

尤一鸣 傅景义 王俊省 编著

单片机总线扩展技术

北京航空航天大学出版社

单片机总线扩展技术

尤一鸣 傅景义 王俊省 编著

北京航空航天大学出版社

(京)新登字 166 号

图书在版编目(CIP)数据

单片机总线扩展技术/尤一鸣等编著. —北京:北京航空航天大学出版社, 1993. 12
ISBN 7-81012-426-9

I. 单...

II. 尤...

III. 微处理机-总线式结构

IV. TP368.1

内 容 简 介

本书是一本讲述对单片机系统进行扩展的书籍,介绍了有关单片机总线的基本概念、总线传输以及其它的相关知识,系统而详尽地介绍了当前最常见、应用最广泛的几种单片机总线系统,如STD总线、PC/AT总线、BIT BUS总线、I²C总线、MICROWIRE总线等;讨论了常用的总线接口标准;另外,列举了一些典型的系统扩展实例,旨在使读者对单片机总线及其扩展技术有一个深入的了解。

对于工程技术人员、从事单片机系统开发人员及单片机爱好者,此书是一本很好的技术资料。

单片机总线扩展技术

DANPIANJI ZONGXIAN KUOZHAN JISHU

*

尤一鸣 傅景义 王俊省 编著

责任编辑 冯学民

北京航空航天大学出版社出版

新华书店总店科技发行所发行 各地新华书店经销

朝阳科普印刷厂印装

787×1092 1/16 印张:13.125 字数:336千字

1993年11月第一版 1993年11月第一次印刷 印数:10000册

ISBN 7-81012-426-9/TP·092 定价:10.50元

前 言

随着工业生产的现代化和人民生活水平的不断提高,对于工业与民用产品的要求也越来越高,尤其对多功能化、智能化方面的要求更为突出。应用单片机是满足与实现这一要求的最佳方案。在国外,目前单片机的年产量数以亿计,其数量之大,应用之广,是任何其它类型计算机所不可比拟的。

单片机进入我国市场的时间还不足十年,但已被各个领域的工程技术人员所接受,并应用于各个领域之中。尤其在智能仪表、工业控制器、家用电器领域中,更受到青睐。

人们最初接触到的是少数几个品种规格的单片机芯片,它们具有4~8位CPU,少量存储器和输出输入接口。随着单片机技术的发展,越来越多的单片机芯片增强了通讯接口的功能,从而使单片机的最佳工作方式不仅仅限于单片或最小扩展方式,还具有总线通讯扩展方式。单片机已成为系统控制家族中的一员。

当前,各大公司生产的单片机芯片规格各异,总线接口硬、软件的联接与约定也不尽相同。本书的主要目的在于把当前常见、应用较广的几种单片机总线型式介绍给读者,并对其应用作典型说明。如飞利浦公司的I²C总线,英特尔公司的位总线,美国国家半导体公司的MICROWIRE总线和莫托罗拉公司的68系列单片机SPI总线。

通过总线通讯扩展,使分散的单片机控制设备非常方便地构成网络 and 控制系统,而且还可以和其它微机系统构成多级控制系统及管理与控制相结合的综合系统。由此更进一步扩大了单片机的应用范围。

另一个方面,由于用户系统往往需要扩充或变更部分功能以形成新的系统。在这种情况下,选择相适应的单片机品种、规格来变更系统是最佳的技术方案,但往往受条件、资金等方面的制约而不能实现。为此,通过总线扩展来增强部分功能是当前国内广大单片机用户需要采用的必然手段;而当单片机用于系统控制之中时,采用标准的总线接口则给进一步扩展带来许多方便。据此,本书也以一定的篇幅介绍了STD总线、PC总线、FD总线和常用的并行与串行标准接口。

由于作者水平与能力所限,错误与疏漏在所难免,谨希广大读者见谅,并望提出宝贵意见。

本书由天津纺织工学院尤一鸣、傅景义、王俊省同志编写,北京航空航天大学何立民同志审阅了全书,天津纺织工学院的赵可萍、王金海同志参加了书中应用实例的研制与调试工作。

编 者

一九九二年十月

目 录

前 言

第一章 绪 论	(1)
1.1 有关总线的一般概念	(2)
1.1.1 内部总线与外部总线	(2)
1.1.2 局部总线与系统总线	(2)
1.1.3 总线驱动	(3)
1.1.4 总线功能	(4)
1.1.5 总线占用权的判定	(5)
1.1.6 总线握手	(5)
1.1.7 总线约定或协议	(6)
1.2 同步总线传输与非同步总线传输	(6)
1.2.1 同步总线传输	(6)
1.2.2 异步总线传输	(8)
1.2.3 半同步总线传输	(10)
第二章 STD 总线系统	(12)
2.1 STD 总线结构	(13)
2.1.1 引脚定义	(13)
2.1.2 信号描述	(15)
2.1.3 机械特性	(18)
2.2 单片机 CPU 模板	(19)
2.2.1 8031 CPU 模板	(19)
2.2.2 8098 CPU 模板	(21)
2.3 单片机 STD 总线系统的组成和支持软件	(22)
2.3.1 系统的组成	(22)
2.3.2 支持软件	(23)
第三章 PC 总线与 AT 总线系统	(54)
3.1 PC 总线结构	(54)
3.2 AT 总线结构	(56)
3.3 8086/8088 CPU 和 80286 CPU 引脚功能	(58)
3.3.1 8086/8088 的引脚信号及功能	(58)
3.3.2 80286 的引脚信号与功能	(58)
3.4 8086/8088 CPU 与 PC 总线的系统连接	(61)

2 目 录

3.4.1	8282 锁存器与 8086/8088 的连接	(62)
3.4.2	数据收发器 8286 与 8086/8088 的连接	(63)
3.4.3	总线控制器 8288 与 8086/8088 的连接	(63)
3.4.4	PC 总线系统及其扩展应用	(66)
第四章 BIT BUS 位总线系统		(70)
4.1	BIT BUS 通讯规程	(72)
4.1.1	BIT BUS 通讯物理层(电气接口)	(72)
4.1.2	BIT BUS 通讯数据链路层	(73)
4.1.3	BIT BUS 通讯消息层	(77)
4.1.4	BIT BUS 通讯应用层	(79)
4.2	8044 硬件结构	(80)
4.2.1	串行通讯接口单元 SIU 结构分析	(81)
4.2.2	8044 特殊功能寄存器	(82)
4.2.3	SIU 运行状态	(85)
4.3	BIT BUS 应用设计	(86)
4.3.1	硬件设计	(86)
4.3.2	软件设计	(87)
第五章 I²C 总线系统		(90)
5.1	I ² C 总线结构	(90)
5.1.1	I ² C 总线硬件结构和术语	(90)
5.1.2	I ² C 总线的数据传送	(92)
5.1.3	I ² C 总线规范	(100)
5.2	I ² C 总线的应用	(103)
5.2.1	I ² C 兼容芯片的应用示例	(103)
5.2.2	通用微处理器与 I ² C 总线的接口	(104)
第六章 NS 公司单片机的 MICROWIRE 串行总线		(128)
6.1	概述	(128)
6.1.1	简介	(128)
6.1.2	MICROWIRE 总线的数据传送	(130)
6.2	COP400 系列单片机的 MICROWIRE 总线	(131)
6.2.1	硬件	(131)
6.2.2	软件	(133)
6.2.3	应用示例	(133)
6.3	MICROWIRE/PLUS 总线	(142)
6.3.1	MICROWIRE/PLUS 的操作	(142)
6.3.2	应用示例	(144)
第七章 MOTOROLA 单片机的 SPI/SCI 串行总线		(150)
7.1	SPI 串行总线	(150)

7.1.1 概述	(150)
7.1.2 SPI 接口电路及操作	(151)
7.1.3 SPI 应用示例	(155)
7.2 SCI 串行总线	(157)
第八章 FD-BASE 总线	(161)
8.1 FD-BASE 总线概述	(161)
8.2 FD-BASE 总线规范	(162)
8.2.1 信号定义	(162)
8.2.2 电气特性	(165)
8.2.3 机械特性	(166)
8.3 FD-BASE 总线模板	(167)
8.3.1 BASE 总线模板分类	(167)
8.3.2 FD-BASE 8801 CPU 板简介	(167)
8.3.3 BASE 8807-8097BH 主机板简介	(173)
8.4 FD-BASE 总线人-机接口	(175)
8.4.1 BASE 8807 中西文 CRT 接口板	(175)
8.4.2 BASE 8851 键盘显示板	(179)
第九章 常用总线接口标准	(180)
9.1 串行总线接口标准	(181)
9.1.1 RS-232-C	(181)
9.1.2 RS-449 与 RS-423/422/485	(183)
9.1.3 20 mA 电流环	(186)
9.2 并行总线接口标准	(187)
9.2.1 IEEE-488 并行总线接口标准	(188)
9.2.2 IEEE-488 总线接口电路	(194)
附录	(199)
参考书目	(201)

第一章

绪 论

单片机是把微型计算机的主要组成部分集成在一块芯片上，因而其具有体积小、功能强、集成度高、抗干扰性能好等优点。在工业、农业、通讯、交通、家用电器等各个领域中被广泛应用作为智能控制器件。因而单片机又称作单片微控制器（Single Chip Micro-Controller）。

由于被控制的对象多种多样、千差万别，便形成了多种类型的应用系统，然而不可能有相应类型的单片机与之一一对应。因此，构成一个完善的应用系统时，不可避免地要在一个单片机的基础上进行适当的外部扩展，从而满足应用系统的要求。常见的单片机外部扩展包括：存储器扩展，I/O 接口扩展，定时/计数器扩展以及其它特殊功能的扩展。

在另一些情况下，若干个单片机控制的相对独立的应用系统之间，或者某些单片机控制的应用系统和由其它微型计算机控制的应用系统之间，需要进行某些信息的相互传送。这样，就不可避免地出现了单片机与单片机之间，或单片机与其它 CPU 之间的通讯问题。

无论是系统外部扩展还是片间通讯，当被传送的信息按照单片机或 CPU 数据线的位数同时进行传送时，称做并行传送或并行传输；当被传送的信息按照一定的速率按位传送时，称做串行传送或串行传输。

在微型计算机系统中，信息传输的通路通常利用“总线”结构。总线是微型计算机系统中模块到模块之间传送信息的一束信号线。单片机的信息传输也不例外。当采用并行传输时要应用并行总线，当采用串行传输时要应用串行总线。被传输的信息可以是数据信息、地址信息或命令控制信息，为了使信息发送部分和信息接收部分能进行完整、正确地通讯，不产生信息丢失或出错，需要对总线的工作进行管理，即对硬件和软件做必要的规定。所谓硬件的规定包括总线接口引脚的定义，速率的设定，驱动能力的限制等等；所谓软件的规定包括时序的安排，信息的格式约定等等。

在微型计算机应用技术发展的过程中，已经创建了一些性能良好、被广泛应用的总线格式。如 STD BUS, MULTI BUS, PC BUS, S-100 BUS, Z-BUS 等等，它们都具有一定的标准规格和软件约定，从而成为可通用的总线系统，为用户进行系统扩展和通讯带来了方便。单片机的外部扩展和通讯，除同样可以应用这些标准总线系统外，还有近年来国内外新创建的、应用于单片机系统中的 I²C BUS, CAN BUS, BASE BUS, TF BUS 等非标准总线。

在本章内，我们将对总线的功能和有关概念进行必要的阐述。在随后的各章内，将分别讨论各种不同的总线格式，并介绍其应用于单片机系统扩展及通讯中的设计方法与实例，以便使广大单片机用户能选择恰当的总线格式为自己的应用系统服务。

1.1 有关总线的一般概念

1.1.1 内部总线与外部总线

总线是计算机系统中模块到模块之间传送信息的一束信号线,在单片机的内部和外部都要用到。我们把芯片内部各个模块间传送信息的总线叫做内部总线,把芯片外部各个模块间传送信息的总线叫做外部总线。

单片机芯片内部信息传送时,一般采取分时占用总线的方式,也就是说内部总线上所传送的信息在不同的时刻可以是不同类型的信息(即数据信息、地址信息和控制信息分时占用同一组总线),这种结构叫单总线结构。通常在芯片内部,都采用并行单总线结构,使数据信息的各个位同时传送,以提高传输速度。

在芯片外部使用的总线结构是多种多样的,但统称外部总线。通常在距离较远,要求速度不甚高时,采用串行总线。这样,数据信息的传送仅需一条或两条线路即可(当发送的同时还要接收,则需两条线)。另外再通过一条或几条控制信息传送线路,保证传输工作的正确和可靠(具体介绍在随后各章内阐述)。当距离较近或要求速度较高时,往往采用并行总线,尤其是以单片机芯片作为 CPU 构成一个微机系统的时候,通常采用标准结构的并行总线形式,即数据信息、地址信息和控制信息各自独立使用各自的总线,称为三总线结构。

1.1.2 局部总线与系统总线

在一个较复杂的单片机系统或多个单片机组成的复杂系统中,由于扩展了较多的存储器、I/O 接口,或是多个单片机之间频繁地进行通讯联络时,共用一组外部总线就容易造成总线的堵塞,从而降低系统的吞吐率。较为理想的办法是将这样一个复杂系统,划分为若干个功能相对独立的模块,每个模块建立自己的总线叫做局部总线。各个模块之间建立联系总线叫做系统总线。在局部总线上可以挂有局部存储器和局部 I/O 接口,而系统总线上挂有公共存储器和公共 I/O 接口。这样,很大一部分存储器读写操作或 I/O 操作是通过局部总线来完成的。只有访问公共存储器和公共 I/O 接口时,才用到系统总线,从而避免了系统堵塞现象。与此同时,也为各个模块并行工作提供了条件。

当前,单片机系统设计中,为了便于用户扩展和与其它 CPU 联机,或者共享某些硬件模块使之具有兼容性,以减少单片机用户的硬件设计任务,常常采用规格化的通用系统总线结构。将其做成插槽(如 PC BUS, STD BUS)或接口插头(如 RS-232-C, IEEE-488)形式。在这样的设计中,必须遵照各个规格化总线结构的规定、定义和协议标准,否则将导致系统不能正常工作甚至发生短路事故。图 1-1 表明了局部总线 and 系统总线之间的关系。

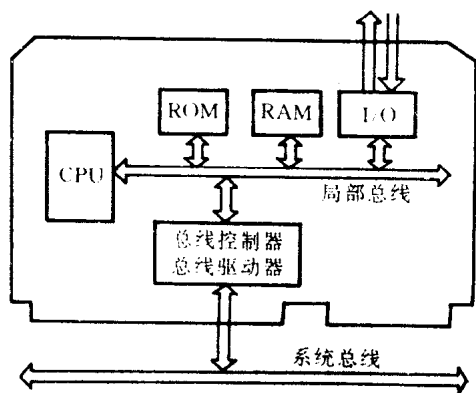


图 1-1 局部总线 and 系统总线

1.1.3 总线驱动

任何一个芯片工作时, 都要求一定的工作电流, 而任一种单片机芯片的任一引脚, 能够提供的输出电流是有限的。如 MCS-51 系列单片机, 其 P_0 口可以驱动 8 个 LS TTL 门电路, 其它端口仅能驱动 4 个 LS TTL 门电路。由于通过系统总线可以扩展的外部模块在理论上是无限的, 因此, 当系统过大、负载过重时, 将导致驱动能力不够, 系统不能正常工作。为了保证系统总线具有一定的驱动能力, 常规设计中, 在局部总线到系统总线之间, 需要设立总线驱动器。在图 1-1 中, 已示出了这一环节。

依据局部总线 and 系统总线间的关系, 选用单向驱动器或双向驱动器, 但无论是单向或双向驱动器, 必须具有三态输出功能。

常用的单向总线驱动器有 74LS240、74LS241 和 74LS244, 其中尤以 74LS244 最为常用。图 1-2 为 74LS244 引脚及在 MCS-51 单片机系统总线扩展中做高八位地址总线驱动器时的管脚联接。

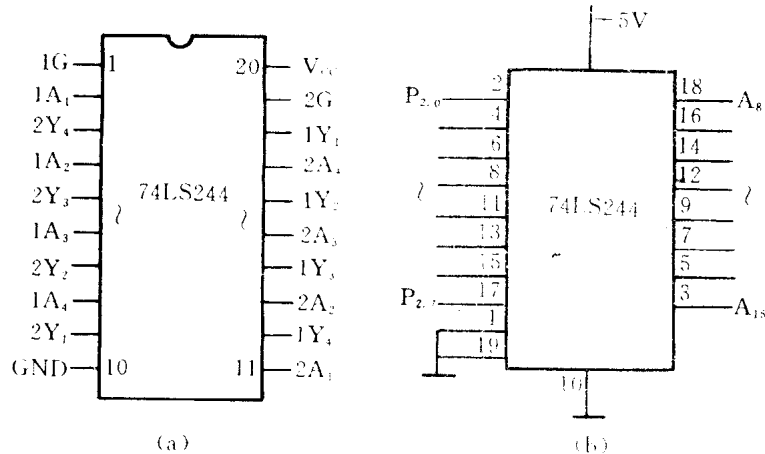


图 1-2 单向总线驱动器 74LS244

由于 MCS-51 的 P_2 口只作为高八位地址输出, 因此 74LS244 的驱动门控制端 $\bar{G}_1$ 、 $\bar{G}_2$ 接地, 以便保持对总线的单向输出驱动, 其逻辑输出状态, 由 P_2 端口状态决定。

常用的双向总线驱动器有 8286 和 74LS245。图 1-3 为 74LS245 引脚及其在 MCS-51 单片机系统总线扩展中作为 P_0 端口数据输入/输出和低八位地址输出复用的双向驱动器。

74LS245 的门控信号 $\bar{G}$ 为低电平有效, 双向控制端 DIR 为高时, 执行输出驱动, DIR 为低时, 执行输入驱动。工作时, 使 $\bar{G}$ 接地, 保持 74LS245 处于工作状态, DIR 引脚由单片机 MCS-51 的数据存储器“读”控制引脚($\overline{RD}$)和程序存储器的取指令控制引脚($\overline{PSEN}$)通过与门控制。当 $\overline{RD}$ 或 $\overline{PSEN}$ 有效时, 使 74LS245 执行总线输入驱动, 除此之外, 都执行输出驱动。图 1-4 示出了以 MCS-51 单片机为例的总线驱动原理图。

和上述单向驱动器的原理一样, 在执行输出驱动时, 其逻辑输出状态由 P_0 端口的状态决定。

在远距离信息传输时, 一般采用串行传输方式, 通常只用 1~2 条信息传输线路。为了使信号在传送途中不因线路损耗造成的衰减而出现信息接收错误, 也为了能更有效地在传输过

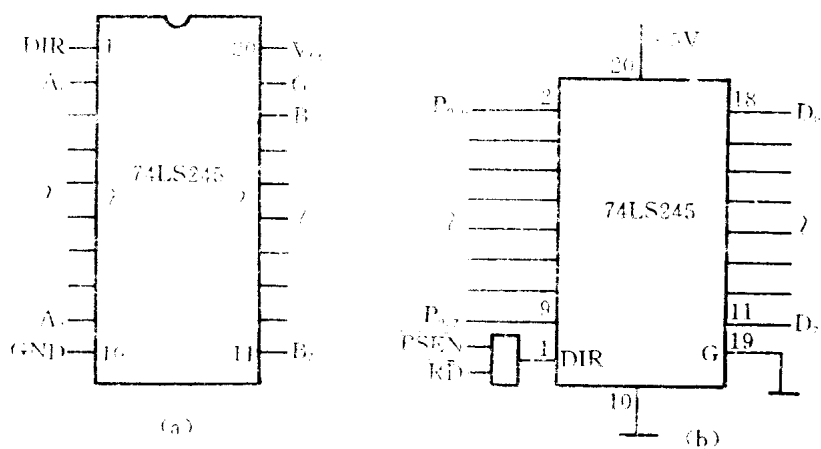


图 1-3 双向总线驱动器 74LS245

程中抵制各种噪声干扰, 需要设置驱动环节。如在 RS-232C 中采用 1488 芯片作为电平转换器件, 将 5V 信号转为 12V 传输, 还有的采用电流环等方式, 这都是为了有效地增大总线驱动能力。在随后各章中, 将分别对此作具体介绍, 此处不再赘述。

1.1.4 总线功能

我们将通过总线向目标模块传递的信息叫做基本信息, 它们可能是地址信息、数据信息或命令信息。为了保证在总线操作期间, 这些基本信息得到准确无误地发送、接收和识别, 就需要在总线操作期间增加一组辅助信息。这些辅助信息可能是时钟信息、

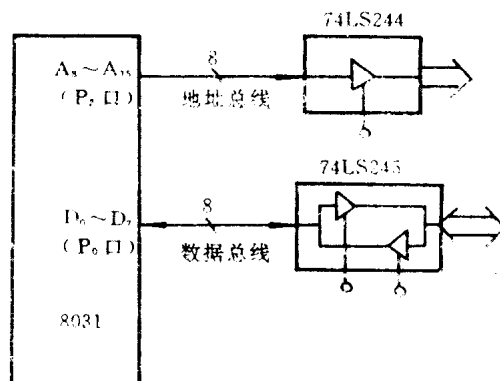


图 1-4 MCS-51 单片机总线驱动原理图

读写控制信息、状态就绪否信息及由谁占用总线的判断信息等。在不同的情况下, 所需要的辅助信息可以包含上述内容的一部分或全部乃至更多的辅助信息。这些辅助信息, 可以通过总线系统中的一组专用线来传送, 也可通过传送基本信息的线路附加在基本信息的前面或后面来传送。在并行总线系统中, 通常采用一组专用线实现; 而在串行总线系统中, 采用专用线和通过基本信息前、后附加辅助信息相结合的方式实现。总之, 总线的功能在于在辅助信息的协调下, 实现对基本信息正确传输。

为了保证总线功能的实现, 必须十分注意以下两个方面:

一、连接在同一系统总线上的若干模块, 在任一时刻最多只能有一个模块享有向总线输出的权力, 即向总线执行“写”操作, 此时, 其它所有模块, 只能处于“读”或高阻浮空状态, 否则, 将出现多个模块同时向系统总线的某一条位线进行信息输出的情况, 导致系统不能正常工作, 甚至发生短路事故。这种情况称为总线冲突。为了保证总线的正常工作, 需要从硬、软件两方面制定相应的措施, 通常称为总线裁决。

二、规定信息的传输形式。这里包含传输的速率, 写操作与读操作控制和数据信息的配合, 传输开始和结束的标志, 以及对信息处理的有关规定等等。在同一系统总线上, 只有这

些规定取得所有模块的一致遵从,才能保证信息传输的正常进行,即实现总线应有的功能。在随后各章中,将分别阐述各种总线的有关规定,在此,我们首先对共同性的一般概念加以说明,以便对总线功能的实现有较明确的认识。

1.1.5 总线占用权的判定

若干个共用同一总线系统的模块,若需要同时从总线上读取其传送的信息,此时仅仅是加重了总线的负载,而不影响总线上的信息状态,因而没有总线占用权的判定。

当在一个总线系统上,只有一个单片机(或 CPU)作为主控元件时,总线的占用权由该单片机独享(只有在 DMA 方式下让出总线)。当在一个总线系统上,具有两个或两个以上主控元件时,必须通过总线占用判断来决定哪个操作应在总线上进行。这种判断的必要性是由两个或两个以上模块企图同时通过总线送出信息(写)引起的。若模块 A 向总线送出一个逻辑“0”,而模块 B 在同一时刻向同一条线路送出一个逻辑“1”,于是就产生了总线冲突。在冲突的时候,一个或两个以上模块就会丢失数据,从而导致传输故障。

实现总线占用判断的方式很多,最简单的方式是“菊花链”方式,如图 1-5 所示。

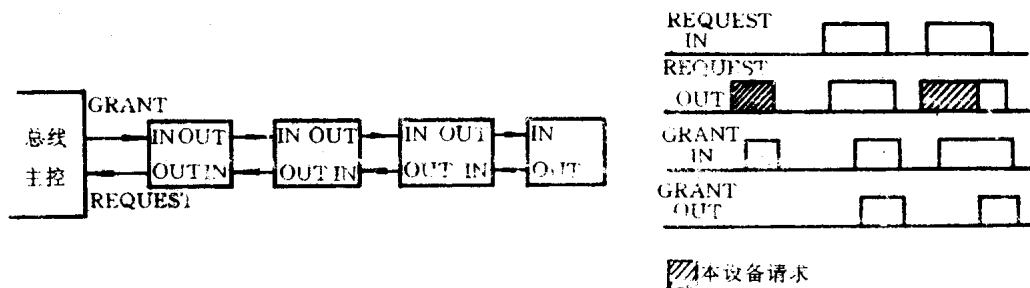


图 1-5 菊花链总线判定

菊花链总线判定的原理是:由判决器发出一个 GRANT 信号,作为对 REQUEST 信号的回答,这个信号按顺序传送,经过其它所有的总线主控部分。接收到 GRANT 信号的第一个请求占用总线的模块,便取得了对总线的控制。在这个模块已经控制了总线的情况下,它将不再把 GRANT 信号往总线上的下一个模块传送,因此,其它模块就不可能对总线进行操作。

此外,还有许多其它的判定方式,如后边要介绍的 I²C 总线占用判定等,在此不一一赘述。由此可见,对于在同一总线上有多个单片机的多机系统,总线判定是必不可少的环节。

1.1.6 总线握手

总线握手信息是总线辅助信息中用于控制基本信息传送同步的信号。它包含控制基本信息传送开始和结束信号两个主要部分。

由于握手信息的主要功能是表明基本信息传送的开始和结束。因此,握手信息必须以某种形式的电位变化来标志这些时刻。有时,在一组总线上传送的基本信息具有非常复杂的意义,因此要求其具有相应复杂的时序同步控制,即能够标明整个总线操作周期的起始与结束,并能标明在此期间的各个子周期的开始与结束。

最简单的握手信号是采用系统时钟信号,以时钟的上升沿和下降沿来标明一个总线周期的起始与结束。但有许多复杂情况,以这种简单的同步信号作为握手信号不能满足要求,因

而设计出多种不同的非同步握手信号,以满足传输的需要。

总线握手信息还用在对条件状态是否就绪的识别中,以便有效地保证在发送方发送信息时,接收方已做好接收准备。或者说,只有在接收方已做好了接收准备的条件下,发送方才进行发送操作,并在确认信号已被接收的情况下,才结束这个操作周期。

1.1.7 总线约定或协议

总线约定或称总线协议包含了诸多方面的内容,但其作用归结为一点就是保证对传送的信息进行正确识别和可靠、无丢失的接收。

首先涉及到的是传输速率。对于串行总线,要保证发送的位速率(波特率)和接收的位速率严格一致;对于并行总线,则要保证前一次发送至下一次发送之间的间隔符合接收的要求。在这里,为了协调快速的单片机与慢速的外部接口器件间的速度差异,经常采用总线缓冲技术,即设立发送或接收缓冲器以协调二者之间的速度。

其次涉及到的是传输格式,包括基本信息的格式和辅助信息的格式。当信息格式符合预先的约定时,被认定执行某种操作或某种识别。例如在多个主控元件的总线传输系统中,当总线系统上的任意两个模块之间进行信息传输时,首先要通过判断信息的格式确定总线是否已被占用,再通过握手信息的格式认定起始信号,然后才能进行实质的传送。

格式还被广泛用于对传送内容的纠错和检验,如累加和校验、奇偶校验都应用预订的格式来进行,在一组信息发送完毕之后,附加送出对这一组信息的一个校验信息。接收方在接收了全部信息(包含校验信息)之后,按照预约的格式,对基本信息进行相应的校检处理,并将校验结果与发送来的校验信息比较,若一致,即表明发送与接收是正确的。由此可见,此时格式必须严格一致。

1.2 同步总线传输与非同步总线传输

系统总线的主要任务是使各模块之间进行可靠的数据传输。在系统总线上,一个传输周期一般分为四个阶段:

1. 申请阶段:要求占有总线的若干模块提出申请,由总线裁决部分确定把总线的使用权授给哪一个模块。当系统总线上只有一个主控模块时,则无需这一阶段。
2. 寻址阶段:总线上传送的基本信息是地址信息,以启动另一个或几个从属模块,使之参与下一阶段的数据传输。
3. 传数阶段:主模块和从属模块之间进行数据传输,总线上的基本信息是数据信息。
4. 结束阶段:占用总线的主模块将传输信息从系统总线上撤除,让出总线。

在完成一个传输周期的过程中,依据基本信息和辅助信息的时序关系,可以分为同步传输、异步传输和半同步传输三种模式,目的是:保证主控模块和从属模块的协调配合,最经济有效地完成传输任务。

1.2.1 同步总线传输

在系统总线上,模块间的传输周期(完成一次传输所占用的时间)是严格固定的。传输过程中,时钟脉冲是唯一的控制各项传输操作的标准。这种总线传输方式叫同步总线传输。

如图 1-6 所示, MCS-51 系列单片机 8031 执行指令 `ADD A, #DATA` 时, 要由程序存储

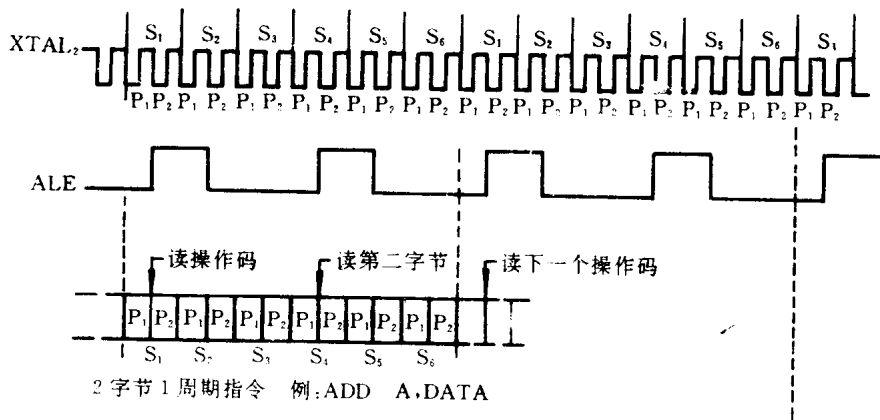


图 1-6 MCS-51 的取指/执行时序

区内取入两个字节代码, 占用一个机器周期。每个机器周期固定由 12 个时钟周期 P (或 6 个 S) 所组成, 且读取第一个字节代码发生在时钟周期 S_1 的 P_2 处, 读取第二个字节代码发生在时钟周期 S_4 的 P_2 处是严格不变的。为此, 单片机 8031 外部扩展的程序存储器工作速率必须满足主控模块单片机 8031 的要求, 此时, 程序存储器作为系统总线上的从属模块。只有符合这一要求, 才能保证传输正确。

由此可见, 同步总线传输的基本特点就在于每个时钟周期所执行的操作都有明确不变的规定。主控模块和从属模块之间的时间配合是强制同步的, 必须在限定的时刻完成所规定的操作, 且对所有的从属模块都是统一的。

在同步总线传输过程中, 要使读、写操作控制与数据状态协调一致以保证传输正确, 就要求数据状态具有一段相对稳定的时间。如图 1-7 所示。图中抽象绘出了对系统总线上存储器进行读、写两种操作时, 地址信息、数据信息与时钟周期 T 的定时关系。

图示表明: 若在时钟周期 T 的下降沿处对存储器执行“写”操作, 在此之前, 地址信息与数据信息均应稳定在相应的地址总线和数据总线上, 直至“写”操作完成之后, 即 T 的下降沿之后的一段时间, 地址信息 and 数据信息才可从各自总线上撤除。图中阴影部分是由于信息传输过程中产生的各种延迟造成的, 如译码时间, 缓冲器上逻辑建立所需的时间等等。

图中还表明了, 在“读”操作时, 数据状态稳定时间较短, 且到来也迟 (相对于“写”操作)。这是由于“读”操作是在地址信息有效译码选通之后才引起数据状态的有效, 而“写”操作的数据信息是由主控模块提供的, 可提前“放置”到数据总线上。

同步总线传输的优点体现在: 规定明确、统一, 模块之间的配合简单、一致。它的缺点在于对所有模块都强求采用同一时限, 使设计缺乏灵活性, 从而对外部器件要求相对比较严

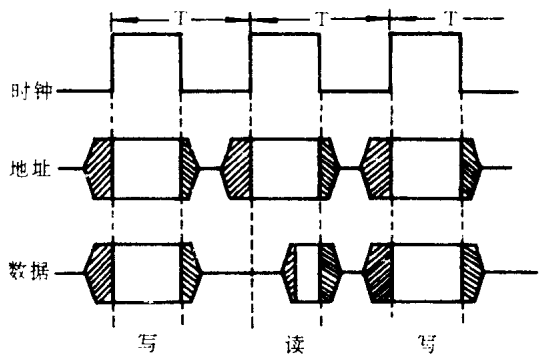


图 1-7 同步总线传输的定时关系

格甚至苛刻。有些器件有时表现在某一方面或某一操作速率不能满足要求,而在其它方面或其它操作上尚可以满足主控模块的要求时,往往造成设计者在系统设计时发生失误,错认为该器件可用或不可用。例如,使用外部可编程并行接口芯片 8155 做为 MCS-51 单片机的外部扩展接口芯片时,8155 在读、写操作时序上(输入或输出)均能和 8031 保持同步方式传输,但 8155 芯片的复位操作却不能和 8031 保持同步。于是当设计的用户系统需要使用 8155 作为输入/输出接口,且上电复位之后立即对 8155 进行方式设定及读、写(输入/输出)操作时,往往会失败。而在主控模块 8031 完成了某一操作,相对延缓了一段时间之后,再对 8155 进行操作,一般都会得到满意的结果。这样的例子还有许多,因此当采用同步传输时需要特别引起注意。

1.2.2 异步总线传输

同步总线传输方式要求总线上的所有模块都必须采用同一的速率,对不同速率的模块适应能力很差。异步总线传输方式则克服了这一缺点,允许各个模块有灵活的选择余地,即快速的模块可以快速地完成了读、写命令,而慢速的模块可以用较慢的速率来完成读、写命令。协调这一关系的关键环节是通过“请求”和“应答”两条信号线来实现的。

下面分别按读、写两种操作来分析异步总线传输的时序关系。

设主控模块 A 为高速模块,从属模块 B 为低速模块,按并行传输异步方式传送数据。

当主控模块 A 要由从属模块 B 中读出某个数据时,主控模块先送出地址信息和读操作请求信息(设为低电平有效),在总线上的所有模块经地址译码等判断之后,B 模块被选中。B 模块送出数据到数据总线上,在数据送出至数据稳定这一阶段完全按照 B 模块的速率进行,直至数据在数据线上稳定之后,B 模块才发出应答信息至应答线上(设为低电平有效)。主控模块 A 收到应答有效信号后,执行读操作,从数据总线上读出数据,并使读操作请求失效(变高)。读操作请求失效使 B 模块应答信号失效(变高),表明 B 模块已“明白”送至总线上的数据已被取走,从而结束了一个总线传送周期。由于读操作请求和读操作信号在时序上可以完全同步,因而也可简化为以读操作信号代表读操作请求信号,图 1-8 示出了上述过程的时序关系,并以①②……⑦表明了其相互作用的顺序。

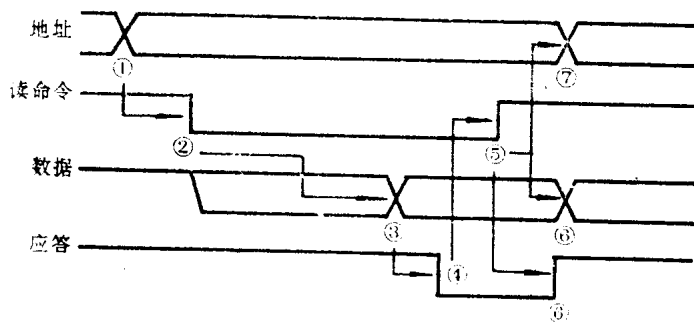


图 1-8 异步总线传输读操作时序

写操作的异步总线传输与读操作有类似之处。如图 1-9 所示。

地址信号总是由主控模块提供的。此时主控模块 A 送出的操作请求为写操作请求,由于所写的的数据由主控模块提供,因而可以在地址信息送出之后,“提前”将数据信息送至数据总

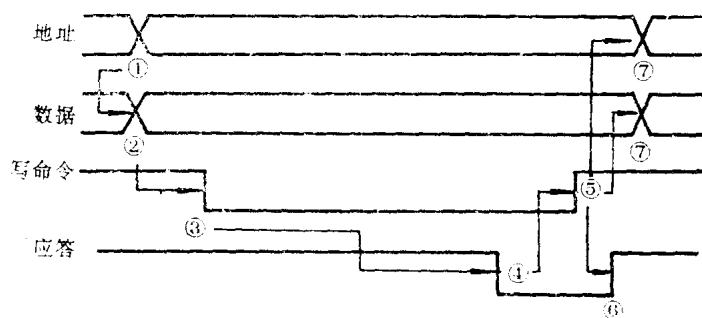


图 1-9 异步总线传输写操作时序

线上, 但此时慢速的 B 模块并不能同步接受这一写入操作, 在 B 模块经地址译码判断选通之后, 发出应答信号(低有效), 表明已作好接受写操作的准备。主控模块 A 收到应答有效信号后, 执行写操作, 然后将写操作请求变高使之失效。此处以写信号代表写操作请求, 即在请求信号失效的同时, 实现了用其后沿(上升沿)锁存数据的操作, 从而完成了一个异步总线传输的写操作周期。

由以上可以看出, “请求”与“应答”是成对出现的, 两条线上的信号电平变化相互牵制, 互为因果, 一条线状态的变化引起另一条线的变化。如上述两种操作中, 若“应答”信号线在有效之后, 没有恢复高电平变为失效之前, 主模块就不可能发出下一个请求信号。对这种关系, 称为互锁关系。这种异步总线传输方式, 称为全互锁异步总线传输。

我们将全互锁控制信号描绘为如图 1-10 的所示关系。

全互锁异步总线传输的优点是可以根据各个模块的速率不同, 自行控制传输速率, 从而使总线上的传输周期可变。其缺点是控制请求与应答信号要往来四次传送, 才能完成一个总线操作周期, 耗费许多时间。因而又出现了一种部分互锁异步总线传输方式。如图 1-11 所示为部分互锁异步总线传输控制信息关系, 在这里我们不再详细对它的传输过程进行分析。其特点是每个控制信号(请求或应答)只有一个有效沿, 而其另一个沿是无效的, 从而可以减少控制请求与应答信息传送的次数。但由图可知: 当主控模块发出请求信号且得到应答之后, 即由主控模块自行安排随后的操作。而不再“听取”应答状况。而下一个总线周期的起始, 也由主模块决定。这样, 有时会造成传输错误。

通过上述分析可知, 全互锁的数据传输是非常可靠的, 而部分互锁则未必。因而全互锁方式被广泛采用。

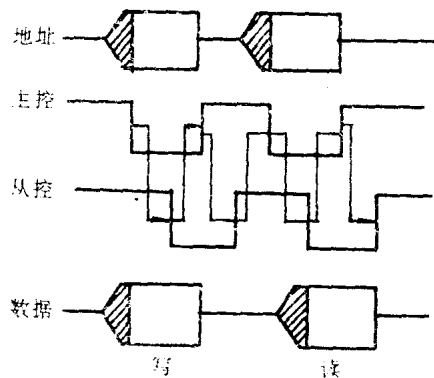


图 1-10 全互锁异步总线传输控制信号

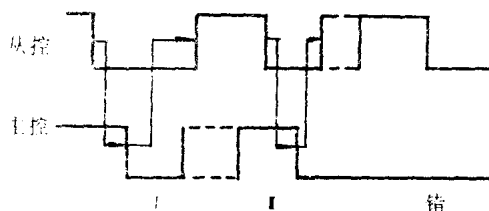


图 1-11 部分互锁异步总线传输控制信号

1.2.3 半同步总线传输

同步总线传输较为苛刻的要求,使得非同一时钟频率的模块不能按同步总线传输方式工作。而异步总线传输又要求主控模块与从属模块之间,不加区分地以“请求”和“应答”来做为传输联络信号,增加了传输线路,也增加了传输的附加时间。半同步总线传输是介于二者之间的一种折衷方式,它保留了同步总线传输的基本特点:对地址、命令或数据的发出时刻,都严格参照系统时钟脉冲规定“某个”脉冲的前沿或后沿的发出,并且也规定了在“某个”时钟脉冲的前沿或后沿进行判断、检测或接收。从整体上来看是一个“同步”总线传输。然而,这里所用的“同步”是被改变了速率后的伪同步,也就是半同步。

具体处理方式是增加一条附加信号线 $\overline{\text{WAIT}}$ (等待)信号。下面通过一个例子来说明。

若系统总线上主控模块A在 T_1 时刻发出地址信息, T_2 时刻发出命令信息, T_3 时刻进行数据传输, T_4 时刻结束;但接受这一传输的从属模块B工作速度慢,无法响应这一速率传输,因而必须在接到主控模块A的地址、命令信息之后,在 T_3 时刻之前通知主控模块A请其等待($\overline{\text{WAIT}}$),直至模块B做好发送或接收传输数据(“读”或“写”,依命令而定)的准备之后,再撤除等待($\overline{\text{WAIT}}$)信号。这样,主控模块A发出地址、命令信息之后,不是接着进行 T_3 时刻的数据传输,而是采样($\overline{\text{WAIT}}$)信号,若当等待信号有效时,则自动插入等待周期,直至等待信号失效,再执行 T_3 时刻的数据传输。这样,整体上是 $T_1 \sim T_4$ 的所谓同步传输,中间加入了等待信号的异步传输,但又不以互锁信号相联系,而把由“请求”和“应答”两条联络信号传递的是否就绪信息改用一条“等待”信号作单方向的状态传递,使之按延长传送周期的方式进行数据传输,从而实现了半同步总线传输。

图 1-12 示出了半同步传输总线时序状态。

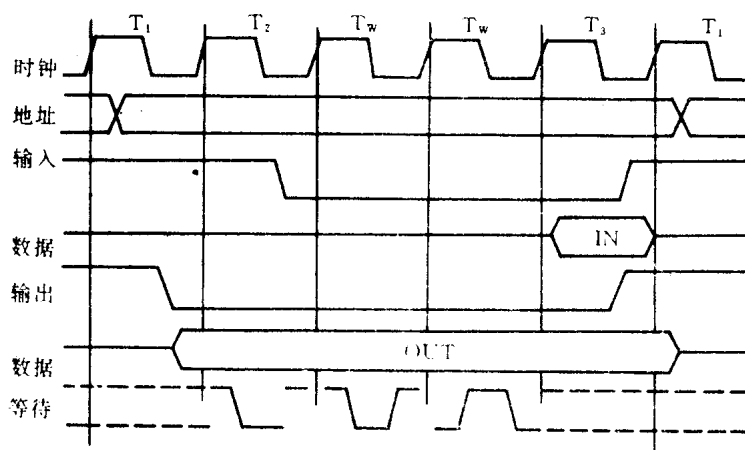


图 1-12 半同步总线传输时序示例

按照半同步总线传输方式,在系统总线上能够和主控模块采用相同速率进行数据传输的从属模块,可以实现真正的同步总线传输。而对于那些慢速的从属模块,则可以通过($\overline{\text{WAIT}}$)等待信号,强制主控模块延迟操作,插入等待周期(每个等待周期为一个时钟周期),以保证主控模块在固定的脉冲沿采样等待信号。根据等待信号的有效时间,可以插入若干等待周期,从而延长了整个传输周期的时间间隔。