

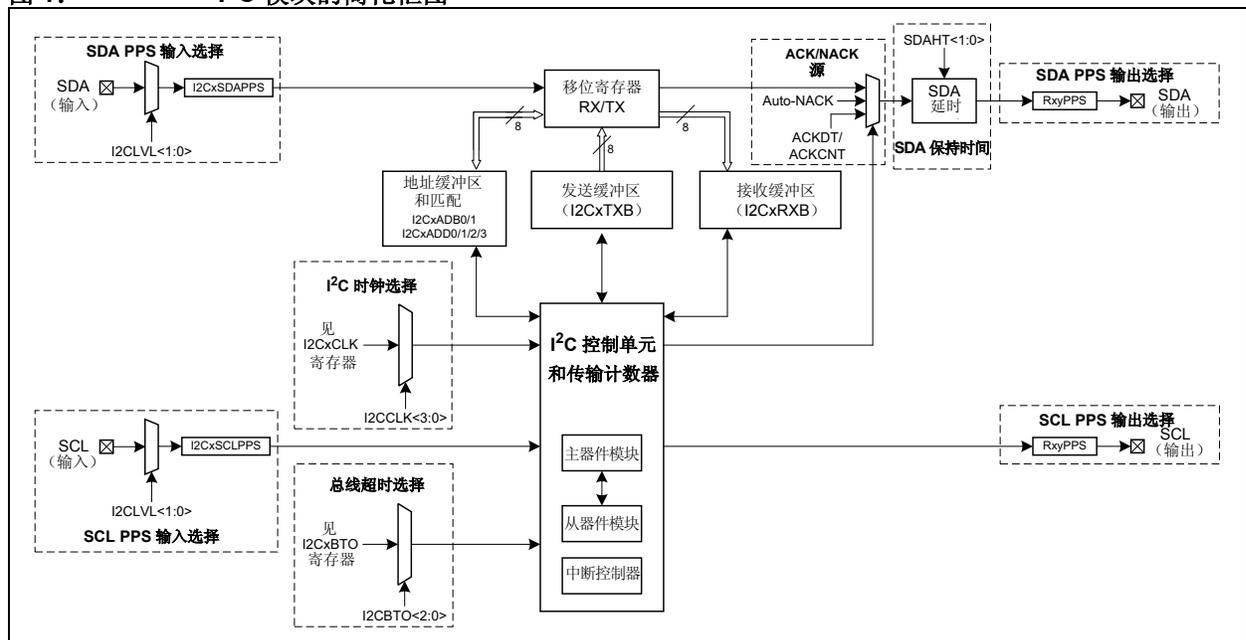
8 位 PIC[®] 单片机上具有硬件协议加速功能的 I²C 通信

作者: Mary Iva Rosario Salimbao
Microchip Technology Inc.

简介

本技术简介讨论了 I²C 模块及其特性和基本功能。图 1 给出了 I²C 模块的简化框图。

图 1: I²C 模块的简化框图

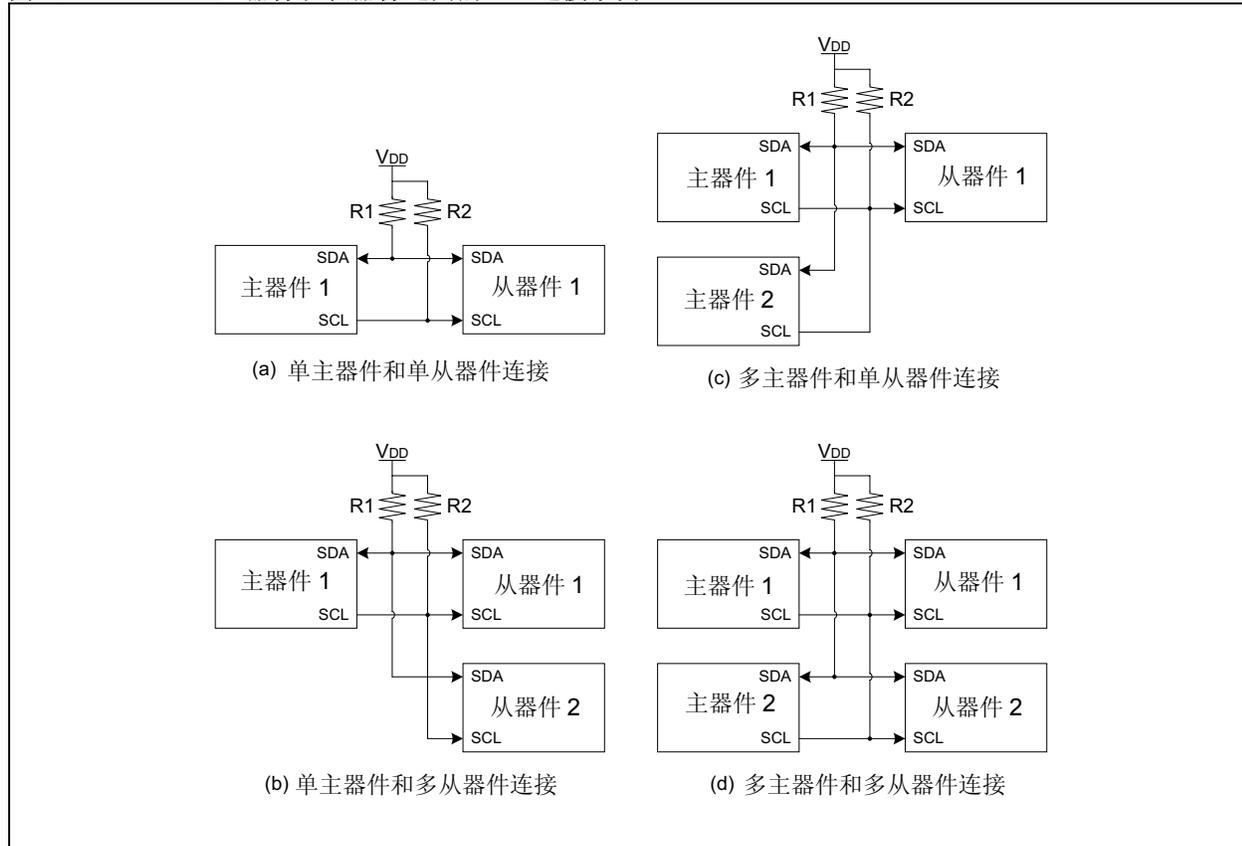


在图 1 中，控制单元包含用于主/从工作模式的主器件模块和从器件模块以及用于监视模块状态的中断控制器。传输计数器通过自动计数功能自动传输数据。缓冲区保存数据或地址的一个字节，而另一个字节由移位寄存器在 SDA 引脚上移入或移出。ACK/NACK 源将第 9 位添加到传输来表示应答。这 9 位在由 SDA 延时设置的时间间隔内移出。时钟和总线超时选择分别控制 SCL 和超时的时序。SDA 和 SCL 引脚通过外设引脚选择 (Peripheral Pin Select, PPS) 配置。

I²C 协议概述

I²C 模块遵循 Phillips® I²C 规范。模块在 PIC® 单片机和其他支持 I²C 的器件之间提供一个双向的主器件 / 从器件同步接口。这些器件通过两线式串行总线连接，从而允许多个主器件与多个从器件通信。图 2 显示了主器件和从器件之间的不同连接类型。

图 2: 主器件和从器件之间的 I²C 连接示例

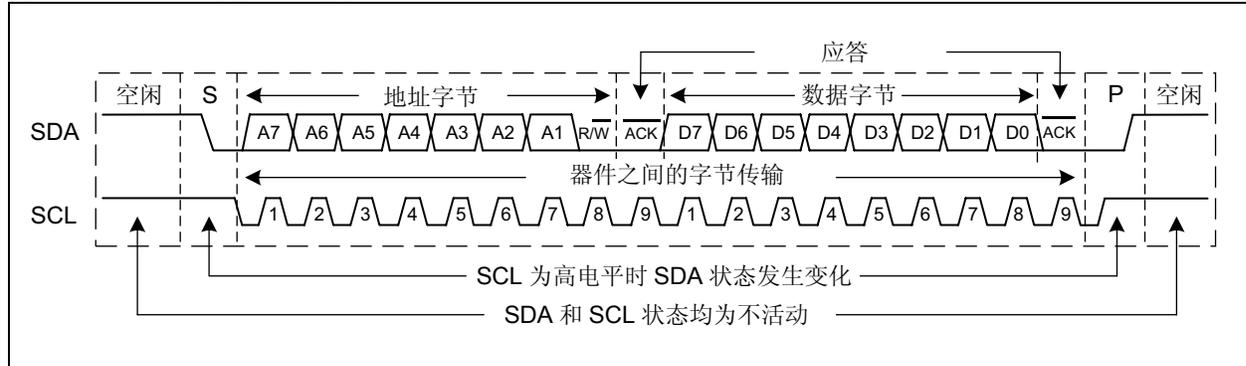


总线

I²C 总线由串行时钟 (SCL) 线和串行数据 (SDA) 线组成。时钟信号由主器件产生, 通过 SCL 线发送来控制数据传输。SDA 线传输主器件 / 从器件发送 / 接收的数据。每个信号线都需要一个上拉电阻实现漏极开路连接。当总线空闲时, 这两个信号线最初均悬浮为高电平, 并且必须按特定顺序驱动为低电平来启动 I²C 器件之间的通信。SDA 线的跳变总是在 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条件和停止条件。

图 3 显示了典型 I²C 报文中 SDA 和 SCL 信号的状态。该图还显示了 I²C 报文的基本组成部分: 启动条件和停止条件、地址和数据字节以及应答。

图 3: 典型 I²C 报文

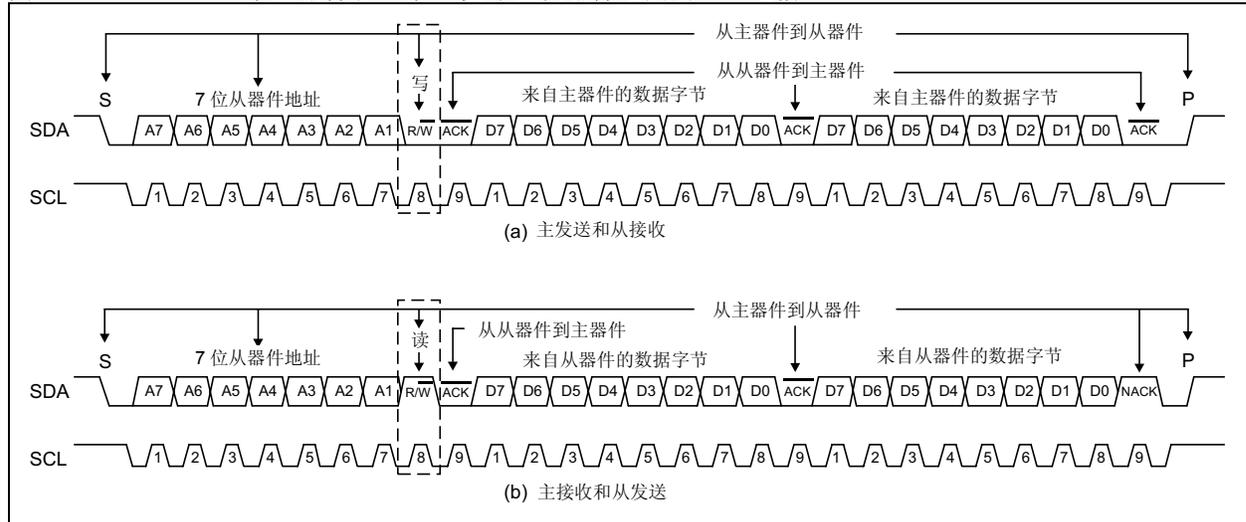


I²C 操作

支持 I²C 器件可工作在主发送 / 接收或从发送 / 接收模式下。启动条件和停止条件之间的传输以 9 位分段发送, 由一个地址或数据字节后跟 ACK/NACK 条件组成。

图 4 显示了在发送和接收操作下两个数据字节的简单 I²C 传输。(a) 和 (b) 之间的主要区别在于 R/W 位的状态。该位确定传输的寻址阶段的操作。

图 4: 一个主器件和一个 7 位寻址从器件之间的 I²C 通信



I²C 模块特性

下面列出了 I²C 模块提供的特性。

- 主模式
- 从模式（字节无应答）
- 多主器件模式
- I²C 主器件和从器件硬件支持
 - 专用地址、接收和发送缓冲区
 - 最多 4 个从器件地址匹配⁽¹⁾
 - 广播呼叫地址匹配
 - 7 位和 10 位寻址模式支持掩码
 - 启动、重复启动、停止、地址、读、写、ACK 和字节计数中断
 - 总线超时、总线冲突和 NACK 检测错误中断
- 基于硬件的时钟延长用于：
 - 接收缓冲区满时
 - 发送缓冲区空时
 - 地址、写和 ACK 之后
- 自动字节计数
- 可选择时钟
- 总线冲突检测
- 带可编程源的总线超时检测
- SDA 保持时间选择
- 可编程总线空闲时间选择
- I²C、SMBus 和 1.8V 输入电平选择
- 支持 DMA⁽²⁾ 和 PMD

注 1: 仅在 7 位寻址模式下支持四个从器件地址匹配。10 位寻址的地址匹配最多只能有两个地址。

2: 直接存储器访问（Direct Memory Access, DMA）模块是否可用取决于具体器件。请参见数据手册检查器件是否提供这些特性。

为了与 SMBus 和 PMBus™ 兼容，总线超时允许模块按照在 I2Cx^BT0 寄存器中选定的超时复位。当模块未在使用时，还可通过 PMD 禁止。主器件模块和从器件模块中的数据自动处理降低了代码开销并提高了模块的整体性能。

多主器件模式

对于多主器件支持特性，总线空闲（BFRE）状态位允许主器件在将启动条件置为有效之前确定总线是否空闲。主器件先等待 8-64 个 I²C 时钟脉冲（BFRET<1:0>）的总线空闲时间，然后再将 BFRE 位置 1。这可防止主器件试图在另一个主器件正在使用总线时控制总线。如果两个主器件同时将启动条件置为有效，将在寻址阶段进行总线仲裁。

I²C 模块操作

本节按照 I²C 模块特性讨论其工作方式。关于与该模块相关的寄存器和位的汇总，请参见表 A-1。关于这些寄存器的更详细讨论，请参见器件数据手册。

设置 I²C 总线

SCL 和 SDA 均是数字的、双向漏极开路线路。分配给这些信号的 I/O 引脚必须分别通过 ANSELx 和 ODCONx 寄存器以及 PPS 进行配置。PPS 使这些引脚用作外设输入（I2CxSCLPPS 和 I2CxSDAPPS）和输出（RxyPPS）。输入阈值、压摆率和内部上拉设置在 RxyI2C 控制寄存器中设置。SCL 时钟通过 I²C 时钟选择配置，而 SDA 保持时间可通过 I2CxCON2 寄存器的 SDAHT 位设置为 30/100/300 ns。例 1 给出了一个 K42 器件上 SDA 和 SCL 引脚的示例配置。

例 1: 配置 SCL 和 SDA 引脚

```
// 将引脚配置为数字引脚
ANSELbits.ANSEL3 = 0;
ANSELbits.ANSEL4 = 0;

// PPS 解锁序列
PPSLOCK = 0x55;
PPSLOCK = 0xAA;
PPSLOCKbits.PPSLOCKED = 0x00;

// 将 RC4 设置为 SDA
RC4PPS = 0x22;
I2C1SDAPPS = 0x14;

// 将 RC3 设置为 SCL
RC3PPS = 0x21;
I2C1SCLPPS = 0x13;

// PPS 锁定序列
PPSLOCK = 0x55;
PPSLOCK = 0xAA;
PPSLOCKbits.PPSLOCKED = 0x01;

// 将引脚配置为漏极开路
ODCONbits.ODCC3 = 1;
ODCONbits.ODCC4 = 1;

// 设置 I2C 电平
RC3I2Cbits.TH = 1;
RC4I2Cbits.TH = 1;

// 将引脚配置为输出引脚
TRISbits.TRISC3 = 0;
TRISbits.TRISC4 = 0;
```

时钟选择

时钟源 (I2CxCLK) 寄存器为主器件配置 I²C 时钟。可通过该寄存器选择内部 CPU 振荡器和其他外设提供的时钟源。快速模式使能 (FMEN) 位在由硬件驱动之前通过控制 SCL 引脚的采样来提供额外配置。使能或禁止快速模式分别将 SCL 频率设置为 FCLK/4 或 FCLK/5。

模式选择

I²C 模块具有 I²C 模式选择, 通过寻址类型定义。这些模式通过 I2CxCON0 控制寄存器中的模式选择位定义。选项包括 7 位和 10 位主模式、7 位和 10 位从模式以及 7 位多主器件模式。

条件

I²C 模块观察一些条件, 这些条件指示总线上发生的不同事件。请参见图 4 和图 5 了解 SDA 线上这些条件的示例。

1. 启动条件 (S):

主器件产生启动条件以指示开始传输, 总线的状态从空闲变为活动。启动条件定义为在 SCL 线为高电平时, SDA 线从高电平变为低电平状态。

2. 停止条件 (P):

当传输结束时, 主器件还会产生停止条件以释放对总线的控制并将总线返回为空闲状态。当 SCL 线为高电平时, SDA 线从低电平变为高电平状态。

3. 重复启动 (RS):

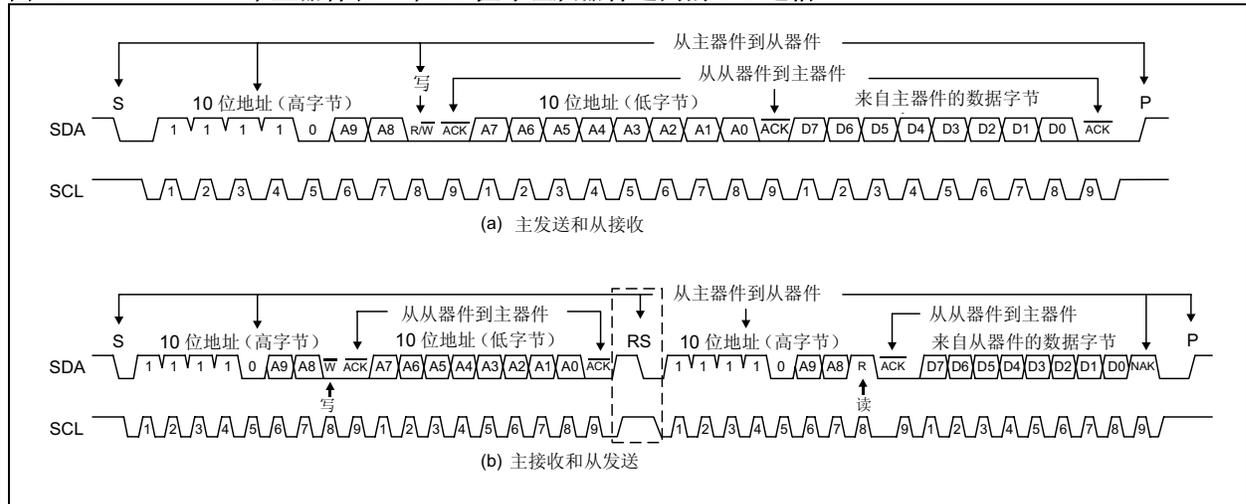
如果主器件在传输后仍然需要控制总线, 它发出重复启动条件。这可防止其他主器件在传输之间控制总线。

4. ACK/NACK 条件:

ACK 条件对从器件控制或数据传输成功作出应答。该条件由传输的接收器在 SCL 的第 9 个时钟脉冲将 SDA 线拉为低电平发送。否则, 接收器将使 SDA 线悬浮为高电平, 指示 NACK 条件。主器件还会发送 NACK 以终止从从器件接收数据。

启动、停止和重复启动条件始终由主器件产生。主器件在总线仍处于空闲状态时通过将启动位置 1 或者写入发送缓冲区发出启动条件。发生启动条件后, 主器件模块现在处于活动状态。将重复启动使能 (RSEN) 位置 1 使能重复启动条件。主器件模块处于活动状态且启动 (S) 位置 1 后, 如果字节计数为零或者未收到 ACK 信号, 主器件将发出重复启动条件。接收或发出 NACK 或字节计数达到零时, 发生停止条件。在局部 I²C 外设中断标志寄存器中为这些条件均分配了中断标志。

图 5: 一个主器件和一个 10 位寻址从器件之间的 I²C 通信



应答序列

ACK/NACK 序列结果存储在 ACK 状态 (ACKSTAT) 位中。第 9 位始终保存 ACK/NACK 信息。ACK 条件将状态位清零，而 NACK 将该位设置为高电平。在接收操作中，数据字节计数寄存器 (I2CXCNT)、应答数据位 (ACKDT) 和计数结束位 (ACKCNT) 确定该序列的结果。如果计数仍未达到 0，接收器发出 ACKDT 位，否则，发送 ACKCNT。

在从模式下，如果保持使能位置 1 且发生地址匹配，则启动时钟延长。这允许用户设置发回发送器的 ACK 值。其他条件也会导致 NACK，例如发生发送和接收缓冲区错误。主器件硬件在检测到 NACK 时自动发送停止条件。

寻址

从器件可具有 7 位或 10 位地址。在该模块中，可以有四个 7 位地址和两个 10 位地址。启动条件之后，主器件始终首先发出地址字节。该地址字节保存主器件尝试与其通信的从器件的地址以及读 / 写 (R/W) 位。

1. 7 位寻址：

主器件发送单个地址字节，其中从器件地址占用 bit 7 到 1。地址字节的 Bit 0 保存 R/W 位。(见图 4。)

2. 10 位寻址：

第一个字节由组合 11110xx 和 $\overline{R/W}$ 位组成。xxx 是 10 位地址的两个最高有效位 (MSb)。第二个字节包含地址的其余 8 位。(见图 5。)

R/W 位确定数据的方向。如果主器件希望发送数据，则 SDA 线悬浮为高电平。对于数据接收，SDA 线拉为低电平。如果总线上存在从器件，则以 ACK 响应。然而，如果主器件未能连接到从器件，则返回 NACK。

10 位寻址模式的主接收和从发送操作需要重复启动条件以将数据从从器件移出。主器件首先发出两个地址字节进行写操作。如果从器件应答两个地址字节，主器件软件将启动位置 1 发出重复启动条件。主器件将重新发送高地址字节进行读操作，等待 ACK 条件，然后读取从器件的数据。

模块提供两个地址数据缓冲区 (I2CXADBx) 和四个地址 (I2CXADRx) 寄存器。地址数据缓冲区充当发送或接收缓冲区 (见表 1)。在主模式下，缓冲区包含要移出总线的地址字节。在从模式下，缓冲区保存接收到的地址字节并将地址与地址寄存器匹配。地址寄存器包含用于从模式下匹配和掩码的从模式地址。需要多个地址寄存器来支持 SMBus 和 PMBus 通信。

表 1：不同 I²C 模式下地址数据缓冲区的方向

模式	I2CXADB0	I2CXADB1
从模式 (7 位，带或不带掩码)	RX	—
从模式 (10 位，带或不带掩码)	RX	RX
主模式 (7 位)	—	TX
主模式 (10 位)	TX	TX
多主器件模式 (7 位)	RX	TX

支持广播呼叫地址

I²C 模块支持广播呼叫寻址，这样主器件可寻址所有从器件并接收 ACK。广播呼叫地址定义为 0x00 且需要软件使能广播呼叫地址使能位。无需定义任何地址寄存器来保存广播呼叫地址。

数据传输

通信继续发送数据字节。如果数据字节正常发送或接收，则返回 ACK 位。I²C 总线在检测到停止条件之后被释放，并且返回其空闲状态直至出现下一个启动条件。

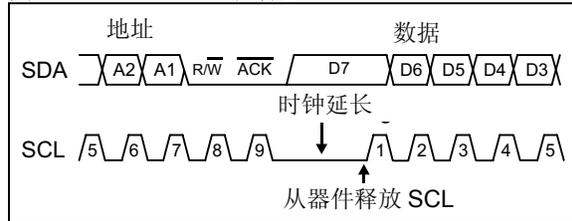
专用缓冲区

I²C 模块有两个数据缓冲区，一个用于发送 (TXB)，另一个用于接收 (RXB)。这些缓冲区可通过软件或 DMA 访问。I²C 模块需要用户通过这些缓冲区的状态 (TXBE/RXBF) 或中断 (TXIF/RXIF) 标志监视其状态，从而确保正确的读 / 写操作。当缓冲区满时，用户不得将新字节装入到 TXB。如果 RXB 为空或其保存旧数据，则不能读取其中的数据。不正确的读 / 写操作会导致设置错误的条件标志。

时钟延长

如果从器件无法与主器件的数据速率保持同步，可通过使用时钟延长来延迟传输。从器件可在传输的任意时刻将 SCL 线保持为低电平，直到它准备好继续通信为止（见图 6）。当主器件检测到 SCL 线保持为低电平时，它将等待直至从器件释放线路以发送或接收剩余的数据。

图 6: 时钟延长



当发送缓冲区空 (TXBE) 位或接收缓冲区满 (RXBF) 位置 1 且计数寄存器不为零时，发生时钟延长。将地址匹配 (ADRIE)、数据写入 (WRIE) 和应答状态 (ACKTIE) 的专用中断和保持位置 1 也会为这些操作使用时钟延长。时钟延长可通过将时钟延长禁止 (CSTRDIS) 位置 1 来禁止。

自动字节计数

字节计数 (I2CxCNT) 寄存器是硬件计数器，用于控制 I²C 总线的数据传输的长度。该寄存器保存要发送或接收的数据字节数，不包括地址字节。在每个发送字节的第 9 个 SCL 下降沿，该寄存器针对总线上每一组字节 +ACK 自动递减。当寄存器递减到 0 时，计数中断标志 (CNTIF) 置 1 且主器件硬件将发出停止条件。如果字节数超过 256，可中途重写寄存器。

中断、状态和错误检测

I²C 模块有多种方式来监视其工作状态。用户可观察状态寄存器 (I2CxSTAT0/1) 或允许相关中断源。大多数状态位都是只读的，并显示 I²C 总线和模块 (活动主器件或活动从器件) 的当前状态。

局部 I²C 中断包含 I²C 总线条件、时钟延长选项和字节计数寄存器的 (I2CxPIR) 标志。将任一标志置 1 会将主 I²C 中断标志 (I2CxIF) 置 1。

I²C 错误 (I2CxERR) 寄存器包含由于通信错误而设置的中断。这些错误包括总线超时、总线冲突和 NACK 检测。检测到任一错误时也会将主 I²C 错误中断标志 (I2CxEIF) 置 1。

发送和接收缓冲区也有其自己的中断允许和标志位。如果允许缓冲区空 (TXIF) 和满 (RXIF) 中断，在开始下一次传输前需要先处理这些中断。对于向量中断控制器，也可以为这些中断源设置中断优先级 (Interrupt Priority Level, IPL)。请参见器件数据手册了解完整中断列表。

I²C 模块应用示例

主模式

例 2 给出了 I²C 模块作为 I²C 主器件工作的软件配置。使能 500 kHz 时钟源和快速模式，模块以 125 kHz 的 SCL 频率进行通信。例 3 和例 4 显示了对 I²C 从器件 EEPROM 进行的简单主器件写操作和读操作。地址缓冲区 I2C1ADB1 保存 7 位从器件地址 EE_SLAVE_ADDRESS。计数寄存器 I2C1CNT 保存要发送或接收的字节总数。

例 2: I²C 主模式初始化

```
// 7 位主模式 (MODE = 4)
I2C1CON0 = 0x04;

// I2C 时钟 = MFINTOSC (500 kHz)
I2C1CLK = 0x03;

// 对每个有效字节发送 ACK (ACKDT = 0)
// 发送 NACK 以结束读操作 (ACKCNT = 1)
I2C1CON1 = 0x80;

// 禁止自动计数 (ACNT = 0)
// 禁止广播呼叫 (GCEN = 0)
// 使能快速模式
// (FME = 1; SCL = I2CCLK/4)
// 使用 ADB1 地址缓冲区 (ADB = 0)
// SDA 保持时间为 300 ns (SDAHT = 0)
// 总线空闲时间为 16 个 I2C 时钟脉冲
// (BFRET = 1)
I2C1CON2 = 0x21;

// 使能 I2C 模块
I2C1CON0bits.EN = 1;
```

由于此时 I²C 总线仍处于空闲状态，因此写发送缓冲区将启动传输。模块将启动条件置为有效并在 SDA 线上移出从器件地址和写位。接收到 ACK 之后，模块移出数据指针所指的第一个数据字节（EEPROM 存储地址字节）。将最后一个字节加载到缓冲区后，模块仍然需要一些时间才能将数据移出并发送停止条件。

例 3: I²C 主器件写入 EEPROM

```
// 写操作 (W/R 位 = 0)
I2C1ADB1 = (EE_SLAVE_ADDRESS<<1)|0;
totalByteLength = EE_ADDR_BYTE_LENGTH
                  +DATA_BYTE_LENGTH;
I2C1CNT = totalByteLength;

for(i = 0; i < totalByteLength; i++){
    // 写入 TXB 开始通信
    I2C1TXB = *dataBlock;
    dataBlock++;

    // 等待直至 TXB 为空
    while(!I2C1STAT1bits.TXBE);
}
```

典型的 EEPROM 读取操作由主器件对 EEPROM 存储地址的写操作和对数据字节的读操作组成。传输存储地址字节后，重写 I2C1ADB1 和 I2C1CNT 进行读操作。模块通过将启动位置 1 发出另一个启动条件并移出 EE_SLAVE_ADDRESS 和读位。从器件返回 ACK 后，主器件移入第一个数据字节并将其存储在接收缓冲区中。主器件必须通过软件读取接收缓冲区以清零 RXBF 进行下一个读取操作。读取最后一个字节后，主器件将按照例 2 中的配置发送 NACK。

例 4: I²C 主器件读取 EEPROM

```
// 读操作 (W/R 位 = 1)
I2C1ADB1 = (EE_SLAVE_ADDRESS<<1)|1;
I2C1CNT = READ_BYTE_LENGTH;
I2C1CON0bits.S = 1;

for(i = 0; i < READ_BYTE_LENGTH; i++){
    while (!I2C1STAT1bits.RXBF);
    *readBlock = I2C1RXB;
    readBlock++;
}
```

从模式

例 5 显示了 I²C 模块作为具有最多 4 个 7 位地址的从器件工作的配置。地址寄存器可能保存相似或完全不同的地址。如果主器件请求 4 个地址中的任意一个，从器件模块将响应 ACK。

由于 I²C 从器件等待 I²C 主器件启动通信，因此从器件设置为使用中断和状态位来检测是发送还是接收数据。中断还允许从器件通过软件控制其下一个操作。在使能模块之前允许这些中断以确保在中断程序中处理 I²C 主器件的任何即时通信。

例 5: I²C 从模式初始化

```
// 7 位从模式 (MODE = 0)
I2C1CON0 = 0x00;

// 从器件地址匹配
I2C1ADR0 = 0x98;
I2C1ADR1 = 0x98;
I2C1ADR2 = 0x98;
I2C1ADR3 = 0x98;

// 对每个有效字节发送 ACK (ACKDT = 0)
// 读操作结束时发送 ACK (ACKCNT = 0)
// 使能时钟延长 (CSTRDIS = 0)
I2C1CON1 = 0x00;

// 禁止自动计数 (ACNT = 0)
// 禁止广播呼叫 (GCEN = 0)
// 使能快速模式 (FME = 1)
// 使用 ADB0 地址缓冲区 (ADB = 0)
// SDA 保持时间为 30 ns (SDAHT = 2)
// 总线空闲时间为 8 个 I2C 时钟脉冲
// (BFRET = 1)
I2C1CON2 = 0x28;

// 清零所有 I2C 标志
PIR3bits.I2C1F = 0;
I2C1PIR = 0x00;

// 允许全局和 I2C 中断
INTCON0bits.IPEN = 1;
INTCON0bits.GIEH = 1;
PIE3bits.I2C1IE = 1;
PIE3bits.I2C1IP = 1;

// 允许 ACK 序列局部中断
I2C1PIE = 0x40;

// 使能 I2C 模块
I2C1CON0bits.EN = 1;
```

I²C 从器件模块中断服务程序有多个中断选项。在例 5 中，使能 ACK 中断保持。如果触发中断事件，允许从器件软件访问数据缓冲区，同时进行时钟延长（见例 6）。同样，也可在地址保持、数据保持、缓冲区满和空中断中执行这些操作。对于保持中断，还可修改 ACKDT 位，并清零 CSTR 释放时钟。对于缓冲区中断，读或写缓冲区将释放 SCL 并恢复通信。

例 6: I²C 从器件中断服务

```
void interrupt I2CSLAVE_ISR (void){
    if (PIR3bits.I2C1IF){
        // 清零 I2C 中断标志
        PIR3bits.I2C1IF = 0;

        // 检测 ACK 序列中断
        if (I2C1PIRbits.ACKTIF){
            // 清零中断标志
            I2C1PIRbits.ACKTIF = 0;

            // 对于从器件读操作 / 主器件写操作
            if (!I2C1STAT0bits.R){
                // 接收数据字节
                if (I2C1STAT0bits.D){
                    // 读取 RXB
                    *readBlock = I2C1RXB;
                    readBlock++;
                }
            }
            // 对于从器件写操作 / 主器件读操作
            else {
                // 写入 TXB
                I2C1TXB = *dataBlock;
                dataBlock++;
            }
            // 释放 SCL
            I2C1CON0bits.CSTR = 0;
        }
    }else {
        // 其他中断 ...
    }
}
```

结论

I²C 模块是双线同步和双向接口，支持多个主器件和从器件之间的通信。硬件协议加速功能通过额外的硬件支持特性使模块能够以最小的软件开销运行。

额外的硬件支持允许模块对每个数据字节仅使用一个中断。通过添加 DMA，模块对每个消息仅需要处理单个 CPU 中断。

附录 A: 与 I²C 模块相关的寄存器和位

表 A-1: I²C 模块寄存器和位汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
I2CxBTO	—	—	—	—	—	BTO<2:0>		
I2CxCLK	—	—	—	—	—	CLK<2:0>		
I2CxPIE	CNTIE	ACKTIE	—	WRIE	ADRIE	PCIE	RSCIE	SCIE
I2CxPIR	CNTIF	ACKTIF	—	WRIF	ADRIF	PCIF	RSCIF	SCIF
I2CxERR	—	BTOIF	BCLIF	NACKIF	—	BTOIE	BCLIE	NACKIE
I2CxSTAT0	BFRE	SMA	MMA	R	D	—	—	—
I2CxSTAT1	TXWE	—	TXBE	—	RXRE	CLRBF	—	RXBF
I2CxCON0	EN	RSEN	S	CSTR	MDR	MODE<2:0>		
I2CxCON1	ACKCNT	ACKDT	ACKSTAT	ACKT	—	RXOV	TXU	CSD
I2CxCON2	ACNT	GCEN	FME	ADB	SDAHT<3:2>		BFRET<1:0>	
I2CxADR0	ADR<7:0>							
I2CxADR1	ADR<7:1>							—
I2CxADR2	ADR<7:0>							
I2CxADR3	ADR<7:1>							—
I2CxADB0	ADB<7:0>							
I2CxADB1	ADB<7:0>							
I2xCNT	CNT<7:0>							
I2cRXB	RXB<7:0>							
I2cTXB	TXB<7:0>							
I2CxSDAPPS	—	—	—	I2CxSDAPPS<4:0>				
I2CxSCLPPS	—	—	—	I2CxSCLPPS<4:0>				
RxyI2C	—	SLEW	PU<1:0>		—	—	TH<1:0>	

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字千年版权法案 (Digital Millennium Copyright Act)》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适用性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。除非另外声明，在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2009 认证。Microchip 的 PIC® MCU 与 dsPIC® DSC、KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器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。

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949 ==

商标

Microchip 的名称和徽标组合、Microchip 徽标、AnyRate、AVR 徽标、AVR Freaks、BeaconThings、BitCloud、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KEELOQ、KEELOQ 徽标、Kleer、LANCheck、LINK MD、maXStylus、maXTouch、MediaLB、megaAVR、MOST、MOST 徽标、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32 徽标、Prochip Designer、QTouch、RightTouch、SAM-BA、SpyNIC、SST、SST 徽标、SuperFlash、tinyAVR、UNI/O 及 XMEGA 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTouch、Precision Edge 和 Quiet-Wire 均为 Microchip Technology Inc. 在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BodyCom、chipKIT、chipKIT 徽标、CodeGuard、CryptoAuthentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherGREEN、In-Circuit Serial Programming、ICSP、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNet 徽标、Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PureSilicon、QMatrix、RightTouch 徽标、REAL ICE、Ripple Blocker、SAM-ICE、Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、ViewSpan、WiperLock、Wireless DNA 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 为 Microchip Technology Inc. 在美国的服务标记。

Silicon Storage Technology 为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

GestIC 为 Microchip Technology Inc. 的子公司 Microchip Technology Germany II GmbH & Co. & KG 在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2017, Microchip Technology Inc. 版权所有。

ISBN: 978-1-5224-2303-4

全球销售及服务中心

美洲

公司总部 **Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277

技术支持:
<http://www.microchip.com/support>

网址: www.microchip.com

亚特兰大 **Atlanta** Duluth, GA

Tel: 1-678-957-9614
Fax: 1-678-957-1455

奥斯汀 **Austin, TX** Tel: 1-512-257-3370

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Novi, MI
Tel: 1-248-848-4000

休斯敦 Houston, TX
Tel: 1-281-894-5983

印第安纳波利斯 Indianapolis
Noblesville, IN
Tel: 1-317-773-8323
Fax: 1-317-773-5453
Tel: 1-317-536-2380

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608
Tel: 1-951-273-7800

罗利 Raleigh, NC
Tel: 1-919-844-7510

纽约 New York, NY
Tel: 1-631-435-6000

圣何塞 San Jose, CA
Tel: 1-408-735-9110
Tel: 1-408-436-4270

加拿大多伦多 Toronto
Tel: 1-905-695-1980
Fax: 1-905-695-2078

亚太地区

中国 - 北京
Tel: 86-10-8569-7000

中国 - 成都
Tel: 86-28-8665-5511

中国 - 重庆
Tel: 86-23-8980-9588

中国 - 东莞
Tel: 86-769-8702-9880

中国 - 广州
Tel: 86-20-8755-8029

中国 - 杭州
Tel: 86-571-8792-8115

中国 - 南京
Tel: 86-25-8473-2460

中国 - 青岛
Tel: 86-532-8502-7355

中国 - 上海
Tel: 86-21-3326-8000

中国 - 沈阳
Tel: 86-24-2334-2829

中国 - 深圳
Tel: 86-755-8864-2200

中国 - 苏州
Tel: 86-186-6233-1526

中国 - 武汉
Tel: 86-27-5980-5300

中国 - 西安
Tel: 86-29-8833-7252

中国 - 厦门
Tel: 86-592-238-8138

中国 - 香港特别行政区
Tel: 852-2943-5100

中国 - 珠海
Tel: 86-756-321-0040

台湾地区 - 高雄
Tel: 886-7-213-7830

台湾地区 - 台北
Tel: 886-2-2508-8600

台湾地区 - 新竹
Tel: 886-3-577-8366

亚太地区

澳大利亚 **Australia - Sydney**
Tel: 61-2-9868-6733

印度 **India - Bangalore**
Tel: 91-80-3090-4444

印度 **India - New Delhi**
Tel: 91-11-4160-8631

印度 **India - Pune**
Tel: 91-20-4121-0141

日本 **Japan - Osaka**
Tel: 81-6-6152-7160

日本 **Japan - Tokyo**
Tel: 81-3-6880-3770

韩国 **Korea - Daegu**
Tel: 82-53-744-4301

韩国 **Korea - Seoul**
Tel: 82-2-554-7200

马来西亚
Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

马来西亚 **Malaysia - Penang**
Tel: 60-4-227-8870

菲律宾 **Philippines - Manila**
Tel: 63-2-634-9065

新加坡 **Singapore**
Tel: 65-6334-8870

泰国 **Thailand - Bangkok**
Tel: 66-2-694-1351

越南 **Vietnam - Ho Chi Minh**
Tel: 84-28-5448-2100

欧洲

奥地利 **Austria - Wels**
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦
Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

芬兰 **Finland - Espoo**
Tel: 358-9-4520-820

法国 **France - Paris**
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 **Germany - Garching**
Tel: 49-8931-9700

德国 **Germany - Haan**
Tel: 49-2129-3766400

德国 **Germany - Heilbronn**
Tel: 49-7131-67-3636

德国 **Germany - Karlsruhe**
Tel: 49-721-625370

德国 **Germany - Munich**
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

德国 **Germany - Rosenheim**
Tel: 49-8031-354-560

以色列 **Israel - Ra'anana**
Tel: 972-9-744-7705

意大利 **Italy - Milan**
Tel: 39-0331-742611
Fax: 39-0331-466781

意大利 **Italy - Padova**
Tel: 39-049-7625286

荷兰 **Netherlands - Drunen**
Tel: 31-416-690399
Fax: 31-416-690340

挪威 **Norway - Trondheim**
Tel: 47-7289-7561

波兰 **Poland - Warsaw**
Tel: 48-22-3325737

罗马尼亚
Romania - Bucharest
Tel: 40-21-407-87-50

西班牙 **Spain - Madrid**
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

瑞典 **Sweden - Gothenberg**
Tel: 46-31-704-60-40

瑞典 **Sweden - Stockholm**
Tel: 46-8-5090-4654

英国 **UK - Wokingham**
Tel: 44-118-921-5800
Fax: 44-118-921-5820