

# I<sup>2</sup>C通信协议：了解I<sup>2</sup>C Primer、PMBus和SMBus

Mary Grace Legaspi, 固件工程师  
Eric Peña, 固件工程经理

## 摘要

I<sup>2</sup>C，即Inter-Integrated Circuit，是一种常用的串行通信协议，用于在器件之间——特别是两个或两个以上不同电路之间建立通信。I<sup>2</sup>C Primer是最常用的I<sup>2</sup>C。本文将介绍I<sup>2</sup>C Primer的基本特性和标准，并重点说明在通信实现过程中如何正确使用该协议。从I<sup>2</sup>C的基本原理出发，我们将介绍其变体子集——系统管理总线(SMBus)和电源管理总线(PMBus)——的可用性及其区别。这三种协议各有专门的功能，旨在满足不同的客户需求。

## 为何重要？

I<sup>2</sup>C有利于设计人员在系统的众多节点之间建立简单、双向、灵活的通信。I<sup>2</sup>C仅使用两条双向线来发送和接收信息，从而降低了复杂性。它还允许设计人员配置多个主节点系统IC之间的通信。I<sup>2</sup>C对管理系统和电源的开发人员也很有利，让他们能够在尽可能短的时间内创建高质量的产品。

“沟通（通信）对那些致力于沟通（通信）的人有用。”

—John Powell

通信协议在组织设备之间通信时扮演着重要角色。它基于系统要求而以不同方式进行设计。此类协议具有明确的、为实现成功通信而协商一致的规则。

如果您曾经使用LED显示器、传感器甚至加速度计模块之类的东西构建过系统，那么您很可能已经使用过I<sup>2</sup>C。I<sup>2</sup>C支持多个节点连接单个主器件和多个主器件连接多个节点的特性。如果您希望物尽其用，让一个微控制器将数据记录到单个存储卡，或向单个LCD显示文本，则此特性非常有用。

除了最常用的I<sup>2</sup>C Primer，I<sup>2</sup>C还有两个变体，分别专注于系统和电源应用，称为系统管理总线(SMBus)和电源管理总线(PMBus)。

根据定义，Inter-Integrated Circuit (I<sup>2</sup>C)——也称为Inter IC——是一种硬件通信协议，它通过一条多主器件、多节点、串行通信总线进行同步通信。同步通信意味着两个（或两个以上）交换数据的器件共享一条公共时钟线。I<sup>2</sup>C广泛用于将低速外围IC连接到处理器和微控制器。I<sup>2</sup>C总线由飞利浦公司设计，它让位于同一电路板上的器件之间可以轻松通信。

## I<sup>2</sup>C Primer

### 接口

使用一条串行数据(SDA)线、一条串行时钟(SCL)线和一个公共接地来承载所有通信，最大程度地减少连接。

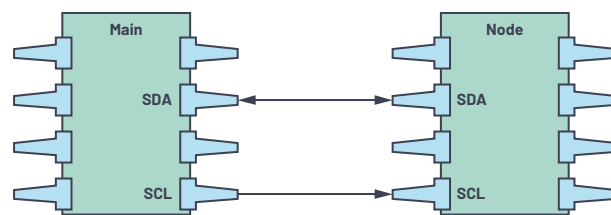


图 1. 集成电路彼此直接通信

每个I<sup>2</sup>C器件有两条线路：

- ▶ SDA是供主器件和节点发送和接收数据的线路。
- ▶ SCL是承载时钟信号的线路。SCL总是由I<sup>2</sup>C主器件生成。规范对时钟信号的低相位和高相位有最短周期要求。

I<sup>2</sup>C总线仅使用两条双向线路：每个器件的SDA和SCL用于简单的IC间通信。

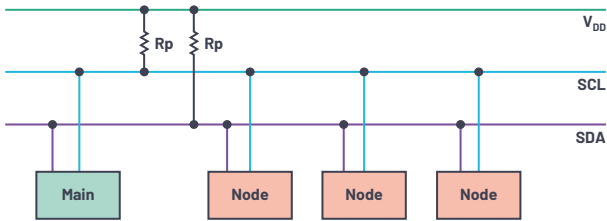


图 2. I<sup>2</sup>C 上拉电阻连接

硬件最重要的注意是在SDA和SCL上加入上拉电阻。I<sup>2</sup>C器件通过开集或开漏引脚连接到总线，将线路拉低。当没有数据传输时，I<sup>2</sup>C总线处于高电平空闲状态；线路被被动拉高。要传输数据，须切换线路，即先拉低再释放（又变为高电平）。数据位在时钟下降沿传输。

开漏输出需要一个上拉电阻（图2中的Rp）才能正确输出高电平。上拉电阻连接在输出引脚和高电平所需的输出电压（图2中的V<sub>DD</sub>）之间。

对于V<sub>CC</sub>和V<sub>DD</sub> (5 V)的典型值，4700 Ω是最常用的上拉电阻值。

作为参考，屏蔽2 AWG双绞线电缆的电容范围为100 pF/m至240pF/m。因此，I<sup>2</sup>C链路的最大总线长度约为1米（100 kBaud时）或10米（10 kBaud时）。非屏蔽电缆的电容通常要小得多，但只能用在以其他方式加以屏蔽的外壳内。

表1总结了I<sup>2</sup>C的关键特性。

表1. I<sup>2</sup>C汇总

特性	规格
导线	2
最大速度	标准模式 = 100 kbps 快速模式 = 400 kbps 高速模式 = 3.4 Mbps 超快速模式 = 5 Mbps
同步或异步?	同步
串行或并行?	串行
最大主器件数	无限制
最大节点数	1008

理论上，寻址模式的最大节点数为2<sup>7</sup>或2<sup>10</sup>，但有16个地址保留用于特殊用途。

I<sup>2</sup>C是同步的，因此位的输出通过主器件和节点之间共享的时钟信号与位的采样同步。时钟信号始终由主器件控制。



图 3. I<sup>2</sup>C 消息

### 保留I<sup>2</sup>C节点地址

有16个保留I<sup>2</sup>C地址。这些地址对应于以下两种模式之一：0000 XXX或1111 XXX。表2显示了为特殊目的而保留的I<sup>2</sup>C地址。

表2. I<sup>2</sup>C保留地址

I <sup>2</sup> C节点地址	R/W	位功能描述
0000 000	0	广播地址
0000 000	1	起始字节
0000 001	X	CBUS地址
0000 010	X	保留用于不同总线格式
0000 011	X	保留供未来使用
0000 1XX	X	高速模式主代码
1111 1XX	X	保留供未来使用
1111 0XX	X	10位节点地址

### I<sup>2</sup>C工作原理

I<sup>2</sup>C数据在消息中传输，消息被分解为数据帧。读写协议包含地址帧（即节点的二进制地址）和另一个数据帧，后者包含所传输的数据、开始和停止条件、重复起始位、读/写位以及每个数据帧之间的应答/不应答位。

### 时序规格表

I<sup>2</sup>C时序表也很重要，因为工程师利用它可以设计出与总线要求兼容的IC。每个数据速率都有自己的时序规格，主器件和节点必须遵守该规格才能正确传输数据。

表3显示了时序规格表上给出的符号和参数。

表3. I<sup>2</sup>C时序规格表示例

符号	参数	单位
f <sub>SCL</sub>	SCL时钟频率	kHz
t <sub>HD(STA)</sub>	(重复) 起始条件的保持时间	μs
t <sub>LOW</sub>	SCL引脚的低电平周期	μs
t <sub>HIGH</sub>	SCL引脚的高电平周期	μs
t <sub>SU(STA)</sub>	重复起始条件的建立时间	μs
t <sub>HD(DAT)</sub>	数据保持时间	μs
t <sub>SU(DAT)</sub>	数据建立时间	ns
t <sub>r</sub>	SDA信号的上升时间	ns
t <sub>f</sub>	SDA信号的下降时间	ns
t <sub>SU(STO)</sub>	停止条件的建立时间	μs

# I<sup>2</sup>C传输子协议

总线上的传输要么是读操作，要么是写操作。读取和写入协议建立在一系列子协议之上，例如起始和停止条件、重复起始位、地址字节、数据传输位和应答/不应答位。

## 起始条件

顾名思义，起始条件总是在传输开始时出现，并由主器件发起。这样做是为了唤醒总线上的空闲节点器件。SDA线从高电平切换到低电平，然后SCL线从高电平切换到低电平。参见图4。

## 重复起始条件

在不发出停止条件的情况下，起始条件可以在传输期间重复。这是一种特殊情况，称为重复起始，用于改变数据传输方向、重复尝试传输、同步多个IC，甚至控制串行存储器。参见图5。

## 地址帧

地址帧包含7位或10位序列，具体取决于可用性（参见数据手册）。参见图6。

不像SPI，I<sup>2</sup>C没有节点选择线路，因此它需要另一种方法来让节点知道数据正向其发送，而不是向另一个节点发送。这是通过寻址来实现的。地址帧始终是新消息中起始位之后的第一帧。

主器件将其想要与之通信的节点地址发送到其所连接的每个节点。然后，每个节点将主器件所发送的地址与其自己的地址进行比较。如果地址匹配，它便向主器件发送一个低电压ACK位。如果地址不匹配，则节点什么也不做，SDA线保持高电平。

## 读/写位

地址帧的最后一位告知节点，主器件是想要将数据写入其中还是从中接收数据。如果主器件希望将数据发送到节点，则读/写位处于低电平。如果主器件请求从节点得到数据，则该位处于高电平。参见图7。

## ACK/NACK位

消息中的每一帧后面都跟随一个应答/不应答位。如果成功接收到一个地址帧或数据帧，则接收器件会向发件者返回一个ACK位。

图例：在下面的图中，白色框表示节点，蓝色框表示主器件。参见图8。

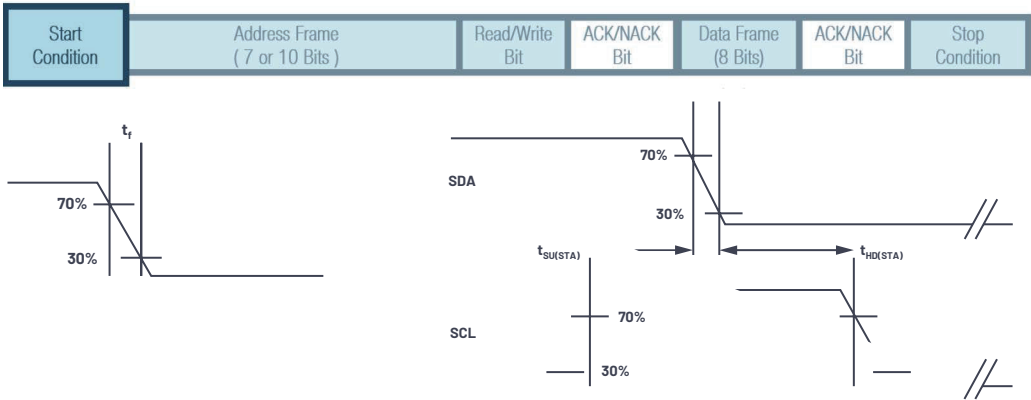


图 4. 起始条件



图 5. 重复起始条件



图 6. 地址帧



图 7. 读 / 写位



图 8. 应答 / 不应答位

## 数据帧

主器件检测到来自节点的ACK位之后，就准备发送第一数据帧。数据帧总是8位长，并以MSB优先方式发送。每个数据帧之后紧接着一个ACK/NACK位，以验证该帧是否已成功接收。主器件或节点（取决于谁发送数据）必须收到ACK位，然后才能发送下一数据帧。参见图9。

## 停止条件

发送完所有数据帧之后，主器件可以向节点发送停止条件以停止传输。停止条件是指SCL线上的电压从低电平变为高电平，然后在SCL线保持高电平的情况下，SDA线上的电压从低电平变为高电平。

在SCL线从低电平切换到高电平后，SDA线从低电平切换到高电平。参见图10。

## I<sup>2</sup>C传输步骤：写入

有关写入单个数据的I<sup>2</sup>C传输的示例，请参见图11。

## 第1步

主器件将SDA线从高电平切换到低电平，然后将SCL线从高电平切换到低电平，以将起始条件发送到每个相连的节点。

## 第2步

主器件将其想要与之通信的节点的7位或10位地址以及写操作位发送给每个节点。

例如，7位地址为0x2D，加上写操作位（相当于0），结果将是0x5A。

### 第3步

每个节点将主器件所发送的地址与其自己的地址进行比较。如果地址匹配，节点便将SDA线拉低一位的时间，以返回一个ACK位。如果来自主器件的地址与节点自己的地址不匹配，则节点让SDA线保持高电平不变。

在SCL的第九个脉冲期间拉低SDA线路可以发送ACK位，保持浮空高电平则为NACK。

## 第4步

主器件发送或接收数据帧。

## 第5步

传输完每个数据帧之后，接收器件再向发送者返回一个ACK位，以确认成功接收该帧。

## 第6步

若要停止数据传输，主器件应将SCL切换为高电平，然后将SDA切换为高电平，从而发送停止条件。

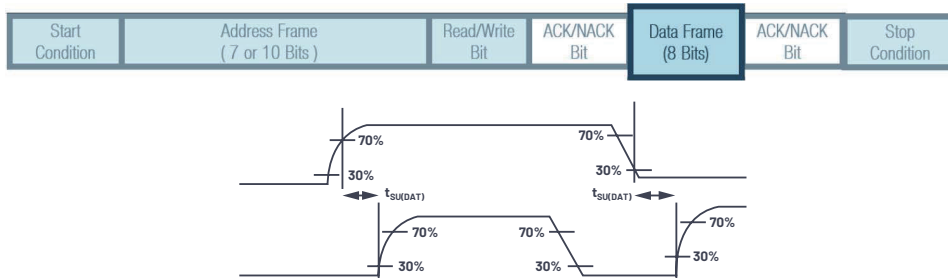


图 9. 数据帧

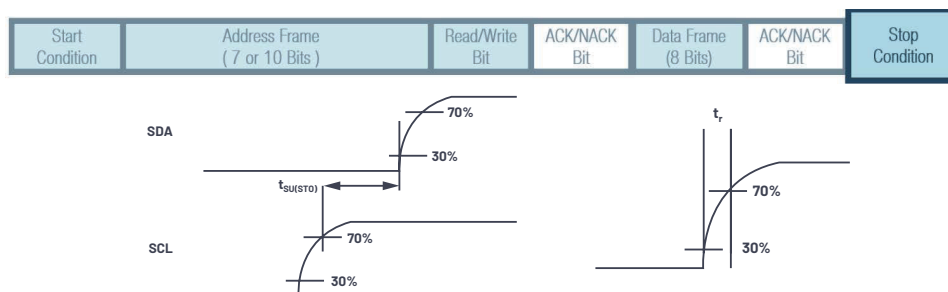


图 10. 停止条件

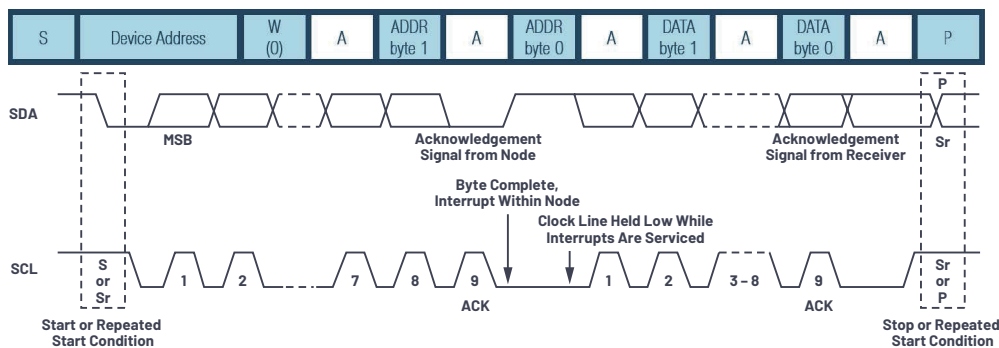
图 11. 写入单个位置的  $^2C$  传输的数据手册示例



图 12. 读取单个位置的 I<sup>2</sup>C 传输的数据手册示例

## I<sup>2</sup>C 数据传输步骤：读取

### 第1步

主器件将 SDA 线从高电平切换到低电平，然后将 SCL 线从高电平切换到低电平，以将起始条件发送到每个相连的节点。

### 第2步

主器件将其想要与之通信的节点的 7 位或 10 位地址以及写操作位发送给每个节点。

例如，7 位地址为 0x2D，加上写操作位（相当于 0），结果将是 0x5A。

### 第3步

每个节点将主器件所发送的地址与其自己的地址进行比较。如果地址匹配，节点便将 SDA 线拉低一位的时间，以返回一个 ACK 位。如果来自主器件的地址与节点自己的地址不匹配，则节点让 SDA 线保持高电平不变。

### 第4步

经过初始启动、寻址和应答之后，主器件已经知道目标节点及指向的地址，因此某些器件具有重复起始条件来清理事务。

注意：仅用于阅读目的！

### 第5步

主器件将其想要与之通信的节点的 7 位或 10 位地址以及读操作位发送给每个节点。

例如，7 位地址为 0x2D，加上读操作位（相当于 1），结果将是 0x5B。

### 第6步

每个节点将主器件所发送的地址与其自己的地址进行比较。如果地址匹配，节点便将 SDA 线拉低一位的时间，以返回一个 ACK 位。如果来自主器件的地址与节点自己的地址不匹配，则节点让 SDA 线保持高电平不变。

### 第7步

得到 ACK 位之后，主器件接收来自节点的数据帧。

### 第8步

传输完每个数据帧之后，主器件再向发送者返回一个 ACK 位，以确认成功接收该帧，或者如果读取请求已经完成，则主器件返回 NACK。

### 第9步

若要停止数据传输，主器件应将 SCL 切换为高电平，然后将 SDA 切换为高电平，从而发送停止条件。

## 单个主器件和多个节点

I<sup>2</sup>C 使用寻址，所以单个主器件可以控制多个节点。使用 7 位地址可提供 128 (2<sup>7</sup>) 个唯一地址。使用 10 位地址很罕见，但可提供 1024 (2<sup>10</sup>) 个唯一地址。要将多个节点连接到单个主器件，请使用 4.7 kΩ 上拉电阻连接这些节点，并将 SDA 和 SCL 线连接到 V<sub>CC</sub>。

## 多个主器件和多个节点

多个主器件可以连接到单个节点或多个节点。如果同一系统中有多个主器件，那么当两个主器件争着在同一时间通过 SDA 线发送或接收数据时，就会出现问题。

为了解决这个问题，每个主器件在传输消息之前，需要检测 SDA 线是低电平还是高电平。

如果 SDA 线为低电平，则说明总线由另一个主器件控制，该主器件应等待。如果 SDA 线为高电平，则它可以安全传输消息。要将多个主器件连接到多个节点，请按照图 13 所示，使用 4.7 kΩ 上拉电阻将 SDA 和 SCL 线连接到 V<sub>CC</sub>。

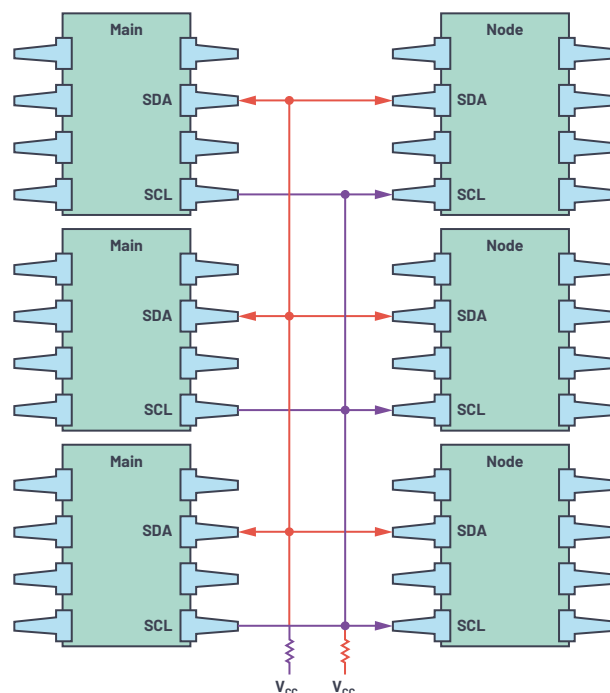


图 13. 连接多个节点的多个主器件

## 仲裁

若干 I<sup>2</sup>C 多主器件可以连接到同一 I<sup>2</sup>C 总线并同时运行。通过不断监视 SDA 和 SCL 有无起始和停止条件，它们可以确定总线是否空闲。如果总线正忙，主器件将延迟挂起的 I<sup>2</sup>C 传输，直至停止条件指示总线再次空闲。



但是，可能发生两个主器件同时开始传输的情况。在传输过程中，主器件不断监视SDA和SCL。如果其中一个检测到SDA为低电平，而它应该为高电平，则该主器件将认为另一主器件处于活动状态，因而立即停止传输。此过程称为仲裁。两个主器件都会生成起始位并继续各自的传输。

如果主器件恰好选择相同的逻辑电平，则什么也不会发生。

一旦主器件尝试施加不同的逻辑电平，则将信号拉低的主器件将被宣布为获胜者；失败者将检测到逻辑不匹配，因而放弃传输。

请花点时间理解一下这种安排的简单性和有效性：

- ▶ 获胜者继续传输而不中断——没有数据损坏，没有驱动器争用，不需要重新启动事务。
- ▶ 理论上，失败者可以在仲裁过程中监视节点地址，如果恰好是被寻址的节点，它可以做出适当的响应。
- ▶ 如果相互竞争的主器件均请求同一节点的数据，则仲裁过程不会不必要地中断任一事务——不会检测到不匹配，节点会将其数据输出到总线，多个主器件可以接收到数据。

时钟延展

也称为时钟同步。

注意：I<sup>2</sup>C规范没有为时钟延展规定任何超时条件——也就是说，任何器件都可以根据需保持SCL。

在I<sup>2</sup>C通信协议中，时钟速度和信号始终由主器件产生。I<sup>2</sup>C主器件产生的信号提供主器件和节点连接之间的同步。

在某些情况下，节点或子节点不是以全状态工作，在接收主器件生成的时钟之前，需要减慢速度。这是通过一种称为“时钟延展”的机制来实现的。

在时钟延展期间，为了降低总线速度，允许节点压低时钟。而在主器件方面，在其变为高电平状态后，必须回读时钟信号。然后，它必须等待，直至线路达到高电平状态。

带宽

虽然时钟延展是一种常见做法，但它对带宽有影响。使用时钟延展时，共享总线的总带宽可能会显著降低。即使使用这种技术，总线性能仍然必须可靠且快速。有必要考虑使用时钟延展的估计影响，尤其是在多个器件共享I<sup>2</sup>C总线的情况下。

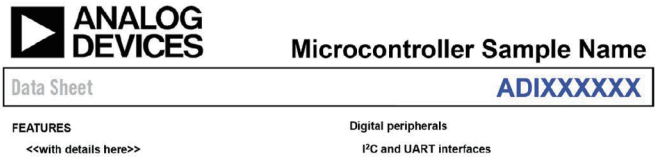


图 14. 微控制器数据手册

通过时钟延展，I<sup>2</sup>C节点器件可以强制主器件进入等待状态。当节点器件需要更多时间来管理数据时，例如存储接收到的数据或准备发送另一字节的数据时，它可能会执行时钟延展。这通常发生在节点器件接收并确认收到一个字节的数

哪些I<sup>2</sup>C节点器件需要时钟延展？

是否需要时钟延展取决于节点器件的功能。这里有两个例子：

- ▶ 处理器件（如微处理器或微控制器）可能需要额外的时间来处理中断，接收和管理数据，以及执行适当的功能。
- ▶ 较简单的器件（如EEPROM）不在内部处理数据，因此不需要时钟延展来执行任何功能。

I<sup>2</sup>C数据手册示例概述

不同公司和制造商采用不同方法来创建数据手册。图13显示了一个简单的数据手册示例和基本I<sup>2</sup>C细节，包括寄存器和电子规格。

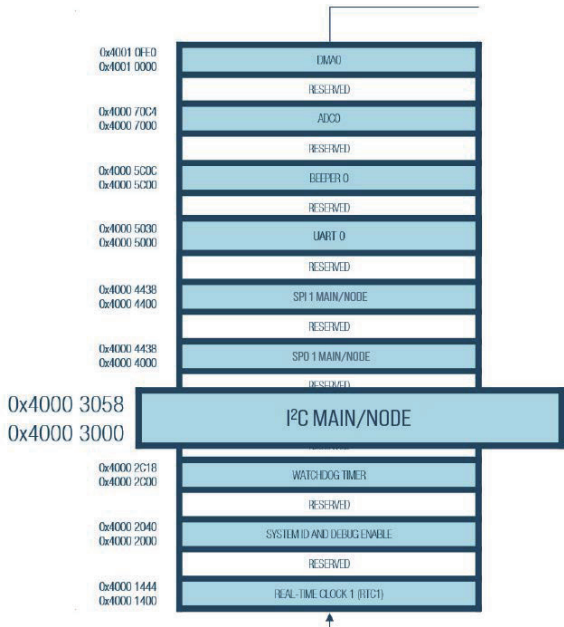


图15. 微控制器存储器映射

表4显示了最常用的I<sup>2</sup>C寄存器。名称和描述可能因数据手册而异，但功能和用法相同。

表4. I<sup>2</sup>C寄存器描述

名称	描述
I2C_ADDR1	主器件地址字节1
I2C_ADDR2	主器件地址字节2
I2C_BYT	起始字节
I2C_ID	节点地址器件ID
I2C_MCTL	主器件控制
I2C_MRX	主器件接收数据
I2C_SCTL	节点控制
I2C_SRX	节点接收
I2C_STAT	主器件和节点FIFO状态

I<sup>2</sup>C的创建可能因使用情况而异。表5显示了基本I<sup>2</sup>C驱动程序API要求的示例。

表5. I<sup>2</sup>C驱动开发

主器件	节点
初始化	
发送处理程序	发送处理程序
接收处理程序	接收处理程序
事件中断	
错误中断	

## SMBus

众所周知，SMBus可用于需要对参数进行关键监控的应用。它最常见的应用是计算机主板和嵌入式系统。对于温度、电源电压、风扇监控和/或控制集成芯片，它有额外的监控规范。

SMBus是一种2线总线，类似于飞利浦公司于1980年代开发的I<sup>2</sup>C总线。两个主要信号是时钟(SMBCLK)和数据(SMBDAT)。I<sup>2</sup>C Primer和SMBus相互兼容，但存在明显差异，例如：

- ▶ SMBus逻辑电平阈值是固定的，与器件的电源电压不成比例。因此，具有不同电源电压的器件可以在同一Primer上运行。例如，一个SMBus可能具有多个由1.8 V、3.3 V和5 V电源供电的器件。
- ▶ 它们都以最高100 kHz的相同速度运行，但I<sup>2</sup>C Primer有400 kHz和2 MHz两个版本。
- ▶ SMBus规定了最低时钟速度，并限制了时钟在一个事务中可以延展的量。违反超时限制会导致所有SMBus器件复位其I/O逻辑以允许总线重启。这种设计增强了总线的鲁棒性。
- ▶ 二者的超时也不同。I<sup>2</sup>C Primer没有超时，而SMBus有超时——对于10 kHz最低时钟速度，可以考虑35 ms的超时。
- ▶ 分组差错校验(PEC)最初是为SMBus定义的。在每个事务的末尾添加一个分组错误码字节。
- ▶ 其余的一些差异涉及传输类型、警报线、暂停线、关断或上电。

SMBus器件每次收到其自己的地址时，无论在做什么，它都必须应答(ACK)，这是一个明确要求，目的是确保主器件可以准确地判断总线上哪些器件处于活动状态。

所有SMBus事务都通过指定的SMBus协议之一执行。

SMBus还有一个可选信号SMBALERT#，节点器件可以使用该信号快速通知主器件或系统主机，它有主器件需要的信息，例如报告故障情况。

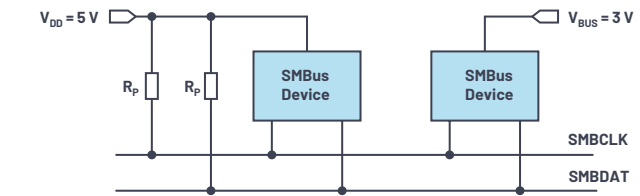


图16. SMBus拓扑结构

## SMBus上拉电路

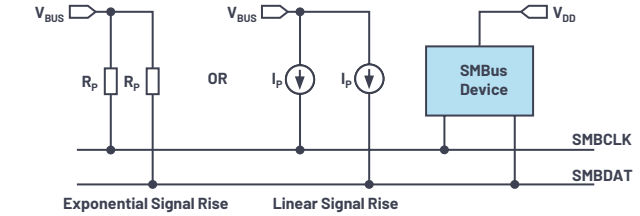


图17. SMBus上拉电路

## SMBus地址

SMBus地址有7个二进制位，通常表示为前4位、后3位以及最后一个字母b，例如0001 110b。这些地址占据总线上一个8位字段的高7位。然而，该字段的最低位另有含义，不属于SMBus地址的范围。

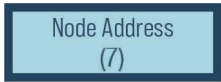


图18. 节点地址

7位目标地址从主器件发送到总线上的一个或多个器件（例如通过广播地址）。

请注意，起始条件和停止条件是转换，而不是位，在符号上方未显示位计数。在事务图中显示时，重复起始也是一个转换，而不是一位，在符号上方也不显示位计数。

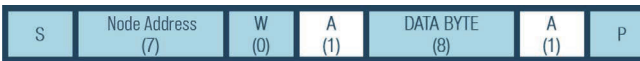


图19. SMBus消息

## SMBus时序测量

表6. SMBus参数

符号	参数	单位
$f_{SMB}$	SMBus工作频率	kHz
$t_{BUF}$	停止与起始条件之间的总线空闲时间	$\mu s$
$T_{HD-STA}$	(重复) 起始条件之后的保持时间	$\mu s$
$T_{SU-STA}$	重复起始条件的建立时间	$\mu s$
$t_{SU(STO)}$	停止条件的建立时间	$\mu s$
$t_{HD(DAT)}$	数据保持时间	ns
$t_{SU(DAT)}$	数据建立时间	ns
$t_{TIMEOUT}$	检测时钟低电平超时	ms
$t_{LOW}$	时钟低电平周期	$\mu s$
$t_{HIGH}$	时钟高电平周期	$\mu s$

PMBus：重新定义电源管理

除了SMBus之外，还有一个变体PMBus，它是一种开放标准电源管理协议。这种灵活且高度通用的标准允许基于模拟和数字技术的器件之间进行通信，并提供真正的互操作性，由此将能降低电源系统设计的复杂性并缩短产品上市时间。

PMBus用于带有电源控制和管理器件的电源的数字管理。它具有支持电源管理要求的命令和结构。这意味着I<sup>2</sup>C Primer和PMBus在电气要求和命令语义上是兼容和可互操作的。

电源管理的基本参数之一是过压电平监控，PMBus提供了设置和读取该值的命令。PMBus可以附加在I<sup>2</sup>C Primer和SMBus的已有特性上，充当现有标准（尤其是SMBus）之上的协议层。

I<sup>2</sup>C规范仅描述了2线总线的物理层、时序和流控制。I<sup>2</sup>C规范没有（像SMBus协议那样）描述消息的格式，也没有描述消息的内容。

PMBus规范是一个完整的电源管理协议。它说明了如何将比特和字节从一个器件传送到另一个器件（即传输）。它还描述了一种命令语言，赋予这些比特和字节以意义。

寻址

对于冗余系统，一旦电源安装到系统中，最多有三个信号来设置电源的地址位置：地址2、地址1和地址0。对于非冗余系统，电源器件地址位置应为B0h。

硬件

针对基于I<sup>2</sup>C V<sub>DD</sub>的电源和驱动（对于V<sub>DD</sub> = 3.3 V），电源中的器件应与SMBus 2.0高功率规范兼容。该总线应以3.3 V运行。

电源

电源内部的电路应从备用输出获得电源。对于冗余电源，器件应从“逻辑或”器件的系统侧供电。只要系统中的电源或并联冗余电源接通交流电源，PMBus器件就应处于开启状态。

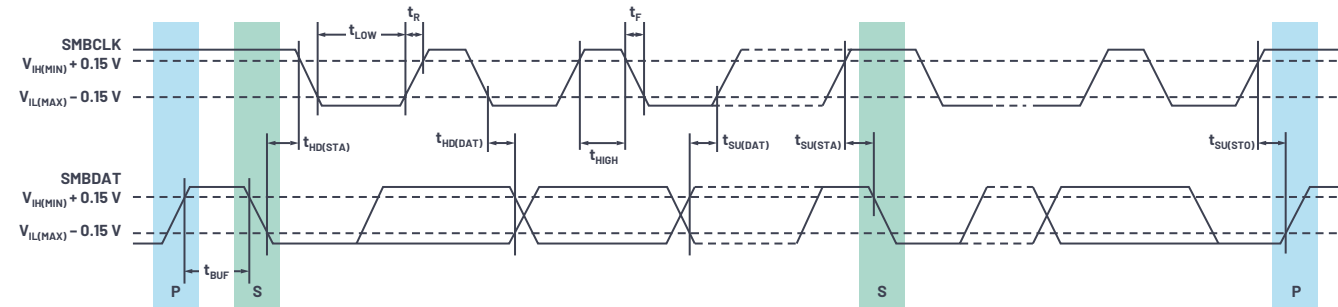


图20. SMBus时序测量

表7. PMBus寻址

所用地址	主要寻址，用于具有两个寻址引脚的大多数服务器电源				附加寻址，前提是电源上提供了三个寻址引脚			
系统寻址 地址2/地址1/地址0	0/0/0	0/0/1	0/1/1	0/1/1	1/0/0	1/0/1	1/1/0	1/1/1
PMBus器件读取地址	B0h/B1h	B2h/B3h	B4h/B5h	B6h/B7h	B8h/B9h	BAh/BBh	BCh/BDh	BEh/BFh

上拉电阻

电源内部的SCL或SDA线上只能使用弱上拉电阻。主要上拉电阻由系统提供，可以连接到3.3 V或5 V。对于系统设计，主要上拉电阻应位于电源外部，并从备用电源轨获取电源。

数据速度

电源中的PMBus器件应以100 kbps SMBus全速运行，并尽可能避免使用时钟延展，因为它会减慢总线速度。

总结

表8概述并总结了I<sup>2</sup>C Primer、SMBus（高功率和低功率）、PMBus的信号、时序和电气规格。

I<sup>2</sup>C Primer、SMBus和PMBus有何关系？

SMBus最初开发用于协助电池管理系统，使用I<sup>2</sup>C硬件，但增加了第二级软件，最终允许器件热插拔，而无需重新启动系统。PMBus扩展了SMBus，定义了一组专门用于管理功率转换器的器件命令，暴露了器件的测量电压、电流、温度等属性。一般而言，I<sup>2</sup>C Primer、SMBus和PMBus器件可以共享总线而不会发生什么大问题。

I<sup>2</sup>C、SMB、PMB的优势

- ▶ 仅使用两条线
- ▶ 具有ACK/NACK位
- ▶ 广为人知的协议
- ▶ 支持多个主器件和多个节点
- ▶ 硬件不如UART复杂
- ▶ 广泛使用的方法

缺点

- ▶ 数据传输速率比SPI慢
- ▶ 数据帧的大小限制为8位
- ▶ 实现所需的硬件比SPI复杂



用例

- ▶ 传感器读取
- ▶ 传感器写入
- ▶ EEPROM、温度传感器、触摸屏、接近传感器
- ▶ 传输和控制用户指引的操作
- ▶ 与多个微控制器通信
- ▶ 消费类电子设备
- ▶ 系统管理
- ▶ 电源管理
- ▶ 调试

参考文献

“I<sup>2</sup>C通信的优势和局限”。Total Phase，2016年8月。

Afzal, Sal。 “I<sup>2</sup>C Primer：什么是I<sup>2</sup>C？（第一部分）”。ADI公司。

Afzal, Sal。 “I<sup>2</sup>C时序：定义和规范指南（第2部分）”。ADI公司。

Campbell, Scott。 “I<sup>2</sup>C通信协议基础”。Circuit Basics。

I<sup>2</sup>C快速指南。ADI公司。

“I<sup>2</sup>C是什么？” I<sup>2</sup>C总线。

表8. I<sup>2</sup>C Primer、SMBus和PMBus规格总结

规格		I <sup>2</sup> C Primer	SMBus 高功率   低功率		PMBus
信号	分组差错校验（可选）	—	—	—	—
	SMBALERT（可选）	—	—	—	—
	块大小限值	—	32 字节	32 字节	255 字节
时序	数据速率： • 标准模式 • 快速模式 • 快速模式+ • 高速模式	100 kbps 400 kbps 1 Mbps 3.4 Mbps 0 Hz 至 3.4 MHz	100 kbps — — — 10 kHz 至 100 kHz	100 kbps — — — 10 kHz 至 100 kHz	100 kbps 400 kbps — — 10 kHz 至 400 kHz
	时钟速度	—	25 ms 至 35 ms	25 ms 至 35 ms	25 ms 至 35 ms
	总线超时	—	50 μs	50 μs	50 μs
	总线主器件请求延迟（最小值）	—	—	—	—
	SCL保持时间（最大值）	—	2 ms	2 ms	2 ms
	数据保持时间（最小值）	—	300 ns	300 ns	300 ns
	—	—	—	—	—
电气	每个总线段的容性负载（最大值）	400 pF	400 pF	—	400 pF
	上升时间（最大值）	1 μs（100 kHz时），300 ns（400 kHz时）	1 μs	1 μs	1 μs（100 kHz时），300 ns（400 kHz时）
	0.4 V时的上拉电流（最大值）	3 mA（标准和快速模式）	4 mA	350 μA	4 mA
	每个器件的漏电流（最大值）	±10 μA	±10 μA	±5 μA	±10 μA
	VIL输入逻辑低电平阈值（最大值）	0.3 V <sub>DD</sub> 或 1.5 V	0.8 V	0.8 V	0.8 V
	VIH输入逻辑高电平阈值（最小值）	0.7 V <sub>CC</sub> 或 3 V	2.1 V	2.1 V	2.1 V
VOL输出逻辑低电平阈值（最大值）	—	0.4 V	2.4 V	0.4 V	0.4 V



作者简介

Mary Grace Legaspi是一名固件工程师，隶属设计与布局团队，在ADI公司的消费者软件工程部门工作。2018年9月加入ADI公司，工作地点位于菲律宾甲米地。毕业于塔拉克州立大学，获电子工程学士学位。目前正在攻读菲律宾大学管理硕士学位。联系方式：[mary.legaspi@analog.com](mailto:mary.legaspi@analog.com)。



作者简介

Eric Peña是设计与布局团队的固件工程经理，在ADI公司的工业平台与网络部门工作。2019年4月加入ADI公司，工作地点位于菲律宾甲米地。毕业于马尼拉亚当森大学，获计算机工程学士学位。Eric之前曾在Technology Enabler Designer担任固件工程师，还曾在Fujitsu Ten Solutions担任系统工程师。联系方式：[eric.pena@analog.com](mailto:eric.pena@analog.com)。

