零	HS = 硬件置 1 位      C = 用户清零位

bit 7      **WCOL:** 写冲突检测位

#### 主模式:

1 = I<sup>2</sup>C 不满足启动发送数据的条件时, 试图向 SSPBUF 寄存器写入数据  
0 = 未发生冲突

#### 从模式:

1 = 正在发送前一个字时, 又有数据写入 SSPBUF 寄存器 (必须用软件清零)  
0 = 未发生冲突

bit 6      **SSPOV:** 接收溢出指示位 <sup>(1)</sup>

1 = SSPBUF 寄存器仍持有前一字节时, 又接收到一个字节。在发送模式下, SSPOV 是“无关位”  
(必须用软件清零)。  
0 = 无溢出

bit 5      **SSPEN:** 同步串行端口使能位

在两种模式下, 当使能时, 必须将这些引脚正确地配置为输入或输出  
1 = 使能串口并将 SDA 和 SCL 配置为串口引脚源 <sup>(2)</sup>  
0 = 禁止串口并将上述引脚配置为 I/O 端口引脚

bit 4      **CKP:** 时钟极性选择位

#### 在 I<sup>2</sup>C 从模式下:

SCL 释放控制  
1 = 使能时钟  
0 = 保持时钟为低电平 (时钟延长) (用于确保数据建立时间。)

#### 在 I<sup>2</sup>C 主模式下:

在此模式下未使用

- 注 1: 在主模式下, 溢出位不会被置 1, 因为每次接收 (和发送) 新数据都是通过写入 SSPBUF 寄存器启动的。  
2: 当使能时, 必须将 SDA 和 SCL 引脚配置为输入引脚。  
3: 对于 I<sup>2</sup>C 模式, 不支持 SSPADD 值为 0、1 或 2。

**寄存器 28-2: SSPCON1: SSP 控制寄存器 1 (续)**bit 3-0 **SSPM<3:0>**: 同步串口模式选择位

0000 = 保留

0001 = 保留

0010 = 保留

0011 = 保留

0100 = 保留

0101 = 保留

0110 = I<sup>2</sup>C 从模式, 7 位地址0111 = I<sup>2</sup>C 从模式, 10 位地址1000 = I<sup>2</sup>C 主模式, 时钟 = F<sub>OSC</sub>/(4 \* (SSPADD+1))<sup>(3)</sup>

1001 = 保留

1010 = 保留

1011 = I<sup>2</sup>C 固件控制主模式 (从器件空闲)

1100 = 保留

1101 = 保留

1110 = I<sup>2</sup>C 从模式, 7 位地址, 并允许启动位和停止位中断1111 = I<sup>2</sup>C 从模式, 10 位地址, 并允许启动位和停止位中断

- 注 1:** 在主模式下, 溢出位不会被置 1, 因为每次接收 (和发送) 新数据都是通过写入 SSPBUF 寄存器启动的。
- 2:** 当使能时, 必须将 SDA 和 SCL 引脚配置为输入引脚。
- 3:** 对于 I<sup>2</sup>C 模式, 不支持 SSPADD 值为 0、1 或 2。

# MCP19114/5

## 寄存器 28-3: SSPCON2: SSP 控制寄存器 2

R/W-0/0	R-0/0	R/W-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/W/HS-0/0
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

### 图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	H = 由硬件置 1                      S = 由用户置 1

- bit 7      **GCEN:** 广播呼叫使能位 (仅限 I<sup>2</sup>C 从模式)  
1 = 当 SSPSR 寄存器接收到广播呼叫地址 (0x00 或 00h) 时允许中断  
0 = 禁止广播呼叫地址
- bit 6      **ACKSTAT:** 应答状态位 (仅限 I<sup>2</sup>C 模式)  
1 = 未接收到应答  
0 = 接收到应答
- bit 5      **ACKDT:** 应答数据位 (仅限 I<sup>2</sup>C 模式)  
在接收模式下:  
当用户在接收结束时发出一个应答序列时要发送的值  
1 = 无应答  
0 = 应答
- bit 4      **ACKEN:** 应答序列使能位 (仅限 I<sup>2</sup>C 主模式)  
在主接收模式下:  
1 = 在 SDA 和 SCL 引脚上发出应答序列, 并发送 ACKDT 数据位。由硬件自动清零。  
0 = 应答序列空闲
- bit 3      **RCEN:** 接收使能位 (仅限 I<sup>2</sup>C 主模式)  
1 = 使能 I<sup>2</sup>C 接收模式  
0 = 接收空闲
- bit 2      **PEN:** 停止条件使能位 (仅限 I<sup>2</sup>C 主模式)  
SCK 释放控制:  
1 = 在 SDA 和 SCL 引脚上发出停止条件。由硬件自动清零。  
0 = 停止条件空闲
- bit 1      **RSEN:** 重复启动条件使能位 (仅限 I<sup>2</sup>C 主模式)  
1 = 在 SDA 和 SCL 引脚上发出重复启动条件。由硬件自动清零。  
0 = 重复启动条件空闲
- bit 0      **SEN:** 启动条件使能位 (仅限 I<sup>2</sup>C 主模式)  
在主模式下:  
1 = 在 SDA 和 SCL 引脚上发出启动条件。由硬件自动清零。  
0 = 启动条件空闲  
在从模式下:  
1 = 使能从发送和从接收的时钟延长 (已使能延长)  
0 = 禁止时钟延长

注 1: 对于 ACKEN、RCEN、PEN、RSEN 和 SEN 位: 如果 I<sup>2</sup>C 模块不处于空闲模式, 该位可能不会被置 1 (不支持并行操作), 并且可能不会写入 SSPBUF (或禁止写入 SSPBUF)。

## 寄存器 28-4: SSPCON3: SSP 控制寄存器 3

R-0/0	R/W-0/0						
ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

## 图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7 **ACKTIM:** 应答时间状态位 (仅限 I<sup>2</sup>C 模式) <sup>(2)</sup>  
1 = 指示 I<sup>2</sup>C 总线处于应答序列中, 在 SCL 时钟的第 8 个下降沿置 1  
0 = 不处于应答序列中, 在 SCL 时钟的第 9 个上升沿清零
- bit 6 **PCIE:** 停止条件中断允许位 (仅限 I<sup>2</sup>C 模式)  
1 = 在检测到停止条件时允许中断  
0 = 禁止在检测到停止条件时中断 <sup>(1)</sup>
- bit 5 **SCIE:** 启动条件中断允许位 (仅限 I<sup>2</sup>C 模式)  
1 = 在检测到启动或重复启动条件时允许中断  
0 = 禁止在检测到启动或重复启动条件时中断 <sup>(1)</sup>
- bit 4 **BOEN:** 缓冲区改写使能位  
在 I<sup>2</sup>C 主模式下:  
该位被忽略。  
在 I<sup>2</sup>C 从模式下:  
1 = 仅当 BF 位 = 0 时, 在接收到地址 / 数据字节时, 更新 SSPBUF 并生成  $\overline{\text{ACK}}$  信号, 并忽略 SSPOV 位的状态。  
0 = 仅当 SSPOV 位清零时才更新 SSPBUF
- bit 3 **SDAHT:** SDA 保持时间选择位  
1 = 在 SCL 的下降沿之后, 在 SDA 上最少有 300 ns 的保持时间  
0 = 在 SCL 的下降沿之后, 在 SDA 上最少有 100 ns 的保持时间
- bit 2 **SBCDE:** 从模式总线冲突检测使能位 (仅限 I<sup>2</sup>C 从模式)  
如果在 SCL 的上升沿, 在模块输出高电平状态时采样到 SDA 为低电平, 则 PIR1 寄存器的 BCLIF 位会置 1, 总线会变为空闲状态  
1 = 允许从器件总线冲突中断  
0 = 禁止从器件总线冲突中断
- bit 1 **AHEN:** 地址保持使能位 (仅限 I<sup>2</sup>C 从模式)  
1 = 在所接收匹配地址字节的第 8 个 SCL 下降沿之后, SSPCON1 寄存器的 CKP 位将清零, SCL 将保持低电平。  
0 = 禁止地址保持
- bit 0 **DHEN:** 数据保持使能位 (仅限 I<sup>2</sup>C 从模式)  
1 = 在所接收数据字节的第 8 个 SCL 下降沿之后, 从器件硬件清零 SSPCON1 寄存器的 CKP 位, 而 SCL 则保持低电平。  
0 = 禁止数据保持

注 1: 在启动和停止条件检测明确列为使能的从模式下, 该位没有任何作用。

2: ACKTIM 状态位仅在 AHEN 位或 DHEN 位置 1 时有效。

# MCP19114/5

## 寄存器 28-5: SSPMSK1: SSP 掩码寄存器 1

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

### 图注:

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

- bit 7-1 **MSK<7:1>**: 掩码位  
1 = 接收到的地址 bit n 与 SSPADD<n> 相比较来检测 I<sup>2</sup>C 模式下地址是否匹配  
0 = 接收到的地址 bit n 不用于检测 I<sup>2</sup>C 模式下地址是否匹配
- bit 0 **MSK<0>**: 用于 I<sup>2</sup>C 从模式, 10 位地址的掩码位  
I<sup>2</sup>C 从模式, 10 位地址 (SSPM<3:0> = 0111 或 1111):  
1 = 接收到的地址 bit 0 与 SSPADD<0> 相比较来检测 I<sup>2</sup>C 模式下地址是否匹配  
0 = 接收到的地址 bit 0 不用于检测 I<sup>2</sup>C 模式下地址是否匹配  
I<sup>2</sup>C 从模式, 7 位地址, 该位被忽略

## 寄存器 28-6: SSPADD: MSSP 地址和波特率寄存器 1

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

### 图注:

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

### 主模式:

- bit 7-0 **ADD<7:0>**: 波特率时钟分频比位  
 $SCL \text{ 引脚时钟周期} = ((ADD<7:0> + 1) * 4) / F_{OSC}$

### 10 位从模式——地址高字节:

- bit 7-3 **未使用**: 地址高字节中不使用的位。该寄存器的位状态为“无关”。主器件发送的位格式由 I<sup>2</sup>C 规范确定, 必须等于 11110。但是, 那些位通过硬件进行比较, 并且不受该寄存器中的值影响。
- bit 2-1 **ADD<2:1>**: 10 位地址的高 2 位
- bit 0 **未使用**: 在此模式下未使用。位状态为“无关”。

### 10 位从模式——地址低字节:

- bit 7-0 **ADD<7:0>**: 10 位地址的低 8 位

### 7 位从模式:

- bit 7-1 **ADD<7:1>**: 7 位地址
- bit 0 **未使用**: 在此模式下未使用。位状态为“无关”。

## 寄存器 28-7: SSPMSK2: SSP 掩码寄存器 2

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK2<7:0>							
bit 7							bit 0

### 图注:

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

- bit 7-1      **MSK2<7:1>**: 掩码位  
 1 = 接收到的地址 bit n 与 SSPADD2<n> 相比较来检测 I<sup>2</sup>C 模式下地址是否匹配  
 0 = 接收到的地址 bit n 不用于检测 I<sup>2</sup>C 模式下地址是否匹配
- bit 0      **MSK2<0>**: 用于 I<sup>2</sup>C 从模式, 10 位地址的掩码位  
 I<sup>2</sup>C 从模式, 10 位地址 (SSPM<3:0> = 0111 或 1111):  
 1 = 接收到的地址 bit 0 与 SSPADD2<0> 相比较来检测 I<sup>2</sup>C 模式下地址是否匹配  
 0 = 接收到的地址 bit 0 不用于检测 I<sup>2</sup>C 模式下地址是否匹配  
 I<sup>2</sup>C 从模式, 7 位地址, 该位被忽略

## 寄存器 28-8: SSPADD2: MSSP 地址和波特率寄存器 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
ADD2<7:0>							
bit 7							bit 0

### 图注:

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

### 主模式:

- bit 7-0      **ADD2<7:0>**: 波特率时钟分频比位  
 SCL 引脚时钟周期 = ((ADD<7:0> + 1) \* 4)/F<sub>OSC</sub>

### 10 位从模式——地址高字节:

- bit 7-3      **未使用**: 地址高字节中不使用的位。该寄存器的位状态为“无关”。主器件发送的位格式由 I<sup>2</sup>C 规范确定, 必须等于 11110。但是, 那些位通过硬件进行比较, 并且不受该寄存器中的值影响。
- bit 2-1      **ADD2<2:1>**: 10 位地址的高 2 位
- bit 0      **未使用**: 在此模式下未使用。位状态为“无关”。

### 10 位从模式——地址低字节:

- bit 7-0      **ADD2<7:0>**: 10 位地址的低 8 位

### 7 位从模式:

- bit 7-1      **ADD2<7:1>**: 7 位地址
- bit 0      **未使用**: 在此模式下未使用。位状态为“无关”。

# MCP19114/5

---

注:



# MCP19114/5

表 29-2: MCP19114/5 指令集

助记符, 操作数	说明	周期数	14 位操作码				受影响的状态位	注	
			MSb		LSb				
<b>面向字节的文件寄存器操作</b>									
ADDWF	f, d	W 和 f 相加	1	00	0111	dfff	ffff	C、DC 和 Z	1, 2
ANDWF	f, d	W 和 f 作逻辑与运算	1	00	0101	dfff	ffff	Z	1, 2
CLRF	f	将 f 清零	1	00	0001	1fff	ffff	Z	2
CLRWF	-	将 W 清零	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	f 取反	1	00	1001	dfff	ffff	Z	1, 2
DECF	f, d	f 递减 1	1	00	0011	dfff	ffff	Z	1, 2
DECFSZ	f, d	f 递减 1, 为 0 则跳过	1(2)	00	1011	dfff	ffff		1, 2, 3
INCF	f, d	f 递增 1	1	00	1010	dfff	ffff	Z	1, 2
INCFSZ	f, d	f 递增 1, 为 0 则跳过	1(2)	00	1111	dfff	ffff		1, 2, 3
IORWF	f, d	W 与 f 作逻辑或运算	1	00	0100	dfff	ffff	Z	1, 2
MOVF	f, d	传送 f	1	00	1000	dfff	ffff	Z	1, 2
MOVWF	f	将 W 的内容送到 f	1	00	0000	1fff	ffff		
NOP	-	空操作	1	00	0000	0xx0	0000		
RLF	f, d	对 f 执行带进位的循环左移	1	00	1101	dfff	ffff	C	1, 2
RRF	f, d	对 f 执行带进位的循环右移	1	00	1100	dfff	ffff	C	1, 2
SUBWF	f, d	f 减去 W	1	00	0010	dfff	ffff	C、DC 和 Z	1, 2
SWAPF	f, d	将 f 中的两个半字节进行交换	1	00	1110	dfff	ffff		1, 2
XORWF	f, d	W 与 f 作逻辑异或运算	1	00	0110	dfff	ffff	Z	1, 2
<b>面向位的文件寄存器操作</b>									
BCF	f, b	将 f 中的某位清零	1	01	00bb	bfff	ffff		1, 2
BSF	f, b	将 f 中的某位置 1	1	01	01bb	bfff	ffff		1, 2
BTFSC	f, b	检测 f 中的某位, 为 0 则跳过	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f, b	检测 f 中的某位, 为 1 则跳过	1 (2)	01	11bb	bfff	ffff		3
<b>立即数和控制操作类指令</b>									
ADDLW	k	立即数和 W 相加	1	11	111x	kkkk	kkkk	C、DC 和 Z	
ANDLW	k	立即数与 W 作逻辑与运算	1	11	1001	kkkk	kkkk	Z	
CALL	k	调用子程序	2	10	0kkk	kkkk	kkkk		
CLRWD <sub>T</sub>	-	清零看门狗定时器	1	00	0000	0110	0100	$\overline{TO}$ 和 $\overline{PD}$	
GOTO	k	跳转到地址	2	10	1kkk	kkkk	kkkk		
IORLW	k	立即数与 W 作逻辑或运算	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	将立即数传送到 W	1	11	00xx	kkkk	kkkk		
RETFIE	-	从中断返回	2	00	0000	0000	1001		
RETLW	k	返回并将立即数传送到 W	2	11	01xx	kkkk	kkkk		
RETURN	-	从子程序返回	2	00	0000	0000	1000		
SLEEP	-	进入待机模式	1	00	0000	0110	0011	$\overline{TO}$ 和 $\overline{PD}$	
SUBLW	k	从立即数中减去 W 的内容	1	11	110x	kkkk	kkkk	C、DC 和 Z	
XORLW	k	立即数与 W 作逻辑异或运算	1	11	1010	kkkk	kkkk	Z	

注 1: 当 I/O 寄存器修改自身时 (例如: MOVF PORTA, 1), 修改时使用的值是引脚上的当前值。例如, 如果将一引脚配置为输入, 其对应数据锁存器中的值将为 1, 但此时若有外部器件将该引脚驱动为低电平, 则被写回数据锁存器的数据值将是 0。

2: 当对 TMR0 寄存器执行该指令 (并且 d = 1) 时, 如果已为 Timer0 模块分配了预分频器, 则将该预分频器清零。

3: 如果程序计数器 (PC) 被修改或条件测试为真, 则执行该指令需要两个周期。第二个周期执行一条 NOP 指令。

## 29.2 指令说明

**ADDLW**      立即数和 W 相加

语法:            [ 标号 ] ADDLW   k  
 操作数:         $0 \leq k \leq 255$   
 操作:             $(W) + k \rightarrow (W)$   
 受影响的状态位: C、DC 和 Z  
 说明:            将 W 寄存器的内容与 8 位立即数 k 相加, 结果存放到 W 寄存器。

**ADDWF**      W 和 f 相加

语法:            [ 标号 ] ADDWF   f,d  
 操作数:         $0 \leq f \leq 127$   
                    $d \in [0,1]$   
 操作:             $(W) + (f) \rightarrow (\text{目标寄存器})$   
 受影响的状态位: C、DC 和 Z  
 说明:            将 W 寄存器的内容与寄存器 f 的内容相加。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存放回 f 寄存器。

**ANDLW**      立即数与 W 作逻辑与运算

语法:            [ 标号 ] ANDLW   k  
 操作数:         $0 \leq k \leq 255$   
 操作:             $(W) .AND.(k) \rightarrow (W)$   
 受影响的状态位: Z  
 说明:            将 W 寄存器的内容与 8 位立即数 k 进行逻辑与运算。结果存放到 W 寄存器。

**ANDWF**      W 和 f 作逻辑与运算

语法:            [ 标号 ] ANDWF   f,d  
 操作数:         $0 \leq f \leq 127$   
                    $d \in [0,1]$   
 操作:             $(W) .AND.(f) \rightarrow (\text{目标寄存器})$   
 受影响的状态位: Z  
 说明:            将 W 寄存器的内容与寄存器 f 的内容进行逻辑与运算。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存放回 f 寄存器。

**BCF**            将 f 中的某位清零

语法:            [ 标号 ] BCF    f,b  
 操作数:         $0 \leq f \leq 127$   
                    $0 \leq b \leq 7$   
 操作:             $0 \rightarrow (f<b>)$   
 受影响的状态位: 无  
 说明:            将寄存器 f 中的位 b 清零。

**BSF**            将 f 中的某位置 1

语法:            [ 标号 ] BSF    f,b  
 操作数:         $0 \leq f \leq 127$   
                    $0 \leq b \leq 7$   
 操作:             $1 \rightarrow (f<b>)$   
 受影响的状态位: 无  
 说明:            将寄存器 f 中的位 b 置 1。

**BTFSC**        检测 f 中的某位, 为 0 则跳过

语法:            [ 标号 ] BTFSC f,b  
 操作数:         $0 \leq f \leq 127$   
                    $0 \leq b \leq 7$   
 操作:            如果  $(f<b>) = 0$ , 则跳过  
 受影响的状态位: 无  
 说明:            如果寄存器 f 中的位 b 为 1, 则执行下一条指令。  
                   如果寄存器 f 中的位 b 为 0, 则丢弃下一条指令, 转而执行一条 NOP 指令, 从而使该指令成为双周期指令。

# MCP19114/5

<b>BTFSS</b>	<b>检测 f 中的某位，为 1 则跳过</b>
语法:	[ 标号] BTFSS f,b
操作数:	$0 \leq f \leq 127$ $0 \leq b < 7$
操作:	如果 $(f < b) = 1$ ，则跳过
受影响的状态位:	无
说明:	如果寄存器 f 中的位 b 为 0，则执行下一条指令。 如果位 b 为 1，则丢弃下一条指令，转而执行一条 NOP 指令，从而使该指令成为双周期指令。

<b>CALL</b>	<b>调用子程序</b>
语法:	[ 标号] CALL k
操作数:	$0 \leq k \leq 2047$
操作:	$(PC) + 1 \rightarrow TOS$ , $k \rightarrow PC < 10:0 >$ , $(PCLATH < 4:3 >) \rightarrow PC < 12:11 >$
受影响的状态位:	无
说明:	调用子程序。首先，将返回地址 $(PC + 1)$ 压入堆栈。11 位立即数地址装入 PC 位 $< 10:0 >$ 。将 PCLATH 的内容装入 PC 的高位。CALL 是双周期指令。

<b>CLRF</b>	<b>将 f 清零</b>
语法:	[ 标号] CLRF f
操作数:	$0 \leq f \leq 127$
操作:	$00h \rightarrow (f)$ $1 \rightarrow Z$
受影响的状态位:	Z
说明:	寄存器 f 的内容被清零，并且 Z 位被置 1。

<b>CLRW</b>	<b>将 W 清零</b>
语法:	[ 标号] CLRW
操作数:	无
操作:	$00h \rightarrow (W)$ $1 \rightarrow Z$
受影响的状态位:	Z
说明:	W 寄存器被清零。全零标志位 (Z) 置 1。

<b>CLRWDT</b>	<b>清零看门狗定时器</b>
语法:	[ 标号] CLRWDT
操作数:	无
操作:	$00h \rightarrow WDT$ $0 \rightarrow \overline{WDT}$ 预分频器, $1 \rightarrow \overline{TO}$ $1 \rightarrow \overline{PD}$
受影响的状态位:	$\overline{TO}$ 和 $\overline{PD}$
说明:	CLRWDT 指令复位看门狗定时器。还将复位 WDT 的预分频器。STATUS 寄存器中的 $\overline{TO}$ 和 $\overline{PD}$ 位置 1。

<b>COMF</b>	<b>f 取反</b>
语法:	[ 标号] COMF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(\bar{f}) \rightarrow (\text{目标寄存器})$
受影响的状态位:	Z
说明:	将寄存器 f 的内容取反。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

<b>DECF</b>	<b>f 递减 1</b>
语法:	[ 标号] DECF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - 1 \rightarrow (\text{目标寄存器})$
受影响的状态位:	Z
说明:	将寄存器 f 的内容减 1。如果 d 为 0，结果存放到 W 寄存器。如果 d 为 1，结果存放回 f 寄存器。

<b>DECFSZ</b>	<b>f 递减 1, 为 0 则跳过</b>
语法:	[ 标号] DECFSZ f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - 1 \rightarrow$ ( 目标寄存器 ); 结果 = 0 则跳过
受影响的状态位:	无
说明:	将寄存器 f 的内容递减 1。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。如果结果为 1, 则执行下一条指令。如果结果为 0, 则转而执行一条 NOP 指令, 从而使该指令成为双周期指令。

<b>GOTO</b>	<b>无条件跳转</b>
语法:	[ 标号] GOTO k
操作数:	$0 \leq k \leq 2047$
操作:	$k \rightarrow PC<10:0>$ $PCLATH<4:3> \rightarrow PC<12:11>$
受影响的状态位:	无
说明:	GOTO 是无条件跳转指令。11 位立即数被装入 PC 位 $<10:0>$ 。将 PCLATH $<4:3>$ 的内容装入 PC 的高位。GOTO 是双周期指令。

<b>INCF</b>	<b>f 递增 1</b>
语法:	[ 标号] INCF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) + 1 \rightarrow$ ( 目标寄存器 )
受影响的状态位:	Z
说明:	将寄存器 f 的内容递增 1。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。

<b>INCFSZ</b>	<b>f 递增 1, 为 0 则跳过</b>
语法:	[ 标号] INCFSZ f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) + 1 \rightarrow$ ( 目标寄存器 ), 结果 = 0 则跳过
受影响的状态位:	无
说明:	将寄存器 f 的内容递增 1。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。如果结果为 1, 则执行下一条指令。如果结果为 0, 则转而执行一条 NOP 指令, 从而使该指令成为双周期指令。

<b>IORLW</b>	<b>立即数与 W 作逻辑或运算</b>
语法:	[ 标号] IORLW k
操作数:	$0 \leq k \leq 255$
操作:	$(W) .OR. k \rightarrow (W)$
受影响的状态位:	Z
说明:	将 W 寄存器的内容与 8 位立即数 k 进行逻辑或运算。结果存放到 W 寄存器。

<b>IORWF</b>	<b>W 与 f 作逻辑或运算</b>
语法:	[ 标号] IORWF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(W) .OR. (f) \rightarrow$ ( 目标寄存器 )
受影响的状态位:	Z
说明:	将 W 寄存器的内容与寄存器 f 的内容进行逻辑或运算。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。

# MCP19114/5

<b>MOVF</b>	<b>传送 f</b>
语法:	[ 标号] MOVF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	(f) → (目标寄存器)
受影响的状态位:	Z
说明:	根据 d 的状态, 将寄存器 f 的内容送入目标寄存器。如果 d = 0, 则目标寄存器为 W。如果 d = 1, 目标寄存器为文件寄存器 f 本身。由于状态标志位 Z 会受影响, 可用 d = 1 对文件寄存器进行检测。
指令字数:	1
指令周期数:	1
示例:	MOVF FSR, 0 执行指令后 W = FSR 寄存器中的值 Z = 1

<b>MOVLW</b>	<b>将立即数传送到 W</b>
语法:	[ 标号] MOVLW k
操作数:	$0 \leq k \leq 255$
操作:	$k \rightarrow (W)$
受影响的状态位:	无
说明:	将 8 位立即数 k 装入 W 寄存器。其余“无关位”均汇编为 0。
指令字数:	1
指令周期数:	1
示例:	MOVLW 0x5A 执行指令后 W = 0x5A

<b>MOVWF</b>	<b>将 W 的内容送到 f</b>
语法:	[ 标号] MOVWF f
操作数:	$0 \leq f \leq 127$
操作:	(W) → (f)
受影响的状态位:	无
说明:	将 W 寄存器中的数据传送到寄存器 f。
指令字数:	1
指令周期数:	1
示例:	MOVWF OPTION F 执行指令前 OPTION = 0xFF W = 0x4F 执行指令后 OPTION = 0x4F W = 0x4F

<b>NOP</b>	<b>空操作</b>
语法:	[ 标号] NOP
操作数:	无
操作:	空操作
受影响的状态位:	无
说明:	空操作。
指令字数:	1
指令周期数:	1
示例:	NOP

## RETFIE 从中断返回

语法: [标号] RETFIE

操作数: 无

操作: TOS → PC,  
1 → GIE

受影响的状态位: 无

说明: 从中断返回。执行出栈操作, 将栈顶 (Top-of-Stack, TOS) 的内容装入 PC。通过将全局中断允许位 GIE (INTCON<7>) 置 1, 来允许中断。这是一条双周期指令。

指令字数: 1

指令周期数: 2

示例: RETFIE

中断后

```
PC = TOS
GIE = 1
```

## RETLW 返回并将立即数传送到 W

语法: [标号] RETLW k

操作数:  $0 \leq k \leq 255$

操作:  $k \rightarrow (W)$ ;  
TOS → PC

受影响的状态位: 无

说明: 将 8 位立即数 k 装入 W 寄存器。将栈顶内容 (返回地址) 装入程序计数器。这是一条双周期指令。

指令字数: 1

指令周期数: 2

示例: CALL TABLE;W contains  
;table offset  
;value

```
GOTO DONE
TABLE
•
•
ADDWF PC ;W = offset
RETLW k1 ;Begin table
RETLW k2 ;
•
•
•
RETLW kn ;End of table
DONE
```

执行指令前

```
W = 0x07
```

执行指令后

```
W = k8 的值
```

## RETURN 从子程序返回

语法: [标号] RETURN

操作数: 无

操作: TOS → PC

受影响的状态位: 无

说明: 从子程序返回。执行出栈操作, 将栈顶 (TOS) 的内容装入程序计数器。这是一条双周期指令。

## RLF 对 f 执行带进位的循环左移

语法: [标号] RLF f,d

操作数:  $0 \leq f \leq 127$   
 $d \in [0,1]$

操作: 参见下面的说明

受影响的状态位: C

说明: 将寄存器 f 的内容连同进位标志位一起循环左移 1 位。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。



指令字数: 1

指令周期数: 1

示例: RLF REG1,0

执行指令前

```
REG1 = 1110 0110
C = 0
```

执行指令后

```
REG1 = 1110 0110
W = 1100 1100
C = 1
```

# MCP19114/5

**RRF**                    **对 f 执行带进位的循环右移**

语法:                    [ 标号 ] RRF f,d

操作数:                 $0 \leq f \leq 127$   
 $d \in [0,1]$

操作:                    参见下面的说明

受影响的状态位:    C

说明:                    将寄存器 f 的内容连同进位标志位一起循环右移 1 位。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。



**SUBWF**                **f 减去 W**

语法:                    [ 标号 ] SUBWF f,d

操作数:                 $0 \leq f \leq 127$   
 $d \in [0,1]$

操作:                     $(f) - (W) \rightarrow$ ( 目标寄存器 )

受影响的状态位:    C、DC 和 Z

说明:                    用寄存器 f 的内容减去 W 寄存器的内容 (采用二进制补码方法进行运算)。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。

C = 0	W > f
C = 1	W ≤ f
DC = 0	W<3:0> > f<3:0>
DC = 1	W<3:0> ≤ f<3:0>

**SLEEP**                **进入休眠模式**

语法:                    [ 标号 ] SLEEP

操作数:                无

操作:                    00h → WDT,  
 $0 \rightarrow \overline{WDT}$  预分频器,  
 $1 \rightarrow \overline{TO}$ ,  
 $0 \rightarrow \overline{PD}$

受影响的状态位:     $\overline{TO}$  和  $\overline{PD}$

说明:                    掉电状态位  $\overline{PD}$  清零。超时状态位  $\overline{TO}$  置 1。看门狗定时器及其预分频器清零。振荡器停振, 处理器进入休眠模式。

**SWAPF**                **将 f 中的两个半字节进行交换**

语法:                    [ 标号 ] SWAPF f,d

操作数:                 $0 \leq f \leq 127$   
 $d \in [0,1]$

操作:                     $(f<3:0>) \rightarrow$ ( 目标寄存器 <7:4>),  
 $(f<7:4>) \rightarrow$ ( 目标寄存器 <3:0>)

受影响的状态位:    无

说明:                    将寄存器 f 的高半字节和低半字节交换。如果 d 为 0, 结果存放到 W 寄存器。如果 d 为 1, 结果存放到 f 寄存器。

**SUBLW**                **从立即数中减去 W 的内容**

语法:                    [ 标号 ] SUBLW k

操作数:                 $0 \leq k \leq 255$

操作:                     $k - (W) \rightarrow (W)$

受影响的状态位:    C、DC 和 Z

说明:                    从 8 位立即数 k 中减去 W 寄存器的内容 (采用二进制补码方法进行运算)。结果存放到 W 寄存器。

结果	条件
C = 0	W > k
C = 1	W ≤ k
DC = 0	W<3:0> > k<3:0>
DC = 1	W<3:0> ≤ k<3:0>

**XORLW**                **立即数与 W 作逻辑异或运算**

语法:                    [ 标号 ] XORLW k

操作数:                 $0 \leq k \leq 255$

操作:                     $(W) .XOR. k \rightarrow (W)$

受影响的状态位:    Z

说明:                    将 W 寄存器中的内容与 8 位立即数 k 进行逻辑异或运算。结果存放到 W 寄存器。

<b>XORWF</b>	<b>W 和 f 作逻辑异或运算</b>
语法:	[ 标号 ] XORWF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	(W) .XOR.(f) →( 目标寄存器 )
受影响的状态位:	Z
说明:	将 W 寄存器的内容与寄存器 f 的内容进行逻辑异或运算。如果 d 为 0，结果存放到 W 寄存器。如果 d 为 1，结果存回 f 寄存器。

# MCP19114/5

---

注:

## 30.0 在线串行编程 (ICSP™)

ICSP 允许用户在生产电路板时使用未编程器件。编程可以在装配流程之后完成，从而可以使用最新版本的固件或者定制固件对器件编程。ICSP 编程需要 5 个引脚：

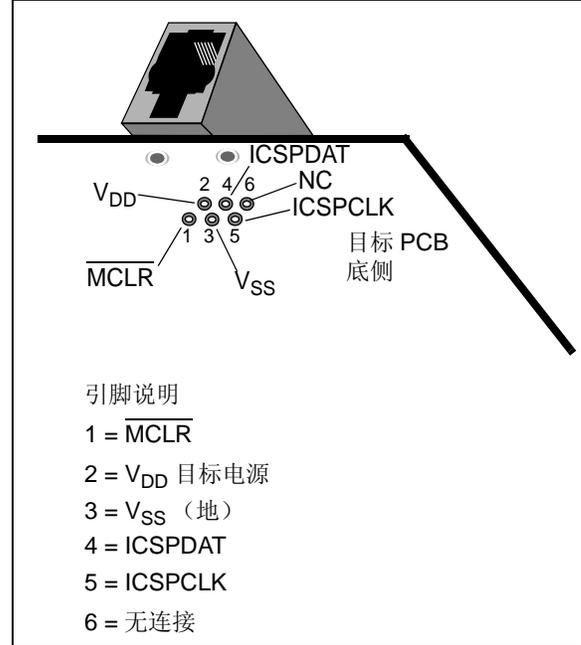
- ICSPCLK
- ICSPDAT
- MCLR
- V<sub>DD</sub>
- V<sub>SS</sub> (A<sub>GND</sub>)

在编程 / 校验模式下，通过串行通信对程序存储器、用户 ID 和配置字进行编程。ICSPDAT 引脚是用于传输串行数据的双向 I/O，ICSPCLK 引脚是时钟输入引脚。通过将 ICSPDAT 和 ICSPCLK 引脚保持为低电平，同时将 MCLR 引脚上的电压从 V<sub>IL</sub> 升至 V<sub>IH</sub>，可将器件置于编程 / 校验模式。

## 30.1 通用编程接口

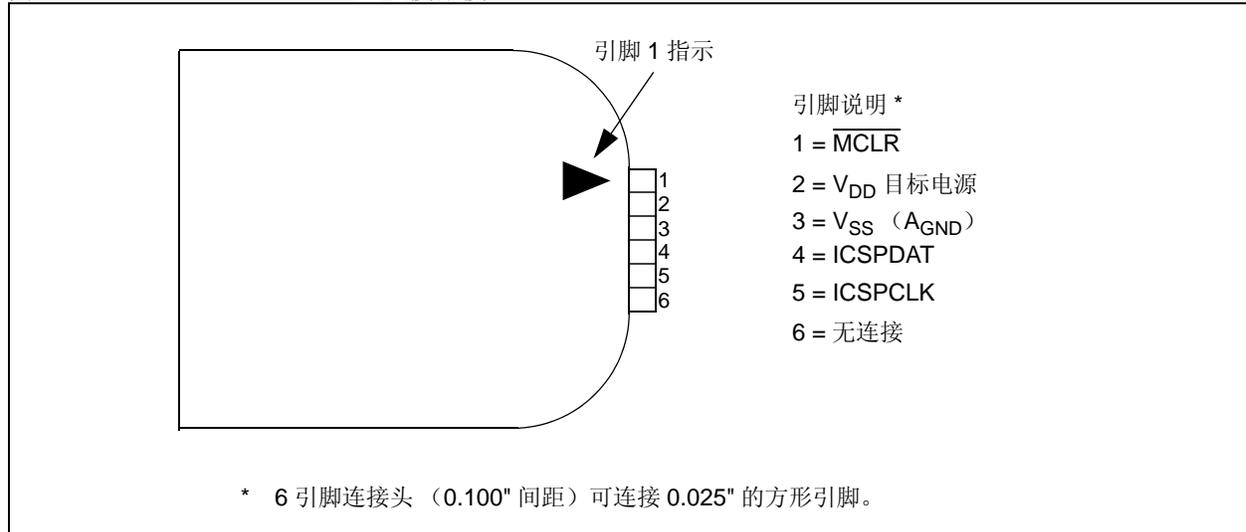
通常，与目标器件的连接通过一个 ICSP 连接头完成。开发工具上常见的连接器是采用 6P6C（6 个引脚和 6 个连接器）配置的 RJ-11。请参见图 30-1。

图 30-1: ICD RJ-11 型连接器接口



另一种常用于 PICKIT™ 编程器的连接器是间距为 0.1 英寸的标准 6 引脚连接头。请参见图 30-2。

图 30-2: PICKIT™ 型连接器接口

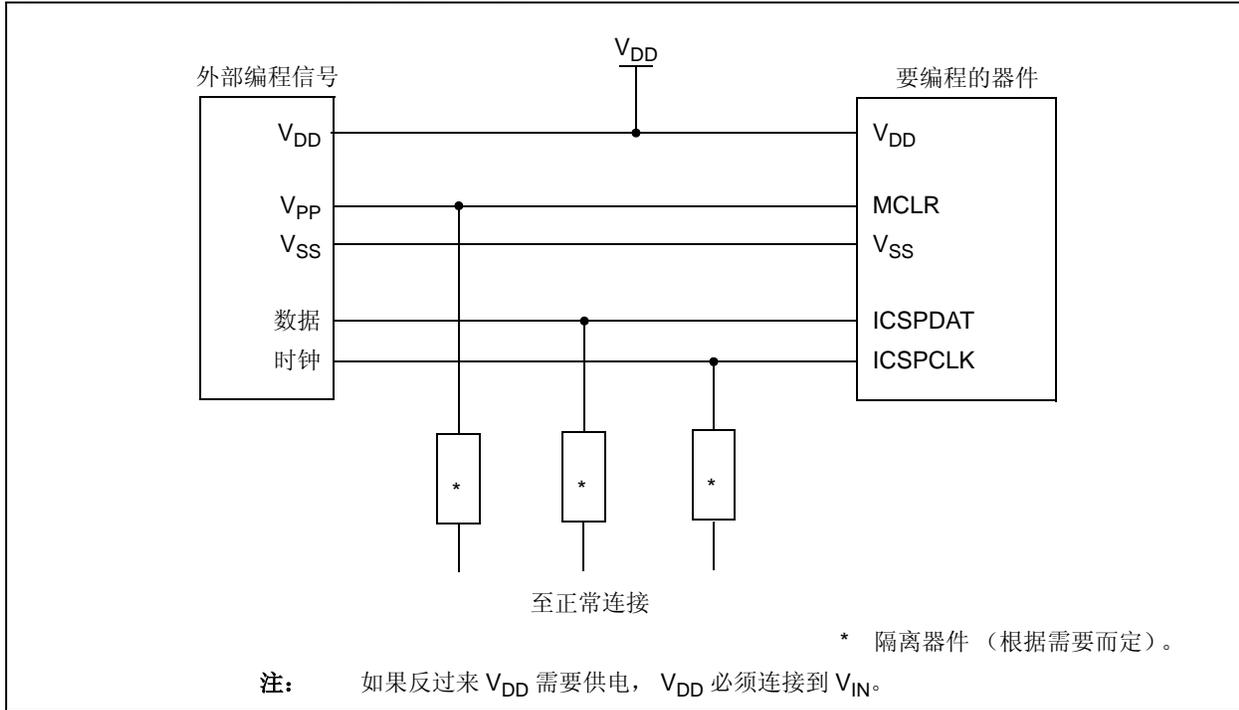


# MCP19114/5

关于其他推荐接口，在进行 PCB 设计之前，请参见具体的器件编程器手册。

建议使用隔离器件将编程引脚与其他电路隔离。隔离类型主要取决于具体应用，可能包括电阻、二极管甚至跳线之类的器件。更多信息，请参见图 30-3。

图 30-3: ICSP™ 编程的典型连接



## 31.0 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

## 31.1 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

## 31.2 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

## 31.3 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

## 31.4 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

## 31.5 适用于各种器件系列的 MPLAB 汇编器、链接器和库管理器

MPLAB汇编器为PIC24和PIC32 MCU以及dsPIC DSC器件从符号汇编语言生成可重定位机器码。MPLAB XC编译器使用该汇编器生成目标文件。汇编器产生可重定位目标文件之后，可将这些目标文件存档，或与其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特性：

- 支持整个器件指令集
- 支持定点和浮点数据
- 命令行接口
- 丰富的指令集
- 与MPLAB X IDE兼容

## 31.6 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。

## 31.7 MPLAB REAL ICE 在线仿真器系统

MPLAB REAL ICE 在线仿真器系统是 Microchip 针对其闪存 DSC 和 MCU 器件推出的新一代高速仿真器。结合 MPLAB X IDE 易于使用且功能强大的图形用户界面，该仿真器可对所有 8 位、16 位和 32 位 MCU 及 DSC 器件进行调试和编程。

该仿真器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与在线调试器系统兼容的连接器和新型抗噪声、高速低压差分信号 (LVDS) 互连电缆 (CAT5) 与目标板相连。

可通过 MPLAB X IDE 下载将来版本的固件，对该仿真器进行现场升级。在同类仿真器中，MPLAB REAL ICE 的优势十分明显：全速仿真、运行时变量观察、跟踪分析、复杂断点、逻辑探针、耐用的探针接口及较长（长达 3 米）的互连电缆。

## 31.8 MPLAB ICD 3 在线调试器系统

MPLAB ICD 3 在线调试器系统是 Microchip 成本效益最高的高速硬件调试器 / 编程器，适用于 Microchip 的闪存 DSC 和 MCU 器件。结合 MPLAB X IDE 功能强大但易于使用的图形用户界面，该调试器可对 PIC 闪存单片机和 dsPIC DSC 进行调试和编程。

MPLAB ICD 3 在线调试器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与 MPLAB ICD 2 或 MPLAB REAL ICE 系统兼容的连接器和目标板相连。MPLAB ICD 3 支持所有 MPLAB ICD 2 连接器。

## 31.9 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、编译器和 MPLAB IDE 软件)。

## 31.10 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。

## 31.11 演示 / 开发板、评估工具包及入门工具包

有许多演示、开发和评估板可用于各种 PIC MCU 和 dsPIC DSC，实现对全功能系统的快速应用开发。大多数的演示、开发和评估板都有实验布线区，供用户添加定制电路；还有应用固件和源代码，用于检查和修改。

这些板支持多种功能部件，包括 LED、温度传感器、开关、扬声器、RS-232 接口、LCD 显示器、电位计和附加 EEPROM 存储器。

演示和开发板可用于教学环境，在实验布线区设计定制电路，从而掌握各种单片机应用。

除了 PICDEM™ 和 dsPICDEM™ 演示 / 开发板系列电路外，Microchip 还有一系列评估工具包和演示软件，适用于模拟滤波器设计、KEELOQ® 数据安全产品 IC、CAN、IrDA®、PowerSmart 电池管理、SEEVAL® 评估系统、 $\Sigma$ - $\Delta$  ADC、流速传感器，等等。

同时还提供入门工具包，其中包含体验指定器件功能所需的所有软硬件。通常提供单个应用以及调试功能，都包含在一块电路板上。

有关演示、开发和评估工具包的完整列表，请访问 Microchip 网站 ([www.microchip.com](http://www.microchip.com))。

## 31.12 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。

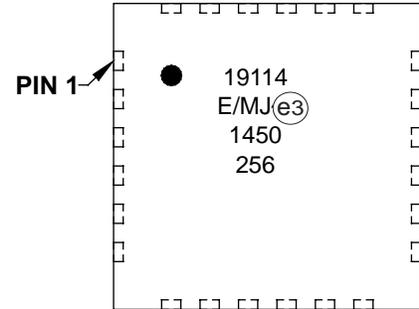
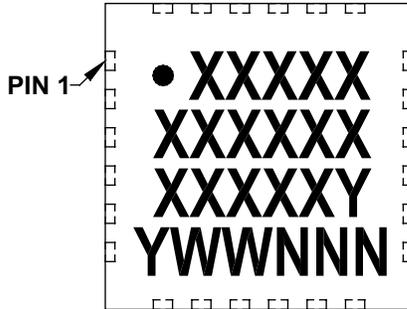
- SoftLog 和 CCS 等公司提供的器件编程器和量产编程器
- Gimpel 和 Trace Systems 等公司提供的软件工具
- Saleae 和 Total Phase 等公司提供的协议分析器
- MikroElektronika、Digilent® 和 Olimex 等公司提供的演示板
- EZ Web Lynx、WIZnet 和 ILogika® 等公司提供的嵌入式以太网解决方案

## 32.0 封装信息

### 32.1 封装标识信息

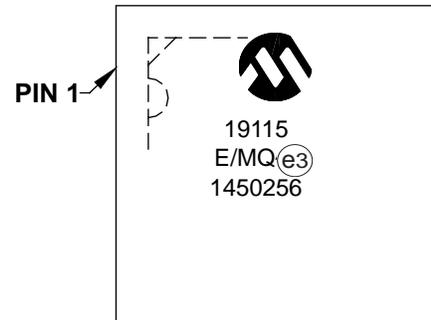
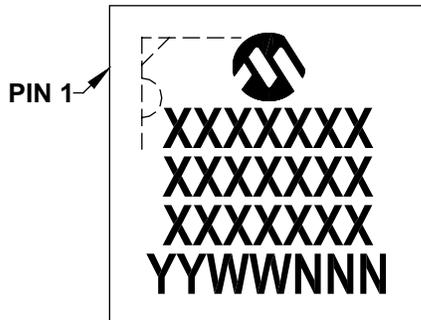
24 引脚 QFN (4x4x0.9 mm) (仅 MCP19114)

示例



28 引脚 QFN (5x5x0.9 mm)

示例

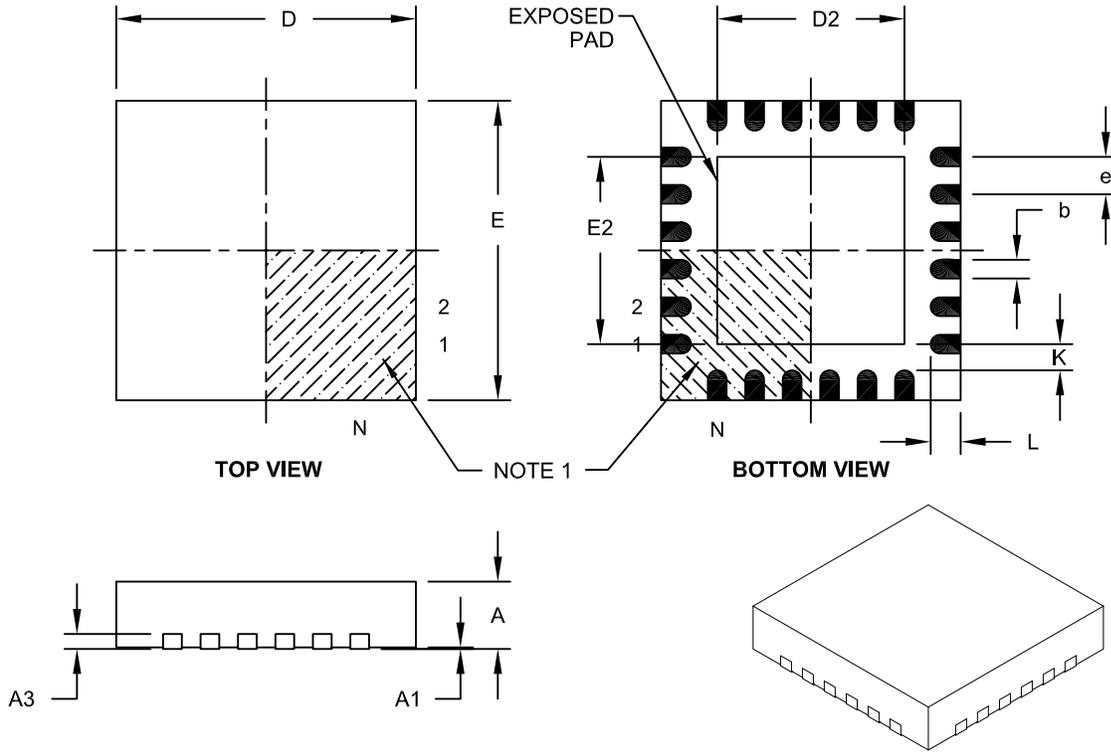


<b>图注:</b>	XX...X 客户指定信息 Y 年份代码 (日历年的最后一位数字) YY 年份代码 (日历年的最后两位数字) WW 星期代码 (一月一日的星期代码为“01”) NNN 以字母数字排序的追踪代码 (e3) 雾锡 (Matte Tin, Sn) 的 JEDEC® 无铅标志 * 表示无铅封装。JEDEC 无铅标志 (e3) 标示于此种封装的外包装上。
<b>注:</b>	Microchip 器件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户信息的字符数。

# MCP19114/5

## 24 引脚塑封四方扁平无脚封装 (MJ) —— 主体 4x4x0.9 mm [QFN]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	24		
Pitch	e	0.50 BSC		
Overall Height	A	0.80	0.85	0.90
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Width	E	4.00 BSC		
Exposed Pad Width	E2	2.40	2.50	2.60
Overall Length	D	4.00 BSC		
Exposed Pad Length	D2	2.40	2.50	2.60
Contact Width	b	0.20	0.25	0.30
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.

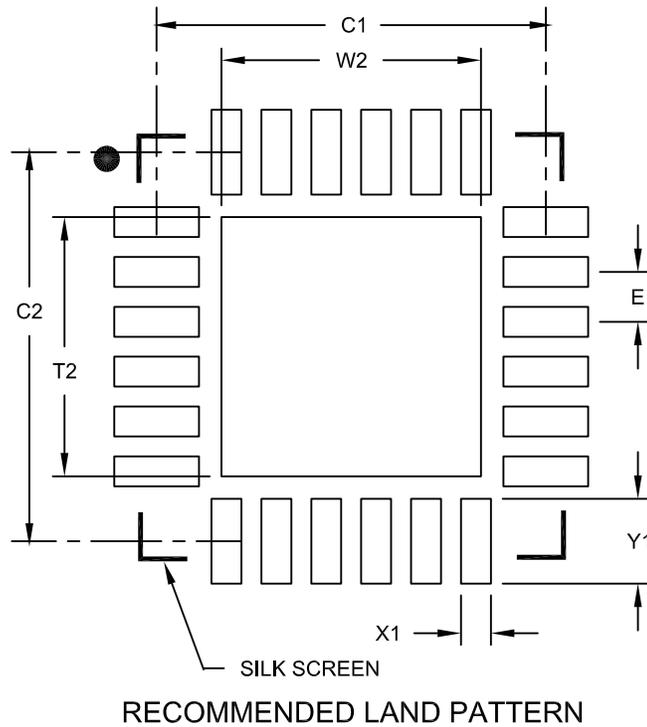
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-143A

## 24 引脚塑封四方扁平无脚封装 (MJ) ——主体 4x4 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		0.50 BSC		
Optional Center Pad Width	W2				2.60
Optional Center Pad Length	T2				2.60
Contact Pad Spacing	C1			3.90	
Contact Pad Spacing	C2			3.90	
Contact Pad Width	X1				0.30
Contact Pad Length	Y1				0.85

**Notes:**

1. Dimensioning and tolerancing per ASME Y14.5M

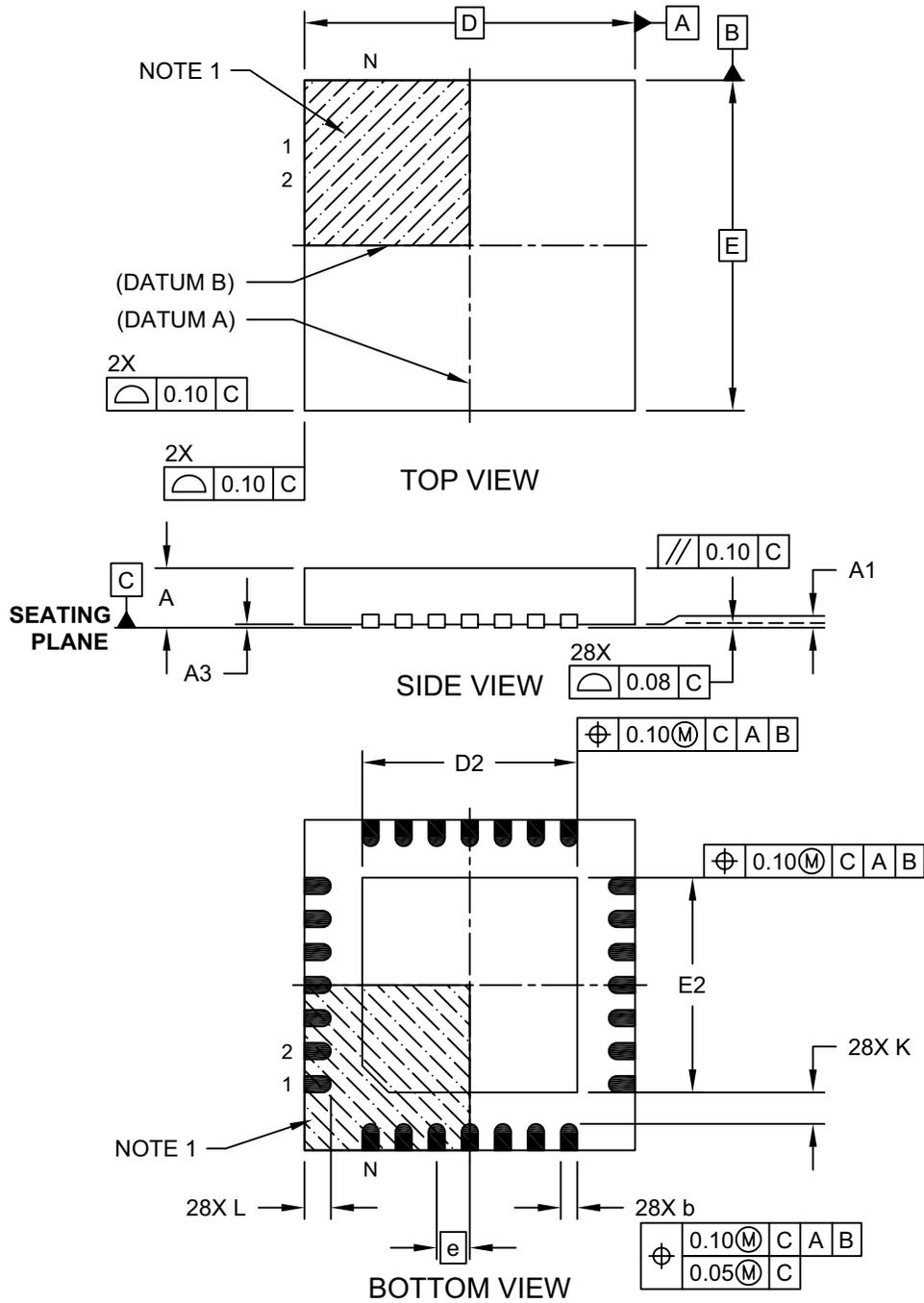
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2143B

# MCP19114/5

## 28 引脚塑封四方扁平无脚封装 (MQ) —— 主体 5x5x0.9 mm [QFN 或 VQFN]

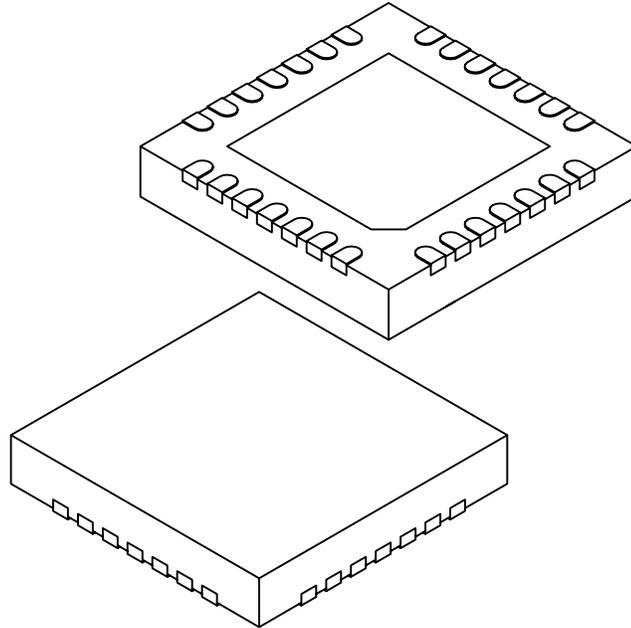
注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Microchip Technology Drawing C04-140C Sheet 1 of 2

## 28 引脚塑封四方扁平无脚封装 (MQ) —— 主体 5x5x0.9 mm [QFN 或 VQFN]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	0.50 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Width	E	5.00 BSC		
Exposed Pad Width	E2	3.15	3.25	3.35
Overall Length	D	5.00 BSC		
Exposed Pad Length	D2	3.15	3.25	3.35
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.35	0.40	0.45
Contact-to-Exposed Pad	K	0.20	-	-

### Notes:

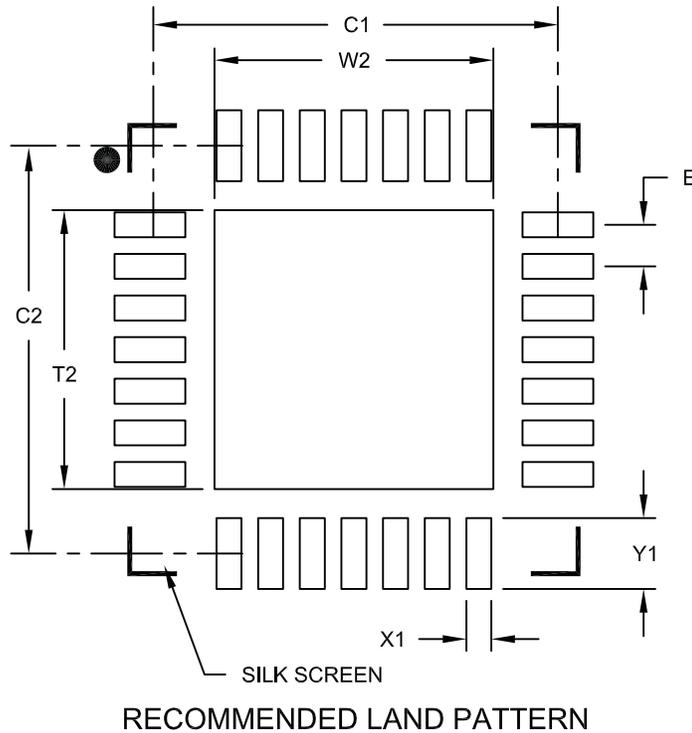
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.
  - BSC: Basic Dimension. Theoretically exact value shown without tolerances.
  - REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-140C Sheet 2 of 2

# MCP19114/5

28 引脚塑封四方扁平无脚封装 (MQ) —— 主体 5x5 mm [QFN] 焊盘布局, 触点长度 0.55 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	0.50 BSC		
Optional Center Pad Width	W2			3.35
Optional Center Pad Length	T2			3.35
Contact Pad Spacing	C1		4.90	
Contact Pad Spacing	C2		4.90	
Contact Pad Width (X28)	X1			0.30
Contact Pad Length (X28)	Y1			0.85

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2140A

## 附录 A： 版本历史

### 版本 B（2015 年 2 月）

以下为修改列表：

1. 更新了特性部分。
2. 更新了电气特性表。
3. 更新了表 5-5 和 5-6。
4. 更新了第 2.0 节“引脚说明”中关于 GPIO 上电的注释。
5. 更新了第 6.1 节“输入欠压和过压锁定（UVLO 和 OVLO）”。
6. 在第 7.0 节“典型性能曲线”中更新了图 7-14 并添加了图 7-15 和 7-16。
7. 更新了第 13.1 节“上电复位（POR）”和第 13.3 节“欠压复位（BOR）”。
8. 更新了第 15.0 节“掉电模式（休眠）”。
9. 更新了第 20.2 节“温度输出”。
10. 更新了公式 20-1。

### 版本 A（2014 年 3 月）

- 本文档的初始版本。

# MCP19114/5

---

注:

## 索引

## A

A/D	
规范	3, 5, 35
A/D 转换。请参见 ADC	
ABECON 寄存器	49
ACKSTAT	
状态标志	179
ADC	127
10 位结果格式	129
采集要求	133
端口配置	128
工作原理	130
寄存器定义	131
计算采集时间	133
框图	127
内部采样开关 (R <sub>SS</sub> ) 阻抗	133
配置	128
配置中断	130
时序图	36
通道选择	128
相关寄存器	135
信号源阻抗 (R <sub>S</sub> )	133
休眠期间的工作	130
要求	35
中断	129
转换时钟	128
转换步骤	130
ADCON0 寄存器	57, 131
ADCON1 寄存器	132
ADRESH 寄存器	132
ADRESL 寄存器	132
ANSELA 寄存器	114
ANSELB 寄存器	119

## B

BF	181
状态标志	179
BF 状态标志	181
比较模式	
框图	150
编程, 器件指令	199
变更通知客户服务	229
捕捉模式	
框图	149
捕捉 / 比较 / PWM 模块	
捕捉模式	149
CCP1IF	149
CCX 引脚配置	149
预分频器	149
软件中断	149
Timer1 模式选择	149
比较模式	
CCP1IF	150
CMPX 引脚配置	150
软件中断	150
特殊事件触发器	150
Timer1 模式选择	150
寄存器	151

## C

CCDCON 寄存器	151
C 编译器	
MPLAB XC	212

操作码字段说明	199
存储器构成	69
数据	70
程序	69
程序存储器	
映射和堆栈 (MCP19114)	69
构成	69
保护	82
初级输入电流失调调节	42

## D

DEADCON 寄存器	45
DESATCON 寄存器	41
代码示例	
间接寻址	80
代码示例	
A/D 转换	130
将预分频器分配给 Timer0	138
将预分频器分配给 WDT	138
初始化 PORTGPA	111
将 STATUS 和 W 寄存器的值保存在 RAM 中	100
电流检测	19
电平变化中断	121
相关寄存器	123
清零中断标志	121
使能模块	121
独立的引脚配置	121
休眠期间的工作	121
寄存器	122
电气特性	22
典型性能曲线	53
掉电模式 (休眠)	101
相关寄存器	102
定时器	
Timer1 (T1CON)	139
Timer2 (T2CON)	143
读 - 修改 - 写操作	199
堆栈	79

## F

FSR	
寄存器	79
峰值电流模式	19
封装	215
标识	215
规格	216
复位	85
看门狗定时器	88
欠压	87
确定复位原因	89
上电	86
相关寄存器	91

## G

GPR	
寄存器	70
固件指令	199
过压锁定	
输入	37

## H

汇编器	
MPASM 汇编器	212

# MCP19114/5

## I

### I/O

端口 ..... 111

### I<sup>2</sup>C 模式 (MSSP)

#### 从模式

SSPMSK1 寄存器 ..... 175

10 位地址接收 ..... 170

发送 ..... 166

工作原理 ..... 160

时钟延长 ..... 174

时钟同步 ..... 174

广播呼叫地址支持 ..... 175

总线冲突 ..... 166

操作 ..... 158

读 / 写位信息 (R/W 位) ..... 160

多主器件通信、总线冲突和总线仲裁 ..... 184

多主器件模式 ..... 184

复位的影响 ..... 184

概述 ..... 156

使用 BRG 的 I<sup>2</sup>C 时钟速率 ..... 190

停止条件时序 ..... 183

相关寄存器 ..... 189

休眠模式下的操作 ..... 184

应答序列 ..... 160

应答序列时序 ..... 183

总线冲突

重复启动条件期间 ..... 187

启动条件期间 ..... 185

停止条件期间 ..... 188

主模式 ..... 176

重复启动条件时序 ..... 178

发送 ..... 179

工作原理 ..... 176

接收 ..... 181

启动条件时序 ..... 177

时钟仲裁 ..... 176

ICLBCON 寄存器 ..... 43

ICOACON 寄存器 ..... 42

### INDF

寄存器 ..... 79

INTCON 寄存器 ..... 95

IOCA 寄存器 ..... 122

IOCB 寄存器 ..... 122

## J

### 寄存器

ABECON(模拟模块使能控制) ..... 49

ADCON0 (A/D 控制寄存器 0) ..... 131

ADCON0 (模数控制寄存器) ..... 57

ADCON1 (A/D 控制寄存器 1) ..... 132

ADRESH (ADC 结果寄存器的高字节) ..... 132

ADRESL (ADC 结果寄存器的低字节) ..... 132

ANSELA(模拟选择 GPA) ..... 114

ANSELB(模拟选择 GPB) ..... 119

CALWD1 (校准字 1) ..... 59

CALWD10 (校准字 10) ..... 66

CALWD11 (校准字 11) ..... 67

CALWD2 (校准字 2) ..... 60

CALWD3 (校准字 3) ..... 61

CALWD4 (校准字 4) ..... 61

CALWD5 (校准字 5) ..... 62

CALWD6 (校准字 6) ..... 62

CALWD7 (校准字 7) ..... 63

CALWD8 (校准字 8) ..... 64

CALWD9 (校准字 9) ..... 65

CCDCON(双捕捉 / 比较控制模块) ..... 151

CONFIG(配置字) ..... 81

DEADCON(驱动器死区控制) ..... 45

DESATCON(退饱和和比较器控制) ..... 41

FSR(文件选择寄存器) ..... 79

ICLEBCON(输入电流前沿消隐控制) ..... 43

ICOACON(输入电流失凋调节控制) ..... 42

INDF ..... 79

INTCON(中断控制) ..... 95

IOCA(中断电平变化 PORTGPA) ..... 122

IOCB(中断电平变化 PORTGPB) ..... 122

MODECON(主器件 / 从器件和 RFB 多路开关控制) ..... 51

OPTION\_REG(选项) ..... 78

OSCTUNE(振荡器调节) ..... 83

OVCN(输出过压比较器控制) ..... 39

OVREFCON(输出过压检测电平) ..... 39

PCON(电源控制) ..... 88, 91

PE1(模拟外设使能控制) ..... 48

PIE1(外设中断允许 1) ..... 96

PIE2(外设中断允许 2) ..... 97

PIR1(外设中断标志 1) ..... 98

PIR2(外设中断标志 2) ..... 99

PMADRH(程序存储器地址高字节) ..... 105, 107

PMADRL(程序存储器地址低字节) ..... 105, 106

PMCON1(程序存储器控制 1) ..... 105, 107

PMCON2(程序存储器控制 2) ..... 105

PMDATH(程序存储器数据高字节) ..... 106

PMDATL(程序存储器数据低字节) ..... 106

PORTGPA ..... 111, 113

PORTGPB ..... 116, 117

SLPCRCON(斜率补偿斜升控制) ..... 44

SSPADD(MSSP 地址和波特率 1) ..... 196

SSPADD2(MSSP 地址 2) ..... 197

SSPCON1(SSP 控制 1) ..... 192

SSPCON2(SSP 控制 2) ..... 194

SSPCON3(SSP 控制 3) ..... 195

SSPMSK(SSP 掩码 1) ..... 196

SSPMSK1(SSP 掩码) ..... 175

SSPMSK2(SSP 掩码 2) ..... 197

SSPSTAT(SSP 状态) ..... 191

STATUS ..... 71

T1CON(Timer1 控制) ..... 140

T2CON(Timer2 控制) ..... 144

TRISGPA(PORTGPA 三态) ..... 111, 113

TRISGPB(PORTGPB 三态) ..... 116, 118

VINCON(UVLO 和 OVLO 比较器控制) ..... 37

VINOVLO(输入过压锁定) ..... 38

VINUVLO(输入欠压锁定) ..... 38

VREF2CON(VREF2 电压设定值) ..... 47

VREFCON(电流 / 电压调节设定值控制) ..... 46

WPUGPA(弱上拉 PORTGPA) ..... 114

WPUGPB(弱上拉 PORTGPB) ..... 118

特殊功能 ..... 70

特殊功能寄存器汇总

Bank 0 ..... 74

Bank 1 ..... 75

Bank 2 ..... 76

Bank 3 ..... 77

通用寄存器 ..... 70

计算函数调用 ..... 79

计算 GOTO ..... 79

基准测试

系统 ..... 57

间接寻址 ..... 79, 80

交流特性 ..... 30

绝对最大值 .....	22	模式和 RFB 多路开关控制 .....	51
<b>K</b>		<b>N</b>	
开发支持 .....	211	内部采样开关 (R <sub>SS</sub> ) 阻抗 .....	133
看门狗定时器。请参见 WDT		<b>O</b>	
勘误表 .....	7	OPTION_REG 寄存器 .....	78
客户支持 .....	229	OSCTUNE 寄存器 .....	83
框图		OVCN 寄存器 .....	39
ADC .....	127	OVREFCON 寄存器 .....	39
ADC 传递函数 .....	134	<b>P</b>	
MCP19114/5 反激同步准谐振 .....	10	PCL .....	79
MSSP (I <sup>2</sup> C 主模式) .....	155	修改 .....	79
MSSP (I <sup>2</sup> C 从模式) .....	156	PCLATH .....	79
Timer0 .....	137	PCON 寄存器 .....	88, 91
Timer1 .....	139	PE1 寄存器 .....	48
Timer2 .....	143	PICKIT 3 在线调试器 / 编程器 .....	213
比较模式工作原理 .....	150	PIE1 寄存器 .....	96
波特率发生器 .....	190	PIE2 寄存器 .....	97
捕捉模式工作原理 .....	149	PIR1 寄存器 .....	98
带共享预分频器的看门狗定时器 .....	103	PIR2 寄存器 .....	99
脉宽调制 (PWM) .....	146	PMADRH 寄存器 .....	105, 107
模拟输入模型 .....	134	PMADRL 寄存器 .....	105, 106
建议的 MCLR 电路 .....	86	PMCON1 寄存器 .....	105, 107
单片机内核 .....	12	PMCON2 寄存器 .....	105
片上复位电路 .....	85	PMDATH 寄存器 .....	106
中断逻辑 .....	94	PMDATL 寄存器 .....	106
<b>M</b>		PORTGPA .....	111, 121
MCLR .....	86	ANSELA 寄存器 .....	112
内部 .....	86	电平变化中断 .....	111
Microchip 网站 .....	229	功能和输出优先级 .....	112
MODECON 寄存器 .....	51	寄存器 .....	111, 113
MOSFET .....	17	弱上拉 .....	111
栅极驱动器使能 .....	48	输出优先级 .....	112
驱动器		相关寄存器 .....	115
死区 .....	48	PORTGPB .....	116, 121
可编程死区 .....	45	ANSELB 寄存器 .....	116
欠压锁定选择 .....	49	电平变化中断 .....	116
MPLAB 汇编器、链接器和库管理器 .....	212	功能和输出优先级 .....	117
MPLAB ICD 3 在线调试器系统 .....	213	寄存器 .....	116, 117
MPLAB 集成开发环境软件 .....	211	弱上拉 .....	116
MPLAB PM3 器件编程器 .....	213	输出优先级 .....	117
MPLAB REAL ICE 在线仿真器系统 .....	213	相关寄存器 .....	119
MPLAB X SIM 软件模拟器 .....	213	<b>PWM</b>	34
MPLINK 目标链接器 / MPLIB 目标库管理器 .....	212	标准模式 .....	145
MSSP .....	155	独立模式 .....	145
I <sup>2</sup> C 总线术语 .....	158	固定频率 .....	20
I <sup>2</sup> C 主模式 .....	176	简化框图 .....	146
I <sup>2</sup> C 模式 .....	156	控制 .....	48
I <sup>2</sup> C 模式操作 .....	158	控制逻辑 .....	153
I <sup>2</sup> C 从模式操作 .....	160	开关频率同步模式 .....	145
波特率发生器 .....	190	时序图 .....	34
概述 .....	155	输出 .....	146
框图 (I <sup>2</sup> C 主模式) .....	155	相关寄存器 .....	147
框图 (I <sup>2</sup> C 从模式) .....	156	休眠期间的工作 .....	147
时钟延长 .....	157	要求 .....	34
仲裁 .....	157	增强型模块 .....	145
脉宽调制		占空比 .....	147
控制逻辑 .....	153	周期 .....	146
脉宽调制。见 PWM		配置字 .....	81
模拟模块使能控制 .....	49	看门狗定时器 .....	104
误差放大器禁止 .....	49	与时钟源相关的寄存器 .....	84
模拟外设控制 .....	48	<b>Q</b>	
脉宽调制控制 .....	48	启动 .....	20
次级电流正检测上拉 .....	48		
模数转换器。见 ADC			

# MCP19114/5

启动序列.....	88	情形 3.....	89
器件校准.....	59	重复启动条件期间的总线冲突（情形 1）.....	187
CALWD1（校准字 1）.....	59	重复启动条件期间的总线冲突（情形 2）.....	187
CALWD10（校准字 10）.....	66	重复启动条件.....	178
CALWD11（校准字 11）.....	67	带有时钟仲裁的波特率发生器.....	177
CALWD2（校准字 2）.....	60	第一个启动位时序.....	177
CALWD3（校准字 3）.....	61	发送和应答时的总线冲突.....	184
CALWD4（校准字 4）.....	61	复位.....	32
CALWD5（校准字 5）.....	62	看门狗定时器.....	32
CALWD6（校准字 6）.....	62	脉宽调制.....	34
CALWD7（校准字 7）.....	63	起振定时器.....	32
CALWD8（校准字 8）.....	64	启动条件期间由 SDA 仲裁引起的 BRG 复位.....	186
CALWD9（校准字 9）.....	65	启动条件期间的总线冲突（SCL=0）.....	186
器件配置.....	37, 81	启动条件期间的总线冲突（仅用于 SDA）.....	185
ID 地址单元.....	82	时钟同步.....	174
代码保护.....	82	上电延时定时器.....	32
配置字.....	81	停止条件期间的总线冲突（情形 1）.....	188
写保护.....	82	停止条件期间的总线冲突（情形 2）.....	188
器件概述.....	9	停止条件接收或发送模式.....	183
欠压复位（BOR）.....	87	应答序列.....	183
欠压锁定.....		中断唤醒.....	102
输入.....	37	时序参数符号.....	29
选择 MOSFET 驱动器.....	49	时钟切换.....	84
前沿消隐.....	43	输出.....	
驱动器控制电路.....	20	过压.....	39
<b>R</b>		OVCON 寄存器.....	39
热规范.....	28	OVREFCON 寄存器.....	39
软件模拟器（MPLAB SIM）.....	213	保护.....	39
<b>S</b>		类型.....	13
SLPCRCON 寄存器.....	44	驱动电路.....	19
SSPADD 寄存器.....	196	输出稳压参考配置.....	46
SSPADD2 寄存器.....	197	输入.....	22
SSPCON1 寄存器.....	192	过压锁定.....	37
SSPCON2 寄存器.....	194	类型.....	13
SSPCON3 寄存器.....	195	欠压锁定.....	37
SSPMSK 寄存器.....	196	数据存储.....	
SSPMSK1 寄存器.....	175	构成.....	70
SSPMSK2 寄存器.....	197	内核寄存器.....	71
SSPOV.....	181	STATUS 寄存器.....	71
SSPOV 状态标志.....	181	特殊功能寄存器.....	70, 72
SSPSTA 寄存器.....	191	通用寄存器.....	70
R/W 位.....	160	映射.....	73
STATUS 寄存器.....	71	数字电气特性.....	29
闪存程序存储器.....		<b>T</b>	
读.....	108	T1CON 寄存器.....	140
控制.....	105	T2CON 寄存器.....	144
代码保护期间的操作.....	109	Timer0.....	137
防止误写.....	109	T0CKI.....	138
寄存器.....	106	TMR0 寄存器.....	137
写保护期间的操作.....	109	8 位计数器模式.....	137
写入.....	109	8 位定时器模式.....	137
上电复位（POR）.....	86	工作.....	137
上电延时定时器（PWRT）.....	88	休眠期间.....	138
时序图.....		框图.....	137
A/D 转换.....	36	模块.....	137
I/O.....	31	功能.....	137
I <sup>2</sup> C 主模式（发送，7 或 10 位地址）.....	180	切换预分频器.....	138
I <sup>2</sup> C 主模式（接收，7 位地址）.....	182	软件可编程预分频器.....	137
INT 引脚中断.....	94	外部时钟.....	138
Timer0，外部时钟.....	33	要求.....	34
Timer1，外部时钟.....	33	时序.....	33
超时序.....		相关寄存器.....	138
情形 1.....	88	中断.....	138
情形 2.....	89	Timer1.....	139
		TMR1H 寄存器.....	139

TMR1L 寄存器 .....	139	<b>X</b>	
工作 .....	139	系统基准测试 .....	57
休眠期间 .....	140	线性稳压器 .....	19
控制寄存器 .....	140	校准字寄存器	
框图 .....	139	CALWD1 (校准字 1) .....	59
模块 .....	139	CALWD10 (校准字 10) .....	66
功能 .....	139	CALWD11 (校准字 11) .....	67
时钟源选择 .....	139	CALWD2 (校准字 2) .....	60
外部时钟时序 .....	33	CALWD3 (校准字 3) .....	61
相关寄存器 .....	141	CALWD4 (校准字 4) .....	61
预分频器 .....	140	CALWD5 (校准字 5) .....	62
中断 .....	140	CALWD6 (校准字 6) .....	62
<b>Timer2</b>		CALWD7 (校准字 7) .....	63
工作原理 .....	143	CALWD8 (校准字 8) .....	64
控制寄存器 .....	144	CALWD9 (校准字 9) .....	65
框图 .....	143	斜率补偿器 .....	44
模块 .....	143	休眠状态	
功能 .....	143	唤醒 .....	101
相关寄存器 .....	144	通过中断唤醒 .....	102
TRISGPA .....	111	<b>Y</b>	
寄存器 .....	111, 113	要求	
TRISGPB .....	116	A/D 采集 .....	133
寄存器 .....	116, 118	A/D 转换 .....	35
特殊事件触发器		I/O, 时序 .....	31
捕捉 / 比较模块 .....	150	PWM .....	34
特殊功能寄存器 (SFR) .....	70	Timer0 外部时钟 .....	34
特性 .....	1	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 .....	33
单片机特性 .....	1	外部时钟, 时序 .....	30
Timer0 模块 .....	137	引脚图	
Timer1 模块 .....	139	24 引脚 QFN (MCP19114) .....	2
通用寄存器。见寄存器		28 引脚 QFN (MCP19114) .....	4
退磁检测 .....	19	引脚排列说明	
<b>V</b>		汇总 .....	3, 5
VINCON 寄存器 .....	37	表 .....	13
VINOVLO 寄存器 .....	38	因特网地址 .....	229
VINUVLO 寄存器 .....	38	应用图	
V <sub>REF2</sub> 参考电压 .....	47	MCP19114 升压准谐振 .....	11
VREF2CON 寄存器 .....	47	MCP19114 Ćuk 同步同相输出 .....	11
VREFCON 寄存器 .....	46	<b>Z</b>	
<b>W</b>		在线串行编程 (ICSP) .....	209
WCOL .....	183	通用编程接口 .....	209
状态标志 .....	177, 179, 181, 183	振荡器 .....	83
WCOL 状态标志 .....	183	唤醒时的延时 .....	84
WDT .....	103	内部 .....	83
编程注意事项 .....	103	频率调节 .....	83
操作 .....	103	上电时的延时 .....	84
复位 .....	88	校准 .....	83
框图 .....	103	相关寄存器 .....	84
切换预分频器 .....	138	改变基本频率时的延时 .....	84
相关寄存器 .....	104	直流和交流特性 .....	53
与看门狗定时器相关的配置字 .....	104	图表 .....	53
周期 .....	103	直接寻址 .....	80
WPUGPA 寄存器 .....	114	指令格式 .....	199
WPUGPB 寄存器 .....	118	指令集 .....	199
WWW 地址 .....	229	ADDLW .....	201
WWW, 在线支持 .....	7	ADDWF .....	201
外部时钟		ANDLW .....	201
时序 .....	30	ANDWF .....	201
时序要求 .....	30	BCF .....	201
温度指示器模块 .....	125	BSF .....	201
电路工作原理 .....	125	BTFSC .....	201
温度输出 .....	125	BTFSS .....	202
文件选择寄存器。见 FSR			

# MCP19114/5

---

CALL .....	202
CLRF .....	202
CLRW .....	202
CLRWDT .....	202
COMF .....	202
DECF .....	202
DECFSZ .....	203
GOTO .....	203
INCF .....	203
INCFSZ .....	203
IORLW .....	203
IORWF .....	203
MOVF .....	204
MOVLW .....	204
MOVWF .....	204
NOP .....	204
RETFIE .....	205
RETLW .....	205
RETURN .....	205
RLF .....	205
RRF .....	206
SLEEP .....	206
SUBLW .....	206
SUBWF .....	206
SWAPF .....	206
XORLW .....	206
XORWF .....	207
汇总表 .....	200
中断	
ADC .....	130
GPA2/INT .....	93
Timer0 .....	138
Timer1 .....	140
控制寄存器 .....	95
相关寄存器 .....	100
现场保护 .....	100
与时钟源相关的配置字 .....	84
准谐振操作的退饱和和检测 .....	41
主同步串行口。请参见 MSSP	

## MICROCHIP 网站

Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的互联网浏览器即可访问。网站提供以下信息：

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及归档软件
- **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](http://www.microchip.com)。在“支持” (Support) 下，点击“变更通知客户” (Customer Change Notification) 服务后按照注册说明完成注册。

## 客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过<http://microchip.com/support>获得网上技术支持。

# MCP19114/5

---

注:

## 产品标识体系

欲订货或获取价格、交货等信息，请与我公司生产厂或各销售办事处联系。

部件编号	[X] <sup>(1)</sup>	-	X	/XX	XXX	示例:
器件	卷带式选项		温度范围	封装	定制编号	
<b>器件:</b>	MCP19114: 数字增强型 PWM 电源模拟高速控制器 MCP19115: 数字增强型 PWM 电源模拟高速控制器					a) MCP19114-E/MJ: 扩展级温度范围, 24 引脚 QFN 4x4 封装
<b>卷带式选项:</b>	空白 = 标准包装 (料管) T = 卷带式					b) MCP19114T-E/MJ: 卷带式, 扩展级温度范围, 24 引脚 QFN 4x4 封装
<b>温度范围:</b>	E = -40°C 至 +125°C (扩展级)					a) MCP19115-E/MQ: 扩展级温度范围, 28 引脚 QFN 5x5 封装
<b>封装:</b>	MJ = 24 引脚塑封四方扁平无脚封装—— 主体 4x4x0.9 mm (QFN) MQ = 28 引脚塑封四方扁平无脚封装—— 主体 5x5x0.9 mm (QFN)					b) MCP19115T-E/MQ: 卷带式, 扩展级温度范围, 28 引脚 QFN 5x5 封装
						<b>注 1:</b> 卷带式标识符仅出现在产品目录的部件编号描述中。该标识符用于订货目的，不会印刷在器件封装上。关于包装是否提供卷带式选项，请咨询当地的 Microchip 销售办事处。

# MCP19114/5

---

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为安全的产品之一。
- 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，是您自身应尽的责任。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。除非另外声明，在 Microchip 知识产权保护下，不得暗或以其他方式转让任何许可证。

#### 商标

Microchip 的名称和徽标组合、Microchip 徽标、dsPIC、FlashFlex、flexPWR、JukeBlox、KEELOQ、KEELOQ 徽标、Kleer、LANCheck、MediaLB、MOST、MOST 徽标、MPLAB、OptoLyzer、PIC、PICSTART、PIC<sup>32</sup> 徽标、RightTouch、SpyNIC、SST、SST 徽标、SuperFlash 及 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

The Embedded Control Solutions Company 和 mTouch 为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、BodyCom、chipKIT、chipKIT 徽标、CodeGuard、dsPICDEM、dsPICDEM.net、ECAN、In-Circuit Serial Programming、ICSP、Inter-Chip Connectivity、KleerNet、KleerNet 徽标、MiWi、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、RightTouch 徽标、REAL ICE、SQI、Serial Quad I/O、Total Endurance、TSHARC、USBCheck、VariSense、ViewSpan、WiperLock、Wireless DNA 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

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

Silicon Storage Technology 为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

GestIC 为 Microchip Technology Inc. 的子公司 Microchip Technology Germany II GmbH & Co. & KG 在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2015, Microchip Technology Inc. 版权所有。

ISBN: 978-1-63277-441-5

**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<sup>®</sup> MCU 与 dsPIC<sup>®</sup> DSC、KEELOQ<sup>®</sup> 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和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。

## 全球销售及及服务网点

### 美洲

公司总部 **Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 1-480-792-7200  
Fax: 1-480-792-7277

技术支持:

<http://www.microchip.com/support>

网址: [www.microchip.com](http://www.microchip.com)

#### 亚特兰大 Atlanta

Duluth, GA  
Tel: 1-678-957-9614  
Fax: 1-678-957-1455

#### 奥斯汀 Austin, TX

Tel: 1-512-257-3370

#### 波士顿 Boston

Westborough, MA  
Tel: 1-774-760-0087  
Fax: 1-774-760-0088

#### 芝加哥 Chicago

Itasca, IL  
Tel: 1-630-285-0071  
Fax: 1-630-285-0075

#### 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

Novi, MI  
Tel: 1-248-848-4000

#### 休斯敦 Houston, TX

Tel: 1-281-894-5983

#### 印第安纳波利斯

Indianapolis  
Noblesville, IN  
Tel: 1-317-773-8323  
Fax: 1-317-773-5453

#### 洛杉矶 Los Angeles

Mission Viejo, CA  
Tel: 1-949-462-9523  
Fax: 1-949-462-9608

#### 纽约 New York, NY

Tel: 1-631-435-6000

#### 圣何塞 San Jose, CA

Tel: 1-408-735-9110

#### 加拿大多伦多 Toronto

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: 852-2943-5100  
Fax: 852-2401-3431

#### 中国 - 北京

Tel: 86-10-8569-7000  
Fax: 86-10-8528-2104

#### 中国 - 成都

Tel: 86-28-8665-5511  
Fax: 86-28-8665-7889

#### 中国 - 重庆

Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

#### 中国 - 东莞

Tel: 86-769-8702-9880

#### 中国 - 杭州

Tel: 86-571-8792-8115  
Fax: 86-571-8792-8116

#### 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

#### 台湾地区 - 台北

Tel: 886-2-2508-8600  
Fax: 886-2-2508-0102

#### 台湾地区 - 新竹

Tel: 886-3-5778-3666  
Fax: 886-3-5770-955

#### 澳大利亚 Australia - Sydney

Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

#### 印度 India - Bangalore

Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

#### 印度 India - New Delhi

Tel: 91-11-4160-8631  
Fax: 91-11-4160-8632

#### 印度 India - Pune

Tel: 91-20-3019-1500

#### 日本 Japan - Osaka

Tel: 81-6-6152-7160  
Fax: 81-6-6152-9310

#### 日本 Japan - Tokyo

Tel: 81-3-6880-3770  
Fax: 81-3-6880-3771

#### 韩国 Korea - Daegu

Tel: 82-53-744-4301  
Fax: 82-53-744-4302

#### 韩国 Korea - Seoul

Tel: 82-2-554-7200  
Fax: 82-2-558-5932 或  
82-2-558-5934

#### 马来西亚 Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857  
Fax: 60-3-6201-9859

#### 马来西亚 Malaysia - Penang

Tel: 60-4-227-8870  
Fax: 60-4-227-4068

#### 菲律宾 Philippines - Manila

Tel: 63-2-634-9065  
Fax: 63-2-634-9069

#### 新加坡 Singapore

Tel: 65-6334-8870  
Fax: 65-6334-8850

#### 泰国 Thailand - Bangkok

Tel: 66-2-694-1351  
Fax: 66-2-694-1350

### 欧洲

#### 奥地利 Austria - Wels

Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393

#### 丹麦 Denmark - Copenhagen

Tel: 45-4450-2828  
Fax: 45-4485-2829

#### 法国 France - Paris

Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

#### 德国 Germany - Dusseldorf

Tel: 49-2129-3766400

#### 德国 Germany - Munich

Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

#### 德国 Germany - Pforzheim

Tel: 49-7231-424750

#### 意大利 Italy - Milan

Tel: 39-0331-742611  
Fax: 39-0331-466781

#### 意大利 Italy - Venice

Tel: 39-049-7625286

#### 荷兰 Netherlands - Drunen

Tel: 31-416-690399  
Fax: 31-416-690340

#### 波兰 Poland - Warsaw

Tel: 48-22-3325737

#### 西班牙 Spain - Madrid

Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91

#### 瑞典 Sweden - Stockholm

Tel: 46-8-5090-4654

#### 英国 UK - Wokingham

Tel: 44-118-921-5800  
Fax: 44-118-921-5820