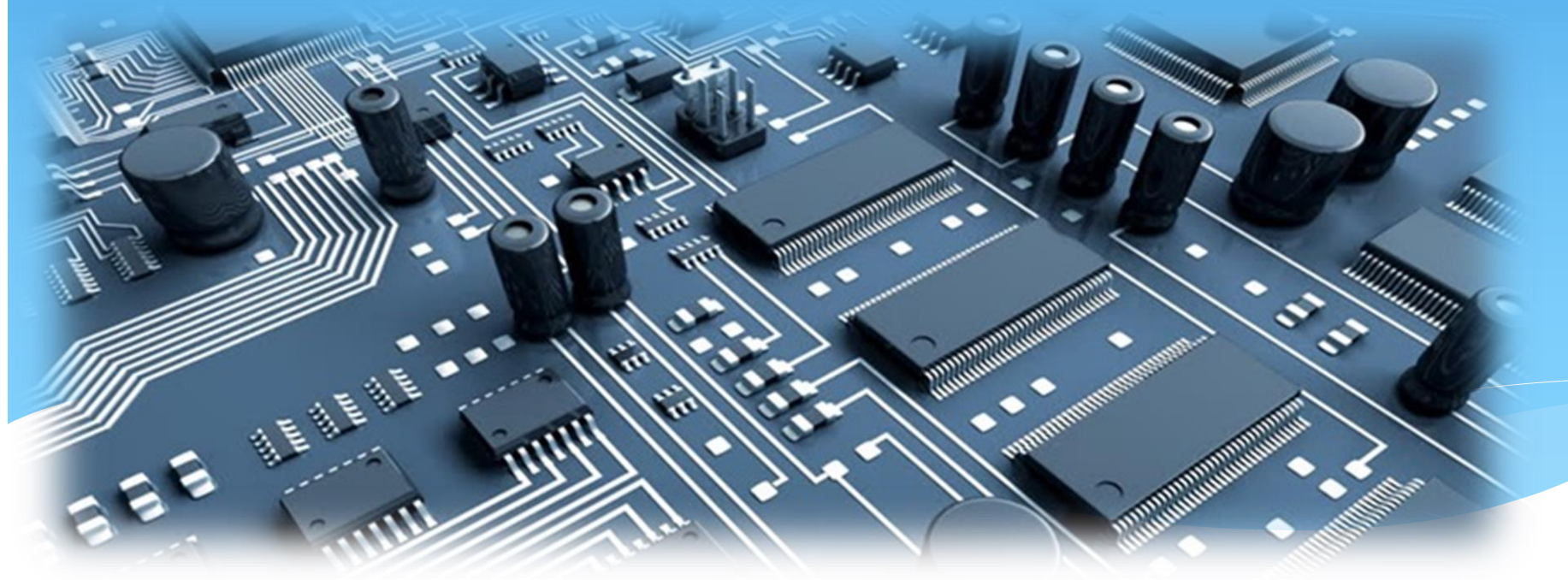


微机原理与接口技术

第15讲 总线技术（2）



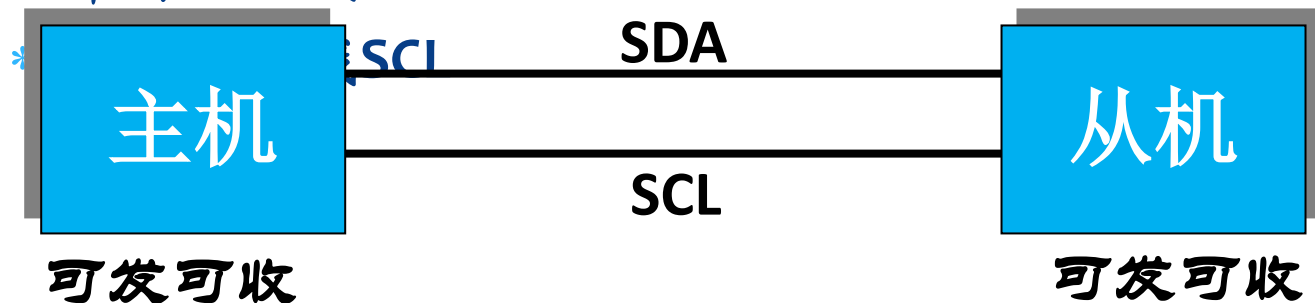
第四节 I²C总线

- * 第一节 总线概述
- * 第二节 PC领域常用的系统总线
- * 第三节 常用的通信总线
- * 第四节 I²C总线

I²C总线概述

- * I²C总线协议只使用2条线

- * 串行数据线SDA



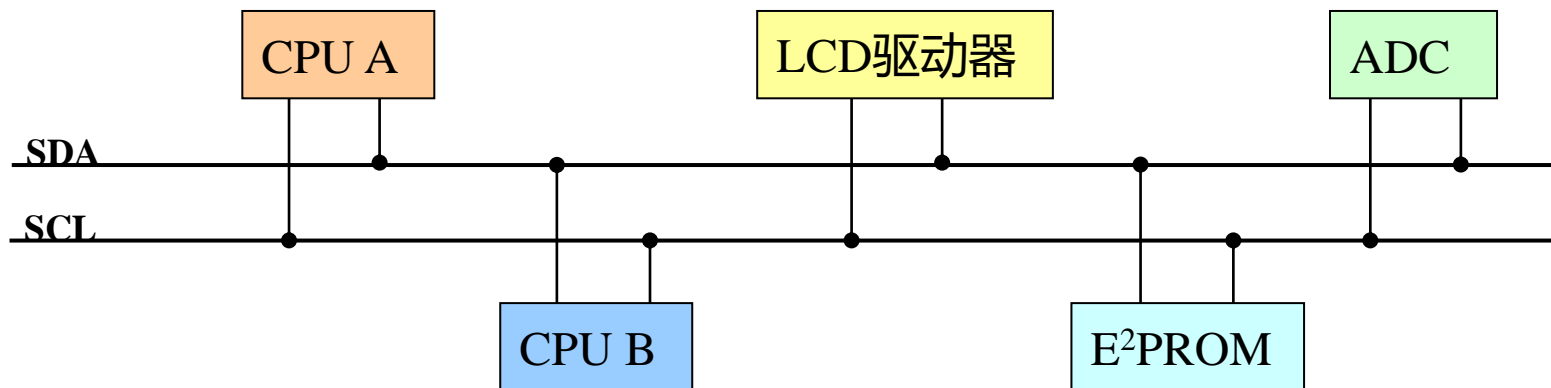
第四节 I²C总线

I²C总线概述

I²C总线，是INTER-IC串行总线的缩写。INTER-IC意思是用于相互作用的集成电路，是Philips推出的串行传输总线。

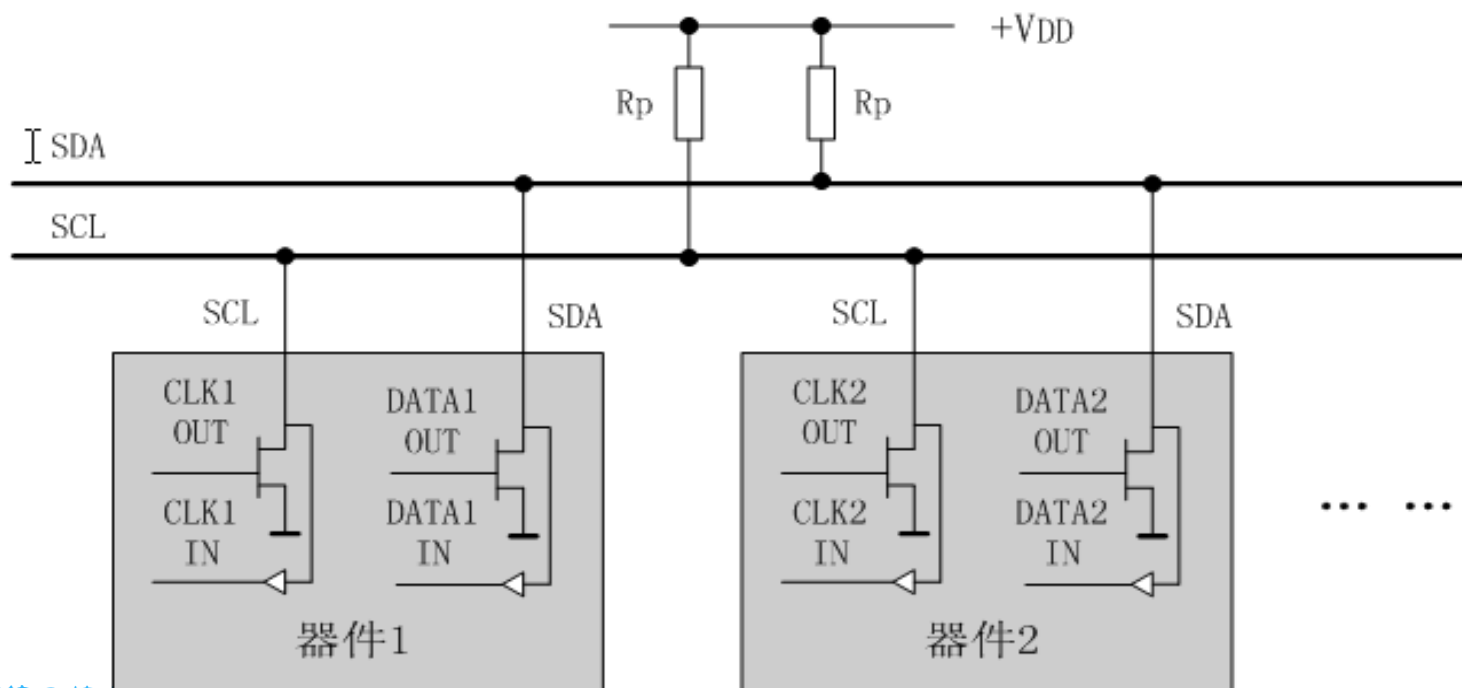
这种集成电路主要由**双向串行时钟线SCL**和**双向串行数据线SDA**两条线路组成。是具备多主机系统所需的包括**总线裁决**和**高低速器件同步功能**的高性能串行总线。以2根连线实现了完善的全双工同步数据传送，可以极方便地构成多机系统和外围器件扩展系统。

各被控设备（模块）均并联在这条总线上，每一个模块既是**接收器**，又是**发送器**，主要取决于它所要完成的功能。



I²C总线概述

I²C总线通过上拉电阻接正电源。当总线空闲时，两根线均为高电平。连到总线上的任一器件输出的低电平，都将使总线的信号变低，即各器件的SDA及SCL都是线“与”关系。



I²C串行总线概述

每个接到I²C总线上的器件都有唯一的地址。主机与其它器件间的数据传送可以由主机发送数据到其它器件，这时主机即为发送器。由总线上接收数据的器件则为接收器。

主机：初始化发送、产生时钟信号和终止发送的器件，它可以是发送器或接收器。主机通常是微处理器。

从机：被主机寻址的器件，它可以是发送器或接收器，

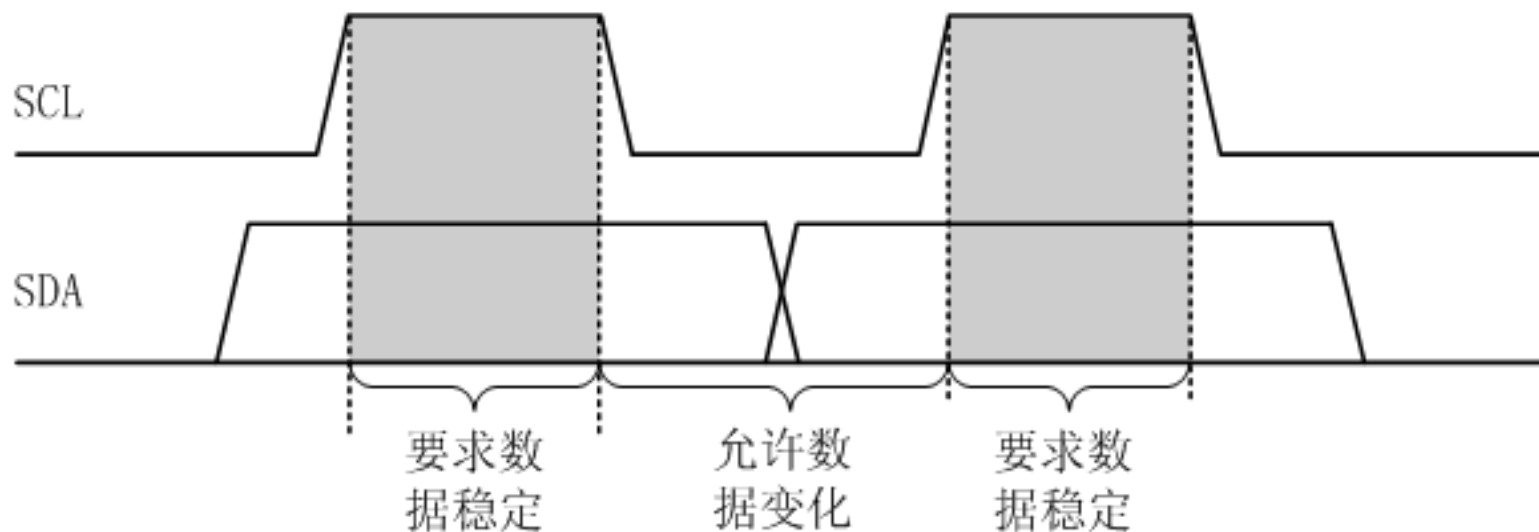
在多主机系统中，可能同时有几个主机企图启动总线传送数据。为了避免混乱，I²C总线要通过总线仲裁，以决定由哪一台主机控制总线。

在单片机应用系统的串行总线扩展中，经常以单片机为主机，其它接口器件为从机。

I²C总线的数据传送

一、数据位的有效性规定

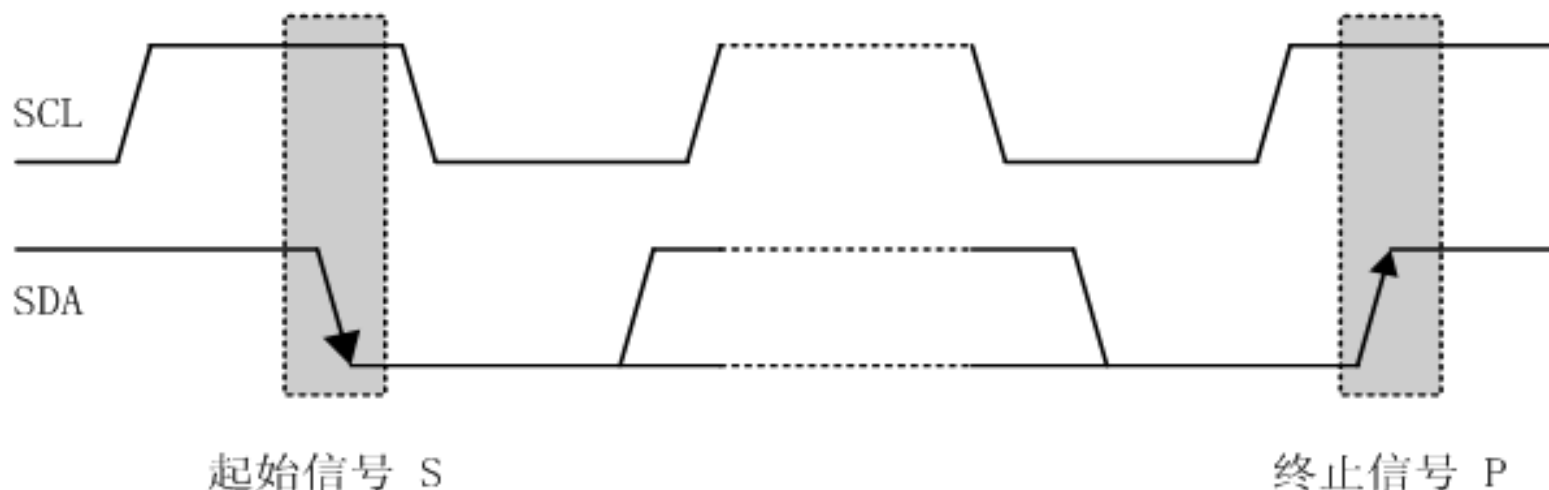
I²C总线进行数据传送时，**时钟信号为高电平期间，数据线上的数据必须保持稳定**，只有在时钟线上的信号为低电平期间，数据线上的高电平或低电平状态才允许变化。



I²C总线的数据传送

二、起始和终止信号

SCL线为高电平期间，SDA线由高电平向低电平的变化表示起始信号；SCL线为高电平期间，SDA线由低电平向高电平的变化表示终止信号。



I²C总线中唯一违反上述数据有效性的是被定义为**起始 (S)** 和**停止 (P)** 条件。

I²C总线的数据传送

起始和终止信号都是由**主机**发出的。

- 在起始信号产生后，总线就处于被占用的状态；
- 在终止信号产生后，总线就处于空闲状态。

连接到I²C总线上的器件

- 若具有I²C总线的硬件接口，则很容易检测到起始和终止信号。
- 对于不具备I²C总线硬件接口的有些单片机来说，为了检测起始和终止信号，必须**保证在每个时钟周期内对数据线SDA采样两次**。

I²C总线的数据传送

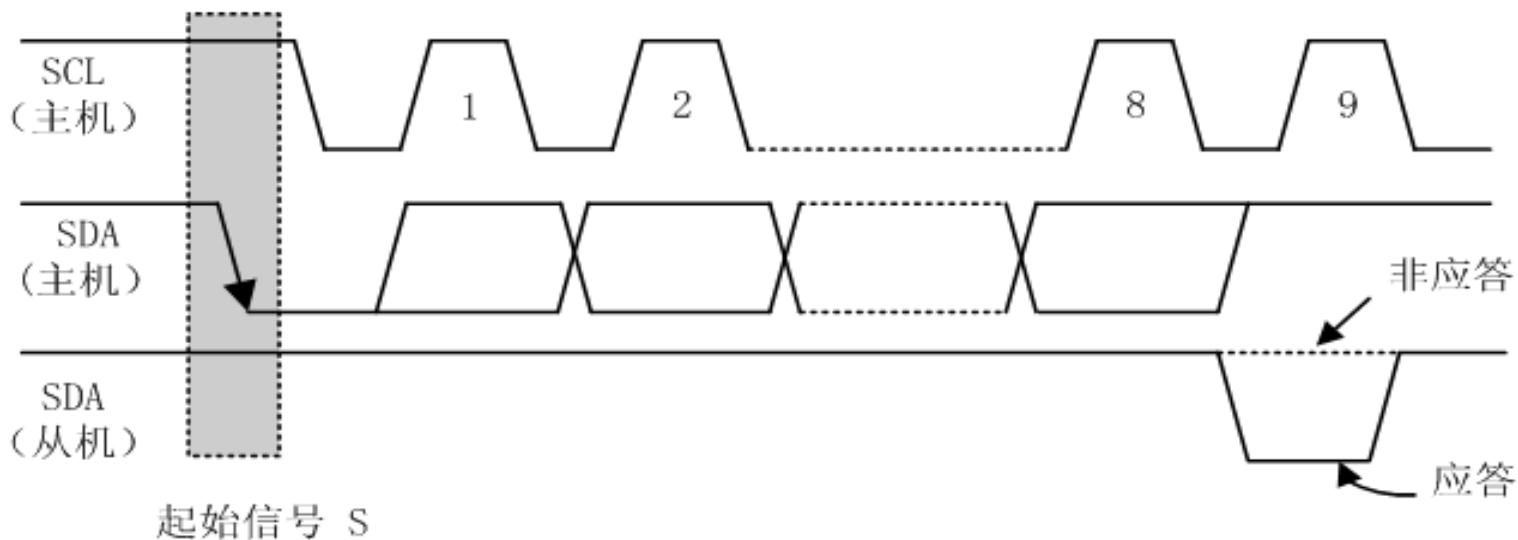
接收器件收到一个完整的数据字节后，有可能需要完成一些其它工作，如处理内部中断服务等，可能无法立刻接收下一个字节，这时接收器件可以将SCL线拉成低电平，从而使主机处于等待状态。直到接收器件准备好接收下一个字节时，再释放SCL线使之为高电平，从而使数据传送可以继续进行。

I²C总线的数据传送

三、数据传送格式

(1) 字节传送与应答

- 每一个字节必须保证是**8位**长度。
- 数据传送时，**先传送最高位（MSB）**，每一个被传送的字节后面都必须跟随**一位应答位**（即**一帧共有9位**）。



I²C总线的数据传送

- 由于某种原因从机不对主机寻址信号应答时（如从机正在进行实时性的处理工作而无法接收总线上的数据），它必须将数据线置于高电平，而由主机产生一个终止信号以结束总线的数据传送。
- 如果从机对主机进行了应答，但在数据传送一段时间后无法继续接收更多的数据时，从机可以通过对无法接收的第一个数据字节的“非应答”通知主机，主机则应发出终止信号以结束数据的继续传送。
- 当主机接收数据时，它收到最后一个数据字节后，必须向从机发出一个结束传送的信号。这个信号是由对从机的“非应答”来实现的。然后，从机释放SDA线，以允许主机产生终止信号。

I²C总线的数据传送

(2) 数据帧格式

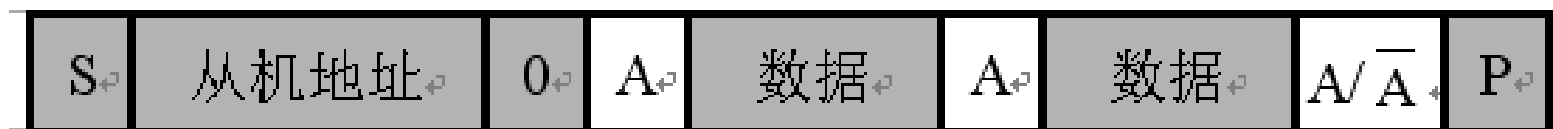
I²C总线上传送的数据信号是广义的，既包括**地址信号**，又包括真正的**数据信号**。

- 在起始信号后必须传送一个从机的**地址**（**7位**）；
- 第**8位**是数据的**传送方向位**（**R/ \overline{W}** ）：
 - ✓ 用“**0**”表示**主机发送数据**（**T**），
 - ✓ 用“**1**”表示**主机接收数据**（**R**）。
- 每次数据传送总是由主机产生的**终止信号**结束。但是，若主机希望继续占用总线进行新的数据传送，则可以不产生终止信号，马上再次发出起始信号对另一从机进行寻址。

I²C总线的数据传送

在总线的一次数据传送过程中，可以有以下几种组合方式：

A、主机向从机发送数据，数据传送方向在整个传送过程中不变。



有阴影部分表示数据由主机向从机传送，无阴影部分则表示数据由从机向主机传送。

A表示应答， \bar{A} 表示非应答（高电平）。S表示起始信号，P表示终止信号。

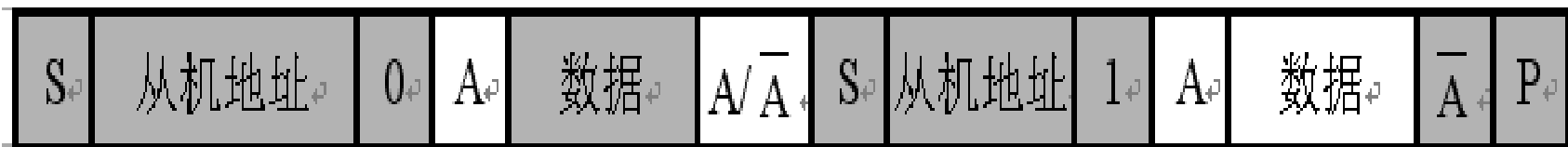
I²C总线的数据传送

B、主机在第一个字节(寻址字节)后，立即由从机读数据。



在从机产生响应时，主机从发送变成接收，从机从接收变成发送。之后，数据由从机发送，主机接收，每个应答由主机产生，时钟信号仍由主机产生。若主机要终止本次传输，则发送一个非应答信号(A)，接着主机产生停止条件。

C、在传送过程中，当需要改变传送方向时，起始信号和从机地址都被重复产生一次，但两次读/写方向位正好反相。



I²C总线的数据传送

四、总线的寻址

I²C总线协议有明确的规定：采用**7位的寻址字节**（寻址字节是起始信号后的第一个字节）。

（1）寻址字节的位定义

位：	7	6	5	4	3	2	1	0
	从机地址							R/ \overline{W}

D7~D1位组成从机的地址。**D0**位是数据**传送方向位**，为“**0**”时表示**主机向从机写数据**，为“**1**”时表示**主机由从机读数据**。

I²C总线的数据传送

- 主机发送地址时，总线上的每个从机都将这7位地址码与自己的地址进行比较，如果相同，则认为自己正被主机寻址，根据R/ \overline{W} 位将自己确定为发送器或接收器。
- 从机的地址由固定部分和可编程部分组成。在一个系统中可能希望接入多个相同的从机，从机地址中可编程部分决定了可接入总线该类器件的最大数目。

如一个从机的7位寻址位有4位是固定位，3位是可编程位，这时仅能寻址8个同样的器件，即可以有8个同样的器件接入到该I²C总线系统中。

I²C总线的数据传送

(2) 寻址字节中的特殊地址

固定地址编号0000和1111已被保留作为特殊用途。

地 址 位							R/ \overline{W}	意 义
0	0	0	0	0	0	0	0	通用呼叫地址
0	0	0	0	0	0	0	1	起始字节
0	0	0	0	0	0	1	X	CBUS 地址
0	0	0	0	0	1	0	X	为不同总线的保留地址
0	0	0	0	0	1	1	X	
0	0	0	0	1	X	X	X	
1	1	1	1	1	X	X	X	
1	1	1	1	0	X	X	X	十位从机地址

I²C总线的数据传送

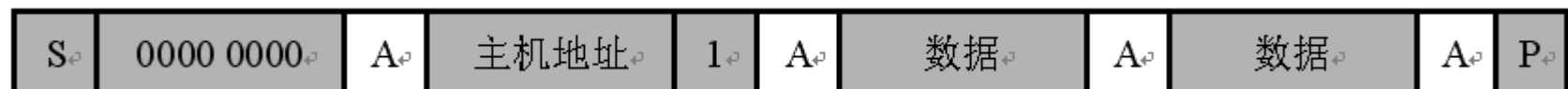
起始信号后的第一字节的8位为“**0000 0000**”时，称为**通用呼叫地址**。通用呼叫地址的用意在第二字节中加以说明。格式为：

第一字节（通用呼叫地址）									第二字节								LSB	
0	0	0	0	0	0	0	0	A	×	×	×	×	×	×	×	×	B	A

- 第二字节为 **06H**时，所有能响应通用呼叫地址的从机器件复位，并由硬件装入从机地址的可编程部分。能响应命令的从机器件复位时不拉低SDA和SCL线，以免堵塞总线。
- 第二字节为 **04H**时，所有能响应通用呼叫地址并通过硬件来定义其可编程地址的从机器件将锁定地址中的可编程位，但不进行复位。

I²C总线的数据传送

- 如果第二字节的方向位为“1”，则这两个字节命令称为**硬件通用呼叫命令**。
- 在这第二字节的高7位说明自己的地址。接在总线上的智能器件，如单片机或其他微处理器能识别这个地址，并与之传送数据。硬件主器件作为从机使用时，也用这个地址作为从机地址。格式为：



- 在系统中另一种选择可能是系统复位时**硬件主机器件工作在从机接收器方式**，这时由系统中的主机先告诉硬件主机器件数据应送往的从机器件地址，当硬件主机器件要发送数据时就可以直接向指定从机器件发送数据了。

I²C总线的数据传送

引导过程由起始信号、起始字节、应答位、重复起始信号（Sr）组成。

- 请求访问总线的主机发出起始信号后，发送起始字节（0000 0001），另一个单片机可以用一个比较低的速率采样SDA线，直到检测到起始字节中的7个“0”中的一个为止。在检测到SDA线上的高电平后，单片机就可以用较高的采样速率，以便寻找作为同步信号使用的第二个起始信号Sr。
- 在起始信号后的应答时钟脉冲仅仅是为了和总线所使用的格式一致，并不要求器件在这个脉冲期间作应答。

SPI总线

* 主要信号

- (1) SCLK: 串行工作时钟, 由主设备发出;
- (2) MOSI: 主设备到从设备的数据传输信号;
- (3) MISO: 从设备到主设备的数据传输信号;
- (4) SS: 从设备选择信号, 由主设备发出。

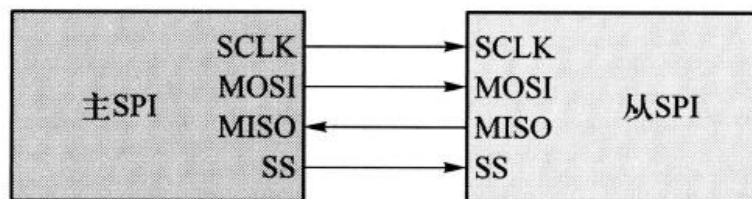


图 6-5-1 SPI 连接

SPI总线

* 多个从设备的连接方式。

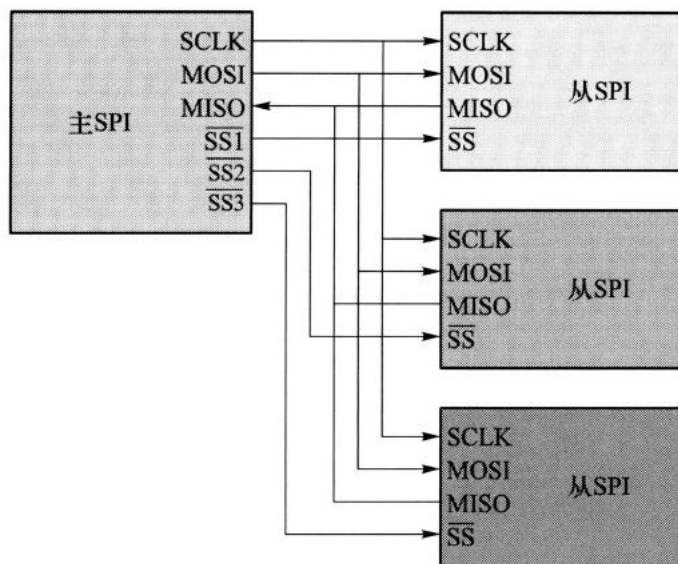


图 6-5-3 SPI 从设备独立连接

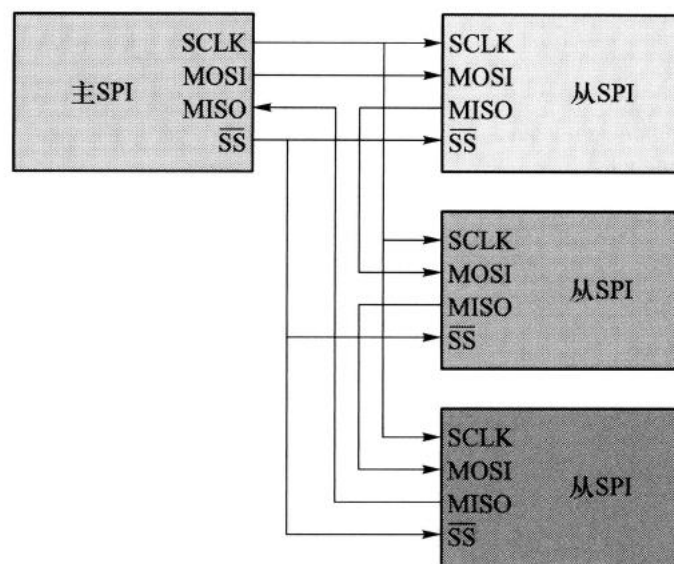


图 6-5-4 SPI 链式连接

ARM 总线

SoC总线：用于芯片内部

arm AMBA (Advanced Microcontroller Bus Architecture)

Key AMBA specifications

AMBA

AMBA 2

AMBA 3

AMBA 4

AMBA 5

CHI

Coherent Hub Interface

Credited coherent protocol
Layered architecture for scalability

CHI

ACE

AXI coherency Extensions

ACE is a superset of AXI – system-wide
coherency across multicore clusters

ACE
+Lite

ACE5
+Lite

AXI

Adv. eXtensible Interface

AXI supports separate A/D phases, bursts,
multiple outstanding addresses, OoO responses

AXI3

AXI4
+Lite, +Stream

AXI5

AHB

Adv. High-performance Bus

AHB supports 64/128 bit multi-managers
AHB-Lite for single managers

AHB

AHB
+Lite

AHB5
+Lite

APB

Advanced Peripheral Bus

System bus for low
bandwidth peripherals

APB

APB2

APB3

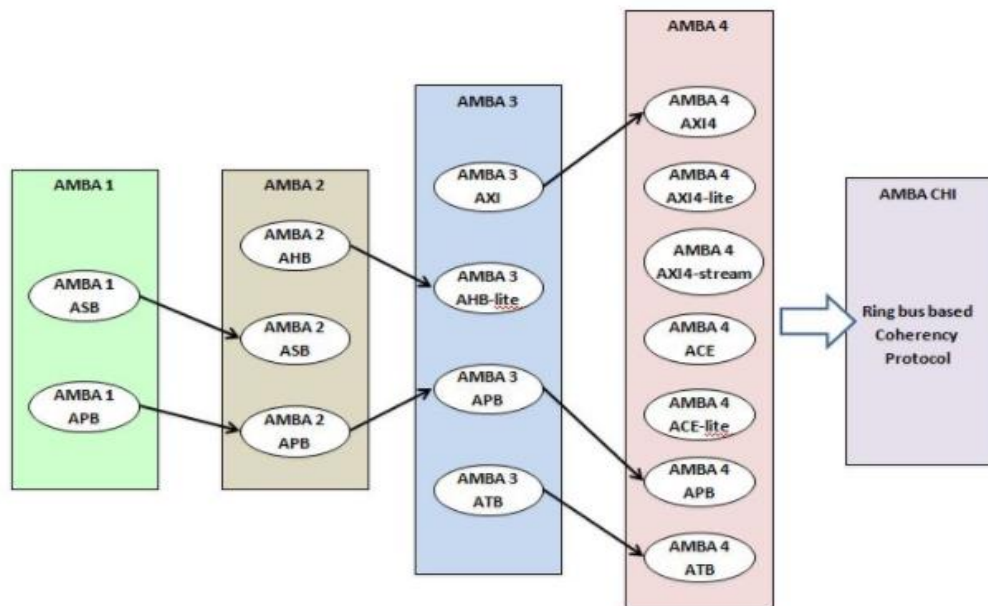
APB4

APB5

AMBA 协议的演进

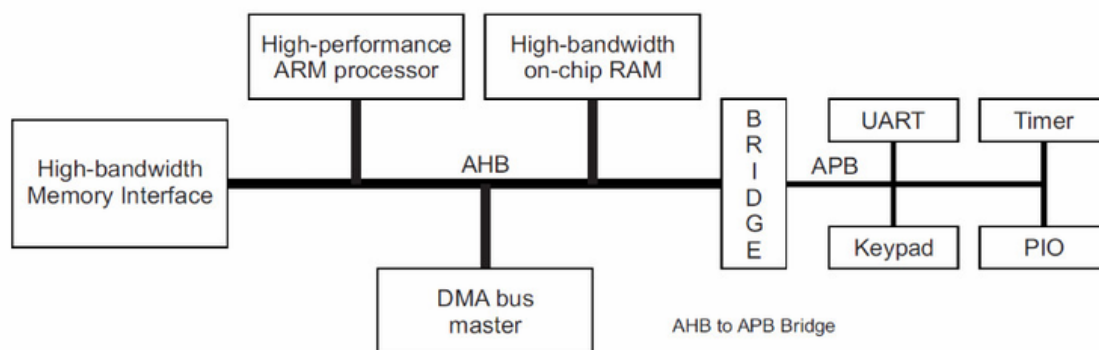
AMBA 1只有ASB和APB协议;

- AMBA 2引入AHB协议用于高速数据传输;
- AMBA 3, 为适应高吞吐量传输和调试引入AXI和ATB, 增加高级可扩展接口, 而AHB协议缩减为AHB-lite, APB协议增加了PREADY和PSLVERR, ASB由于设计复杂而不再使用;
- AMBA 4, AXI得到了增强, 引入QOS和long burst的支持, 根据应用不同可选AXI4, AXI4-lite, AXI4-stream, 同时为满足复杂SOC的操作一致性引入ACE和ACE-lite协议, APB和ATB也同时得到增强, 比如APB加入了PPROT和PSTRB, 另外为改善总线数据传输引入QVN协议;
- 适应更加复杂的高速NOC设计, 引入环形总线协议, 推出的AMBA CHI协议。



AHB/APB

将微控制器（CPU）、高带宽的片上RAM、高带宽的外部存储器接口、DMA总线控制器，以及各种AHB接口的控制器等连接起来构成一个独立的完整SOC系统，还可以通过AHB-APB桥来连接APB总线系统



AHB/APB

一个主设备与多个从设备的连接

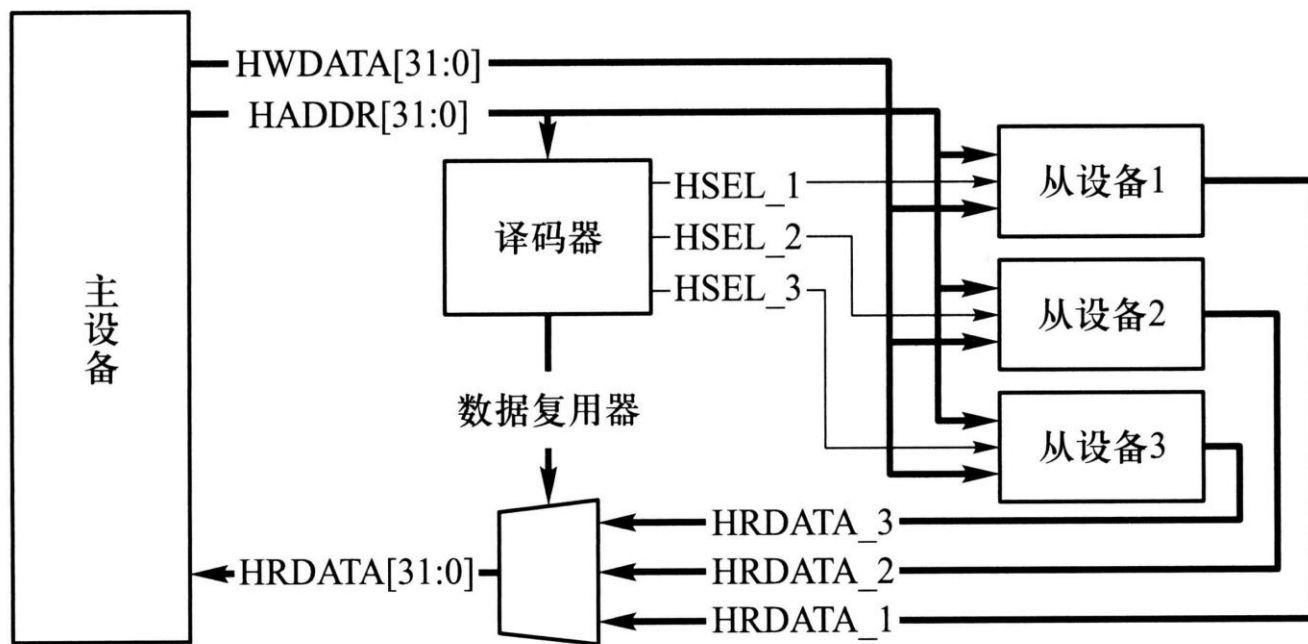


图 6-6-2 AHB 总线连接

AHB/APB

一个主设备与多个从设备的连接

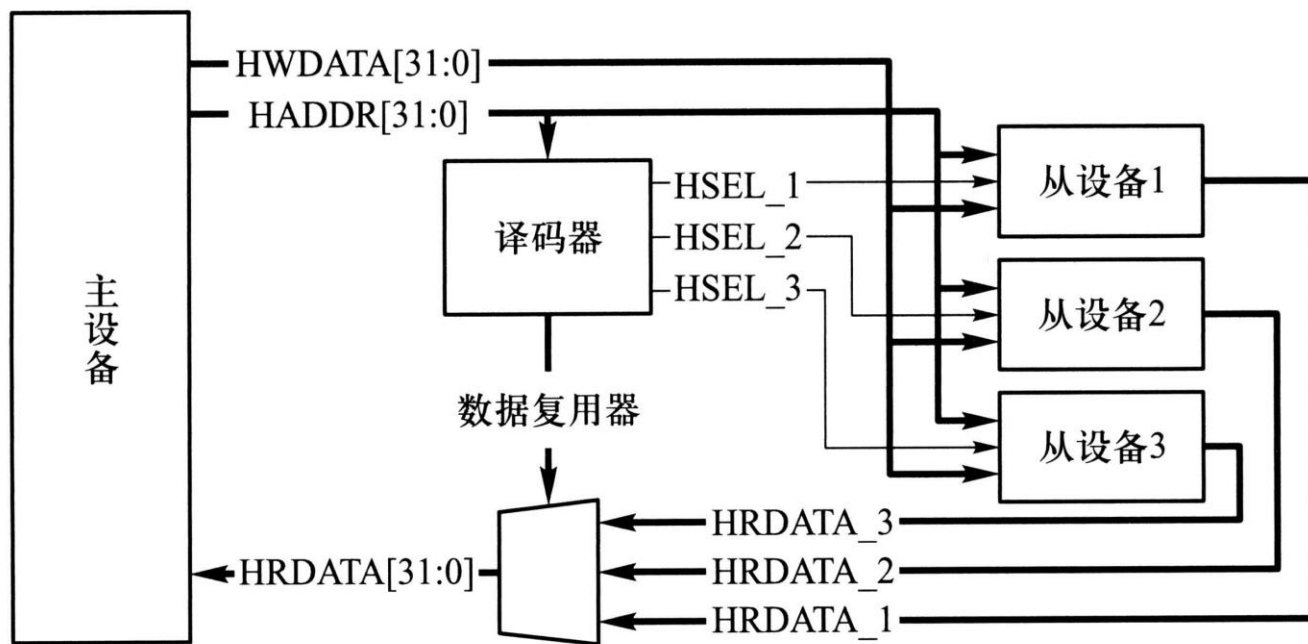


图 6-6-2 AHB 总线连接

AHB/APB

多个主设备与多个从设备的连接

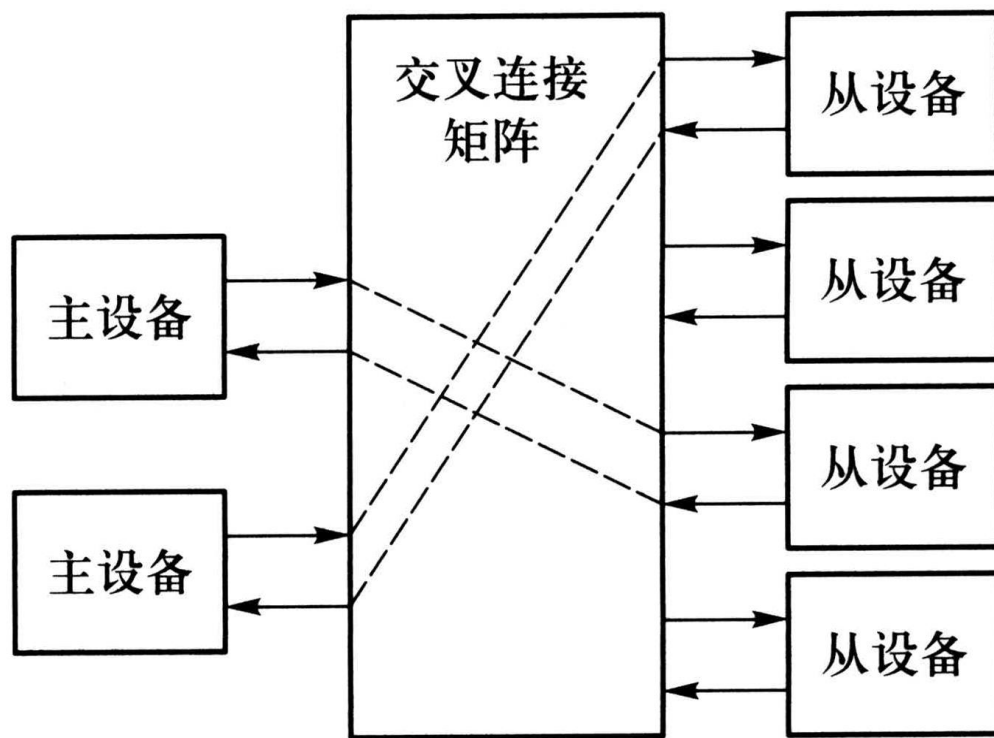
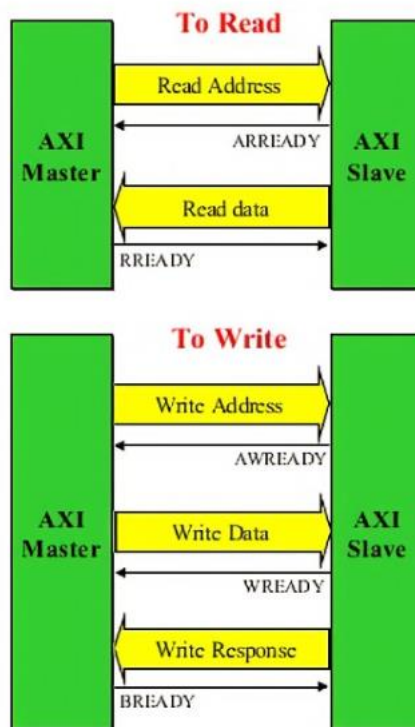


图 6-6-5 AHB 交叉连接

AXI

AXI: 高性能低延迟。点对点的连接



AXI

AXI-APB扩展

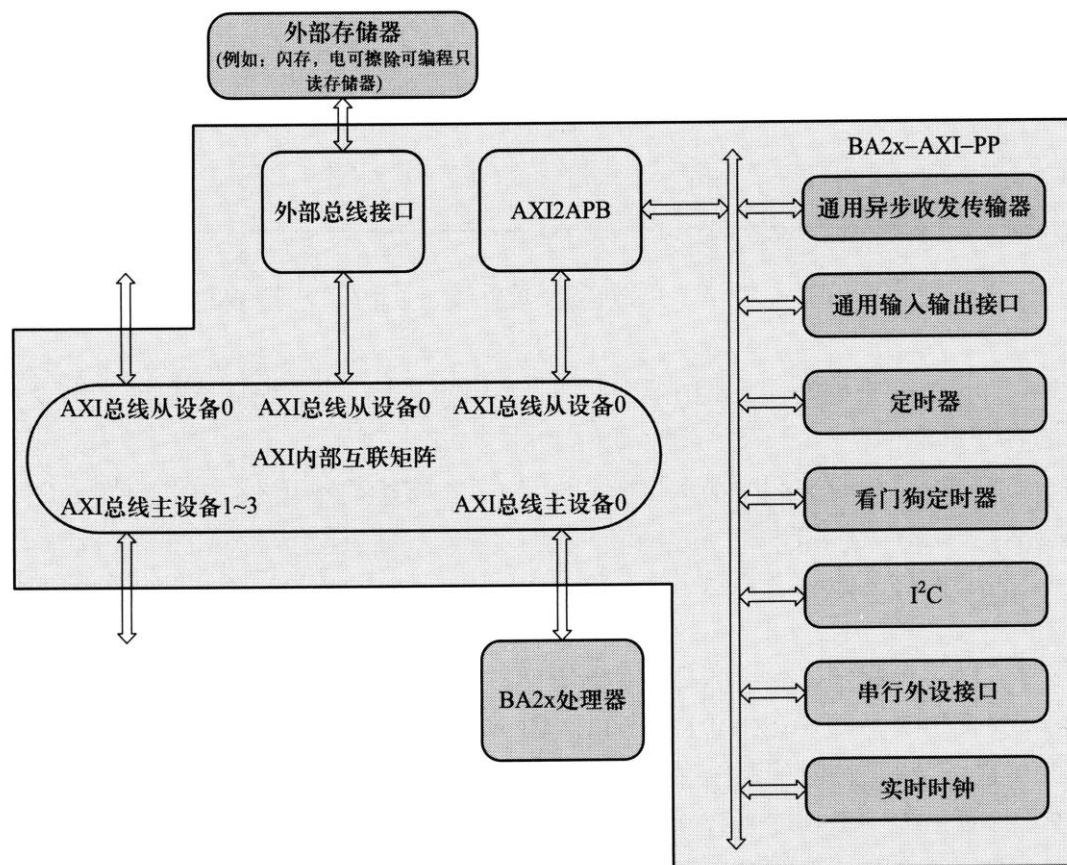


图 6-6-13 AXI-APB 扩展

AMBA总线的性能对比分析

总线	AXI	AHB	APB
总线宽度	8, 16, 32, 64, 128, 256, 512, 1024	32, 64, 128, 256	8, 16, 32
地址宽度	32	32	32
通道特性	读写地址通道、读写数据通道均独立	读写地址通道共用读写数据通道	读写地址通道共用读写数据通道 不支持读写并行操作
体系结构	多主/从设备 仲裁机制	多主/从设备 仲裁机制	单主设备（桥）/多从设备 无仲裁
数据协议	支持流水/分裂传输 支持猝发传输 支持乱序访问 字节/半字/字 大小端对齐 非对齐操作	支持流水/分裂传输 支持猝发传输 支持乱序访问 字节/半字/字 大小端对齐 不支持非对齐操作	一次读/写传输占两个时钟周期 不支持突发传输
传输方式	支持读写并行操作	不支持读写并行操作	不支持读写并行操作
时序	同步	同步	同步
互联	多路	多路	无定义



谢谢！

