

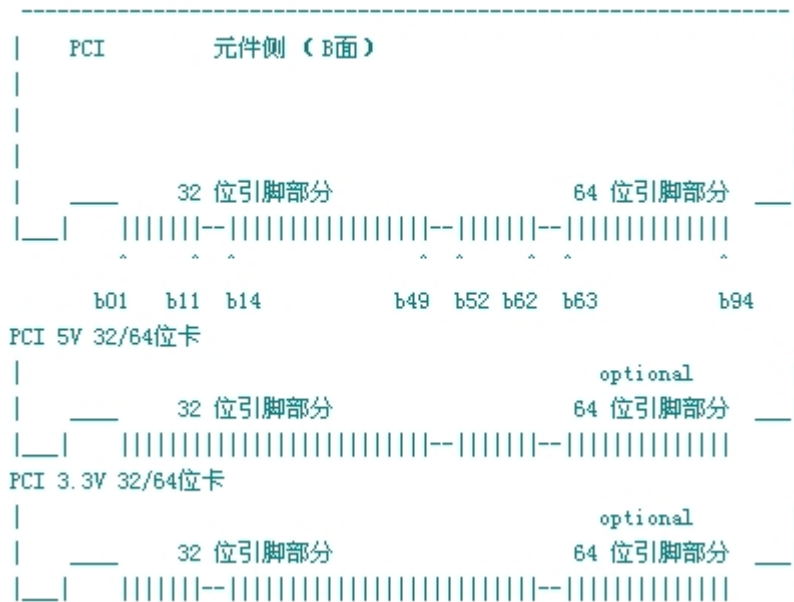
PCI 总线

PCI 总线规范是由 INTEL 公司为首的一个 PCI 特别兴趣小组 (PCI SIG) 制定并维护的一种 32/64 位局部总线,可进行 32 位寻址,也可以进行突发传送。其最高传输能力为 132MB/S (32 位时)。PCI 总线的一个与其它总线最大区别是它有一个特别的地址空间:配置空间,也因此而具备了即插即用 (PNP) 的功能。PCI 总线 (外设互联总线) 与传统的总线标准— ISA 总线 (工业标准结构总线) 相比,具有更高的传输率 (132 MB/s)、支持 32 位处理器、支持 DMA、即插即用等优势。这使之取代 ISA 总线而成为目前台式计算机的事实 I/O 总线标准。如今购买 PC 机而没有 PCI 插槽是不可想象的。对于基于计算机的测试仪器, PCI 总线为应用计算机到新的测试仪器,即满足在插卡和系统存储器中高速传输数据的要求提供了很好的途径。

先把有用的放在前面 其他的介绍就放在最后了 ☺

1.区分 5V 与 3V 的方法

PCI 标准 32位/64位 接口卡



2.标准接口定义

32bit

Pin	+5V	+3.3V	Universal	Description
A1	TRST			Test Logic Reset
A2	+12V			+12 VDC
A3	TMS			Test Mode Select
A4	TDI			Test Data Input
A5	+5V			+5 VDC
A6	INTA			Interrupt A
A7	INTC			Interrupt C
A8	+5V			+5 VDC
A9	RESV01			Reserved VDC
A10	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
A11	RESV03			Reserved VDC
A12	GND03	(OPEN)	(OPEN)	Ground or Open (Key)
A13	GND05	(OPEN)	(OPEN)	Ground or Open (Key)
A14	RESV05			Reserved VDC
A15	RESET			Reset
A16	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
A17	GNT			Grant PCI use
A18	GND08			Ground
A19	RESV06			Reserved VDC
A20	AD30			Address/Data 30
A21	+3.3V01			+3.3 VDC
A22	AD28			Address/Data 28
A23	AD26			Address/Data 26
A24	GND10			Ground
A25	AD24			Address/Data 24
A26	IDSEL			Initialization Device Select
A27	+3.3V03			+3.3 VDC
A28	AD22			Address/Data 22
A29	AD20			Address/Data 20
A30	GND12			Ground
A31	AD18			Address/Data 18
A32	AD16			Address/Data 16
A33	+3.3V05			+3.3 VDC
A34	FRAME			Address or Data phase
A35	GND14			Ground

A36	TRDY			Target Ready
A37	GND15			Ground
A38	STOP			Stop Transfer Cycle
A39	+3.3V07			+3.3 VDC
A40	SDONE			Snoop Done
A41	SBO			Snoop Backoff
A42	GND17			Ground
A43	PAR			Parity
A44	AD15			Address/Data 15
A45	+3.3V10			+3.3 VDC
A46	AD13			Address/Data 13
A47	AD11			Address/Data 11
A48	GND19			Ground
A49	AD9			Address/Data 9
A52	C/BEO			Command, Byte Enable 0
A53	+3.3V11			+3.3 VDC
A54	AD6			Address/Data 6
A55	AD4			Address/Data 4
A56	GND21			Ground
A57	AD2			Address/Data 2
A58	AD0			Address/Data 0
A59	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
A60	REQ64			Request 64 bit ???
A61	VCC11			+5 VDC
A62	VCC13			+5 VDC

B1	-12V			-12 VDC
B2	TCK			Test Clock
B3	GND			Ground
B4	TD0			Test Data Output
B5	+5V			+5 VDC
B6	+5V			+5 VDC
B7	INTB			Interrupt B
B8	INTD			Interrupt D
B9	PRSNT1			Reserved
B10	RES			+V I/O (+5 V or +3.3 V)
B11	PRSNT2			??
B12	GND	(OPEN)	(OPEN)	Ground or Open (Key)
B13	GND	(OPEN)	(OPEN)	Ground or Open (Key)
B14	RES			Reserved VDC
B15	GND			Reset
B16	CLK			Clock
B17	GND			Ground
B18	REQ			Request
B19	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
B20	AD31			Address/Data 31
B21	AD29			Address/Data 29
B22	GND			Ground
B23	AD27			Address/Data 27
B24	AD25			Address/Data 25
B25	+3.3V			+3.3VDC
B26	C/BE3			Command, Byte Enable 3
B27	AD23			Address/Data 23
B28	GND			Ground
B29	AD21			Address/Data 21
B30	AD19			Address/Data 19
B31	+3.3V			+3.3 VDC
B32	AD17			Address/Data 17
B33	C/BE2			Command, Byte Enable 2
B34	GND13			Ground
B35	IRDY			Initiator Ready

B36	+3.3V06			+3.3 VDC
B37	DEVSEL			Device Select
B38	GND16			Ground
B39	LOCK			Lock bus
B40	PERR			Parity Error
B41	+3.3V08			+3.3 VDC
B42	SERR			System Error
B43	+3.3V09			+3.3 VDC
B44	C/BE1			Command, Byte Enable 1
B45	AD14			Address/Data 14
B46	GND18			Ground
B47	AD12			Address/Data 12
B48	AD10			Address/Data 10
B49	GND20			Ground
B50	(OPEN)	GND	(OPEN)	Ground or Open (Key)
B51	(OPEN)	GND	(OPEN)	Ground or Open (Key)
B52	AD8			Address/Data 8
B53	AD7			Address/Data 7
B54	+3.3V12			+3.3 VDC
B55	AD5			Address/Data 5
B56	AD3			Address/Data 3
B57	GND22			Ground
B58	AD1			Address/Data 1
B59	VCC08			+5 VDC
B60	ACK64			Acknowledge 64 bit ???
B61	VCC10			+5 VDC
B62	VCC12			+5 VDC

64bit

A63	GND			Ground
A64	C/BE[7]#			Command, Byte Enable 7
A65	C/BE[5]#			Command, Byte Enable 5
A66	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
A67	PAR64			Parity 64 ???
A68	AD62			Address/Data 62
A69	GND			Ground
A70	AD60			Address/Data 60
A71	AD58			Address/Data 58
A72	GND			Ground
A73	AD56			Address/Data 56
A74	AD54			Address/Data 54
A75	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
A76	AD52			Address/Data 52
A77	AD50			Address/Data 50
A78	GND			Ground
A79	AD48			Address/Data 48
A80	AD46			Address/Data 46
A81	GND			Ground
A82	AD44			Address/Data 44
A83	AD42			Address/Data 42
A84	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
A85	AD40			Address/Data 40
A86	AD38			Address/Data 38
A87	GND			Ground
A88	AD36			Address/Data 36
A89	AD34			Address/Data 34
A90	GND			Ground
A91	AD32			Address/Data 32
A92	RES			Reserved
A93	GND			Ground
A94	RES			Reserved

B63	RES			Reserved
B64	GND			Ground
B65	C/BE[6]#			Command, Byte Enable 6
B66	C/BE[4]#			Command, Byte Enable 4
B67	GND			Ground
B68	AD63			Address/Data 63
B69	AD61			Address/Data 61
B70	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
B71	AD59			Address/Data 59
B72	AD57			Address/Data 57
B73	GND			Ground
B74	AD55			Address/Data 55
B75	AD53			Address/Data 53
B76	GND			Ground
B77	AD51			Address/Data 51
B78	AD49			Address/Data 49
B79	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
B80	AD47			Address/Data 47
B81	AD45			Address/Data 45
B82	GND			Ground
B83	AD43			Address/Data 43
B84	AD41			Address/Data 41
B85	GND			Ground
B86	AD39			Address/Data 39
B87	AD37			Address/Data 37
B88	+5V	+3.3V	Signal Rail	+V I/O (+5 V or +3.3 V)
B89	AD35			Address/Data 35
B90	AD33			Address/Data 33
B91	GND			Ground
B92	RES			Reserved
B93	RES			Reserved
B94	GND			Ground

一个 MINI PCI 的管脚定义

1	TIP		RING	2
	KEY		KEY	
3	RD+		TD+	4
5	RD-		TD-	6
7	RJ45Tmn7		RJ45Tmn4	8
9	RJ45Tmn8		RJ45Tmn5	10
11	LED1_GRNP		LED2_YELP	12
13	LED1_GRNN		LED2_YELN	14
15	CHSGND		RESERVED5	16
17	INTB#		5V2	18
19	3.3V1		INTA#	20
21	RESERVED1		RESERVED6	22
23	GND1		3.3VAUX1	24
25	CLK		RST#	26
27	GND2		3.3V5	28
29	REQ#		GNT#	30
31	3.3V2		GND10	32
33	AD[31]		PME#	34
35	AD[29]		RESERVED7	36
37	GND3		AD[30]	38
39	AD[27]		3.3V6	40
41	AD[25]		AD[28]	42
43	RESERVED2		AD[26]	44
45	C/BE[3]#		AD[24]	46
47	AD[23]		IDSEL	48
49	GND4		GND11	50
51	AD[21]		AD[22]	52
53	AD[19]		AD[20]	54
55	GND5		PAR	56
57	AD[17]		AD[18]	58
59	C/BE[2]#		AD[16]	60
61	IRDY#		GND12	62
63	3.3V3		FRAME#	64
65	CLKRUN#		TRDY#	66
67	SERR#		STOP#	68
69	GND6		3.3V7	70
71	PERR#		DEVSEL#	72
73	C/BE[1]#		GND13	74
75	AD[14]		AD[15]	76
77	GND7		AD[13]	78
79	AD[12]		AD[11]	80
81	AD[10]		GND14	82
83	GND8		AD[09]	84
85	AD[08]		C/BE[0]#	86
87	AD[07]		3.3V8	88
89	3.3V4		AD[06]	90

MINI-PCI CONN

TYPE III

(2 X 62)

89	AD[07]		3.3V8	90
91	3.3V4		AD[06]	92
93	AD[05]		AD[04]	94
95	RESERVED3		AD[02]	96
97	AD[03]		AD[00]	98
99	5V1		RESERVED_WIP1	100
101	AD[01]		RESERVED_WIP2	102
103	GND9		GND15	104
105	AC_SYNC		M66EN	106
107	AC_SDADA_IN		AC_SDATA_OUT	108
109	AC_BIT_CLK		AC_CODEC_ID0#	110
111	AC_CODEC_ID1#		AC_RESET#	112
113	MOD_AUDIO_MON		RESERVED8	114
115	AUDIO_GND1		GND16	116
117	SYS_AUDIO_OUT		SYS_AUDIO_IN	118
119	SYS_AUDIO_OUT_GND		SYS_AUDIO_IN_GND	120
121	AUDIO_GND2		AUDIO_GND3	122
123	RESERVED4	TAB	MPCIACT#	124
	VCC5VA	TAB	3.3VAUX2	

下面是一些其他的介绍

1. AD[31:0] (PCI ADDRESS / DATA BUS)

地址与数据总线讯号,在 FRAME#启动后地址才有效,在 PCLK 第一个 CLOCK 动作初始化时,FRAME#动作后,输出为地址与数据,写入周期,输入为数据,读取周期 TRDY# 与 IRDY# 会动作,高阻抗时,为数据转换周期或 RESET#动作

2. C/BE[3:0]# (PCI COMMAND /BYTE ENABLES)

FRAME#启动后,CLOCK 第一个 CLOCK,周期为 PCI 命令,再下一个周期为允许命令,命令在 FRAME#后有效,数据在 TRDY#与 IRDY#后有效

3. DEVSEL# (PCI DEVSEL SELECT)

确定外部外围连结之响应讯号,高阻抗时,为停止周期或 RESET#动作时

4. FRAME# (PCI CYCLE FRAME)

PCI 总线起始讯号

5. GNT[4:0]# (PCI BUS GRANT)

PCI 总线控制认可讯号

6. IRDY# (INITIATOR READY)

数据读取写入讯号

7. LOCK# (PCI BUS LOCK)

总线锁住讯号

8. PAR (PCI BUS PARITY)

地址与位传送之同位检错讯号

9. PCLK (PCI CLOCK)

PCI 时脉讯号

10.PGNT# (PCI GRANT TO PERIPHERAL BUS CONTROLLER)

PCI 总线对外部外围装置之需求同意认可讯号

11. PERQ# (PCI REQUEST FROM PERIPHERAL BUS CONTROLLER)

外围处理器对 PCI 总线要求讯号

12. REQ[4:0]# (PCI BUS REQUEST)

PCI 总线需求讯号

13. RESET# (RESET)

系统重置讯号

14. SERR# (SYSTEM ERROR)

系统错误侦测讯号 可产生 NMI 不可屏蔽中断

15. STOP# (PCI BUS STOP)

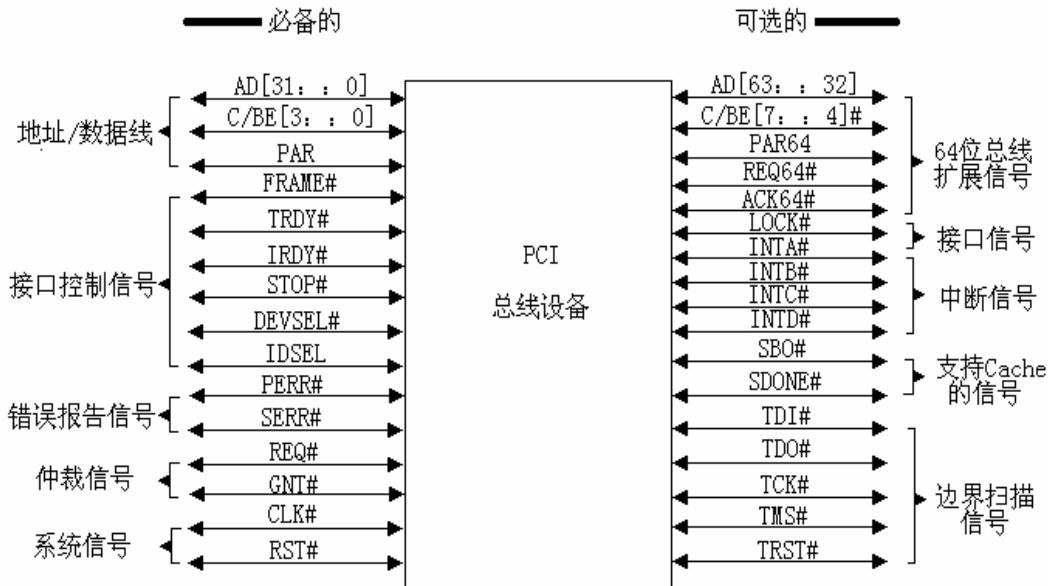
PCI 总线放弃或重试数据传送之讯号

16. TRDY# (TARGET READY)

PCI 总线数据读取传送讯号

17.WSC# (WRITE SNOOP COMPLETE)

I/O APIC 芯片有上时之中断讯息传送讯号



PCI 总线信号的定义

1 系统信号

• **CLK: IN**, 总线时钟输入信号, 其值决定了 PCI 总线的工作频率, 最低可为 0Hz(DC), 最高对 33MHz PCI 总线为 33MHz、对 66MHz PCI 总线为 66MHz。PCI 的其它信号, 除 **RST#**、**INTA~D#**之外, 其余信号都在 CLK 的上升沿同步。

• **RST#**: IN, 复位输入信号, 用来复位 PCI 总线上的接口设备, 使 PCI 专用寄存器和定序器相关的信号恢复到规定的初始状态。对 PCI 的配置寄存器, 其复位状态是 PCI 标准明确规定的 (详见后续部分)。每当总线复位时, PCI 的全部输出信号一般都应驱动到第三态, **SERR#** 信号为高阻状态, **SBD#** 和 **SDONE** 可驱动到低电平 (如果未提供三态输出), **REQ#** 和 **GNT#** 必须同时驱动到第三态, 不能在复位期间为高或为低。为防止 **AD**、**C/BE#** 及 **PAR** 在复位期间浮动, 主板应将它们驱动到逻辑低, 但不能驱动为高电平。**RST#** 可以与 CLK 不同步, 但要保证其撤消边沿没有反弹。当设备请求引导系统时, 将响应复位, 复位后响应系统引导。

2 地址和数据信号

• **AD[31:: 00]**: T/S, 它们是一组 32 位的地址、数据复用双向 (输入/输出) 三态信号。在 **FRAME#**有效后的第一个时钟周期是地址期, **AD[31:: 00]**上传输 32 位地址; 在 **IRDY#**和 **TRDY#**同时有效时是数据期, **AD[31:: 00]**上传输 32 位数据。一个 PCI 总线的传输中包含了一个地址期和接着的一个 (单次读写) 或许多个 (块读写或突发传输) 数据期, PCI 总线支持突发方式的读写功能。地址期为一个时钟周期, 该周期中 **AD[31:: 00]**线上含有一 32 位物理地址。对于 I/O 操作, 它是一个字节地址 (数据为一个字节, 地址仍为 32 位连续的 I/O 地址, 详见总线命令部分); 若是存储器操作和配置寄存器操作则是双字地址 (数据为四字节即双字, 地址为高 30 位, 低两位 **AD0~AD1**: 无用); 在数据期, **AD[07:: 00]**为最低字节, **AD[31:: 24]**为最高字节。**IRDY#**有效时表示写数据稳定有效, **TRDY#**有效时表示读数据稳定有效, 可在时钟的上升沿对数据进行锁存, **IRDY#**和 **TRDY#**均无效时为等待周期。

• **C/BE[3:: 0]#**: (Command/Byte Enable, T/S) 它们是 32 位总线命令和字节使能多路复用三态信号线。在地址期中, 这四条线上传输的是总线命令。PCI 总线用编码方式表示总线命令, 四条线可表示 16 种不同总线命令。在数据期内, 此信号线传输字节使能信号, 用来表示在整个数据期中, **AD[31:: 00]**上哪些字节为有效数据。PCI 的这种表示方式一次可传输任意字节的数据。无论是输入还是输出, 读还是写, **C/BE#**总是由主控来驱动, 目标根据字节使能输出字节数据到总线。

3 接口控制信号

• **FRAME#**: S/T/S, 帧周期信号, 由当前主控驱动, 表示一次数据帧访问的开始和持续时间。**FRAME#**有效预示着总线传输的开始, **FRAME#**开始后的第一个时钟周期为地址期, 之后则为数据期。在 **FRAME#**的有效期间, 意味着数据传输继续进行, 直至 **FRAME#**失效后还有最后一个数据周期。

• **IRDY#**: (Initiator Ready) S/T/S, 主控设备准备好信号。该信号的有效表明发起本次传输的主控已准备好, 否则 (无效) 即为等待周期。在写周期, 该信号有效表示数据已在 **AD[31:: 00]**中, 且稳定有效; 在读周期, 该信号有效表示主控已做好接收数据的准备。

• **TRDY#**: (Target Ready) S/T/S, 目标设备准备好信号。该信号有效表示目标设备 (从设备) 已作好完成当前数据传输的准备工作, 也就是说, 可以进行相应的数据传输。该信号要与 **IRDY#**配合使用, 两者同时有效数据才能完整传输。在读周期中, 该信号有效表示目标已将有效数据提交到 **AD[31:: 00]**中; 在写周期中, 该信号有效表示目标已做好接收数据

的准备。同理，IRDY#和 TRDY#的任何一个无效时都为等待周期。从上可知，PCI 总线可通过 IRDY#或 TRDY#无效在数据传输过程中由主控或目标根据自身的响应速度灵活地插入多个等待周期，以使总线适用各种档次速度的接口设备。

• IDSEL: Identity Select, IN, 初始化设备选择信号。在 PCI 接口配置参数读写传输期间用做片选信号，它是一个主桥到 PCI 插卡的点对点连接信号，一般采用高位地址线实现，由 PnP BIOS 上电时进行驱动，以实现 PCI 接口的自动配置。

4 总线仲裁信号

• REQ#: (Bus Request) T/S, 总线占用请求信号。该信号一旦有效即表明驱动它的设备要求使用总线。它是一个点到点的信号线，任何主设备都有其 REQ# 信号。

• GNT#: (Bus Granted) T/S, 总线占用允许信号。用来向申请占用总线的设备表示其请求已获得批准，可以立刻使用总线。这也是一个点到点的信号线，任何主控都应有自己的 GNT# 信号。

5 错误报告信号

为使数据传输可靠、完整，PCI 局部总线标准要求所有挂于其上的设备都应具有错误报告线。

• PERR#: (Parity Error) S/T/S, 数据奇偶校验错误报告信号。一个设备只有在响应设备选择信号 (DEVSEL#) 和完成数据期之后才能报告一个 PERR #, 即比实际数据传输晚一个时钟周期。对于每个接收数据的设备，如果发现 数据有错误，就应在数据收到后的两个时钟周期内将 PERR# 激活。

• SERR#: (System Error) OD, 系统错误报告信号。该信号的作用是报告在特殊周期中的地址数据奇偶错以及其它可能引起灾难性后果的系统错误。

如果设备不希望产生非屏蔽中断，就应采用其它机制来实现 SERR# 的报告，即 SERR# 相当于 ISA 总线中的非屏蔽中断。

6 中断信号

中断在 PCI 总线中是可选项，不一定必须具有，并且中断信号属电平敏感性，低电平有效，使用漏极开路方式驱动。中断信号的建立和撤消与时钟不同步。对于单功能设备，只有一条中断线，而多功能设备最多可有四条中断线。在前一种情况下，只能使用 INTA#, 其它三条中断线没有意义。所谓的多功能设备是指将几个相互独立的功能集中在一个设备中。

PCI 局部总线总共有四条中断线，分别是：INTA#、INTB#、INTC#和 INTD#，均为 OD (漏极开路)，其作用是实现中断请求。后三个只能用于多功能设备。一个多功能设备上的任何功能都可以连接到四条中断线的任意一条。也就是说，各功能与中断线之间的连接是任意的，没有附加限制，二者的最终对应关系是由中断引脚寄存器来定义的 (见配置寄存器部分)。显然这提供了很大的灵活性。如果一个设备要实现一个中断，就定义为 INTA#, 要实现两个中断，就定义为 INTA#和 INTB#。对于多功能设备，可以多个功能共用同一条中断线，或者各自占一条中断线，或者是两种情况的组合。

PCI 总线命令

总线命令的作用是用来规定主从设备之间信息的传输类型，它出现在地址期的 C/BE[3:: 0]#总线上。这里的主设备（或主控）是指通过仲裁而获得总线控制权的设备；从设备（或目标）是指在 C/BE[3:: 0]#上出现命令的同时，被 AD[31:: 0]总线上的地址所选中的设备。

总线命令编码

总线命令用地址期间 C/BE[3:: 0]#总线上的信号表示，共 16 种，表 3-1 给出了总线命令编码及类型说明。其中，命令编码中的“1”表示高电平，“0”表示低电平。

C/BE[3:: 0]	命令类型说明
0 0 0 0	中断应答（中断识别）(Interrupt Acknowledge)
0 0 0 1	特殊周期 (Special Cycle)
0 0 1 0	I/O 读（从 I/O 口地址中读数据）(I/O Read)
0 0 1 1	I/O 写（向 I/O 口地址中写数据）(I/O Write)
0 1 0 0	保留 (Reserved)
0 1 0 1	保留 (Reserved)
0 1 1 0	存储器读（从内存空间映像中读数据）(Memory Read)
0 1 1 1	存储器写（向内存空间映像中写数据）(Memory Write)
1 0 0 0	保留 (Reserved)
1 0 0 1	保留 (Reserved)
1 0 1 0	配置读 (Configuration Read)
1 0 1 1	配置写 (Configuration Write)
1 1 0 0	存储器多行读 (Memory Read Multiple)
1 1 0 1	双地址周期 (Dual Address Cycle)
1 1 1 0	存储器一行读 (Memory Read Line)
1 1 1 1	存储器写无效 (Memory Write Invalidate)

I/O 读命令

该命令用来从一个映射到 I/O 地址空间的设备中读取数据。AD[31:: 00]上提供一个字节地址，全部 32 位参与译码；而字节使能信号表示传送数据的多少，且必须与字节地址一致。

对 PCI 总线，I/O 地址是 32 位（这一点与 ISA 总线只译码 I/O 地址的低 10 位不同），并且 32 位 I/O 地址精确地寻址一个字节的 I/O 数据，每次 I/O 操作的字节数据在 32 位数据总线上的位置由 C/BE[3:: 0]指示，并且必须与 I/O 操作地址周期中最低两位地址 AD[1:: 0]一一对应。例如，若 I/O 地址最低两位地址 AD[1:: 0]为“00”时，对应的字节数据必须放在数据总线最低 8 位 AD[7:: 0]上传输，同时 C/BE[3:: 0]为“1110”，以示数据在低 8 位总线上；同样，当 AD[1:: 0]为“01”时，C/BE[3:: 0]必须为“1101”，I/O 数据在 AD[15:: 8]总线上，其余依次类推。

在具体的 I/O 访问中，当总线上的一个目标设备被 I/O 地址译码选中后，该目标设备的 DEVSEL#信号就应输出低电平作为响应信号，以告之主控总线上有设备被寻址选中，并通过 TRDY#完成一次 I/O 操作。I/O 读命令操作的时序如图 3-7 所示。

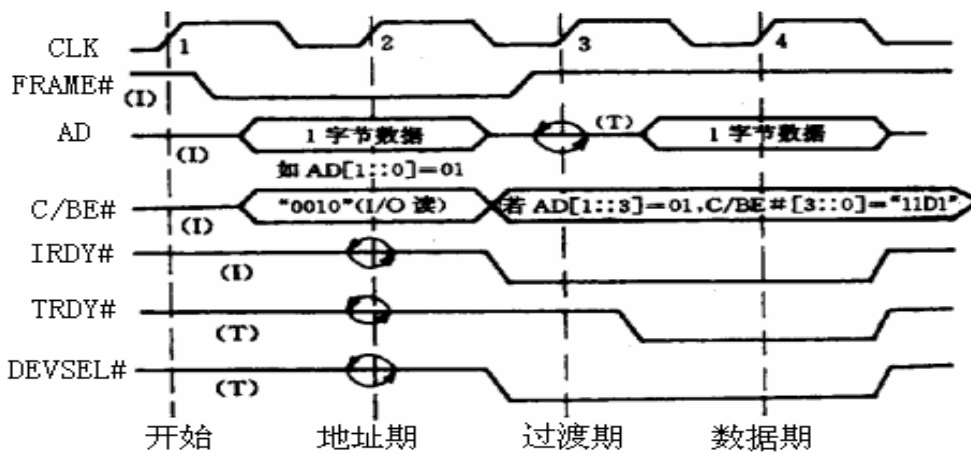


图 3-7 I/O 读时序图

I/O 写命令

该命令用来向一个映射到 I/O 地址空间的设备写入数据。全部 32 位地址必须参加译码，字节使能信号表示字节数据在总线上的位置，且必须和字节地址最低两位一致。I/O 写命令的使用要求、方法和时序与 I/O 读命令是完全一样的。

存储器读命令

该命令用来从一个映射到存储器地址空间中的目标设备读取数据。当 CPU 通过主控发出一次 PCI 存储器读命令时，在 FRAME#有效的同时，主控将 32 位地址通过 AD[31:: 0]输出，同时 C/BE#总线上输出存储器读命令，并在总线时钟上升沿处有效。然后，当主控准备好接收数据时，IRDY#有效。被 32 位存储器地址寻址选中的目标设备必须以 DEVSEL#有效来响应本次存储器读命令过程，同时，经过一个过渡周期后，目标设备将多字节数据输出到 AD[31:: 00]数据总线上，其有效字节的位置由 C/BE#总线相应位指出，并激活 TRDY#有效表示目标数据已准备好，此时主控将在时钟上升沿处将数据读入，完成一次存储器读过程。

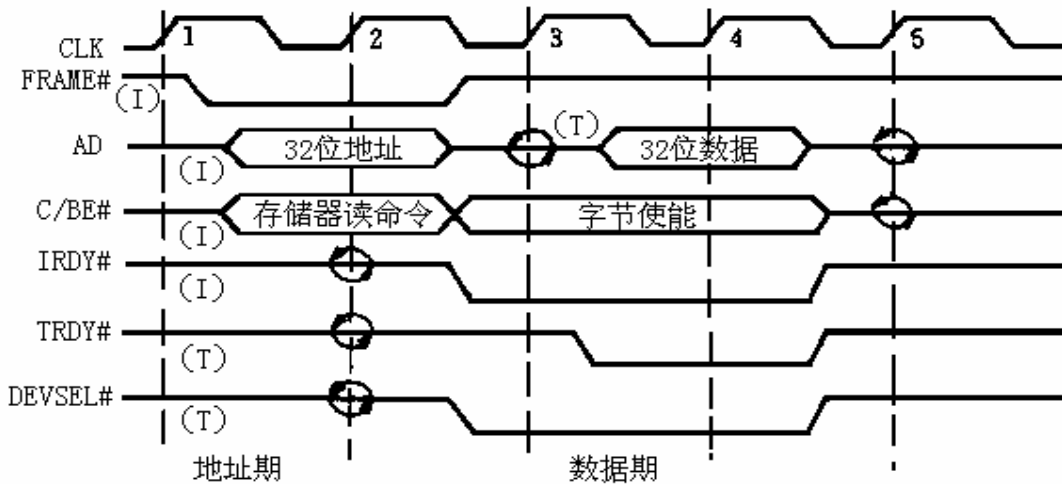


图 3-8 存储器读命令时序

存储器写命令

该命令用来向一个映射到存储器空间的目标设备写入数据。存储器写命令与读命令的要求、特点、使用方法及操作时序完全一样。

3. 3. 3. 6. 配置读命令

该命令用来从每个设备目标的配置空间读取数据。如果一个目标设备的 IDSEL 引脚有效，C/BE#总线上为配置读命令且 AD[1:: 0]=00 时，那么该设备即被选定为配置读命令的目标。在一个配置读命令的地址期内，AD[7:: 2]用于从每个设备的配置空间中的 64 个双字寄存器中选中的一个，AD[31:: 11]无意义，AD[10:: 8]表示一个多功能设备的哪个功能设备被选中。在进行读写时，AD[1:: 0]必须为 00，否则该命令无效。

配置读/写命令除上电时 PnP BIOS 需要使用外，与 PCI 设备对应的驱动程序也需要使用，但设备驱动程序一般只需要配置读命令，如设备驱动程序应知道 PnP 初始化后具体 I/O 地址、内存地址分配在什么区域，中断请求（如需要的话）为多少等。一般不建议在设备驱动程序中使用配置写命令，以免改变系统的配置后导致系统冲突。

3. 3. 3. 7. 配置写命令

该命令用来向每个设备的配置空间写入数据。一个设备被选中的条件是：它的 IDSEL 信号有效并且 $AD[1::0]=00$ 。其余和配置读命令相同。

3. 3. 4 PCI 总线协议

PCI 上的基本总线传输机制是突发分组传输。一个突发分组由一个地址期和一个（多个）数据期组成。PCI 支持存储器空间和 I/O 空间的突发传输。

3. 3. 4. 1 PCI 总线的传输控制

PCI 总线上所有的数据传输基本上都是由以下三条信号线控制的：

- FRAME#：由主设备驱动，指明一次传输的起始和结束。
- IRDY#：由主设备驱动，允许插入等待周期。
- TRDY#：由从设备驱动，允许插入等待周期。

当数据有效时，数据资源需要无条件设置 XRDY#信号（写操作为 IRDY#，读操作为 TRDY#）。接收方可以在适当的时间发出它的 XRDY#信号。FRAME#信号有效后的第一个时钟前沿是地址期的开始，此时传送地址信息和总线命令。下一个时钟前沿开始一个或多个数据期，每逢 IRDY#和 TRDY#共同有效时，所对应的时钟前沿就使数据在主、从设备之间传送，在此期间，可由主设备或从设备分别利用 IRDY# 和 TRDY# 的无效而插入等待周期。

主设备一旦设置了 IRDY#信号，就不能改变 IRDY#和 FRAME#，直到当前的数据期完成为止。而一个从设备一旦设置了 TRDY#信号或 STOP#信号，就不能改变 DEVSEL#、TRDY#或 STOP#，即一旦承诺了数据传输，就必须进行到底。

当到最后一次数据传输时（有时紧接地址期之后）主设备应撤消 FRAME#信号而建立 IRDY#信号，表明主设备已做好了最后一次数据传输的准备，待到从设备发出 TRDY#信号后，就说明最后一次数据传输已完成，FRAME#和 IRDY#信号均可撤消，接口回到空闲状态。总之，PCI 总线传输一般遵循如下管理规则：

- FRAME#和 IRDY#定义了总线的忙/闲状态。当其中一个有效时，总线是忙的；两个都无效时，总线处于空闲状态。
- 一旦 FRAME#信号被置为无效，在同一传输期间不能重新设置。在总线上进行数据传输的时候，一般情况下不能使 FRAME#信号无效。
- 主设备一旦设置了 IRDY#信号，直到当前数据期结束为止，主设备不能改变 IRDY#信号和 FRAME#信号的状态。

3. 3. 4. 2 PCI 的物理地址空间

PCI 总线定义了三个物理地址空间：内存地址空间；I/O 地址空间；配置地址空间。前两个是通常都有的，第三个用以支持 PCI 的硬件配置。

PCI 总线的编址是分布式的，每个设备都有自己的地址译码电路，从而省去了中央译码逻辑。PCI 支持正向和负向两种方式的地址译码，所谓正向译码就是每个设备都监视地址总线上的访问地址是否落在自己的地址范围，因此速度较快；而负向译码是指该设备要接受未被其它设备在正向译码中接受的所有访问，因此，此种译码方式只能由总线上的一个设备来实现，由于它要等到总线上其它所有设备都拒绝之后才能行动，所以速度较慢。然而，负向译码对于像标准扩展总线这类设备却是很有用的，这是因为这类设备必须响应一个很零散的地址空间。但无论是正向译码的设备还是负向译码的设备都不对保留的总线命令发出 DEVSEL#响应信号。

(1) I/O 地址空间

在 I/O 地址空间中，全部 32 位 AD 总线都被用来提供一个完整的地址编码（字节地址），这使得要求地址精确到字节水平的设备不需要多等一个周期就可完成地址译码（产生 DEVSEL#信号），也使负的地址译码节省了一个时钟周期。

在 I/O 访问中，AD[1:: 0]两位很重要，表示传输涉及的最低有效字节，并且要与 C/BE[3:: 0]#相配合。例如，当 C/BE0#有效，那么 AD[1:: 0]必须为“00”；如果 C/BE3#有效时，AD[1:: 0]就应当为“11”。在具体访问中，每当一个从设备被地址译码选中后，便要检查字节使能信号是否与 AD[1:: 0]相符，如果二者矛盾，则整个访问就无法完成，此时从设备不传送任何数据，而是以一个“目标终止”操作来结束访问。表 3-2 给出了 AD[1:: 0]和 C/BE[3:: 0]#的对应关系。其中，1 表示高电平，0 表示低电平，X 表示 0 或 1。任何不同于表 3-2 的组合状态都是非法的。

表 3-2 AD[1:: 0]与 C/BE[3:: 0]#的对应关系表

AD1	AD0	C/BE3#	C/BE2#	C/BE1#	C/BE0#
0	0	X	X	X	0
0	1	X	X	0	1
1	0	X	0	1	1
1	1	0	1	1	1

(2) 内存地址空间

在存储器访问中，地址为双字地址只用 AD[31:: 02]，AD[1:: 0]有特殊用途，所有的目标设备都要检查 AD[1:: 0]，要么提供要求的突发传输顺序，要么执行一目标设备断开操作。对于所有支持突发传输的设备都应能实现线性突发传输，而 Cache 的行切换不一定必须实现。在存储器地址空间中，要用 AD[31:: 02]译码得到一个双字地址的访问。在线性增长方式下，每个数据周期过后地址按一个 DWORD（4 个字）增加，直到对话结束。在存储器访问期间，AD[1:: 0]的含义如下：

当 AD[1:: 0]=00 时，突发传输顺序为线性增长方式；AD[1:: 0]=01 时，为 Cache 行切换方式；AD[1:: 0]=1X 时，为保留。

(3) 配置地址空间

在配置的地址空间中，要用 AD[7:: 2]将访问落实到一个 DWORD 地址，配置空间共 64 双字。当一个设备收到配置命令时，若 IDSEL 信号成立且 AD[1:: 0]=00，则该设备即被选中为访问的目标，否则就不参与当前的对话。如果译码出的命令符合某桥路的编号，且 AD[1:: 0]=01，则说明配置访问是对着该桥后面的设备。具体内容详见下一小节。

3. 3. 4. 3 PCI 总线的配置空间

PCI 定义了三种地址空间：存储器空间、输入输出空间和配置空间。其中最重要的是配置空间。PCI 定义配置空间的目的在于提供一种配置关联，这种关联适合于目前或将来的系统配置机制，从而实现参数自动配置，使所有与 PCI 兼容的设备实现真正的“plug-and-play”。

配置空间为 256 个字节，分为预定义的头域（64Bytes）和依赖于设备的区域（192Bytes）两个部分。PCI 规定，所有与 PCI 兼容的设备都必须定义头域，对依赖于设备的区域则未作定义。表 3-3 为配置空间的头域结构。

头域中“供应商代号”由 PCI SIG 发布。“设备识别”、“修改版本”和“分类代号”三个寄存器用于识别设备及其功能。

头域中“命令”寄存器提供对 PCI 设备产生和响应 PCI 周期的粗略的控制。

头域中“状态”寄存器用来记录与 PCI 总线有关事件的消息。“状态”寄存器的读取与一般寄存器的读取一样，但只能采用写入 1 的方式复位。

头域中“中断线”寄存器用于 X86 微机 IRQ0-IRQ15 的选择，“中断引脚”寄存器则提供 INTA#到 INTD 的选择，以达到无需用户干预的安装。

表 3-3 PCI 配置空间预定义头域

设备识别号		供应商代码	
状态		命令	
分类代号		修改版本	
内含自测试	头标类型	延时计数	Cached 大小
基址寄存器(6 个双字)			
保留			
保留			
扩展 ROM 基址寄存器			
保留			
保留			
Max-Lat	Min-Gnt	中断引脚	中断线

基地址 (Base Address Register) 是用来确定该设备的存储和 I/O 空间的，目的是为了实现在地址空间的重定位。BIOS 在系统自举交给操作系统之前，要建立一个地址映射，这意味着确定多少存储器及 I/O 空间给指定的 PCI 设备。基地址寄存器由 6 个双字 (32bit) 构成，每个双字定义相同。如下图 3-9。

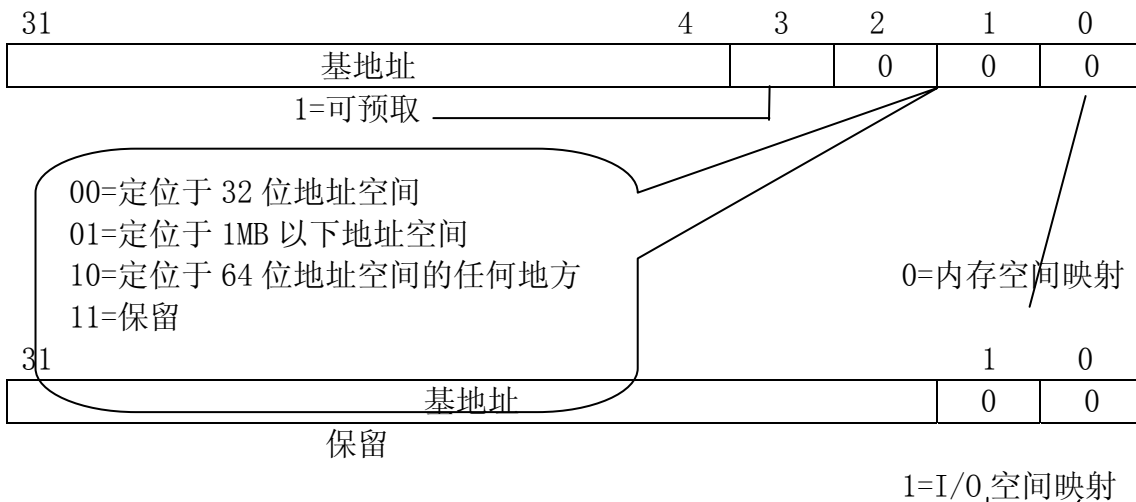


图 3-9 PCI 总线配置空间基地址定义

如果是 I/O 空间映射，有 30 位可用 (bit2-bit31)，I/O 空间映射的大小最少为 4 个字节；若为内存空间，有 28 位可用 (bit4-bit31)，也可用内存空间映射于 1M 字节以下 (bit2=bit1=0)，可保持 DOS 与 WINDOWS 操作系统的兼容性，内存空间映射大小最少为 16 字节。

系统采用以下方法来实现 PCI 设备的地址重定位：

首先，在主机上电自举过程开始时，主机识别系统的每一个 PCI 设备后，将

依次向每个 PCI 设备的每个基址寄存器写入全 ‘1’，然后再将其读回，读回的值即用以确认地址空间的大小（此读回值也即每个 PCI 设备初始化时的上电值，也是我们可以预先定制的数值）。其表现有些特别，对于存储器映射，将最低 4 位屏蔽后，采用自右向左的方向，以 ‘0’ 的个数的多少来确定空间的大小，比如 “FFFFFFF0” 表示大小为 16 个字节，“FFFFFFE0” 表示大小为 32 个字节，“FFFFFFC2” 表示位于 1M 字节以下，大小为 64 个字节的内存块等等；I/O 空间映射除了把 bit0-1 屏蔽外，与以上雷同，如 “FFFFFFFD” 表示 4 个字节大小的 I/O 地址空间等。

其次，在主机依次回读完每个 PCI 设备的基址寄存器后，有 BIOS 根据每个 PCI 设备所占有的 I/O 和存储器空间的大小，结合微机系统资源的剩余量，分配给每个 PCI 设备相应大小的系统资源，接着由主机依次将分配的内存或 I/O 空间的首地址写入相应 PCI 设备的对应基址寄存器。这个起始地址一般是对应块大小的整数倍。例如，若基址寄存器请求分配一 64K 字节大小的区域，则最终写入的起始地址一定是 $64K \times n$ (n 为整数倍)，至此，圆满完成 PCI 设备地址的重定位。

