

ISP1581: 高速 USB 接口器件

1.概述

ISP1581 是一种价格低、功能强的高速通用串行总线 (USB) 接口器件, 它完全符合 USB 2.0 规范, 并为基于微控制器或微处理器的系统提供了高速 USB 通信能力。ISP1581 与系统的微控制器/微处理器的通信是通过一个高速的通用并行接口来实现的。

ISP1581 支持高速 USB 系统的自动检测。最初 USB 规范的返回工作模式允许器件在全速条件下正常工作。ISP1581 是一个通用的 USB 接口器件, 它符合现有的大多数器件的分类规格, 比如: 成像类、海量存储器件、通信器件、打印设备以及人机接口设备。

内部通用 DMA 模块使得数据流很方便的集成。另外, 多种结构的 DMA 模块实现了海量存储的应用。

这种实现 USB 接口的标准组件使得使用者可以在各种不同类型的微控制器中选择出一种最合适的微控制器。通过使用已有的结构和减少固件上的投资缩短了开发时间、减少了开发风险和费用。从而用最快捷的方法实现了最经济的 USB 外设的解决方案。

ISP1581 可理想地用于许多外设, 例如: 打印机、扫描仪、MO、CD、DVD 和 Zip/Jaz 驱动器、数码相机、USB 和以太网的链接、电缆和 DSL 调制解调器等等。另外, ISP1581 所具有的低挂起功耗还可以满足 ACPI™、OnNOW™ 和 USB 电源管理的要求。

此外, ISP1581 内部还集成了许多特性, 包括 SoftConnect™、低频晶体振荡器和集成的终止寄存器。所有这些特性都为系统大大节约了成本, 同时使强大的 USB 功能很容易地用于 PC 机外设。

2.特性

- 直接与 ATA/ATAPI 外设相连;
- 完全符合通用串行总线 (USB) Rev 2.0 规范;
- 符合大多数器件的分类规格;
- 高性能的 USB 接口器件, 集成了串行接口引擎 (SIE)、PIE、FIFO 存储器、数据收发器和 3.3V 的电压调整器;
- 支持高速 USB 的自检工作模式和最初 USB 规范的返回工作模式;
- 高速的 DMA 接口 (12.8M 字/秒);
- 完全自治的多结构 DMA 操作;
- 7 个 IN 端点, 7 个 OUT 端点和 1 个固定的控制 IN/OUT 端点;
- 集成 8K 字节的多结构 FIFO 存储器;
- 端点的双缓冲配置增加了数据吞吐量并轻松实现实时数据传输;
- 同大部分的微控制器/微处理器有单独的总线接口 (12.5M 位/秒);
- 集成了 PLL 的 12MHZ 的晶体振荡器,有着良好的 EMI 特性;
- 集成了 5V 到 3V 的内置电压调整器;
- 可通过软件控制与 USB 总线的连接 (SoftConnect™);
- 符合 ACPI™、OnNOW™ 和 USB 电源管理的要求;
- 可通过内部上电复位和低电压复位电路复位, 也可通过软件复位;
- 工作在扩展 USB 总线电压范围 (4.0~5.5) 内, I/O 端口最大可承受 5V 的电压;
- 操作温度: -40°C~+85°C;
- LQFP64 的封装形式。

3.应用

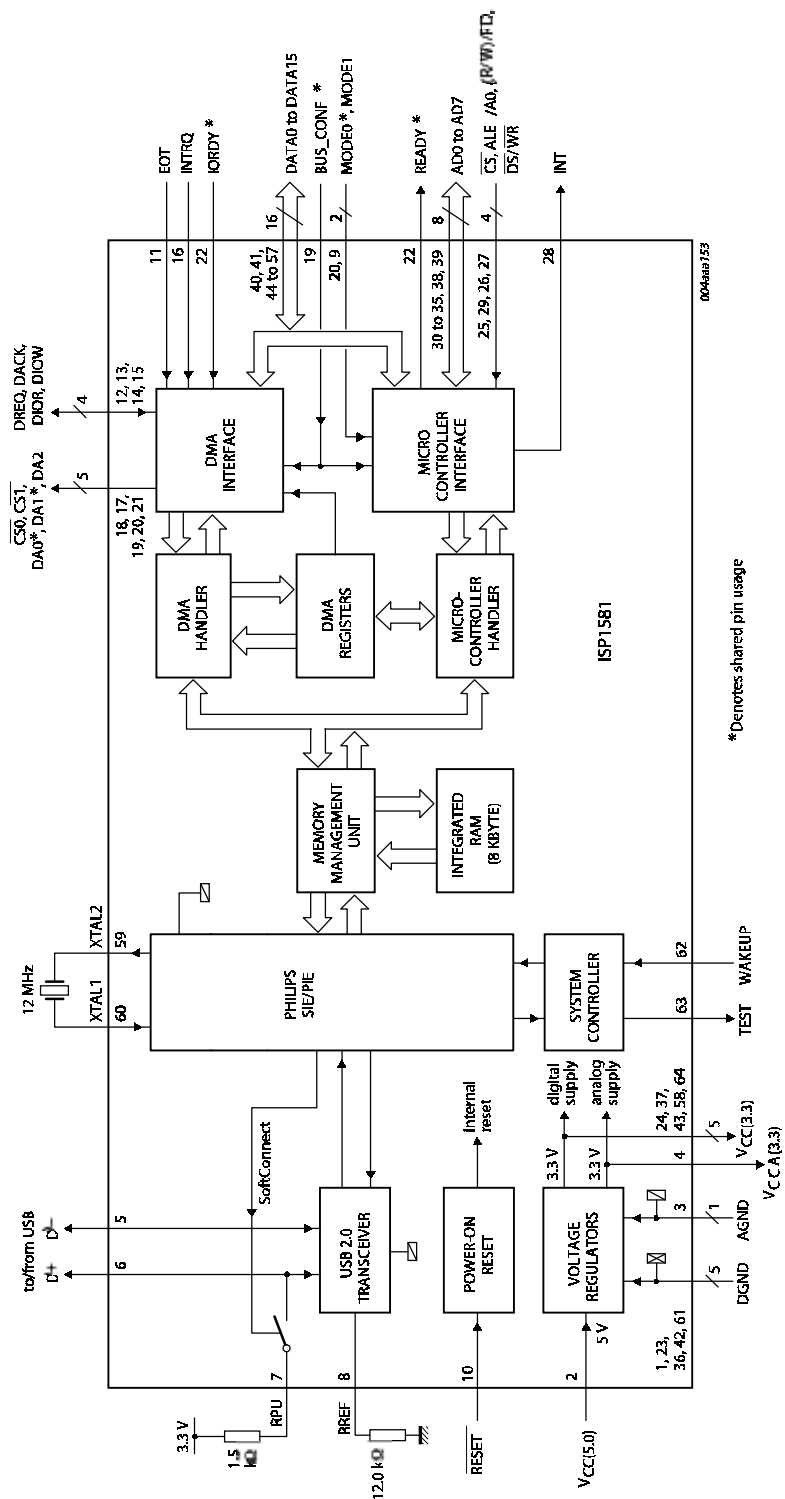
- 个人数字助理 (PDA);
- 海量存储器件, 如 Zip[®], Jaz[®], MO,CD,DVD 驱动器;
- 数字化视频相机;
- 数码相机;
- 3G 移动电话;
- MP3 播放器;
- 通信设备, 例如: 路由器, 调制解调器;
- 打印机;
- 扫描仪。

4.订购信息

表 1 订购信息

编号	封装	描述	版本
ISP1581BD	LQFP64	小塑料方形扁平封装; 64 脚; 外形尺寸: 10×10×1.4mm	SOT314-2

5. 方框图



DREQ, DACK, DIOR 和 DIOW 的方向由 MASTER 位 (DMA 硬件寄存器) 和 ATA_MODE 位 (DMA 配置寄存器) 控制。

图 1 方框图

6. 管脚信息

6.1 管脚

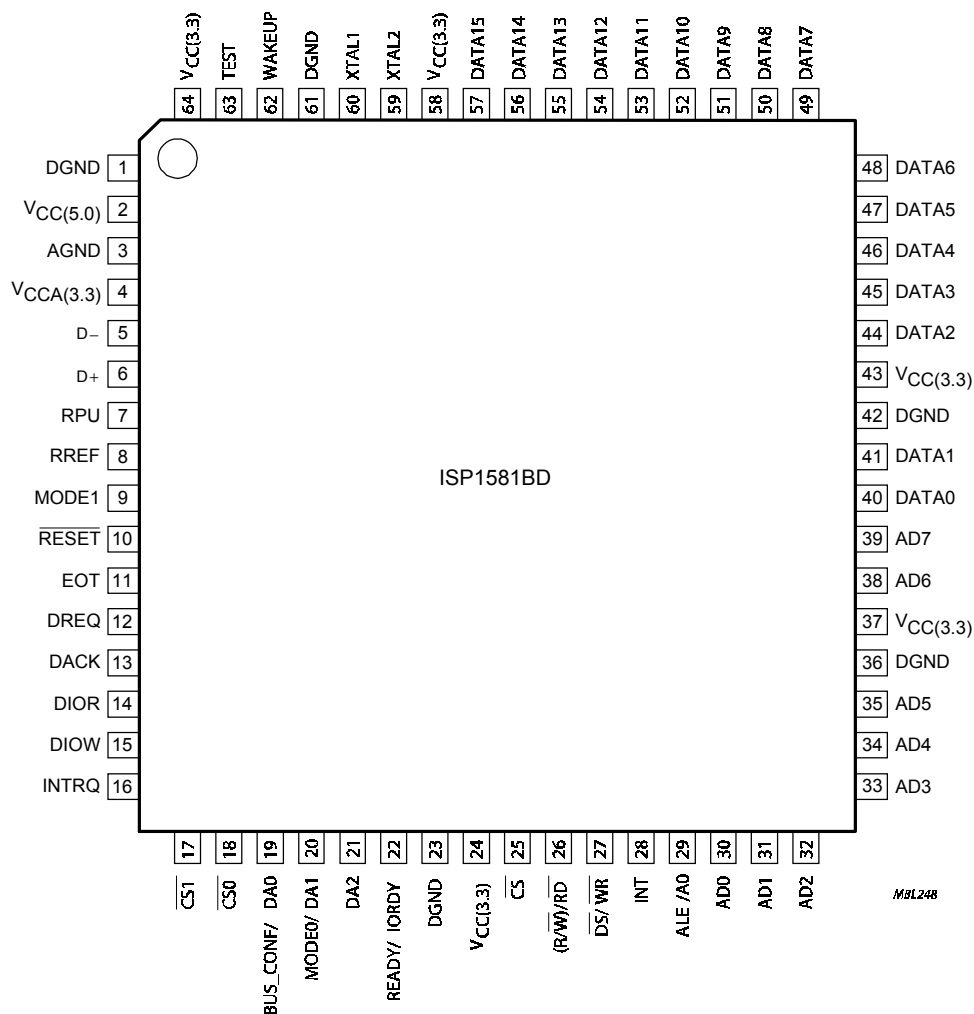


图 2 LQFP64 的管脚配置

6.2 管脚描述

表 2 LQFP64 封装的管脚描述

符号	管脚	类型	描述
DGND	1	—	数字地
$V_{CC(5.0)}$ ^[3]	2	—	供电电压 (3.3V 或 5V)。对于 5.0V 的操作, 只需使用这个引脚。可参考 10 号引脚
AGND	3	—	模拟地
$V_{CCA(3.3)}$ ^[3]	4	—	内部电压调整器输出的供电电压 (3.3V±0.3V); 给内部模拟电路供电; 与去耦电容和 D+线上的 1.5KΩ上拉电阻相连 注: 不能给外部器件供电。参考 10 号引脚功能

符号	管脚	类型	描述
D-	5	A	USB D-连接(模拟)
D+	6	A	USB D+连接 (模拟)
RPU	7	A	USB D+线的外部上拉电阻连接端, 通过一个 1.5KΩ 的电阻与 V _{CCA(3.3)} 相连
RREF	8	A	连接外部偏置电阻; 通过一个 12.0KΩ (±1%) 的电阻同地端相连
MODE1	9	I	ALE/A0 的功能选择 (仅在断开总线工作模式下); 0—ALE 功能 (地址锁存使能) 1—A0 功能 (地址/数据指示) 注: 在通用处理器工作模式下与 V _{CC(5.0)} 相连 输入引脚; TTL 电平; 最大可承受 5V 电压; 内部含下拉电阻
$\overline{\text{RESET}}$	10	I	复位输入端; 为低电平时产生一个异步复位信号; 与 V _{CC} 相连可产生上电复位 (内部的 POR 电路); 滞后的 TTL 电平; 5V 的最大承受电压; 内部含上拉电阻
EOT	11	I	传送终止输入 (触发极性可编程, 见表 37); 在 DMA 的从机工作模式下只用作输入端; TTL; 5V 的最大承受电压; 5ns 的斜率控制
DREQ	12	I/O	DMA 请求 (触发极性可编程); 信号传送方向由 DMA 硬件寄存器 (DMA 主机: 输入, DMA 从机: 输出) 的 MASTER 位来控制; 见表 35 和表 36 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压。
DACK	13	I/O	DMA 应答 (触发极性可编程); 信号传送方向由 DMA 硬件寄存器 (DMA 主机: 输入, DMA 从机: 输出) 的 MASTER 位来控制; 见表 35 和表 36 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DIOR	14	I/O	DMA 读选通信号 (触发极性可编程); 信号传送方向由 DMA 硬件寄存器 (DMA 从机: 输入, DMA 主机: 输出) 的 MASTER 位来控制; 见表 35 和表 36 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DIOW	15	I/O	DMA 写选通信号 (触发极性可编程); 信号传送方向由 DMA 硬件寄存器 (DMA 从机: 输入, DMA 主机: 输出) 的 MASTER 位来控制; 见表 35 和表 36 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
INTRQ	16	I	ATA/ATAPI 外设的中断请求; 输入端口; 滞后的 TTL 电平; 5V 的最大承受电压
$\overline{\text{CS1}}^{[5]}$	17	O	ATA/ATAPI 设备的输出片选; CMOS 输出; 5ns 的斜率控制; 见表 33 和表 34
$\overline{\text{CS0}}^{[5]}$	18	O	ATA/ATAPI 设备的输出片选; CMOS 输出; 5ns 的斜率控制; 见表 33 和表 34

符号	管脚	类型	描述
BUS_CON F/ DA0 ^[5]	19	I/O	<p>上电时: 选择总线结构</p> <p>0— 断开总线模式; AD[7:0]为多路复用的 8 位地址/数据总线, DATA[15:0]^[4]为单独的 DMA 数据总线</p> <p>1— 通用处理器工作模式; AD[7:0]为单独的 8 位地址线, DATA[15:0]为 16 位的处理器数据总线。DMA 多路复用到 DATA[15:0]的处理器数据总线上。</p> <p>正常操作: 地址输出, 用来选择 ATA/ATAPI 设备的任务文件寄存器。数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压; 见表 33 和表 34</p>
MODE0 DA1 ^[5]	20	I/O	<p>上电时: 输入端, 用来选择通用处理器工作模式下的读/写功能</p> <p>0—Motorola 类型: 管脚 26 是 $\overline{R/\overline{W}}$, 管脚 27 是 \overline{DS}</p> <p>1—8051 类型: 管脚 26 是 \overline{RD}, 管脚 27 是 \overline{WR}</p> <p>正常操作: 地址输出, 用来选择 ATA/ATAPI 设备的任务文件寄存器。数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压; 见表 33 和表 34</p>
DA2 ^[5]	21	O	地址输出, 用来选择 ATA/ATAPI 设备的任务文件寄存器; CMOS 输出; 5ns 的斜率控制; 见表 33 和表 34
READY/ IORDY	22	I/O	<p>通用处理器工作模式: 读信号 (READY; 输出)</p> <p>引脚为低电平表明 ISP1581 正在处理上次的命令或数据, 还没有准备好接收新的命令或数据; 引脚为高电平则表明 ISP1581 已经为下一次微处理器的读和写做好了准备。</p> <p>断开总线工作模式: DMA 读信号 (IORDY; 输入); 用来访问 ATA/ATAPI 外围设备 (仅在 PIO 和 UDMA 模式下)</p> <p>数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压</p>
DGND	23	-	数字地
V _{CC(3.3)} ^[3]	24	-	供电电压 (3.3V±0.3V); 它是内部调整器的输出电压, 给内部数字电路供电; 这个电压不能用来驱动外部器件; 见 10 号引脚功能
\overline{CS}	25	I	片选输入; TTL; 5V 的最大承受电压
($\overline{R/\overline{W}}$)/ \overline{RD}	26	I	<p>输入; 功能由上电时的 MODE0 决定</p> <p>MODE0 = 0—$\overline{R/\overline{W}}$ (Motorola 类型)</p> <p>MODE0 = 1—\overline{RD} (8051 类型)</p> <p>输入端; 滞后的 TTL 电平; 5V 的最大承受电压</p>
$\overline{DS}/\overline{WR}$	27	I	<p>输入; 功能由上电时的 MODE0 决定</p> <p>MODE0 = 0—\overline{DS} (Motorola 类型)</p> <p>MODE0 = 1—\overline{WR} (8051 类型)</p> <p>输入端; 滞后的 TTL 电平; 5V 的最大承受电压</p>
INT	28	O	中断输出; 触发极性可编程 (高电平或低电平有效) 和触发方式 (边沿触发或电平触发); CMOS 输出; 5ns 斜率控制

符号	管脚	类型	描述
ALE/A0	29	I	输入；由上电时 MODE1 的输入决定； MODE1 = 0—ALE 的功能(地址锁存使能)；下降沿到来时将多路复用地址/数据总线 (AD[7:0]) 上的地址锁存 MODE1 = 1—A0 的功能(AD[7:0]的地址/数据选择)；在 \overline{WR} 脉冲的上升沿检测到逻辑“1”，将 AD[7:0]作为地址寄存器；反之，为数据寄存器；仅使用在断开总线模式下。 注：在通用处理器模式下与 DGND 相连 输入端；TTL；5V 的最大承受电压。
AD0	30	I/O	多路复用地址/数据线的第 0 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
AD1	31	I/O	多路复用地址/数据线的第 1 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
AD2	32	I/O	多路复用地址/数据线的第 2 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
AD3	33	I/O	多路复用地址/数据线的第 3 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
AD4	34	I/O	多路复用地址/数据线的第 4 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
AD5	35	I/O	多路复用地址/数据线的第 5 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
DGND	36	-	数字地
$V_{CC(3.3)}$ ^[3]	37	-	供电电压 (3.3V±0.3V)；它是内部调整器的输出电压，给内部数字电路供电；这个电压不能用来驱动外部器件；见 10 号引脚功能
AD6	38	I/O	地址/数据复用的第 6 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
AD7	39	I/O	多路复用地址/数据线的第 7 位 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
DATA0	40	I/O	双向数据线第 0 位。 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
DATA1	41	I/O	双向数据线第 1 位。 数据双向传送端；推挽输出；5ns 的斜率控制；TTL；5V 的最大承受电压。
DGND	42	-	数字地

符号	管脚	类型	描述
V _{CC(3.3)} ^[3]	43	-	供电电压 (3.3V±0.3V); 它是内部调整器的输出电压, 给内部数字电路供电; 这个电压不能用来驱动外部器件; 见 10 号引脚功能
DATA2	44	I/O	双向数据线 2 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压。
DATA3	45	I/O	双向数据线第 3 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压。
DATA4	46	I/O	双向数据线第 4 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压。
DATA5	47	I/O	双向数据线第 5 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压。
DATA6	48	I/O	双向数据线第 6 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压。
DATA7	49	I/O	双向数据线第 7 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA8	50	I/O	双向数据线第 8 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA9	51	I/O	双向数据线第 9 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA10	52	I/O	双向数据线第 10 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA11	53	I/O	双向数据线第 11 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA12	54	I/O	双向数据线第 12 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA13	55	I/O	双向数据线第 13 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA14	56	I/O	双向数据线第 14 位。 数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压
DATA15	57	I/O	双向数据线第 15 位。数据双向传送端; 推挽输出; 5ns 的斜率控制; TTL; 5V 的最大承受电压

符号	管脚	类型	描述
V _{CC(3.3)} ^[3]	58	-	供电电压 (3.3V±0.3V); 它是内部调整器的输出电压, 给内部数字电路供电; 这个电压不能用来驱动外部器件; 见 10 号引脚功能
XTAL2	59	O	晶体振荡器输出 (12MHZ); 连接一个基本的并联振荡电路; 当 XTAL1 连接一个外部时钟源时该管脚悬空
XTAL1	60	I	晶体振荡器输入 (12MHZ); 连接一个基本的并联振荡电路或一个外部时钟源 (此时 XTAL2 悬空)
DGND	61	-	数字地
WAKEUP	62	I	唤醒输入 (边沿触发); 输入一个由低到高的电平跳变信号将系统从“挂起”状态唤醒。 输入端; 滞后的 TTL 电平; 内部有一个下拉电阻; 5V 的最大承受电压。
TEST	63	O	检测输出; 该脚只用于检测目的
V _{CC(3.3)} ^[3]	64	-	供电电压 (3.3V±0.3V); 它是内部调整器的输出电压, 给内部数字电路供电; 这个电压不能用来驱动外部器件; 见 10 号引脚功能

[1] 引脚符号的上面有一条横线 (如: $\overline{\text{NAME}}$), 表示低电平有效。

[2] 所有的输出端和 I/O 口都能够提供 4mA 的电流

[3] 在所有的供电引脚上增加一个去耦电容 (0.1uF)。为了得到良好的 EMI 性能, 可再增加一个 0.01uF 的电容, 并联到 0.1uF 的电容两端。

[4] DMA 总线为三态, 直到执行一个 DMA 命令。

[5] 控制信号不是三态的。

7. 功能描述

ISP1581 是一个高速 USB 器件控制器。它实现了高速 USB 和最初 USB 规范的物理层和数据协议层的任务, 并且实现了连同端点 EP0 设置(用于访问设置缓冲器)在内的 16 个 USB 端点的共同协作。USB chapter 9 的有关 USB 协议的处理是由外部固件来完成的。

ISP1581 有一个快速通用接口, 利用它可以实现与大部分类型的微控制器/处理器的通信。这个微控制器的接口由管脚 BUS_CONF, MODE1 和 MODE0 共同设置, 适用于大部分类型的接口。ISP1581 内部含有两种总线结构配置, 上电时由 BUS_CONF 输入管脚进行选择:

● 通用处理器工作模式 (BUS_CONF=1):

—AD[7:0]: 8 位地址总线 (选择目标寄存器)

—DATA[15:0]: 16 位数据总线 (处理器和 DMA 共享)

—控制信号: $\overline{\text{R/W}}$ 和 $\overline{\text{DS}}$ 或 $\overline{\text{RD}}$ 和 $\overline{\text{WR}}$ (由 MODE0 选择), $\overline{\text{CS}}$

—DMA 接口 (仅用在通用从机工作模式下): DATA[15:0]为数据总线, DIOR 和 DIOW 为读/写选通信号

● 断开总线工作模式 (BUS_CONF=0)

—AD[7:0]: 8 位本地微处理器总线 (多路复用地址/数据)

—DATA[15:0]: 16 位 DMA 数据总线

—控制信号: $\overline{\text{CS}}$, ALE 或 A0(由 MODE1 选择), $\overline{\text{R/W}}$ 和 $\overline{\text{DS}}$ 或 $\overline{\text{RD}}$ 和 $\overline{\text{WR}}$ (由 MODE0 选择)

—DMA 接口 (主机或从机模式): DIOR 和 DIOW 为读/写选通信号

ISP1581 和外部存储器或外部设备之间的高带宽的数据传输是通过集成的 DMA 控制器来控制完成的。通过“写”对应的 DMA 寄存器来配置 DMA 接口 (见 9.4)。

ISP1581 支持高速 USB 和最初 USB 规范的信道, USB 信道速度的检测是自动完成的。

ISP1581 含有一个 8K 字节的内部 FIFO 存储器, 供所有正在使用的 USB 端点共享。

ISP1581 含有 7 个 IN 端点, 7 个 OUT 端点和 2 个控制端点, 它们的固定长度是 64K 字节。7 个 IN 和

7 个 OUT 端点可以单独使用或禁止。这些端点的端点类型（中断，同步或批量）和信息包大小根据实际需要进行设置。利用双缓冲器的配置来增加这些数据端点的数据吞吐量。

ISP1581 根据 I/O 口的电压从 3.3V 和 5.0V 中选择一个作为供电电压。内部含有一个 3.3V 的电压调整器，用于模拟收发器的供电。ISP1581 的 I/O 引脚最大能承受 5.0V 的电压。

ISP1581 工作频率为 12MHZ。内部集成的一个 $40 \times \text{PLL}$ 时钟乘法器可产生 480MHZ 的内部抽样时钟。

7.1 高速 USB 收发器

模拟收发器通过集成的终止电阻直接与 USB 电缆相连。通过在管脚 RREF 和地之间接一个外部电阻 ($12.0\text{K}\Omega \pm 1\%$) 使高速收发器得到一个精确的镜电流，利用这个电流来产生 USB 2.0 的电流驱动。全速收发器也类似。由此可以看出，ISP1581 完全符合高速 USB 和最初的 USB 规范，支持高速和全速的物理层。每当速度自动检测完成后，串行接口引擎 (SIE) 就设置上述的收发器来处理高速或全速信号。

7.2 Philips 串行接口引擎 (SIE)

Philips SIE 控制完成所有 USB 协议层的功能，这些功能完全由硬件来实现而不需要固件的参与。SIE 模块主要完成以下功能：同步方式的识别，并行/串行转换，位填充/解除填充，CRC 校验 /产生，分组标识 (PID) 校验/产生，地址识别和握手评估/产生。

7.3 Philips HS (高速) 收发器

7.3.1 Philips 并行接口引擎

在 HS 收发器中，Philips PIE 接口使用了一个 16 位的并行双向数据接口。HS(高速)模块完成位填充/解除填充和 NRZI 编码/解码逻辑功能。

7.3.2 外部电路

在 ISP1581 内部，利用一个带隙标准电路和 RREF 电阻共同产生一个电流，利用得到的这个持续电流来驱动 HS(高速) 传送电路并给其它模拟电路提供一个偏置电压。这个电路还需要在外部连接一个阻值精确 ($12.0\text{K}\Omega \pm 1\%$) 的电阻，将其与模拟地相连。

7.3.3 HS 检测

在 USB 规范下，ISP1581 不单要处理一种电气状态 (FS/HS)。当设备和主控制器已经在 USB 电缆两端连接好了，ISP1581 器件就默认为全速 (FS) 状态直至主控制器发出一个总线复位信号。

在总线复位期间，器件发送一个高速(HS)模式检测信号来判断主控制器支持的是高速 USB 规范还是最初 USB 规范。在对这个信号进行处理时，必须连接一个上拉电阻，同时内部的终止寄存器禁用。如果信号处理结束后得到的是一个 HS 的握手信号，表明有一个 HS 主机与总线相连，那么 ISP1581 就转换到 HS 状态。

在 HS 状态下，ISP1581 要定时地对总线状态进行检测。如果总线未被使用的时间超过 3ms，设备就返回到 FS 状态来检查 USB 总线的 SE0(单端 0)。若检测到的 SE0 达到了指定时间(100us 到 875us, 见 USB 2.0 的 7.1.7.6)，ISP 1581 转换到高速(HS)模式信号检测状态做一个 HS 的握手检测。否则，ISP1581 为 FS 状态，总线挂起。

7.4 电压调整器

片内集成了两个 5V 到 3.3V 的电压调整器分别用于模拟收发器和内部逻辑电路的供电。这两个电压调整器的输出为 $V_{\text{CCA}(3.3)}$ 和 $V_{\text{CC}(3.3)}$ ，两者可以通过是使用在模拟电路还是数字电路中加以区分。 $V_{\text{CCA}(3.3)}$ 还可用于 D_+ 线上的外部上拉电阻的供电。

注： $V_{\text{CCA}(3.3)}$ 和 $V_{\text{CC}(3.3)}$ 不能用于外部器件的供电。

7.5 存储器管理单元 (MMU) 和集成 RAM

MMU 和集成 RAM 实现了 USB 速度 (全速: 12M 比特/秒, 高速: 480M 比特/秒) 和微控制器管理器或 DMA 控制器之间的转化。从 USB 总线上所得的数据存储在集成 RAM 中, 在微控制器或 DMA 控制器已经所有的数据从通信端点的缓冲区中读出或者把所有的数据写入到端点缓冲区之后, 数据被清除。缓冲区也可通过置位控制功能寄存器的 CLBUF 位强制清空。8K 字节的 RAM 也可充当缓冲区使用。

7.6 软件连接

设备与 USB 的连接是通过一个 1.5K Ω 的上拉电阻将 D₋线 (对于全速 USB 器件) 置为高来实现的。在 ISP1581 中, RPU 和 V_{CCA(3,3)} 之间连接了一个 1.5K Ω 的外部上拉电阻。上拉电阻再经 RPU 与 D₊线相连, 此时方式寄存器的 SOFTCT 被置位 (见表 7), 从而实现了软件连接。在一个硬件复位发生后, 上拉电阻默认为断开 (SOFTCT=0)。USB 总线复位时, SOFTCT 位的值仍保持不变。

7.7 微控制器/处理器接口和微控制器/处理器的管理器

微控制器接口可以实现其自身与大部分微控制器的直接连接。上电时, 通过 BUS_CONF, MODE1 和 MODE0 对接口进行设置。

BUS_CONF 设为 1, 微控制器接口为通用处理器模式, 在这种模式下, AD[7:0] 是 8 位地址总线, DATA[15:0] 是独立的 16 位数据总线。BUS_CONF 设为 0, 接口为断开总线模式, 这时 AD[7:0] 为本地微处理器总线 (多路复用地址/数据), DATA[15:0] 只作为 DMA 总线使用。

MODE0 设为 1, 读和写选通信号为 \overline{RD} 和 \overline{WR} (8051 类型)。MODE0 为 0, $\overline{R/\overline{W}}$ 和 \overline{DS} 为方向和数据选通信号 (Motorola 类型)。

MODE1 为 0 时, ALE 将 AD[7:0] 复用的地址锁存。MODE1 为 1 时, A0 用来指示传送的地址或数据。MODE1 只用在断开总线模式: 在通用处理器模式下将它与 V_{CC(5,0)} (逻辑 1) 相连。

微控制器的管理器允许外部微控制器对 Philips SIE 和 DMA 控制器内的寄存器进行访问。DMA 配置的初始化由微控制器的管理器完成。

7.8 DMA 接口和 DMA 控制器

DMA 模块可以细分成两部分: DMA 控制器和 DMA 接口。

固件通过“写”DMA 命令寄存器来启动一次 DMA 传送 (见表 28)。命令操作码决定 DMA 的传送方式: 通用 DMA, PIO, MDMA 或 UDMA。由于控制器和 USB 内核使用的是同一个 FIFO (内部 RAM), 所以 DMA 控制器接收到 DMA 命令后, 可直接控制数据从内部 RAM 传送到外部 DMA 设备或从外部 DMA 设备传送给内部 RAM。

DMA 接口的配置包括 DMA 时序和 DMA 握手。数据传送通过 DIOR 和 DIOW 选通信号或 DACK 和 DREQ 握手信号来实现。通过“写”DMA 配置寄存器来决定不同的 DMA 配置 (见表 33 和表 34)。

对于基于 IDE 的存储接口来说, 用到的 DMA 模式有 PIO (并行 I/O), MDMA (多字 DMA: ATA), 和 UDMA (增强 DMA: ATA)。

对于通用 DMA 接口来说, 用到的 DMA 模式有通用 DMA (从机) 模式和 MDMA (主机) 模式。

8. 运行模式

ISP1581 有两种总线配置方式, 由上电时 BUS_CONF/DA0 来决定:

- 总线断开模式 (BUS_CONF=0): 多路复用 8 位地址/数据总线和单独的 8 位/16 位 DMA 总线
- 通用处理器模式 (BUS_CONF=1): 单独的 8 位地址和 16 位数据总线

每一种模式总线配置的详细情况如表 3 所示, 两者的典型接口电路见 15 部分。

表 3: 总线配置模式

BUS_CONF	PIO 宽度	DMA 宽度		描述
		DMAWD=0	DMAWD=1	
0	AD[7:0]	D[7:0]	D[15:0]	断开总线模式: AD[7:0] 为多路复用地址/数据; DATA[15:0]为单独的 8/16 位 DMA 总线
1	AD[7:0] D[15:0]	D[7:0]	D[15:0]	通用处理器模式: AD[7:0]为单独的 8 位地址; DATA[15:0]为 16 位数据 (PIO 和 DMA)

9. 寄存器描述

表 4: 寄存器汇总

名称	接收者	地址 (Hex)	描述	大小 (字节)
需预置初值的寄存器				
地址	器件	00	USB 器件地址+使能	1
模式	器件	0C	低功耗选择, 全局中断使能, 软件连接	1
中断设置	器件	10	中断源, 触发方式, 输出极性	1
中断使能	器件	14	激活中断源	4
DMA 配置	DMA 控制器	38	见下面的“DMA 寄存器”	2
DMA 硬件	DMA 控制器	3C	见下面的“DMA 寄存器”	1
数据流寄存器				
端点索引	端点	2C	端点选择, 数据流方向	1
控制功能	端点	28	端点缓冲区管理	1
数据端口	端点	20	端点 FIFO 的数据通道	2
缓冲区长度	端点	1C	包大小计数器	2
端点 MaxPacketSize	端点	04	最大数据包长度	2
端点类型	端点	08	选择端点类型: 控制, 同步, 批量或中断	2
短包	端点	24	从 OUT 端点接收的短包	2
DMA 寄存器				
DMA 命令	DMA 控制器	30	控制所有的 DMA 传输	1
DMA 传输计数器	DMA 控制器	34	DMA 传输字节计数	4
DMA 配置	DMA 控制器	38 (字节 0)	选择 GDMA 配置 (计数器使能, 脉冲时间, 数据选通, 总线宽度)	1
		39 (字节 1)	选择 ATA 配置 (IORDY 使能, 方式选择: ATA/UDMA/MDMA/PIO)	1
DMA 硬件	DMA 控制器	3C	主/从选择, DACK,DREQ,DIOW,DIOR 极性选择	1
1F0 任务文件	ATAPI 外设	40	单一地址字寄存器: 先访问字节 0 (低字节)	
1F1 任务文件	ATAPI 外设	48	访问 IDE 器件	1

名称	接收者	地址 (Hex)	描述	大小 (字节)
1F2 任务文件	ATAPI 外设	49	访问 IDE 器件	1
1F3 任务文件	ATAPI 外设	4A	访问 IDE 器件	1
1F4 任务文件	ATAPI 外设	4B	访问 IDE 器件	1
1F5 任务文件	ATAPI 外设	4C	访问 IDE 器件	1
1F6 任务文件	ATAPI 外设	4D	访问 IDE 器件	1
1F7 任务文件	ATAPI 外设	44	访问 IDE 器件 (只能写, 读时返回 FFH)	1
3F6 任务文件	ATAPI 外设	4E	访问 IDE 器件	1
3F7 任务文件	ATAPI 外设	4F	访问 IDE 器件	1
DMA 中断源	DMA 控制器	50(字节 0)	DMA 中断原因 (中断源)	1
		51(字节 1)		1
DMA 中断使能	DMA 控制器	54 (字节 0)	激活 DMA 中断源	1
		55 (字节 1)		1
DMA 端点	DMA 控制器	58	选择 FIFO 端点, 数据流方向	1
DMA 选通时间	DMA 控制器	60	UDMA/MDMA 模式下的选通时间	1
常规寄存器				
中断	器件	18	列出中断源	4
芯片 ID	器件	70	器件 ID 和硬件版本	3
帧数目	器件	74	最后成功接收到的数据的帧开始: 低字节 (byte0) 在前	2
暂存	器件	78	在总线的挂起状态下存储/恢复固件的状态	2
检测模式	PHY	84	直接设置 D+,D_状态, 循环返回模式, 内部收发器检测 (PHY)	1

9.1 寄存器访问

寄存器的访问与使用的总线宽度有关:

- 8 位总线: 多字节寄存器先访问低字节 (LSByte)。
- 16 位总线: 访问单字节寄存器时将高字节忽略 (MSByte)。

端点索引寄存器将端点的一些特定的寄存器编入索引。在对这些寄存器进行访问之前先要选好目标端点:

- 缓冲区长度
- 控制功能
- 数据端口
- 端点 MaxPacketSize
- 端点类型
- 短包

注: 以上所有寄存器保留位的功能都不会执行。由于没有对总线和总线的复位进行定义, 所以设置这些保留位不会对系统产生任何影响。

9.2 需预置初值的寄存器

9.2.1 地址寄存器（地址：00H）

这个寄存器用来设置 USB 的分配地址并激活 USB 设备。它的各位分配如表 5 所示。

只要出现总线复位、上电复位和软件复位三者之一，DEVEN 和 DEVADDR 清零。

为了响应标准 USB 的 SET_ADDRESS 请求，固件必须先将（激活）设备地址写入地址寄存器，再发送一个空包给主机。当主机识别空包后，这个“新”的设备被激活。

表 5：地址寄存器：位分配

位	7	6	5	4	3	2	1	0
符号	DEVEN	DEVADDR[6:0]						
复位	0	00H						
总线复位	0	00H						
访问	R/W	R/W						

表 6：端点设置寄存器：位描述

位	符号	描述
7	DEVEN	为逻辑 1 时激活设备
6 到 0	DEVADDR[6:0]	指定的 USB 设备地址

9.2.2 方式寄存器（地址：0CH）

这是一个单字节的寄存器（位分配：见表 7）。在 16 位总线模式下对其进行访问时忽略高字节。

方式寄存器控制着重新开始、挂起和唤醒行为、中断活动、软件复位、时钟信号和软件连接操作。

表 7：方式寄存器：位分配

位	7	6	5	4	3	2	1	0
符号	CLKAON	SNDRSU	GOSUSP	SFRESET	GLINTENA	WKUPCS	保留	SOFTCT
复位	0	0	0	0	0	0	-	0
总线复位	0	0	0	0	不变	0	-	不变
访问	R/W	R/W	R/W	R/W	R/W	R/W	-	R/W

表 8：方式寄存器：位描述

位	符号	描述
7	CLKAON	时钟始终有效：逻辑 1 时，在“挂起”状态下内部时钟仍然有效。逻辑 0 时，无需使用内部振荡器和 PLL 时，使其停止工作。在“挂起”状态期间，为了满足当前的要求，这一位必须置为 0。GOSUSP 置位后，时钟延时大约 2ms 之后停止工作。
位	符号	描述
6	SNDRSU	发送重新开始信号：输入一个由“1”到“0”的跳变，延时 5ms 后产生一个 10ms 的上行“重新开始”信号。
5	GOSUSP	进入挂起状态：输入一个由“1”到“0”的跳变使系统进入“挂起”状态。
4	SFRESET	软件复位：输入一个由“1”到“0”的跳变激活 ISP1581 产生一次软件复位。软件复位的情况类似于硬件复位方式（经 $\overline{\text{RESET}}$ 管脚）。
3	GLINTE NA	全局中断使能：为“1”时使能所有中断，单个中断可以通过清除中断使能寄存器中相应的位来屏蔽。总线复位时值不变。
2	WKUPCS	片选唤醒：为“1”时向 $\overline{\text{CS}}$ 端输入低电平发送远程唤醒信号
1	-	保留；必须写 0
0	SOFTCT	软件连接：为“1”时将 RPU 的 1.5K Ω 的上拉电阻与 D ₊ 相连。总线复位时值不变

9.2.3 中断配置寄存器（地址：10H）

这是一个单字节的寄存器，它决定了 INT 输出的动作和极性。位分配见表 9。当 USB SIE 接收或产生一个 ACK,NAK 或 STALL，就根据以下三种调试方式位域来产生中断：

- CDBGMOD[1:0]：控制端点 0 中断。
- DDBGMODIN[1:0]：端点 1 到 7 的 DATA IN 中断。
- DDBGMODOUT[1:0]：端点 1 到 7 的 DATA OUT 中断。

CDBGMOD,DDBGMODIN 和 DDBGMODOUT 的调试方式允许用户分别进行设置，操作时 ISP1581 将向外部微处理器发送一个中断请求。表 11 列出了各种可能的组合。

寄存器的 INTPOL 位控制 INT 输出的信号极性（高/低电平有效，上升沿/下降沿）。若选择电平触发方式，INTLVL 必须为 0。若 INTLVL 设为 1 将产生一个 60ns 的脉冲（边沿触发）触发产生中断。

表 9：中断配置寄存器：位分配

位	7	6	5	4	3	2	1	0
符号	CDBGMOD[1:0]		DDBGMODIN[1:0]		DDBGMODOUT[1:0]		INTLVL	INTPOL
复位	03H		03H		03H		0	0
总线复位	03H		03H		03H		不变	不变
访问	R/W		R/W		R/W		R/W	R/W

表 10：中断配置寄存器：位描述

位	符号	描述
7 到 6	CDBGMOD[1:0]	控制 0 调试方式：值见表 11
5 到 4	DDBGMODIN[1:0]	数据调试方式 IN：值见表 11
3 到 2	DDBGMODOUT[1:0]	数据调试方式 OUT：值见表 11
1	INTLVL	中断电平：选择 INT 输出信道方式（0=电平，1=脉冲）。若为脉冲方式，一个中断会产生一个 60ns 的脉冲。总线复位时值不变
0	INTPOL	中断极性：选择 INT 输出信号极性（0=低电平有效，1=高电平有效）。总线复位时值不变。

表 11：调试方式设置

值	CDBGMOD	DDBGMODIN	DDBGMODOUT
00H	中断发生在所有的 ACK 和 NAK 分组	中断发生在所有的 ACK 和 NAK 分组	中断发生在所有的 ACK,NYET 和 NAK 分组
值	CDBGMsOD	DDBGMODIN	DDBGMODOUT
01H	中断发生在所有的 ACK 分组	中断发生在 ACK 分组	中断发生在 ACK 和 NYET 分组
1XH	中断发生在所有的 ACK 分组和第一个 NAK ^[1] 分组	中断发生在所有的 ACK 分组和第一个 NAK ^[1] 分组	中断发生在所有的 ACK,NYET 和第一个 NAK ^[1] 分组

[1]第一个 NAK：发生在 IN 或 OUT 令牌处理的前一个 ACK 响应后的第一个 NAK

9.2.4 中断使能寄存器（地址：14H）

这个寄存器用来激活/禁止单个中断源。寄存器中的 IEPnRX 或 IEPnTX 位（‘n’代表端点号）控制着与之相对应的端点的中断。控制方式寄存器的 GLINTENA 位（见表 7）可以屏蔽所有的中断。

当 USB SIE 在 USB 总线上接收或产生一个 ACK 或 NAK，就产生一次中断。中断的产生还依赖于位域 CDBGMOD,DDBGMODIN 和 DDBGMODOUT 调试方式设置。

所有数据的 IN 处理通过位 DDBGMODIN 控制的发送缓冲区（TX）来实现。所有数据的 OUT 处理则是通过位 DDBGMODOUT 控制的接收缓冲区来实现。而位 CDBGMOD 用于控制端点 0 的传输（IN,OUT

和 SETUP)。

由 USB 总线上的事件 (SOF, 假 SOF, 挂起, 重新开始, 总线复位, Setup 和高速状态) 所产生的中断也能够被分别控制。除位 IEBRST (总线复位) 控制的中断之外, 总线复位信号可以将所有激活的中断禁止。总线复位时 IEBRST 的值不变。

中断使能寄存器包含 4 个字节。它的位分配如表 12 所示。

表 12: 中断使能寄存器: 位分配

位	31 30 29 28 27 26						25	24
符号	保留						IEP7TX	IEP7RX
复位	-	-	-	-	-	-	0	0
总线复位	-	-	-	-	-	-	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	23	22	21	20	19	18	17	16
符号	IEP6TX	IEP6RX	IEP5TX	IEP5RX	IEP4TX	IEP4RX	IEP3TX	IEP3RX
复位	0	0	0	0	0	0	0	0
总线复位	0	0	0	0	0	0	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	15	14	13	12	11	10	9	8
符号	IEP2TX	IEP2RX	IEP1TX	IEP1RX	IEP0TX	IEP0RX	保留	IEP0SETUP
复位	0	0	0	0	0	0	-	0
总线复位	0	0	0	0	0	0	-	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	7	6	5	4	3	2	1	0
符号	保留	IEDMA	IEHS_STA	IERESM	IESUSP	IEPSOF	IESOF	IEBRST
复位	-	0	0	0	0	0	0	0
总线复位	-	0	0	0	0	0	0	不变
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

表 13: 中断使能寄存器: 位描述

位	符号	描述
31 到 26	-	保留; 必须写入 0
25 到 12	IEP7TX 到 IEP1RX	为 1, 激活指定端点的中断
11	IEP0TX	为 1, 激活控制 IN 处理端点 0 的中断
10	IEP0RX	为 1, 激活控制 OUT 处理端点 0 的中断
9	-	保留
8	IEP0SETUP	为 1, 激活端点 0 的启动数据接收中断
7	-	保留
6	IEDMA	为 1, 激活检测到 DMA 状态变化的中断
5	IEHS_STA	为 1, 激活检测到一个高速状态变化的中断
4	IERESM	为 1, 激活检测到一个“重新开始”状态的中断
3	IESUSP	为 1, 激活检测到一个“挂起”状态的中断
2	IEPSOF	为 1, 激活检测到一个伪 SOF 的中断
1	IESOF	为 1, 激活检测到一个 SOF 的中断

位	符号	描述
0	IEBRST	为 1, 激活检测到一个总线复位的中断

9.2.5 DMA 配置寄存器 (地址: 38H)

见 9.4.3。

9.2.6 DMA 硬件寄存器 (地址: 3CH)

见 9.4.4。

9.3 数据流寄存器

9.3.1 端点索引寄存器 (地址: 2CH)

端点寄存器是一个单字节的寄存器, 它为微控制器对寄存器的访问提供了目标端口。位分配如表 14 所示。以下是被编入了索引的寄存器:

- 端点 MaxPacketSize
- 端点类型
- 缓冲区长度
- 数据端口
- 短包
- 控制功能

例如, 向端点索引寄存器写入了 02H, 就可通过数据端口寄存器对端点 1 的 OUT 数据缓冲区进行访问。

表 14: 端点索引寄存器: 位分配

位	7	6	5	4	3	2	1	0
符号	保留		EPOSETUP	ENDPIDX[3:0]			DIR	
复位	-	-	0	00H			0	
总线复位	不变							
访问	R/W	R/W	R/W	R/W			R/W	

表 15: 端点寄存器: 位描述

位	符号	描述
7 到 6	-	保留
5	EPOSETUP	为端点 0 选择 SETUP 缓冲区: 0—EPO 数据缓冲区 1—SETUP 缓冲区
4 到 1	ENDPIDX[3:0]	端点索引: 为缓冲器长度、控制功能、数据端口、端口类型、MaxPacketSize 规定值和短包等寄存器的访问选择目标端点
0	DIR	方向位: 将目标端点设置成 IN 或 OUT 端点: 0—设置成 OUT(RX)FIFO 目标端点 1—设置成 IN(TX)FIFO 目标端点

表 16: 端点 0 缓冲区的编址

缓冲区名称	EPOSETUP	ENDPIDX	DIR
SETUP	1	00H	0
Data OUT	0	00H	0
Data IN	0	00H	1

9.3.2 控制功能寄存器 (地址: 28H)

控制功能寄存器是个单字节的寄存器, 它的作用是管理端点的缓冲区。各位的分配见表 17。这个寄存器的各位可以使任何一个激活的数据端点终止、清除或生效。在访问该寄存器之前, 必须先“写”端点索引寄存器来指定目标端点。

表 17: 控制功能寄存器: 位分配

位	7	6	5	4	3	2	1	0
符号	保留			CLBUF	VENDP	保留	STATUS ^[1]	STALL
复位	-	-	-	0	0	-	0	0
总线复位	-	-	-	0	0	-	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[1]仅应用在控制 IN/OUT。

表 18: 控制功能寄存器: 位描述

位	符号	描述
7 到 5	-	保留
4	CLBUF	清除缓冲区: 为 1, 清除索引端点的 RX 缓冲区; 对 TX 缓冲器无影响。一旦端点内容完全读出, RX 缓冲区自动清零。这一位通常用在需要强制清除缓冲区的情况下。
3	VENDP	端点有效: 为 1, 使一个 IN 端点的 TX FIFO 数据有效, 为下一个 IN 令牌的处理发送做准备。一般而言, 当端点的 FIFO 字节数达到端点 MaxPacketSize 规定值时端点自动生效。这一位通常用在当端点的 FIFO 字节数低于端点 MaxPacketSize 规定值时仍要求端点有效的情况下。
2	-	保留
1	STATUS	状态响应: 这一位用于控制在 SETUP 传输状态阶段 ACK 或 NAK 信号的产生。在该状态结束和接收到一个 SETUP 令牌时自动清除。 0—发送 NAK 1—IN 令牌 (主机到设备) 处理之后发送一个空包或 OUT 令牌处理 (设备到主机) 之后发送 ACK 信号。
0	STALL	终止端点: 为 1, 终止索引的端点。这一位不能用于同步传输。 注: ‘终止’ 一个数据端点会干扰被停止了了的端点的数据触发位, 因为内部逻辑就是从端点终止处获得的。所以, 必须先通过禁止或再次激活相应的端点来复位数据触发位 (通过将端点类型寄存器的 ‘ENABLE’ 位置为 0 或 1), 进而复位 PID。

9.3.3 数据端口寄存器 (地址: 20H)

这是个 2 字节的寄存器, 利用它可以实现一个微控制器对索引端点的 FIFO 的直接访问。总线为 8 位时高字节不使用。它各位的分配见表 19。

设备到主机 (IN 端点): 每执行完一次写操作, 内部计数器的值就自动增加 (16 位通道每次加 2, 8 位通道每次加 1) 到 TX FIFO 的下一个存储单元。当写完所有的字节 (FIFO 字节计数=端点 MaxPacketSize 规定值), 缓冲区自动有效, 数据包在下一个 IN 令牌时发送。如果要使得端点字节数小于 MaxPacketSize 规定值时端点有效, 可通过控制功能寄存器 (VENDP 位) 来实现。

主机到设备 (OUT 端点): 每执行完一个读操作, 内部计数器的值自动减少 (16 位通道每次减 2, 8 位通道每次减 1) 到 RX FIFO 的下一个存储单元。当所有字节内容读出后, 缓冲区的内容自动清除。这样, 一个新的数据包要在下一个 OUT 令牌处理时接收。如果需要强制清除缓冲区的内容, 通过控制功能寄存器 (CLBUF 位) 来实现。

注: 缓冲区的有效或自动清除还可通过缓冲区长度寄存器实现 (见表 21)。

表 19: 数据端口寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	DATAPORT[15:8]							
复位	00H							
总线复位	00H							

访问	R/W							
位	7	6	5	4	3	2	1	0
符号	DATAPORT[15:0]							
复位	00H							
总线复位	00H							
访问	R/W							

表 20: 数据端口寄存器: 位描述

位	符号	描述
15 到 8	DATAPORT[15:8]	数据 (高字节); 不能用在 8 位总线模式下
7 到 0	DATAPORT[7:0]	数据 (低字节)

9.3.4 缓冲区长度寄存器 (地址: 1CH)

这是个 2 字节的寄存器, 它决定了索引端点 FIFO 的当前包大小 (DATACOUNT)。各位的分配见表 21。

当端点 MaxPacketSize 寄存器写入数据后 (见表 22), 缓冲区长度寄存器自动装载 FIFO 的大小。需要时还可将一个更小的值写入到寄存器中。总线复位后缓冲区长度寄存器值为零。

IN 端点: 当传输的数据量倍数于 MaxPacketSize 的值时, 缓冲区长度寄存器就没有意义。只有传输的数据量不是 MaxPacketSize 的值的倍数时才需使用该寄存器。以下的两个例子说明了缓冲区长度寄存器在传输过程中的使用。

例 1: 假设要传输 512 字节的数据, MaxPacketSize 的编程设定值为 64 字节, 则不需要给缓冲区长度寄存器赋值。这是因为此时的传输数据量倍数于 MaxPacketSize 的值, 传输的最后一个数据包就是 MaxPacketSize 的值所指定的, 所以 MaxPacketSize 数据包将自动有效。

例 2: 假设要传输 510 字节的数据, MaxPacketSize 的编程设定值为 64 字节, 则在 MCU 写入最后一个 62 字节的数据包之前将缓冲区长度寄存器赋值 62 字节。这样就可确保最后一个数据包, 62 字节的短包, 自动有效。

以上操作仅适用于 PIO 模式的访问。

OUT 端点: DATACOUNT 的值自动初始化成每一个 ACK 时主机发送的数据字节数。

注: 如果使用的是 16 位的微处理器总线, 一个奇数大小的包的最后一个字节当作低字节输出 (LSByte)。

表 21: 缓冲区长度寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	DATAPORT[15:8]							
复位	00H							
总线复位	00H							
访问	R/W							
位	7	6	5	4	3	2	1	0
符号	DATAPORT[15:0]							
复位	00H							
总线复位	00H							
访问	R/W							

9.3.5 端点 MaxPacketSize 寄存器 (地址: 04H)

该寄存器决定了除控制管道 0 之外的所有端口的最大数据包长度。寄存器包含 2 个字节, 它的位分配如表 22 所示。

每次写完寄存器, 所有端点的缓冲区长度寄存器重新初始化来改变 FFOSZ 域的值。NTRANS 位控制着一个单片微帧 (仅适用于高速同步和端点中断) 的信息处理数量。

表 22: 端点 MaxPacketSize 寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	保留			NTRANS[1:0]		FFOSZ[10:8]		
复位	-	-	-	00H		00H		
总线复位	-	-	-	00H		00H		
访问	R/W	R/W	R/W	R/W		R/W		
位	7	6	5	4	3	2	1	0
符号	FFOSZ[7:0]							
复位	00H							
总线复位	00H							
访问	R/W							

表 23: 端点 MaxPacketSize 寄存器: 位描述

位	符号	描述
15 到 13	保留	保留
12 到 11	NTRANS[1:0]	处理信息数量 (仅适用于 HS 模式): 0—每微帧 1 个信息包 1—每微帧 2 个信息包 2—每微帧 3 个信息包 3—保留 这些位仅应用于同步/中断处理中。
10 到 0	FFOSZ[10:0]	FIFO 大小: 为索引端点以字节形式设置 FIFO 的大小。应用在 HS 和 FS 操作中。

9.3.6 端点类型寄存器 (地址: 08H)

该寄存器用于设置索引端点的端点类型: 同步, 批量或中断。它还可以使能端点和设置双缓冲区, 并可通过位 NOEMPKT 使一个长度为零的 TX 缓冲器空包的自动产生禁止。该寄存器包含两个字节, 它的位分配如表 24 所示。

表 24: 端点类型寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	保留							
复位	-	-	-	-	-	-	-	-
总线复位	-	-	-	-	-	-	-	-
访问	R/W							
位	7	6	5	4	3	2	1	0
符号	保留			NOEMPKT	ENABLE	DBLBUF	ENDPTYP[1:0]	
复位	-	-	-	0	0	0	00H	
总线复位	-	-	-	0	0	0	00H	
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

表 25: 端点类型寄存器: 位描述

位	符号	描述
15 到 5	保留	保留
4	NOEMPKT	无空包: 如果缓冲区长度寄存器或端点 MaxPacketSize 寄存器内容为 0, 该位为 0 表示将一个空包添加到 USB 数据的下一个 IN 令牌。为 1 时这个功能禁止。这位仅应用于 DMA 模式。

位	符号	描述
3	ENABLE	端点使能: 为 1, 使能索引端点的 FIFO。在端点 MaxPacketSize 寄存器中指定存储区大小。为 0 禁用 FIFO。 注: ‘终止’ 一个数据端点会干扰被停止了了的端点的数据触发位, 因为内部逻辑就是从端点终止处获得的。所以, 必须先通过禁止或再次激活相应的端点来复位数据触发位 (通过将端点类型寄存器的 ‘ENABLE’ 位置为 0 或 1), 进而复位 PID。
2	DBLBUF	双缓冲区设置: 为 1, 使能索引端点的双缓冲区设置。为 0 禁止双缓冲区设置
1 到 0	ENDPTYP[1:0]	端点类型: 这 2 位用于选择端点类型: 01H—同步 02H—批量 03H—中断

9.3.7 短包寄存器 (地址: 24H)

该寄存器保留。

9.4 DMA 寄存器

通过往 DMA 命令寄存器写入合适的操作码可以实现两字节的通用 DMA 传输和三字节的 IDE 规范传输。分别在表 26 (通用 DMA 传输) 和表 27 (IDE 规范传输) 给出了它们的控制位。

GDMA 读/写 (操作码=00H/01H) —通用 DMA 从机方式; 通过 DMA 配置寄存器的 MODE[1:0]位进行设置, 数据选通信号是 DACK 或 DIOR/DIOW。两个信号都是由外部 DMA 控制器驱动的。

GDMA 从机方式在计数器模式或 EOT 单独模式下有效。

在计数器模式下, DMA 配置寄存器的 DIS_XFER_CNT 位必须为 0。DMA 传输计数器寄存器必须在 DMA 命令发布之前初始化。DMA 传输计数器的初始化就是 “写” LSByte 到 MSByte (地址: 34H 到 37H) 的值。DMA 传输计数值只有在写完 MSByte 后才能开始更新。一旦 DMA 传输启动, 传输计数器的值开始减少, 计数值变为 ‘0’ 时 DMA_XFER_OK 置位, ISP1581 产生中断。若此时 DMA 主机要终止 DMA 传输, 可以发送一个 EOT 信号给 ISP1581。这个 EOT 信号不考虑传输计数器而随时终止 DMA 传输。

在 EOT 单独模式下, DIS_XFER_CNT 置 1。此时即使对 DMA 传输计数器进行编程也不会对 DMA 传输产生任何影响。DMA 命令一发布, 就启动 DMA 传输。只要出现以下三种情况中的任何一种都会终止此次 DMA 传输:

- 检测到一个外部 EOT 信号
- 检测到一个内部 EOT(OUT 令牌的短包)
- 复位 DMA

有 3 个基本的可编程中断可以区分上述的三种 DMA 传输终止方式: DMA 中断源寄存器的 INT_EOT, EXE_EOT 和 DMA_XFER_OK 位。详细内容参照表 53。

MDAM(主机)读/写 (操作码=06H/07H) —通用 DMA 主机方式; 通过 DMA 配置寄存器中的 MODE[1:0] 进行设置, 数据选通信号是 DACK 或 DIOR/DIOW。这两个信号是由 ISP1581 驱动的。

在主机方式下, 没有使用 DMA 配置寄存器的 BURST[2:0], DIS_XFER_CNT 和外部 EOT 信号。DMA 传输计数器一直有效, 一旦计数器的值为 ‘0’, DMA_XFER_OK 置 ‘1’。

PIO 读/写 (操作码=04H/05H) —IDE 传输的 PIO 方式; 该方式的详细说明见 ATA 规范 Rev.4。数据选通信号是 DIOR 和 DIOW, IORDY 被用于扩展 PIO 周期。

MDMA 读/写 (操作码=06H/07H) —IDE 传输的多字 DMA 模式; 该方式的详细说明见 ATA 规范 Rev.4。数据选通信号是 DIOR 和 DIOW, 握手信号是 DREQ 和 DACK。

UDMA 读/写 (操作码=(02H/03H)) —IDE 传输的增强 DMA 模式; 该方式的详细说明见 ATA 规范 Rev.4。引脚 DA0 到 DA2, CS0 和 CS1 用来选择访问的寄存器。它的控制信号如下所示: DREQ(=DMARQ), DACK(=DMACK), DIOW(=STOP), DIOR(=HDMARDY 或 HSTROBE), IORDY(=DSTROBE)

或 DDMARDY)。

表 26: 通用 DMA 传输的控制位

控制位	描述
GDMA 读/写 (操作码=00H/01H)	
DMA 配置寄存器 (见表 33 和表 34)	
BURST[2:0]	设定 DMA 周期数, DREQ 一直保持有效
MODE[1:0]	设定有效的读/写数据选通信号
WIDTH0	选择 DMA 总线宽度: 8 或 16 位
DIS_XFER_CNT	禁止使用 DMA 传输计数器
ATA_MODE	置零 (非 ATA 传输)
DMA 硬件寄存器 (见表 35 和表 36)	
EOT_POL	选择 EOT 信号的触发极性
ENDIAN[1:0]	设定数据是要进行字节交换或保持正常, 仅使用在 16 位数据总线的方式下
ACK_POL,DREQ_POL WRITE_POL,READ_POL	选择 DMA 握手信号的触发极性
MASTER	置 0 (从机)
MDMA(主机)读/写 (操作码=06H/07H)	
DMA 配置寄存器 (见表 33 和表 34)	
DMA_MODE[1:0]	设定 MDMA 的 DIOR 和 DIOW 有效时间 (UDMA 方式下值为 03H)
MODE[1:0]	设定有效的数据选通 (S)
WIDTH	选择 DMA 总线宽度: 8 或 16 位
DIS_XFER_CNT	禁止使用 DMA 传送计数器
ATA_MODE	置 1 (ATA 传输)
DMA 硬件寄存器 (见表 35 和 36)	
EOT_POL	不使用输入 EOT
ENDIAN[1:0]	设定数据是要进行字节交换或保持正常, 仅使用在 16 位数据总线的方式下
ACK_POL,DREQ_POL WRITE_POL,READ_POL	选择 DMA 握手信号的触发极性
MASTER	置 1 (主机)

表 27: IDE 规范 DMA 传输控制位

控制位	描述
PIO 读/写 (操作码=04H/05H)	
DMA 配置寄存器 (见表 33 和表 34)	
PIO_MODE[2:0]	选择 PIO 模式; 时序与 ATA(PI)兼容
ATA_MODE	置 1 (ATA 传输)
DMA 硬件寄存器 (见表 35 和表 36)	
MASTER	置 0
MDMA 读/写 (操作码=06H/07H)	
DMA 配置寄存器 (见表 33 和表 34)	
DMA_MODE[1:0]	选择 MDMA 模式; 时序与 ATA(PI)兼容
ATA_MODE	置 1 (ATA 传输)

控制位	描述
DMA 硬件寄存器 (见表 35 和表 36)	
MASTER	置 0
UDMA 读/写 (操作码=02H/03H)	
DMA 配置寄存器 (见表 33 和表 34)	
DMA_MODE[1:0]	选择 UDMA 模式; 时序与 ATA(PI)兼容
IGNORE_IORDY	传输期间忽略 IORDY
ATA_MODE	置 1 (ATA 传输)
DMA 硬件寄存器 (见表 35 和表 36)	
MASTER	置 0

注: DMA 总线默认为三态, 直到执行一个 DMA 命令。所有其它的控制信号都不是三态的。

9.4.1 DMA 命令寄存器 (地址: 30H)

这是一个单字节的寄存器, 用于在 DMA 控制器作用下启动所有的 DMA 传输。该寄存器只可写: 读时返回 FFH。

注: DMA 总线为三态, 直到执行一个 DMA 命令。

表 28: DMA 命令寄存器: 位分配

位	7	6	5	4	3	2	1	0
符号	DMA_CMD[7:0]							
复位	FFH							
总线复位	FFH							
访问	W							

表 29: DMA 命令寄存器: 位描述

位	符号	描述
7: 0	DMA_CMD[7:0]	DMA 命令代码, 见表 30

表 30: DMA 命令

代码 (Hex)	名称	描述
00	GDMA 读	通用 DMA IN 令牌传输 (仅主机模式): 数据从外部 DMA 总线传送到内部缓冲区。选通信号: 外部 DMA 控制器控制的 DIOW
01	GDMA 写	通用 DMA OUT 令牌传输 (仅从机模式): 数据从内部缓冲区传送到外部 DMA 总线。选通信号: 外部 DMA 控制器控制的 DIOR
02	UDMA 读	UDMA 读命令: 数据从外部 DMA 总线传送到内部 DMA 总线
03	UDMA 写	UDMA 写命令: 数据在 UDMA 模式下从内部缓冲区传送到外部 DMA 总线
04	PIO 读 ^[1]	ATAPI 设备的 PIO 读命令: 数据在 PIO 模式下从外部 DMA 总线传送到内部缓冲区, IORDY 有效时数据开始传输。忽略 DREQ 和 DACK 输入
05	PIO 写 ^[1]	ATAPI 设备的 PIO 写命令: 数据在 PIO 模式下从内部缓冲区传送到外部 DMA 总线。IORDY 有效时输出开始传输。忽略 DREQ 和 DACK 输入
06	MDMA 读	多字 DMA 读: 数据从外部 DMA 总线传送到内部缓冲区
07	MDMA 写	多字 DMA 写: 数据从内部缓冲区传送到外部 DMA 总线
0A	读 1F0	读 01F0H 中的内容: 启动一次对任务文件 1F0 的 PIO 读操作。在发送此命令前要对任务文件的字节计数值在 1F4H (LSB) 和 1F5H (MSB) 两个地址单元进行编程设置

代码 (Hex)	名称	描述
0B	Poll BSY	ATAPI 设备的 Poll BSY 状态位: 设置重复的 PIO 读操作来查询 ATAPI 设备的 BSY 状态。BSY=0 时查询停止, 并产生一个中断。
0C	读任务文件	读任务文件: 对除 1F0H 和 1F7H 之外的所有任务文件执行读操作。操作结束后产生一个中断。
0D	-	保留
0E	缓冲区有效	缓冲区有效 (仅调试时): 一次 ATA 到 USB 的数据传输结束后, 微控制器发送一个使端点缓冲区有效的请求
0F	清空缓冲区	清空缓冲区: 在一次 USB 到 ATA 的数据传输结束后, 微控制器发送一个清除端点缓冲区内容的请求
10	重新开始	重新开始: 微控制器一个请求, 将缓冲区指针移回端点 FIFO 起始单元
11	复位 DMA	复位 DMA: 初始化 DMA 内核, 使其处于上电复位状态 注: 在执行复位 DMA 命令时 DMA 内核被复位, DREQ,DACK,DIOW 和 DIOR 握手信号暂时都有效。这样会造成 DMA 控制器的混乱。为了防止这种情况的出现, 要求在 DMA 复位结束后再启动外部 DMA 控制器工作
12	MDMA 终止	MDMA 终止: 执行这个命令将立即终止 MDMA 的数据传输。它只应用在代码为 06H 和 07H 的命令中
13 到 FF	-	保留

[1]利用 DMA 命令寄存器启动 PIO 的读或写只用于 16 位的数据传输中。

9.4.2 DMA 传输计数器寄存器 (地址: 34H)

这是个 4 字节的寄存器, 用来设置一次 DMA 传输的总字节数 (DMACR)。它指示的是还未被传送的字节数。它各位的分配如表 31 所示。

传输计数器一般用于 DMA 模式: PIO(命令: 04H,05H)模式,UDMA(命令: 02H,03H)模式,MDMA(命令: 06H,07H)模式和 GDMA(命令: 00H,01H)模式。

对该寄存器执行“写”操作时,应当先写低字节(DMACR1)或字的低半部分 (MSByte: DMACR2,LSByte: DMACR1)。并且, 只有当写完最后一个字节 (DMACR4) 才能对它初始化。

在 GDMA 从机模式中, DMA 配置寄存器的 DIS_XFER_CNT 位可以禁止传输计数器 (见表 33) 的应用。这时, 当数据通过 IN 令牌从外部设备传送到主机时, 通过控制输入信号 EOT 来终止数据的 DMA 传输。若 EOF 引脚有效, FIFO 的最后一个数据包有效。

表 31: DMA 传输计数器寄存器: 位分配

位	31	30	29	28	27	26	25	24
符号	DMACR4=DMACR[31:24]							
复位	00H							
总线复位	00H							
访问	R/W							
位	23	22	21	20	19	18	17	16
符号	DMACR3=DMACR[23:16]							
复位	00H							
总线复位	00H							
访问	R/W							
位	15	14	13	12	11	10	9	8
符号	DMACR2=DMACR[15:8]							
复位	00H							

总线复位	00H							
访问	R/W							
位	7	6	5	4	3	2	1	0
符号	DMACR1=DMACR[7:0]							
复位	00H							
总线复位	00H							
访问	R/W							

表 32: DMA 传输计数器寄存器: 位描述

位	符号	描述
31 到 24	DMACR4, DMACR[31:24]	DMA 传输计数器第 4 个字节 (MSB)
23 到 16	DMACR3, DMACR[23:16]	DMA 传输计数器第 3 个字节
15 到 8	DMACR2, DMACR[15:8]	DMA 传输计数器第 2 个字节
7 到 0	DMACR1, DMACR[7:0]	DMA 传输计数器第 1 个字节 (LSB)

9.4.3 DMA 配置寄存器 (地址: 38H)

这个寄存器定义了通用 DMA 模式 (GDMA) 和增强 DMA 模式的配置。DMA 配置寄存器包含 2 个字节。它的各位分配见表 33。

表 33: DMA 配置寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	保留	IGNORE_ IORDY	ATA_ MODE	DMA_MODE[1:0]		PIO_MODE[2:0]		
复位	-	0	0	00H		00H		
总线复位	-	0	0	00H		00H		
访问	R/W	R/W	R/W	R/W		R/W		
位	7	6	5	4	3	2	1	0
符号	DIS_XFER_ CNT		BURST[2:0]		MODE[1:0]		保留	WIDTH
复位	0		00H		00H		-	1
总线复位	0		00H		00H		-	1
访问	R/W		R/W		R/W		R/W	R/W

表 34: DMA 配置寄存器: 位描述

位	符号	描述
15	-	保留
14	IGNORE_IORDY	为 1, 忽略 IORDY 输入信号 (仅在 UDMA 模式)
13	ATA_MODE	为 1, 配置 ATA 或 MDMA 模式下的 DMA 内核。发送 02H 到 07H, 0AH 到 0CH 的 DMA 命令时使用; 也使用在直接访问任务文件寄存器中。 为 0, 配置非 ATA 模式下的 DMA 内核。在发送 00H 到 01H 的 DMA 命令时使用。

位	符号	描述
12 到 11	DMA_MODE[1:0]	这 2 位的取值会对 UDMA 和 MDMA 模式的时序产生影响： 00H—UDMA/MDMA 模式 0：ATA(PI)时序兼容 01H—UDMA/MDMA 模式 1：ATA(PI)时序兼容 02H—UDMA/MDMA 模式 2：ATA(PI)时序兼容 03H—MDMA 模式 3：在非标准选通持续时间内使能 DMA 选通时间寄存器（见表 37 和表 38）；仅适用于 MDMA 模式。
10 到 8	PIO_MODE[2:0][3]	这 3 位的取值会对 PIO 时序（见表 78）产生影响： 00H 到 04H—PIO 模式 0 到 4：ATA(PI)时序兼容 05H 到 07H—保留
7	DIS_XFER_CNT	为 1，禁止 DMA 传输计数器（见表 31）的应用。通常传输计数器只会在 GDMA 从机模式下被禁止使用；在主机模式下一直保持有效
3 到 2	MODE[1:0]	这 2 位仅影响 GDMA（从机）和 MDMA（主机）的握手信号： 00H—DIOR(主机)或 DIOW(从机)：选通从 DMA 总线传送到 ISP1581 的数据；DIOW(主机)或 DIOR（从机）；将 ISP1581 的数据读取到 DMA 总线上 01H—DIOR(主机)或 DACK(从机)信号选通从 DMA 总线传送到 ISP1581 的数据；利用 DACK（主机）或 DIOR(从机)信号将 ISP1581 的数据读取到 DMA 总线上 02H—DACK（主机和从机）信号选通从 DMA 总线传送到 ISP1581 的数据,并将 ISP1581 数据取到 DMA 总线上（这种模式只适用于 16 位的 DMA,不能用在 8 位的 DMA 中。） 03H—保留。
1	-	保留
0	WIDTH	这 1 位用来选择 GDMA（从机）和 MDMA(主机)模式下的 DMA 总线宽度： 0—8 位数据总线 1—16 位数据总线

[1]当 FIFO 中留有未用的空间（写）或未读取出的数据（读）时，DREQ 有效。

[2]FIFO 为空，传输过程结束。

[3]利用 DMA 命令寄存器启动的 PIO 读或写操作只可用在 16 位的传输中。

9.4.4 DMA 硬件寄存器（地址：3CH）

这是个单字节的寄存器。它的各位分配如表 35 所示。

该寄存器决定了总线控制信号（EOT,DACK,DREQ,DIOR,DIOW）的触发极性和 DMA 模式(主机或从机)。还控制着 GDMA(从机)和 MDMA（主机）模式下总线上数据高低部分的交换（ENDIAN[1:0]位）。

表 35：DMA 硬件寄存器：位分配

位	7	6	5	4	3	2	1	0
符号	ENDIAN[1:0]	EOT_	MASTER	ACK_	DREQ_	WRITE_	READ_	
		POL		POL	POL	POL	POL	
复位	00H	0	0	0	1	0	0	
总线复位	00H	0	0	0	1	0	0	
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

表 36: DMA 硬件寄存器: 位描述

位	符号	描述
7 到 6	ENDIAN[1:0]	这 2 位决定内部 RAM 和 DMA 之间的数据总线是否发生交换: 仅应用在 GDMA(从机)和 MDMA(主机)模式。 00H—表示数据正常 16 位总线: MSB 在 DATA[15:8]上, LSB 在 DATA[7:0]上 01H—表示交换后的数据 16 位总线: MSB 在 DATA[7:0]上, LSB 在 DATA[15:8]上 02H,03H—保留 注: 若使用的是 8 位的数据总线, ENDIAN 位一直为 00H.
5	EOT_POL	选择传送结束信号的触发极性 (仅使用在 GDMA 从机模式下): 0—EOT 低电平有效 1—EOT 高电平有效
4	MASTER	选择 DMA 主/从机模式: 0—GDMA 从机模式 1—MDMA 主机模式
3	ACK_POL	选择 DMA 应答信号的触发极性: 0—DACK 低电平有效 1—DACK 高电平有效
2	DREQ_POL	选择 DMA 请求信号的触发极性 0—DREQ 低电平有效 1—DREQ 高电平有效
1	WRITE_POL	选择 DIOW 选通信号的触发极性 0—DIOW 低电平有效 1—DIOW 高电平有效
0	READ_POL	选择 DIOR 选通信号的触发极性 0—DIOR 低电平有效 1—DIOR 高电平有效

9.4.5 DMA 选通时间寄存器 (地址: 60H)

这是个 1 字节的寄存器, 当 DMA 配置寄存器的 DMA_MODE 为 03H 时, 它控制着 MDMA 模式的选通时间。寄存器的各位分配如表 37 所示。

表 37: DMA 选通时间寄存器: 位分配

位	7	6	5	4	3	2	1	0
符号	保留			DMA_STROBE_CNT[4:0]				
复位	-	-	-	1FH				
总线复位	-	-	-	1FH				
访问	R/W	R/W	R/W	R/W				

表 38: DMA 选通时间寄存器: 位描述

位	符号	描述
7 到 5	-	保留
4 到 0	DMA_STROBE_CNT[4:0]	选择 DMA_MODE=03H (见表 33) 时的选通持续时间。这个选通持续时间是第 (N+1) 个周期, N 是 DMA_STROBE_CNT 的值 (见图 3)。

[1] 这个周期的持续时间等于内部时钟的周期 (33.3ns/cycle)。

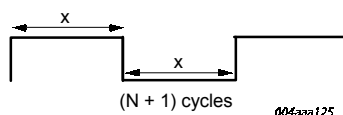


图 3 可编程的选通时序

9.4.6 任务文件寄存器（地址：40H 到 4FH）

这些寄存器允许 PIO 模式下的 ATAPI 外围设备对内部寄存器的直接访问。受支持的任务文件寄存器和它们的功能如表 39 所示。真正的外围寄存器通过 CS1,CS0,DA2,DA1 和 DA0（见表 40）自动编址。

表 39: 任务文件寄存器的各种功能

地址 (Hex)	ATA 功能	ATAPI 功能
1F0	数据 (16 位)	数据 (16 位)
1F1	错误/特性	错误/特性
1F2	扇区计数	中断原因
1F3	扇区数/LBA[7:0]	保留
1F4	低柱面/LBA[15:8]	低柱面
1F5	高柱面/LBA[23:16]	高柱面
1F6	驱动/头/LBA[27:24]	驱动选择
1F7	命令	状态/命令
3F6	改变状态/命令	改变状态/命令
3F7	驱动地址	保留

表 40: ATAPI 外围寄存器编址

任务文件	CS1	CS0	DA2	DA1	DA0
1F0	H	L	L	L	L
1F1	H	L	L	L	H
1F2	H	L	L	H	L
1F3	H	L	L	H	H
1F4	H	L	H	L	L
1F5	H	L	H	L	H
1F6	H	L	H	H	L
1F7	H	L	H	H	H
3F6	L	H	H	H	L
3F7	L	H	H	H	H

在 8 位的总线模式下按以下方法实现对 16 位任务文件寄存器 1F0 的访问: 先对该任务文件寄存器进行 2 次连续的读/写, 再由 IDE 接口产生相应的 PIO 写/读来访问 1F0。整个过程中先对低字节 (LSByte) 操作。其它的任务文件寄存器可以直接访问。

允许以任何次序“写”任务文件寄存器, 但是 1F7 任务文件寄存器要求最后写。

表 41: 任务文件 1F0 (地址: 40H): 位分配

CS1=H,CS0=L,DA2=L,DA1=L,DA0=L

位	7	6	5	4	3	2	1	0
符号	数据 (ATA 或 ATAPI)							
复位	00H							
位	7	6	5	4	3	2	1	0
总线复位	00H							
访问	R/W							

表 42: 任务文件寄存器 1F1 (地址: 48H): 位分配

CS1=H,CS0=L,DA2=L,DA1=L,DA0=H

位	7	6	5	4	3	2	1	0
符号	错误/特性 (ATA 或 ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

表 43: 任务文件寄存器 1F2 (地址: 49H): 位分配

CS1=H,CS0=L,DA2=L,DA1=H,DA0=L

位	7	6	5	4	3	2	1	0
符号	扇区计数(ATA) 或中断原因 (ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

表 44: 任务文件寄存器 1F3 (地址: 4AH): 位分配

CS1=H,CS0=L,DA2=L,DA1=H,DA0=H

位	7	6	5	4	3	2	1	0
符号	扇区数/LBA[7:0](ATA),保留 (ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

表 45: 任务文件寄存器 1F4 (地址: 4BH): 位分配

CS1=H,CS0=L,DA2=H,DA1=L,DA0=L

位	7	6	5	4	3	2	1	0
符号	低柱面/LBA[15:8](ATA) 或低柱面 (ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

表 46: 任务文件寄存器 1F5 (地址: 4CH): 位分配

CS1=H,CS0=L,DA2=H,DA1=L,DA0=H

位	7	6	5	4	3	2	1	0
符号	高柱面 /LBA[23:16](ATA) 或高柱面 (ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

表 47: 任务文件寄存器 1F6 (地址: 4DH): 位分配

CS1=H,CS0=L,DA2=H,DA1=H,DA0=L

位	7	6	5	4	3	2	1	0
符号	驱动/头/LBA[27:24](ATA) 或驱动 (ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

表 48: 任务文件寄存器 1F7 (地址: 44H): 位分配

CS1=H,CS0=L,DA2=H,DA1=H,DA0=H

位	7	6	5	4	3	2	1	0
符号	命令(ATA) 或状态 ^[1] /命令 (ATAPI)							
复位	00H							
总线复位	00H							
访问	W							

[1]任务文件寄存器 1F7 只可写; 读时返回 FFH.

表 49: 任务文件寄存器 3F6 (地址: 4EH): 位分配

CS1=L,CS0=H,DA2=H,DA1=H,DA0=L

位	7	6	5	4	3	2	1	0
符号	状态/命令改变(ATA 或 ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

表 50: 任务文件寄存器 3F7 (地址: 4FH): 位分配

CS1=L,CS0=H,DA2=H,DA1=H,DA0=H

位	7	6	5	4	3	2	1	0
符号	驱动地址(ATA) 或保留(ATAPI)							
复位	00H							
总线复位	00H							
访问	R/W							

9.4.7 DMA 中断源寄存器 (地址: 50H)

这是个 2 字节的寄存器, 它列出了引发一次 DMA 中断的所有可能的中断源。每执行完一个 DMA 命令, 寄存器的每一位就更新一次。中断源通过对寄存器相应的位置 1 来清除。它的各位分配见表 51。

表 51: DMA 中断源寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	保留			ODD_IND	EXT_EOT	INT_EOT	INTRQ_PENDING	DMA_XFER_OK
复位	-	-	-	0	0	0	0	0
总线复位	-	-	-	0	0	0	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	7	6	5	4	3	2	1	0
符号	1F0_WF_E	1F0_WF_F	1F0_RF_E	READ_1F0	BSY_DONE	TF_RD_DONE	CMD_INTRQ_OK	保留
复位	0	0	0	0	0	0	0	-
总线复位	0	0	0	0	0	0	0	-
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

表 52: DMA 中断源寄存器: 位描述

位	符号	描述
15 到 13	-	保留
12	ODD_IND	为 1, 表明最后一个奇数字节包已经从 OUT 令牌缓冲区发送到 DMA。它仅用于 DMA 从机模式下的 OUT 令牌。对于 IN 令牌数据没有意义。可参考文件 Using the Odd Bit Indicator for DMA。
11	EXT_EOT	为 1, 表明检测到一个外部 EOT 信号。仅用于 GDMA 从机模式。
10	INT_EOT	为 1, 表明检测到一个内部 EOT 信号。见表 53。
9	INTRQ_PENDING	为 1, 表明在 INTRQ 引脚上检测到一个未处理的中断。
8	DMA_XFER_OK	为 1, 表明 DMA 传输结束 (DMA 传输计数器的值变为 0)。这一位只使用在 GDMA(从机)模式和 MDMA (主机) 模式。
7	1F0_WF_E	为 1, 表明 1F0 写 FIFO 为空, 要求微控制器启动写数据。
6	1F0_WF_F	为 1, 表明 1F0 写 FIFO 已满, 要求微控制器停止写数据。
5	1F0_RF_E	为 1, 表明 1F0 读 FIFO 为空, 要求微控制器停止读数据。
4	READ_1F0	为 1, 表明 1F0 的 FIFO 内包含未读的数据, 要求微控制器启动读数据。
3	BSY_DONE	为 1, 表明 BSY 状态位变为 0, 对该位的查询已经终止。
2	TF_RD_DONE	为 1, 表明读任务文件命令已被执行。
1	CMD_INTRQ_OK	为 1, 表明 FIFO 的所有字节已发送完毕 (DMA 传输计数器的值为 0), 并在 INTRQ 管脚上检测到一个中断。
0	-	保留

表 53: 内部 EOT 功能与 DMA_XFER_OK 位的关系

INT_EOT	DMA_XFER_OK	描述
1	0	在 DMA 传输过程中, 利用短包提前终止整个过程。
1	1	带有短包的 DMA 传输结束, DMA 传输计数器的值变为 '0'。
0	1	不带任何短包的 DMA 传输结束, DMA 传输计数器的值变为 '0'。

9.4.8 DMA 中断使能寄存器 (地址: 54H)

这是个 2 字节的寄存器, 它控制着 DMA 中断源寄存器(见表 51)中各中断源中断的产生。它的各位分配见表 54, 位描述见表 52。寄存器的某一位为 '1', 就激活它所对应的中断。各位的值在 (总线) 复位后都为 0。

表 54: DMA 中断使能寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	保留			IE_ODD_IND	IE_EXT_EOT	IE_INT_EOF	IE_INTRQ_PENDING	IE_DMA_XFER_OK
复位	-	-	-	0	0	0	0	0
总线复位	-	-	-	0	0	0	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	7	6	5	4	3	2	1	0
符号	IE_1F0_WF_E	IE_1F0_WF_F	IE_1F0_RF_E	IE_READ_1F0	IE_BSY_DONE	IE_TF_RD_DONE	IE_CMD_INTRQ_OK	保留
复位	0	0	0	0	0	0	0	-
总线复位	0	0	0	0	0	0	0	-
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

9.4.9 DMA 端点寄存器 (地址: 58H)

这是个 1 字节的寄存器, 用来决定 FIFO 端点在 DMA 传输中是发送者还是接收者。它的各位分配见表 55。

表 55: DMA 端点寄存器: 位分配

位	7	6	5	4	3	2	1	0
符号	保留				EPIDX[2: 0]			DMADIR
电源复位	-	-	-	-	0	0	0	0
总线复位	-	-	-	-	0	0	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

表 56: DMA 端点寄存器: 位描述

位	符号	描述
7 到 4	-	保留
3 到 1	EPIDX[2:0]	指定进行 DMA 访问使用的端点
0	DMADIR	0—选择 RX/OUT FIFO 作为 DMA 读传输端点 1—选择 TX/IN FIFO 作为 DMA 写传输端点

DMA 端点寄存器在任何情况下都不要使用被编入端点索引寄存器 (02CH) 中的端点。这样是为了防止数据出错。因此, 如果某一端点没有被 DMA 端点寄存器使用, 就把它看成一个未被使用的端点。但是, 一旦把 DMA 端点寄存器把它作为一个有效端点使用, 固件就不能使用端点索引寄存器中与之相同的端点。

9.5 通用寄存器

9.5.1 中断寄存器 (地址: 18H)

这是个 4 字节的寄存器。它的位分配见表 57。

如果中断寄存器中有一位被设置, 这就表明中断产生的硬件条件已经出现。只要中断寄存器的内容不为 ‘0’, INT 输出就有效。在对中断的检测过程中, 外部微处理器可以通过读中断寄存器来确定产生此次中断的中断源。

每一个端点缓冲区都对应一个专门的中断位 (EPnTX, EPnRX)。另外, 各种总线状态也能产生中断: 重新开始、挂起、假 SOF、SOF 和总线复位。DMA 控制器只有一个中断位: DMA 中断的中断源已在 DMA 中断源寄存器中列出 (见表 51)。

每一个中断位通过置 ‘1’ 清除。而要清除 DMA 中断位, 必须将 DMA 中断源寄存器中对应的中断源位和中断寄存器的 DMA 位都置 ‘1’。

表 57: 中断寄存器: 位分配

位	31	30	29	28	27	26	25	24
符号	保留						EP7TX	EP7RX
复位	-	-	-	-	-	-	0	0
总线复位	-	-	-	-	-	-	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	23	22	21	20	19	18	17	16
符号	EP6TX	EP6RX	EP5TX	EP5RX	EP4TX	EP4RX	EP3TX	EP3RX
复位	0	0	0	0	0	0	0	0
总线复位	0	0	0	0	0	0	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	15	14	13	12	11	10	9	8

符号	EP2TX	EP2RX	EP1TX	EP1RX	EP0TX	EP0RX	保留	EP0SETUP
复位	0	0	0	0	0	0	-	0
总线复位	0	0	0	0	0	0	-	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位	7	6	5	4	3	2	1	0
符号	保留	DMA	HS_STAT	RESUME	SUSP	PSOF	SOF	BRE SET
复位	-	0	0	0	0	0	0	0
总线复位	-	0	0	0	0	0	不变	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

表 58: 中断寄存器: 位描述

位	符号	描述
31 到 26	保留	保留; 必须写 0;
25	EP7TX	为 1, 中断源是端点 7 的 TX 缓冲区
24	EP7RX	为 1, 中断源是端点 7 的 RX 缓冲区
23	EP6TX	为 1, 中断源是端点 6 的 TX 缓冲区
22	EP6RX	为 1, 中断源是端点 6 的 RX 缓冲区
21	EP5TX	为 1, 中断源是端点 5 的 TX 缓冲区
20	EP5RX	为 1, 中断源是端点 5 的 RX 缓冲区
19	EP4TX	为 1, 中断源是端点 4 的 TX 缓冲区
18	EP4RX	为 1, 中断源是端点 4 的 RX 缓冲区
17	EP3TX	为 1, 中断源是端点 3 的 TX 缓冲区
16	EP3RX	为 1, 中断源是端点 3 的 RX 缓冲区
15	EP2TX	为 1, 中断源是端点 2 的 TX 缓冲区
14	EP2RX	为 1, 中断源是端点 2 的 RX 缓冲区
13	EP1TX	为 1, 中断源是端点 1 的 TX 缓冲区
12	EP1RX	为 1, 中断源是端点 1 的 RX 缓冲区
11	EP0TX	为 1, 中断源是端点 0 的 TX 缓冲区
10	EP0RX	为 1, 中断源是端点 0 的 RX 缓冲区
9	保留	保留
8	EP0SETUP	为 1, 端点 0 接收到一个 SETUP 令牌
7	保留	保留
6	DMA	DMA 状态: 为 1, DMA 状态寄存器发生了一次状态变化
5	HS_STAT	高速状态: 为 1, 从 FS 模式变为 HS 模式 (HS 连接)。若系统处于 FS 挂起状态, 这一位不能设置。
4	RESUME	重新开始状态: 为 1, 检测到从 ‘挂起’ 状态到 ‘重新开始’ (有效) 状态的变化。
3	SUSP	挂起状态: 为 1, 总线上检测到从正常运行到 ‘挂起’ 的状态改变。
2	PSOF	假 SOF 中断: 为 1, 接收到一个假 SOF 或 uSOF 信号。假 SOF 是一个内部产生的时钟信号 (FS: 周期为 1ms, HS: 周期为 125us), 与 USB 总线的 SOF/uSOF 同步。

位	符号	描述
1	SOF	SOF 中断: 为 1, 接收到一个 SOF/uSOF。
0	BRESET	总线复位: 为 1, 检测到一个 USB 总线复位信号。

9.5.2 芯片 ID 寄存器 (地址: 70H)

这是个只读寄存器, 它的内容包括芯片的标识符和硬件的版本号。固件通过查询这些信息来确定该芯片所支持的功能和含有的特性。这是个 3 字节的寄存器, 它的各位分配见表 59。

表 59: 芯片 ID 寄存器: 位分配

位	23	22	21	20	19	18	17	16
符号	CHIPID[23: 16]							
复位	15H							
总线复位	15H							
访问	R							
位	15	14	13	12	11	10	9	8
符号	CHIPID[15: 8]							
复位	81H							
总线复位	81H							
访问	R							
位	7	6	5	4	3	2	1	0
符号	VERSION[7: 0]							
复位	51H							
总线复位	51H							
访问	R							

表 60: 芯片 ID 寄存器: 位描述

位	符号	描述
23 到 16	CHIPID[23: 16]	芯片 ID: 低字节 (15H)
15 到 8	CHIPID[15: 8]	芯片 ID: 高字节 (81H)
7 到 0	VERSION	版本号 (51H): 随着性能改善和功能扩展了的新版本的出现, 版本可以升级。

9.5.3 帧数目寄存器 (地址: 74H)

这是个只读寄存器, 它的内容是最近成功接收到的数据的帧开始 (SOF) 对应的帧的编号。该寄存器包含 2 个字节, 它的各位分配见表 61。若以 8 位的形式对寄存器进行访问, 先得到低字节的内容。

表 61: 帧数目寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	保留		MICROSOFT[2: 0]			SOFR[10: 8]		
电源复位	-	-	00H			00H		
总线复位	-	-	00H			00H		
访问	R	R	R			R		
位	7	6	5	4	3	2	1	0
符号	SOFR[7: 0]							
电源复位	00H							
总线复位	00H							
访问	R							

表 62: 帧数目寄存器: 位描述

位	符号	描述
13 到 11	MICROSO[2: 0]	微帧数目
10 到 0	SOFR[10: 0]	帧数目

9.5.4 暂存寄存器 (地址: 78H)

这是个 16 位的寄存器, 它常被固件用来存储和恢复信息, 如保存设备进入 ‘挂起’ 状态之前的状态。寄存器的内容不会因为总线复位而发生改变。它的位分配见表 63。

表 63: 暂存寄存器: 位分配

位	15	14	13	12	11	10	9	8
符号	SFIRH[7: 0]							
复位	00H							
总线复位	不变							
访问	R/W							
位	7	6	5	4	3	2	1	0
符号	SFIRL[7: 0]							
复位	00H							
总线复位	不变							
访问	R/W							

表 64: 暂存寄存器: 位描述

位	符号	描述
15 到 8	SFIRH[7: 0]	暂存固件信息寄存器 (高字节)
7 到 0	SFIRL[7: 0]	暂存固件信息寄存器 (低字节)

9.5.5 测试方式寄存器 (地址: 84H)

这是个 1 字节的寄存器, 它允许固件设置 (D+,D-) 线来预定测试的状态。它的各位分配见表 65。

注: 每次只可设置一位。

表 65: 测试方式寄存器: 位分配

位	7	6	5	4	3	2	1	0
符号	FORCEHS	保留	FORCEFS	PRBS	KSTATE	JSTATE	SE0_NAK	
复位	0	-	-	0	0	0	0	0
总线复位	0	-	-	0	0	0	0	0
访问	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

表 66: 测试方式寄存器: 位描述

位	符号	描述
7 ^[1]	FORCEHS	为 1, 强制硬件仅进入高速模式, 使脉冲检测逻辑无效
6 到 5	-	保留
4 ^[1]	FORCEFS	为 1, 强制物理层仅进入全速模式, 使脉冲检测逻辑无效
3 ^[2]	PRBS	为 1, 设置 (D+,D-) 线使状态变换到一个预定的任何状态。
2 ^[2]	KSTATE	写 ‘1’, 设置 (D+,D-) 线进入 K 状态。
1 ^[2]	JSTATE	写 ‘1’, 设置 (D+,D-) 线进入 J 状态。
0 ^[2]	SE0_NAK	写 ‘1’, 设置 (D+,D-) 线进入 HS 静止状态。器件只用一个 NAK 信号来响应一个有效的 HS IN 令牌。

[1]一次只能将 FORCEHS 和 FORCEFS 两者之一置 ‘1’。

[2]在四位 (PRBS,KSTATE,JSTATE 和 SE0_NAK) 中, 一次只能对一位置 ‘1’。

10. 电源

ISP1581 的供电电压为 3.3V 或 5.0V。

利用 $V_{CC}=5.0V$ 的电压对 ISP1581 供电时,集成的电压调整器就给内部逻辑电路和 USB 收发器提供 3.3V 的电压。具体电路的连接见图 4。

ISP1581 还能在 $V_{CC}=3.3V$ 的条件下正常工作。这时,内部的电压调整器不工作,所有的供电引脚都连接到 V_{CC} 端。具体电路的连接见图 5。

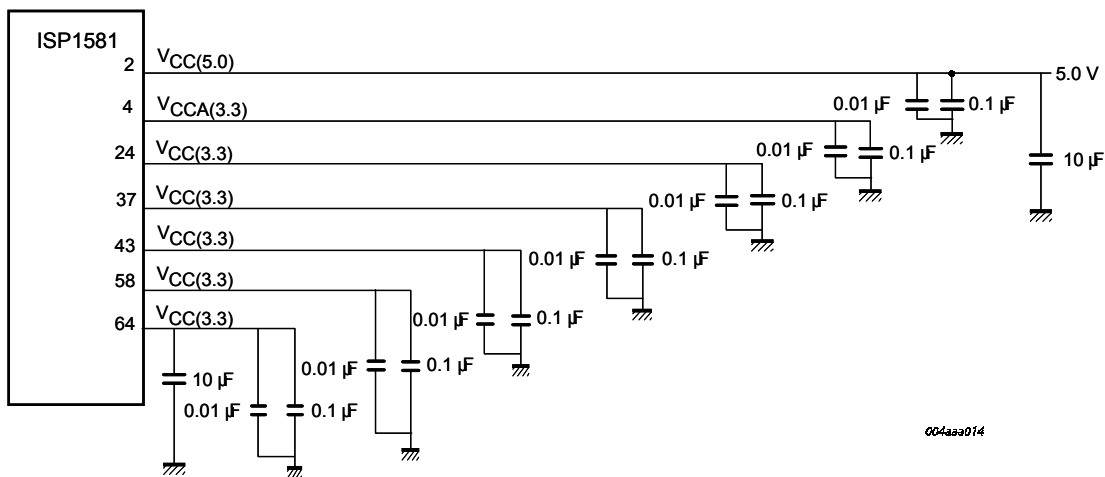


图 4 ISP1581 的 5.0V 供电电压

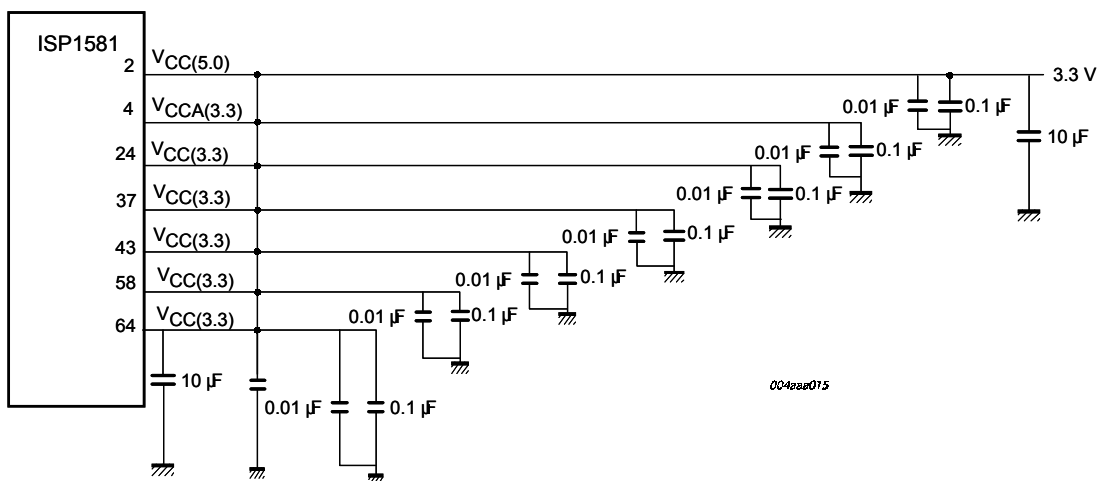


图 5 ISP1581 的 3.3V 供电电压

11. 参数限制

表 67: 最大限度的额定值

以 the Absolute Maximum Rating System (IEC 60134) 为依据。

符号	参数	条件	最小	最大	单位
V_{CC}	供电电压		-0.5	+6.0	V
V_I	输入电压		-0.5	$V_{CC}+0.5$	V

符号	参数	条件	最小	最大	单位
$I_{latchup}$	上锁电流	$V_I < 0$ 或 $V_I > V_{CC}$	-	100	mA
V_{esd}	静电放电电压	$I_{LI} < 1\mu A$			
		引脚 D+, D-, GND 和 $V_{CC(s,0)}$	-	± 4000	V
		其它引脚	-	± 2000	V
T_{stg}	存储温度		-60	+150	°C
P_{tot}	总功率消耗		-	770	mW

12. 推荐操作条件

表 68: 推荐操作条件

符号	参数	条件	最小	最大	单位
V_{CC}	供电电压	有电压转换器	4.0	5.5	V
		无电压转换器	3.0	3.6	V
V_I	输入电压范围		0	5.5	V
$V_{I(AI/O)}$	模拟 I/O 引脚的输入电压 (D+, D-)		0	3.6	V
$V_{O(od)}$	开漏输出上拉电压		0	V_{CC}	V
T_{amb}	操作环境温度		-40	+85	°C

13. 静态特性

表 69: 静态特性; 供电引脚

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^\circ C$; 有特别要求的按要求给定。

符号	参数	条件	最小	类型	最大	单位
$V_{CCA(3.3)}$	可调节的供电电压	有电压转换器	$3.0^{[1]}$	3.3	3.6	V
I_{CC}	操作电流		-	130	-	mA
$I_{CC(susp)}$	挂起电流	引脚 D+ 上无上拉	-	450	-	uA

[1]在‘挂起’模式下最小电压是 2.7V。

表 70: 静态特性: 数字脚

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^\circ C$; 有特别要求的按要求给定。

符号	参数	条件	最小	类型	最大	单位
输入电平						
V_{IL}	低电平输入电压		-	-	0.8	V
V_{IH}	高电平输入电压		2.0	-	-	V
输出电平						
V_{OL}	低点平输出电压	I_{OL} =额定驱动	-	-	0.4	V
V_{OH}	高电平输出电压	I_{OH} =额定驱动	2.6	-	-	V
漏电流						
I_{LI}	输入漏电流		$-5^{[1]}$	-	$+5^{[1]}$	uA

[1] 此参数仅在带晶体管输入的情况下有效。它的取值根据内部的上拉电阻或下拉电阻而不同。

表 71: 静态特性: 模拟 I/O 脚(D+,D-) [1]

V_{CC}=4.0 到 5.5V; V_{GND}=0V; T_{amb}=-40 到+85℃; 有特别要求的按要求给定。

符号	参数	条件	最小	类型	最大	单位
最初 USB 规范收发器 (全速)						
输入电平 (差分接收器)						
V _{DI}	差分输入灵敏度	V _{I(D+)} -V _{I(D-)}	0.2	-	-	V
V _{CM}	差分共模电压	包含 V _{DI} 范围	0.8	-	2.5	V
输入电平 (单端接收器)						
V _{IL}	低电平输入电压		-	-	0.8	V
V _{IH}	高电平输入电压		2.0	-	-	V
V _{hys}	滞后电压		0.4	-	0.7	V
输出电平						
V _{OL}	低电平输出电压	D ₊ ,D ₋ 上拉电阻 R _L =1.5KΩ	0	-	0.4	V
V _{OH}	高电平输出电压	D ₊ ,D ₋ 对 GND 的下拉 电阻 R _L =15KΩ	2.8	-	3.6	V
高速 USB 收发器						
输入电平 (差分接收器)						
V _{HSSQ}	高速静噪检测阈值电压 (差分)	检测到有噪声	-	-	100	mV
		检测到无噪声	150	-	-	mV
V _{HSDSC}	高速断开检测阈值电压 (差分)	检测到断开	625	-	-	mV
		未检测到断开	-	-	525	mV
V _{HSDI}	高速差分输入灵敏度	V _{I(D+)} -V _{I(D-)}	300	-	-	mV
V _{HSCM}	高速数据信道共模电压范围		-50	-	+500	mV
输出电平						
V _{HSOI}	高速空闲电平输出电压 (差分)		-10	-	+10	mV
V _{HSOL}	高速低电平输出电压 (差分)		-10	-	+10	mV
V _{HSOH}	高速高电平输出电压 (差分)		360	-	440	mV
V _{CHIRPJ}	Chirp-J 输出电压 (差分)		[1] 700	-	1100	mV
V _{CHIRPK}	Chirp-K 输出电压 (差分)		[1] -900	-	-500	mV
漏电流						
I _{LZ}	三态漏电流		-	-	±10	uA
电容						
C _{IN}	收发器电容	引脚对地	-	-	20	pF
电阻						
Z _{DRV2} [2]	USB 2.0 和 USB 1.1 的驱动器输出阻抗	稳态驱动	40.5	45	49.5	Ω
Z _{INP}	输入阻抗		10	-	-	MΩ
终止						
V _{TERM}	引脚 RPU 上拉电阻的终止电压		3.0 [3]	-	3.6	V

[1]HS 终止电阻禁用, 连接上拉电阻。仅出现在高速集线器和设备的复位期间。

[2]包括在 D+和 D-上的内部匹配电阻。这个容许范围符合 USB 2.0 规范。

[3]在‘挂起’模式下的最小电压为 2.7V。

14. 动态特性

表 72: 动态特性

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$; 有特别要求的按要求给定。

符号	参数	条件	最小	类型	最大	单位
复位						
$t_w(\overline{RESET})$	输入 \overline{RESET} 的脉冲宽度	晶振运行	500	-	-	us
晶振						
f_{XTAL}	晶振频率		-	12	-	MHz

表 73: 动态特性: 模拟 I/O 脚 (D+,D-) [1]

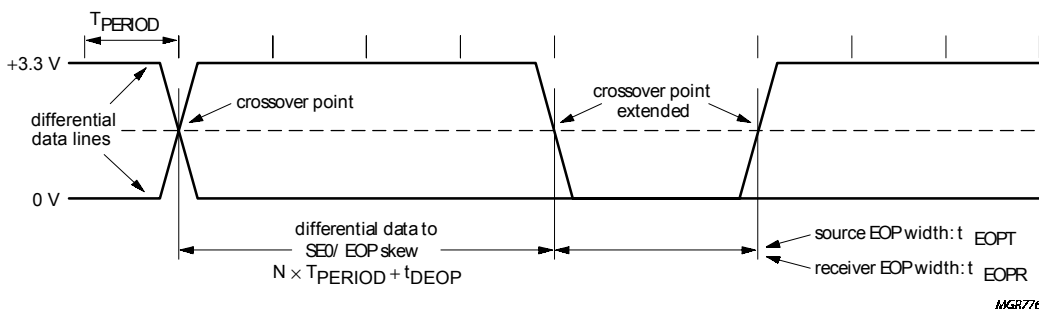
$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$; $C_L=50PF$; D+到 V_{TERM} 连接一个 $R_{PU}=1.5K\Omega$ 的电阻; 有特别要求的按要求给定。

符号	参数	条件	最小	类型	最大	单位
驱动器特性						
全速模式						
t_{FR}	上升时间	$C_L=50PF$; $ V_{OH}-V_{OL} $ 的 10 到 90%	4	-	20	ns
t_{FF}	下降时间	$C_L=50PF$; $ V_{OH}-V_{OL} $ 的 10 到 90%	4	-	20	ns
FRFM	差分上升/下降时间匹配 (t_{FR}/t_{FF})		[2] 90	-	111.11	%
V_{CRS}	输出信号交叉电压		[2][3] 1.3	-	2.0	V
高速模式						
t_{HSR}	高速差分上升时间	有捕获电缆	500	-	-	PS
t_{HSF}	高速差分下降时间	有捕获电缆	500	-	-	PS
数据源时序						
全速模式						
t_{FEOPT}	源 EOP 宽度	见图 6	[3] 160	-	175	ns
t_{FDEOP}	源差分数据到 EOP 的转换	见图 6	[3] -2	-	+5	ns
接收器时序						
全速模式						
t_{JR1}	接收器数据抖动误差到下一次转换	见图 7	[3] -18.5	-	+18.5	ns
t_{JR2}	成对转换的接收器数据抖动误差	见图 7	[3] -9	-	+9	ns
t_{FEOPR}	接收器 SE0 宽度	接收的 EOP 见图 6	[3] 82	-	-	ns

[1]测试电路: 见图 37。

[2]不接收空闲状态的第一次转换

[3]只是一种特性描述，未经测试。真正的限制值仍由实际的设计来决定。



T_{PERIOD} is the bit duration corresponding with the USB data rate.

Full-speed timings have a subscript prefix $\hat{\sigma}$ low-speed timings a prefix $\hat{\sigma}$

图6 源差分数据的 EOP 转换失真和 EOP 宽度

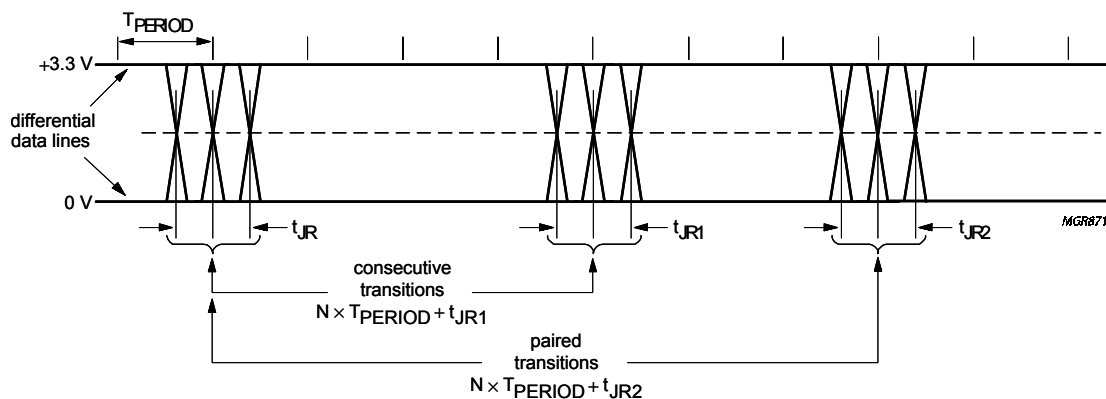


图7 接收器差分数据抖动

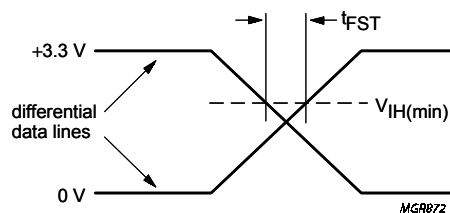


图8 接收器 SE0 宽度误差

14.1 时序符号

表 76: 时序图的特征说明符

符号	描述
时间符号	
t	时间
T	周期 (周期信号)
信号名称	
A	地址; DMA 应答(DACK)

符号	描述
C	时钟; 命令
符号	描述
D	数据输入; 数据
E	芯片使能
G	输出使能
I	指令 (程序存储器内容); 输入 (常规)
L	地址锁存使能 (ALE)
P	程序存储使能 ($\overline{\text{PSEN}}$ 低电平有效); 传输延时
Q	数据输出
R	读信号 $\overline{\text{RD}}$ 低电平有效); 读 (动作); DMA 请求 (DREQ)
S	片选
W	写信号 ($\overline{\text{WR}}$ 低电平有效); 写 (动作); 脉宽
U	未定义
Y	输出 (常规)
逻辑电平	
H	逻辑高
符号	描述
L	逻辑低
P	停止, 无效 (OFF)
S	启动, 有效 (ON)
V	有效的逻辑电平
X	无效的逻辑电平
Z	高阻抗 (悬浮, 三态)

14.2 寄存器访问时序

14.2.1 通用处理器模式 (BUS_CONF=1)

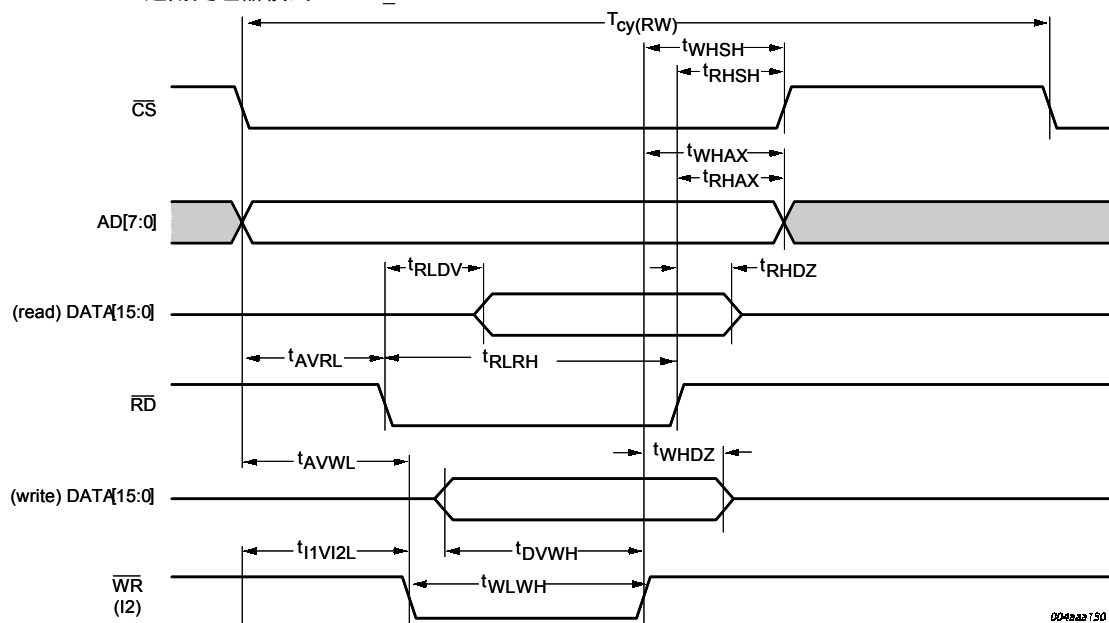


图 9 ISP1581 寄存器访问时序：单独的地址和数据总线 (MODE0=1) s

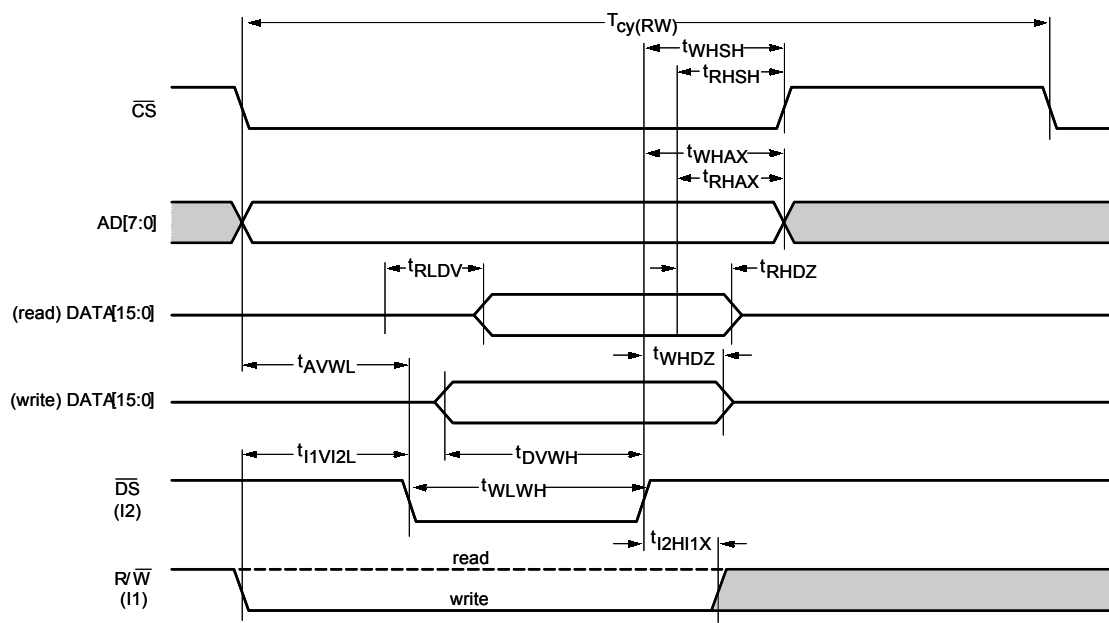


图 10 ISP1581 寄存器访问时序：单独的地址和数据总线 (MODE0=0) .

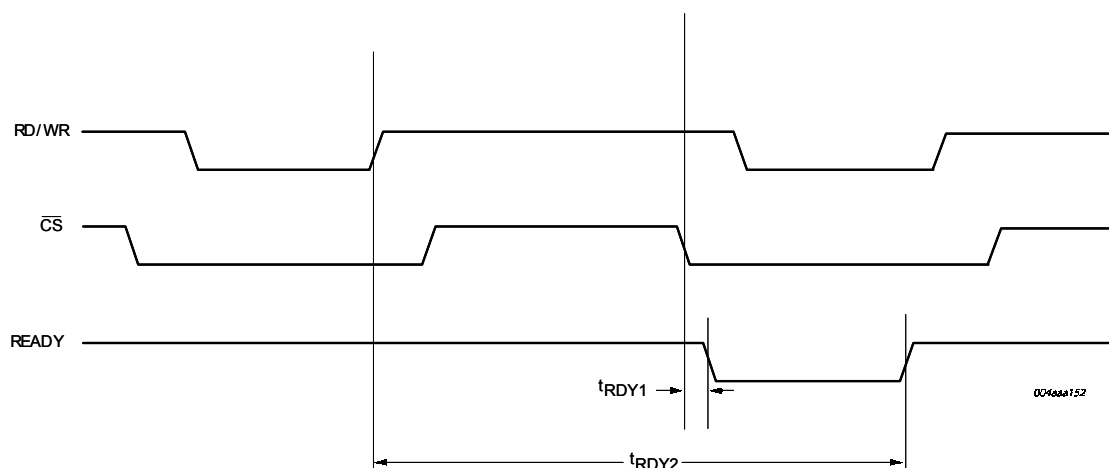


图 11 ISP1581 READY 信号时序

表 75: ISP1581 寄存器访问时序参数: 单独的地址和数据总线

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$.

符号	参数	最小	最大	单位
读				
t_{RLRH}	\overline{RD} 低脉冲宽度	$>t_{RLDV}$	-	ns
t_{AVRL}	\overline{RD} 变低前的地址建立时间	0	-	ns
t_{RHAX}	\overline{RD} 变高后的地址保持时间	0	-	ns
t_{RLDV}	\overline{RD} 低到数据有效的延时	-	26	ns
t_{RHDZ}	\overline{RD} 高到数据输出三态的延时	0	15	ns
t_{RHSH}	\overline{RD} 高到高的延时	0	-	ns
写				
t_{WLWH}	\overline{WR} 低脉冲宽度	15	-	ns
t_{AVWL}	\overline{WR} 变低前的地址建立时间	15	-	ns
t_{WHAX}	\overline{WR} 变高电后的地址保持时间	0	-	ns
t_{DVWH}	\overline{WR} 变高前的数据建立时间	11	-	ns
t_{WHDZ}	\overline{WR} 变高后的数据保持时间	5	-	ns
t_{WHSH}	\overline{WR} 高到 CS 高的延时	0	-	ns
常规				
$T_{cy(RW)}$	读/写周期	80	-	ns
t_{1V12L}	\overline{DS} 变低前的 R/ \overline{W} 建立时间	0	-	ns
t_{12H1X}	\overline{DS} 变高后的 R/ \overline{W} 保持时间	0	-	ns
t_{RDY1}	READY 低到 \overline{CS} 低的延时	-	3	ns
t_{RDY2}	最后一次访问的 READY 高到 $\overline{RD} / \overline{WR}$ 高的时间	-	91	ns

14.2.2 断开总线模式 (BUS_CONF=0)

断开总线模式 (BUS_CONF=0, MODE1=0, MODE0=0/1)

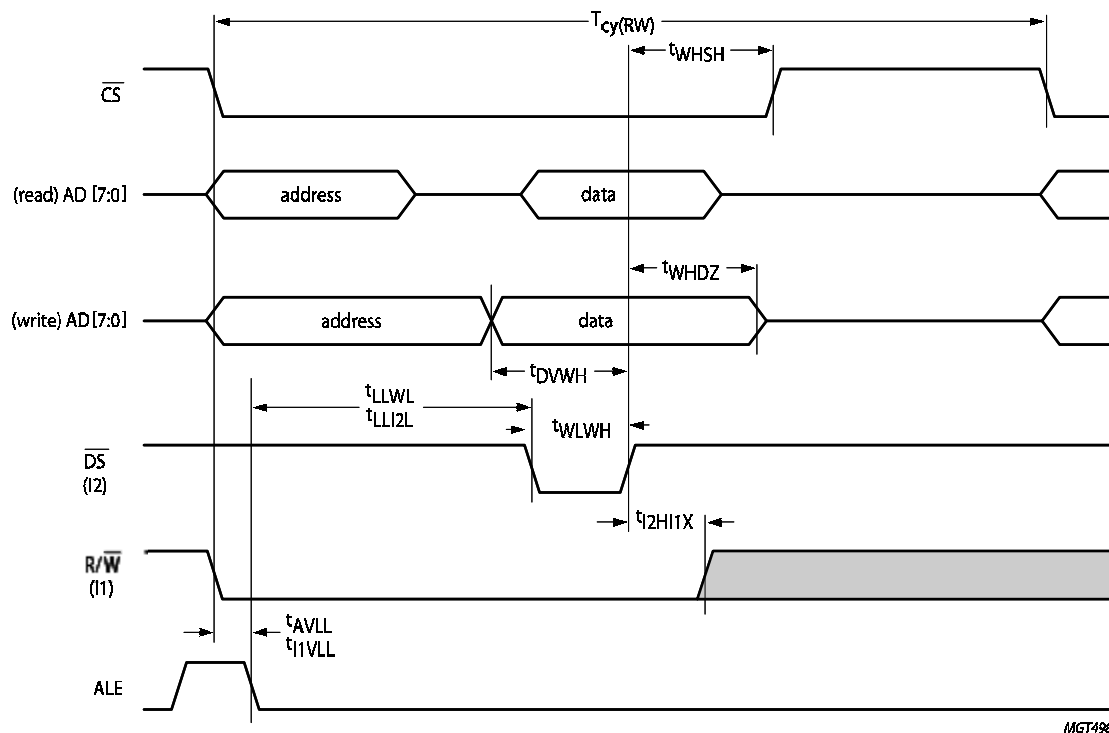


图 12 ISP1581 寄存器访问时序：多路复用地址/数据总线 (BUS_CONF=0, MODE1=0, MODE0=0)

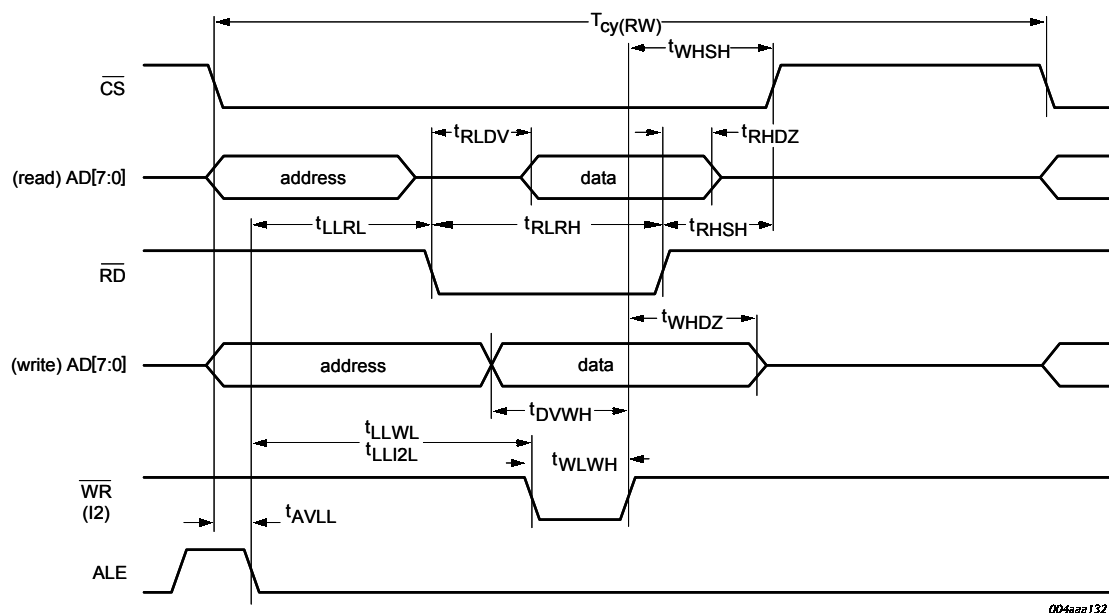


图 13 ISP1581 寄存器访问时序：多路复用地址/数据总线 (BUS_CONF=0, MODE1=0, MODE0=1)

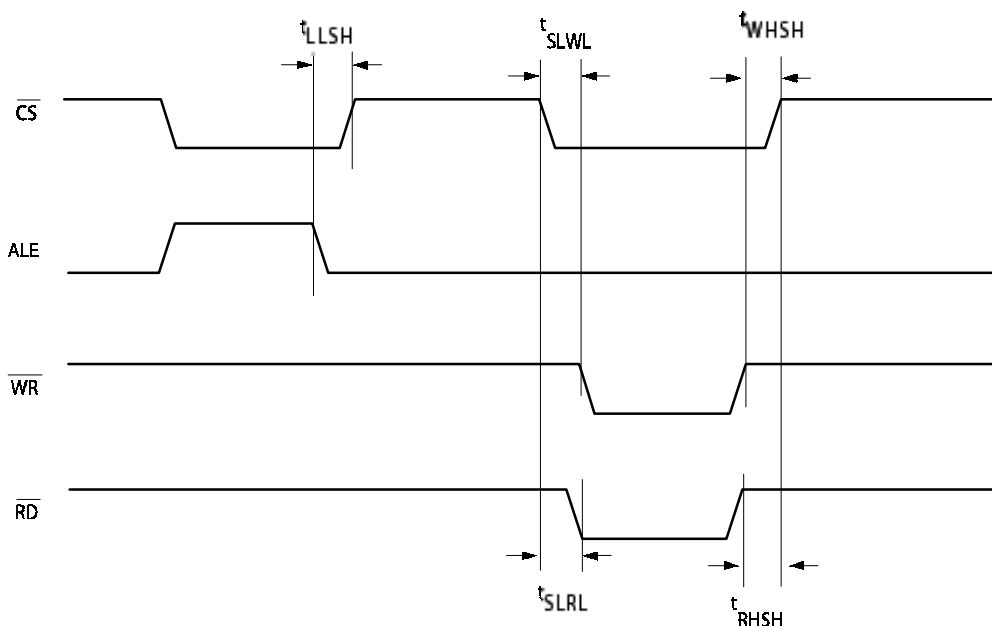


图 14 建立和保持时间

表 76: ISP1581 寄存器访问时序参数: 多路复用地址/数据总线 (MODE1=0)

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$.

符号	参数	最小	最大	单位
读				
t_{RLRH}	\overline{RD} 低脉冲宽度	$>t_{RLDV}$	-	ns
t_{RLDV}	\overline{RD} 低到数据有效的延时	-	25	ns
t_{RHDZ}	\overline{RD} 高到数据输出三态的延时	0	15	ns
t_{RHS}	\overline{RD} 高到 \overline{CS} 高的延时	0	-	ns
t_{LLRL}	\overline{RD} 变低前的 ALE 低建立时间	0	-	ns
写				
t_{WLWH}	$\overline{WR} / \overline{DS}$ 低脉冲宽度	15	-	ns
t_{DVWH}	\overline{WR} 变高前的数据建立时间	5	-	ns
t_{LLWL}	ALE 低到 $\overline{WR} / \overline{DS}$ 低的延时	0	-	ns
t_{WHDZ}	$\overline{WR} / \overline{DS}$ 变高后的数据保持时间	5	-	ns
t_{WHS}	$\overline{WR} / \overline{DS}$ 高到 \overline{CS} 高的延时	0	-	ns
常规				
$T_{CY(RW)}$	读/写周期	80	-	ns
t_{AVLL}	ALE 变低前的地址建立时间	5	-	ns
t_{I1VLL}	ALE 变低前的 R/ \overline{W} 建立时间	5	-	ns
t_{LL1L}	ALE 低到 \overline{DS} 低的延时	5	-	ns
t_{I2H1X}	\overline{DS} 变高后的 R/ \overline{W} 保持时间	5	-	ns
t_{LLSH}	ALE 低到 \overline{CS} 高	0	-	ns
t_{SLWL}	\overline{CS} 低到 \overline{WR} 低	0	-	ns
t_{SLRL}	\overline{CS} 低到 \overline{RD} 低			

断开总线模式 (BUS_CONF=0,MODE1=1,MODE0=0/1)

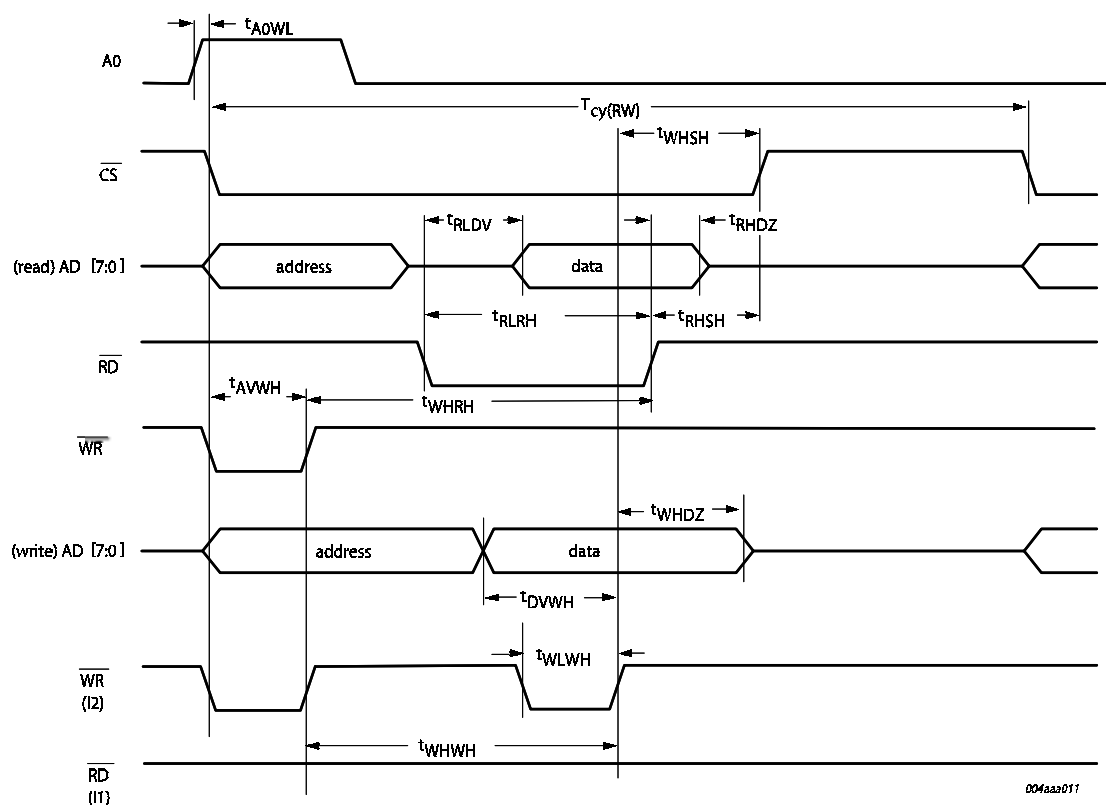


图 15 ISP1581 寄存器访问时序: 多路复用地址/数据总线 (BUS_CONF=0,MODE1=0,MODE0=1)

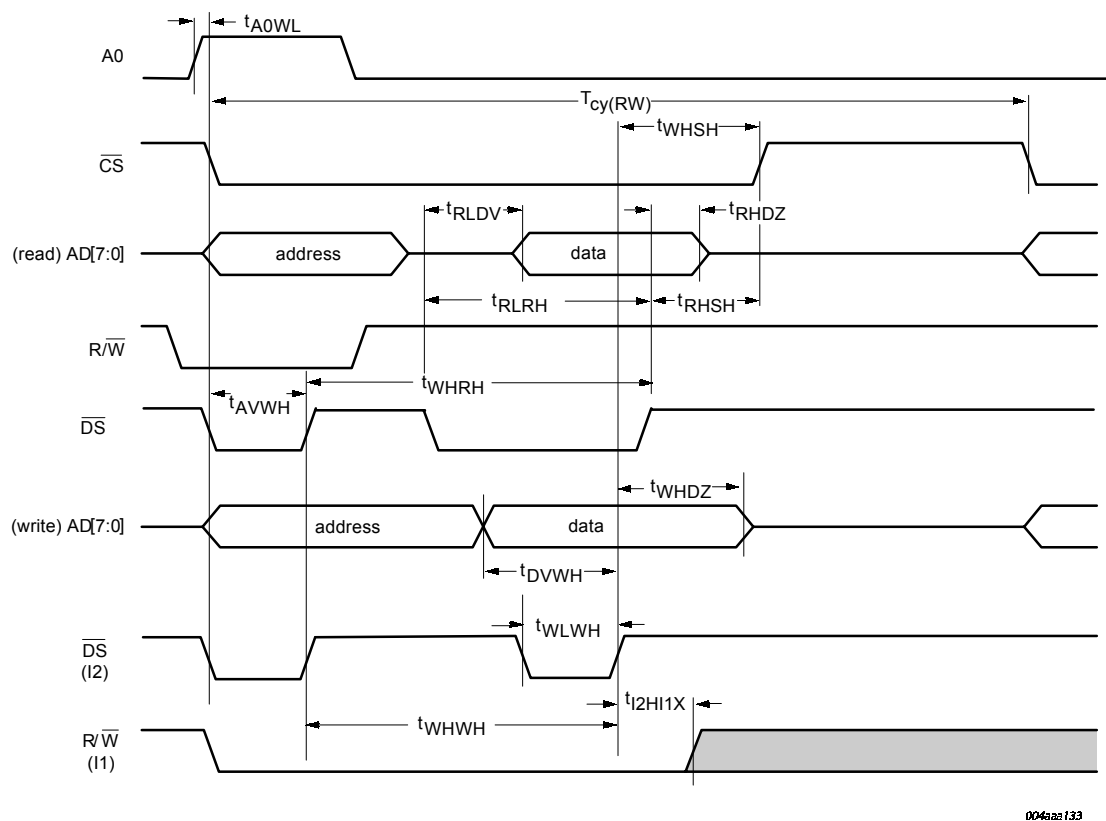


图 16 ISP1581 寄存器访问时序: 多路复用地址/数据总线 (BUS_CONF=0,MODE1=1,MODE0=0)

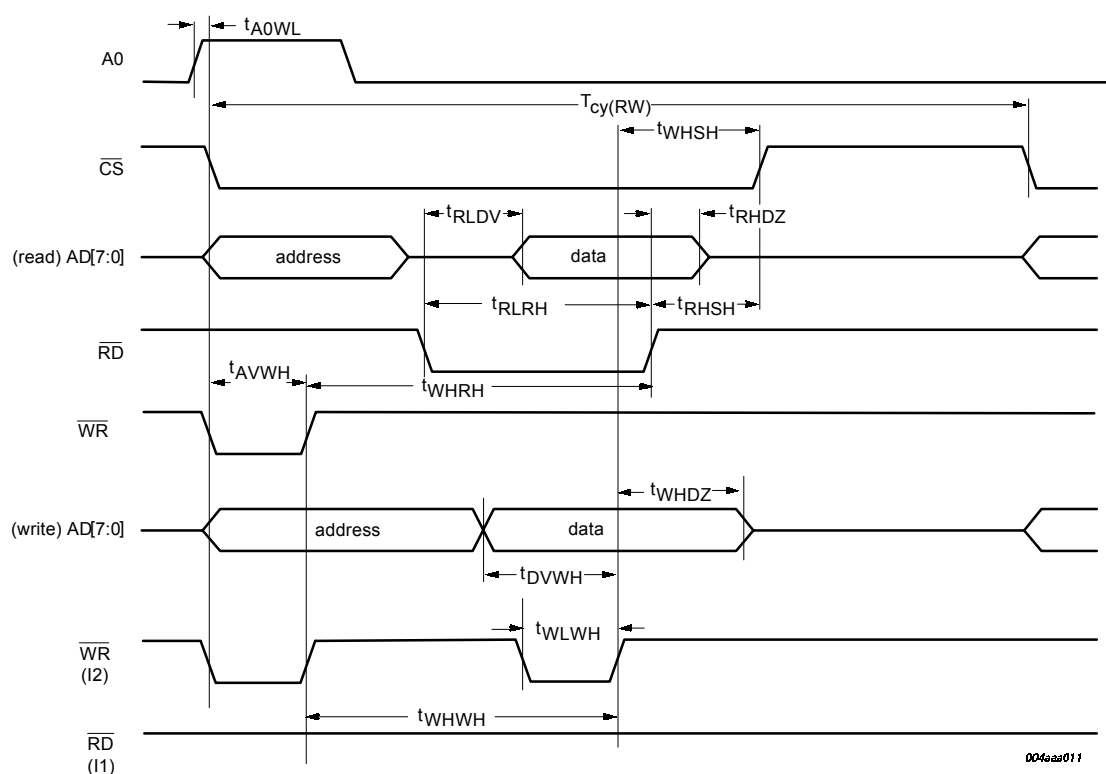
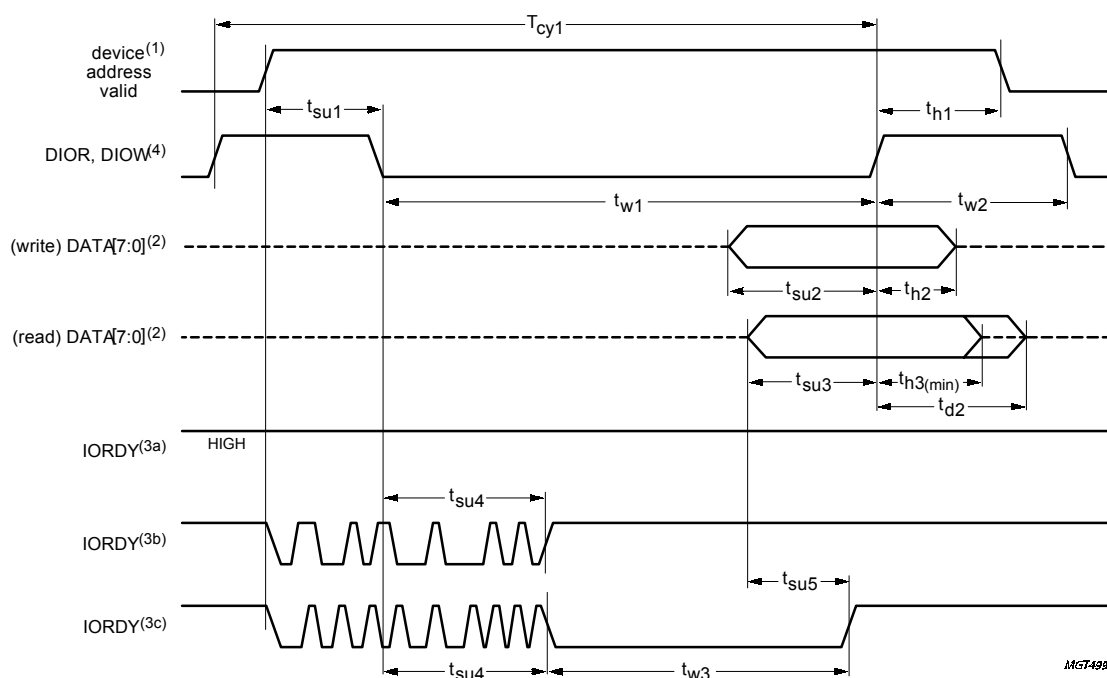


图 17 ISP1581 寄存器访问时序：多路复用地址/数据总线 (BUS_CONF=0,MODE1=1,MODE0=1)
 表 77: ISP1581 寄存器访问时序参数：多路复用地址/数据总线 (MODE1=1)
 $V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$.

符号	参数	最小	最大	单位
读				
t_{RLDV}	\overline{RD} 低到数据有效的延时	-	26	ns
t_{RHDZ}	\overline{RD} 高到数据输出为三态的延时	0	15	ns
t_{RHS}	\overline{RD} 高到 \overline{CS} 高的延时	0	-	ns
t_{RLRH}	\overline{RD} 低脉冲宽度	$>t_{RLDV}$	-	ns
t_{WHRH}	$\overline{WR} / \overline{DS}$ 高到 \overline{RD} 高的延时	40	-	ns
写				
t_{A0WL}	$\overline{WR} / \overline{DS}$ 变低前的 A0 建立时间	0	-	ns
t_{AVWH}	$\overline{WR} / \overline{DS}$ 变高前地址建立时间	5	-	ns
t_{DVWH}	$\overline{WR} / \overline{DS}$ 变高前的数据建立时间	5	-	ns
t_{WHDZ}	$\overline{WR} / \overline{DS}$ 变高后的数据保持时间	5	-	ns
t_{WHS}	$\overline{WR} / \overline{DS}$ 高到 \overline{CS} 高的延时	0	-	ns
t_{WHLWH}	$\overline{WR} / \overline{DS}$ 低脉冲宽度	15	-	ns
t_{WHWH}	$\overline{WR} / \overline{DS}$ 高 (地址) 到 $\overline{WR} / \overline{DS}$ 高 (数据) 的延时	40	-	ns
常规				
$T_{CY(RW)}$	读/写周期	80	-	ns
t_{I2H1X}	\overline{DS} 变高后的 R/ \overline{W} 保持时间	5	-	ns

14.3 DMA 时序

14.3.1 PIO 模式



- (1) 器件地址由信号 $\overline{CS1}$, $\overline{CS0}$, DA2, DA1 和 DA0 组成
- (2) 数据总线的宽度取决于 PIO 存取命令的使用。对任务文件寄存器采用 8 位的存取访问方式 (DATA[7:0], 任务文件 1F0 采用的是 16 位 DATA[15:0])。04H 和 05H DMA 命令也使用 16 位的数据总线。
- (3) 器件可通过 IORDY 的无效以等待状态的形式来延长 PIO 的周期。由主机来决定在 DIOR 或 DIOW 有效后的 t_{su4} 时间后是否需要延长当前的周期。下面分三种情况来说明：
- 器件使 IORDY 一直保持无效（高阻）：不产生等待状态。
 - 在 t_{su4} 时间内器件使 IORDY 无效， t_{su4} 时间到后变为有效：不产生等待状态。
 - 在 t_{su4} 时间内器件使 IORDY 无效， t_{su4} 时间到后的至少 5ns 的时间内仍使 IORDY 保持无效：产生一个等待状态。这个周期在 IORDY 重新有效时结束。为了延长读周期（DIOR 有效），在 IORDY 有效之前的 t_{d1} 时间内从 DATA_n 上读的数据必须一直保持有效。
- (4) DIOR 和 DIOW 含有可编程的触发极性：这里指定它们低电平有效。

图 17 PIO 模式时序

表 78: PIO 模式时序参数

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$ 。

符号	参数	模式 0	模式 1	模式 2	模式 3	模式 4	单位
$T_{cy1(min)}$	读/写周期（最小）	[1]600	383	240	180	120	ns
$t_{su1(min)}$	地址有效到 DIOR/DIOW 的建立时间（最小）	70	50	30	30	25	ns
$t_{w1(min)}$	DIOR/DIOW 脉冲宽度（最小）	[1]165	125	100	80	70	ns
$t_{w2(min)}$	DIOR/DIOW 恢复时间（最小）	[1]-	-	-	70	25	ns
$t_{su2(min)}$	DIOW 无效前的数据建立时间（最小）	60	45	30	30	20	ns
$t_{h2(min)}$	DIOW 无效后的数据保持时间（最小）	30	20	15	10	10	ns

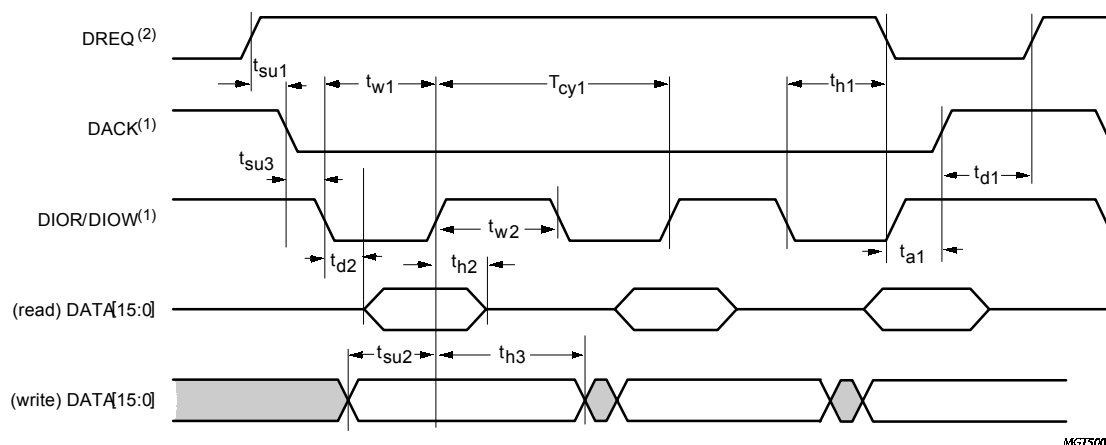
符号	参数	模式 0	模式 1	模式 2	模式 3	模式 4	单位
$t_{su3(min)}$	DIOR 无效前的数据建立时间 (最小)	50	35	20	20	20	ns
$t_{h3(min)}$	DIOR 无效后的数据保持时间 (最小)	5	5	5	5	5	ns
$t_{d2(max)}$	在 DIOR 无效后数据为三态的时间 (最大)	^[2] 30	30	30	30	30	ns
$t_{h1(min)}$	DIOR/DIOW 无效后的地址保持时间 (最小)	20	15	10	10	10	ns
$t_{su4(min)}$	DIOR/DIOW 有效后的 IORDY 建立时间 (最小)	^[3] 35	35	35	35	35	ns
$t_{su5(min)}$	读数据到 IORDY 高的时间 (最小)	^[3] 0	0	0	0	0	ns
$t_{w3(max)}$	IORDY 低脉冲宽度 (最大)	1250	1250	1250	1250	1250	ns

[1] T_{cy1} 是总的周期, 由命令有效时间 t_{w1} 和命令恢复 (=无效) 时间 t_{w2} 两部分组成: $T_{cy1}=t_{w1}+t_{w2}$, 要使得 T_{cy1} 达到最小, 要求 t_{w1} 和 t_{w2} 的匹配。如果 $T_{cy1(min)}$ 要比两者的和大, 那么主机操作的时间就要比 t_{w1} 和 t_{w2} 或两者之一更长, 这样才能确保 T_{cy1} 等于或大于标识设备数据的给定值。设备必须支持合法的主机操作。

[2] t_{d2} 指的是 DIOR 无效后的时间, 在这段时间内设备不再驱动数据总线 (三态)。

[3]如果 IORDY 在 t_{su4} 内为低电平, 则在 PIO 周期结束之前, 主机一直都处于等待状态直至 IORDY 变高。此时, t_{su5} 在时间上就必须满足读数据的要求 (此时 t_{su3} 未用)。当 IORDY 在 t_{su4} 内为高电平时, 则 t_{su5} 在时间上就必须满足读数据的要求 (此时 t_{su5} 未用)。

14.3.2 GDMA 从机模式

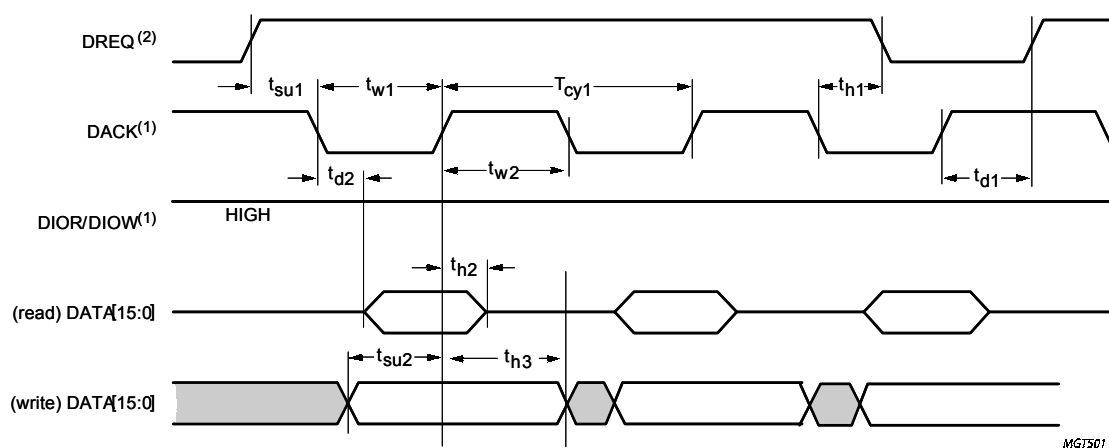


DREQ 信号不断的保持有效直到最后一次传输结束或 FIFO 已满。

数据选通信号: DIOR(读), DIOW (写)。

- (1) 信号的触发极性可编程: 图中所示为低电平有效。
- (2) 信号的触发极性可编程: 图中所示为高电平有效。

图 19 GDMA 从机模式时序 (BURST=00H,MODE=00H)

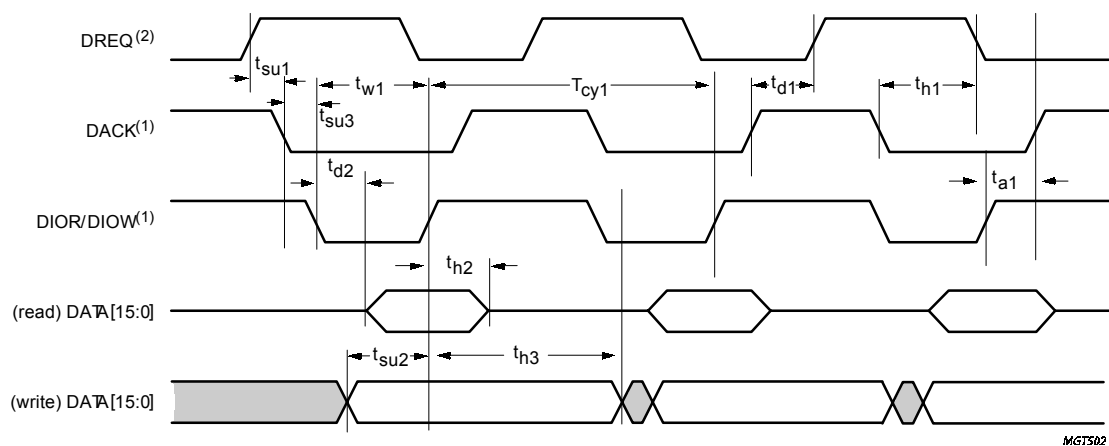


DREQ 信号不断的有效直到最后一次传输结束或 FIFO 已满。

数据选通信号：DACK(读/写)。

- (1) 信号的触发极性可编程：图中所示为低电平有效。
- (2) 信号的触发极性可编程：图中所示为高电平有效。

图 20 GDMA 从机模式时序 (BURST=00H,MODE=02H)

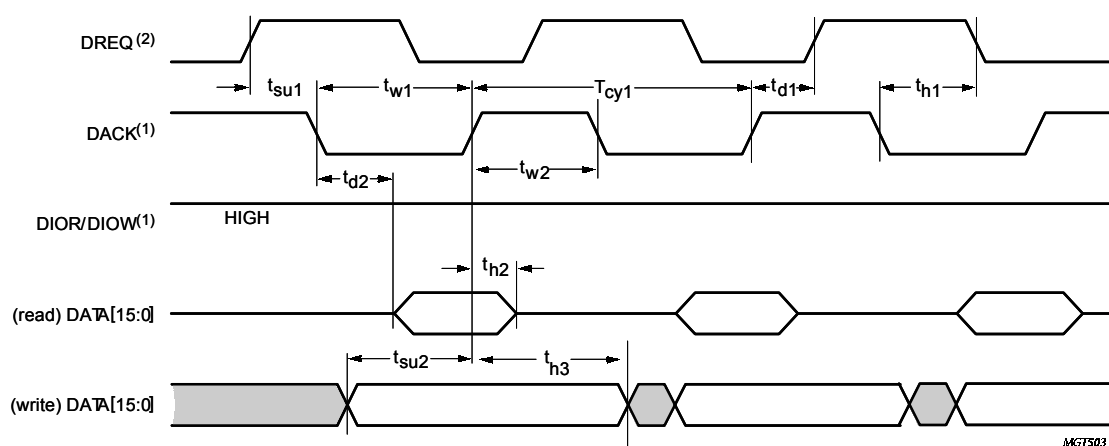


每发生一次传输，DREQ 就有效一次。

数据选通信号：DIOR (读)，DIOW(写)

- (1)信号的触发极性可编程：图中所示为低电平有效
- (2)信号的触发极性可编程：图中所示为高电平有效。

图 21 GDMA 从机模式时序 (BURST=01H,MODE=00H)



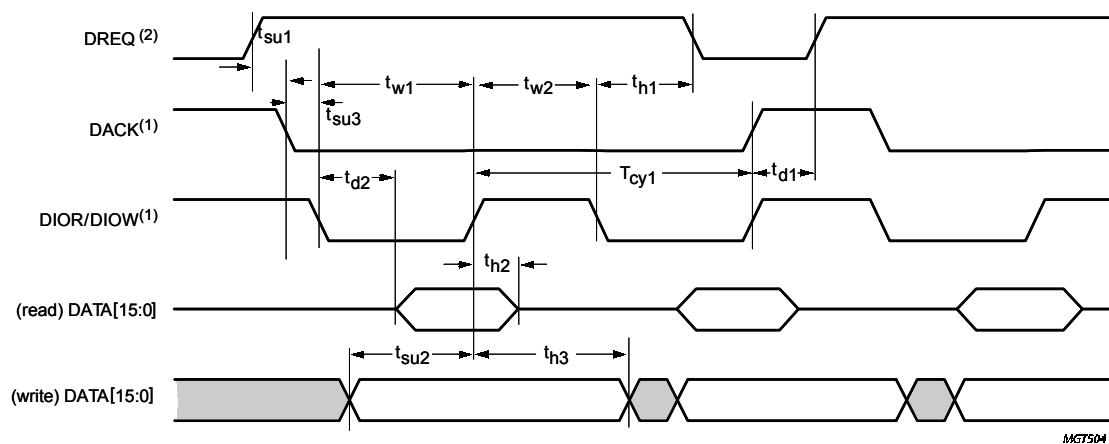
每发生一次传输，DREQ 就有效一次。

数据选通信号：DACK(读/写)。

(1)信号的触发极性可编程：图中所示为低电平有效

(2)信号的触发极性可编程：图中所示为高电平有效。

图 22 GDMA 从机模式时序 (BURST=01H,MODE=02H)



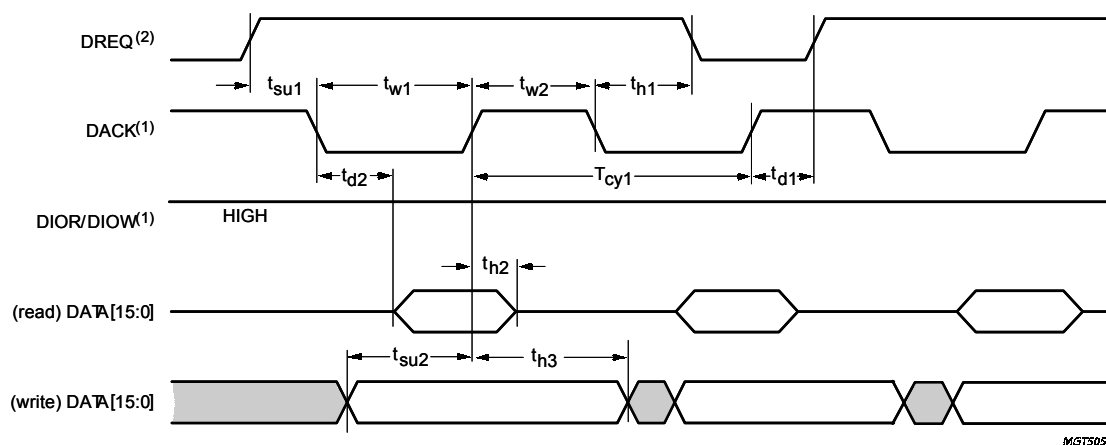
每发生 N 次传输 DREQ 有效一次 (N 由 BURST 的值决定)。在此例中：N=2。

数据选通信号：DIOR (读)，DIOW(写)

(1)信号的触发极性可编程：图中所示为低电平有效

(2)信号的触发极性可编程：图中所示为高电平有效。

图 23 GDMA 从机模式时序 (BURST>01H,MODE=00H)



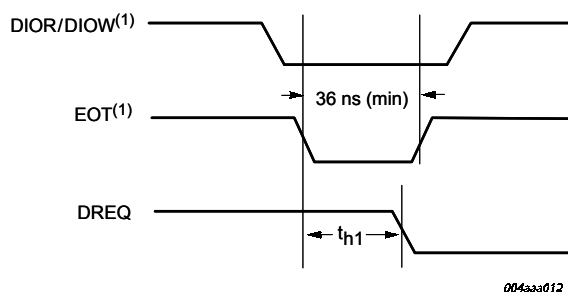
每发生 N 次传输 DREQ 有效一次 (N 由 BURST 的值决定)。在此例中: N=2。

数据选通信号: DIOR (读), DIOW(写)

(1)信号的触发极性可编程: 图中所示为低电平有效

(2)信号的触发极性可编程: 图中所示为高电平有效。

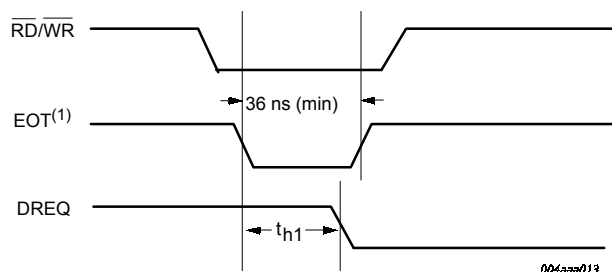
图 24 GDMA 从机模式时序 (BURST>01H,MODE=02H)



(1)信号的触发极性可编程: 图中所示为低电平有效。

注: 当 DIOR/DIOW 有效时, EOT 信号保持有效达 36ns (最小)。

图 25 断开总线模式的 EOT 时序



(1)信号的触发极性可编程: 图中所示为低电平有效。

注: 当 RD/WR 有效时, EOT 信号保持有效达 36ns (最小)。

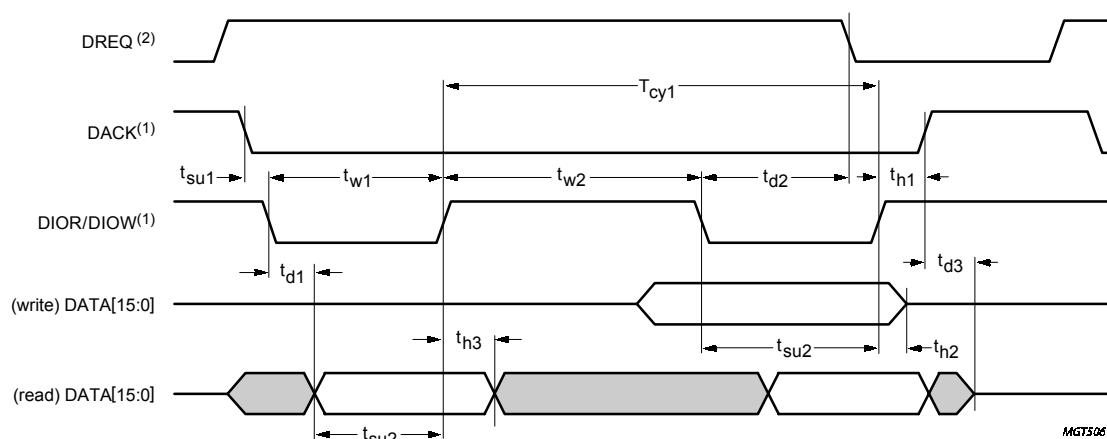
图 26 通用处理器模式的 EOT 时序

表 79: GDMA 从机模式时序参数

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$.

符号	参数	最小	最大	单位
T_{cy1}	读/写周期	78	-	ns
t_{su1}	第一个 DAACK 有效前的 DREQ 建立时间	10	-	ns
t_{d1}	最后的选通无效后 DRREQ 有效的延时	33.33	-	ns
t_{h1}	最后的选通有效后 DREQ 有效保持时间	0	53	ns
t_{w1}	DIOR/DIOW 脉冲宽度	39	600	ns
t_{w2}	DIOR/DIOW 恢复时间	36	-	ns
t_{d2}	选通有效后的读数据有效延时	-	20	ns
t_{h2}	选通无效后的读数据保持时间	-	5	ns
t_{h3}	选通无效后的写数据保持时间	1	-	ns
t_{su2}	选通无效前的写数据建立时间	10	-	ns
t_{su3}	DIOR/DIOW 有效前的 DACK 建立时间	0	-	ns
t_{a1}	DIOR/DIOW 无效到 DACK 无效的时间	0	30	ns

14.3.3 MDMA 模式



(1)信号的触发极性可编程：图中所示为低电平有效。

(2)信号的触发极性可编程：图中所示为高电平有效。

图 27 MDMA 主机模式时序

表 80: MDMA 模式时序参数

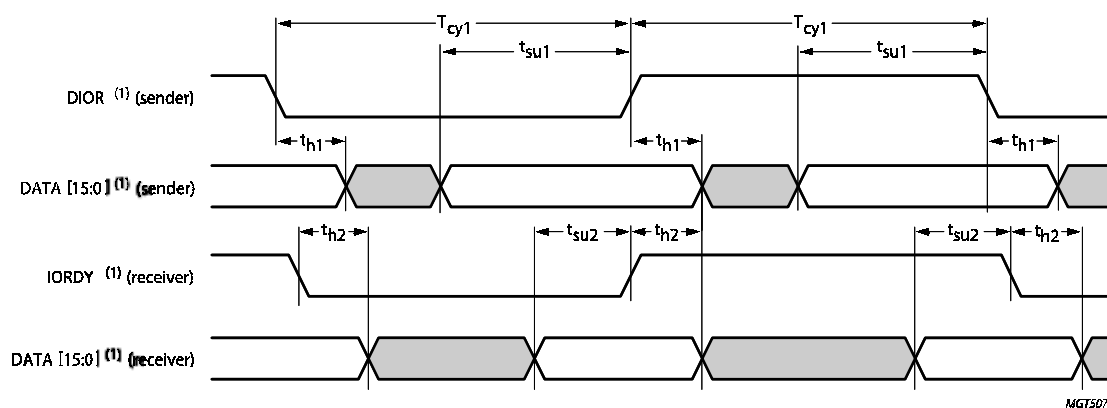
$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$.

符号	参数	模式 0	模式 1	模式 2	单位
$T_{cy1}(\min)$	读/写周期 (最小) [1]	480	150	120	ns
$t_{w1}(\min)$	DIOR/DIOW 脉冲宽度 (最小) [1]	215	80	70	ns
$t_{d1}(\max)$	DIOR 有效后的数据有效延时	150	60	50	ns
$t_{h3}(\min)$	DIOR 无效后的数据保持时间 (最小)	5	5	5	ns
$t_{su2}(\min)$	DIOR/DIOW 无效前的数据建立时间 (最小)	100	30	20	ns
$t_{h2}(\min)$	DIOW 无效后的数据保持时间 (最小)	20	15	10	ns
$t_{su1}(\min)$	DIOR/DIOW 有效前的 DACK 建立时间 (最小)	0	0	0	ns
$t_{h1}(\min)$	DIOR/DIOW 无效后的 DACK 保持时间	20	5	5	ns

符号	参数	模式 0	模式 1	模式 2	单位
tw2(min)	DIOR 恢复时间 (最小) [1]	50	50	25	ns
	DIOW 恢复时间 (最小) [1]	215	50	25	ns
td2(max)	DIOR 有效到 DREQ 无效的延时 (最大)	120	40	35	ns
	DIOW 有效到 DREQ 无效的延时 (最大)	40	40	35	ns
td3(max)	DACK 无效到数据线变为三态的延时 (最大)	20	25	25	ns

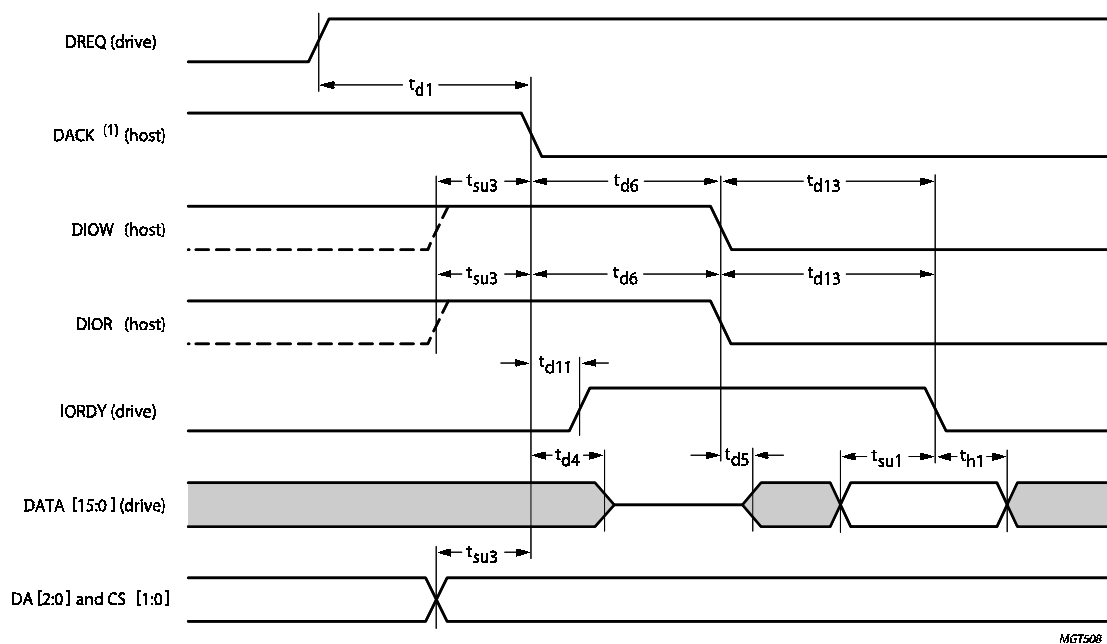
[1] T_{cy1} 是总的周期, 由命令有效时间 t_{w1} 和命令恢复 (=无效) 时间 t_{w2} 两部分组成: $T_{cy1} = t_{w1} + t_{w2}$, 要使得 T_{cy1} 达到最小, 要求 t_{w1} 和 t_{w2} 的匹配。如果 $T_{cy1(min)}$ 要比两者的和大, 那么主机操作的时间就要比 t_{w1} 和 t_{w2} 或两者之一更长, 这样才能确保 T_{cy1} 等于或大于标识设备数据的给定值。设备必须支持合法的主机操作。

14.3.4 UDMA 模式



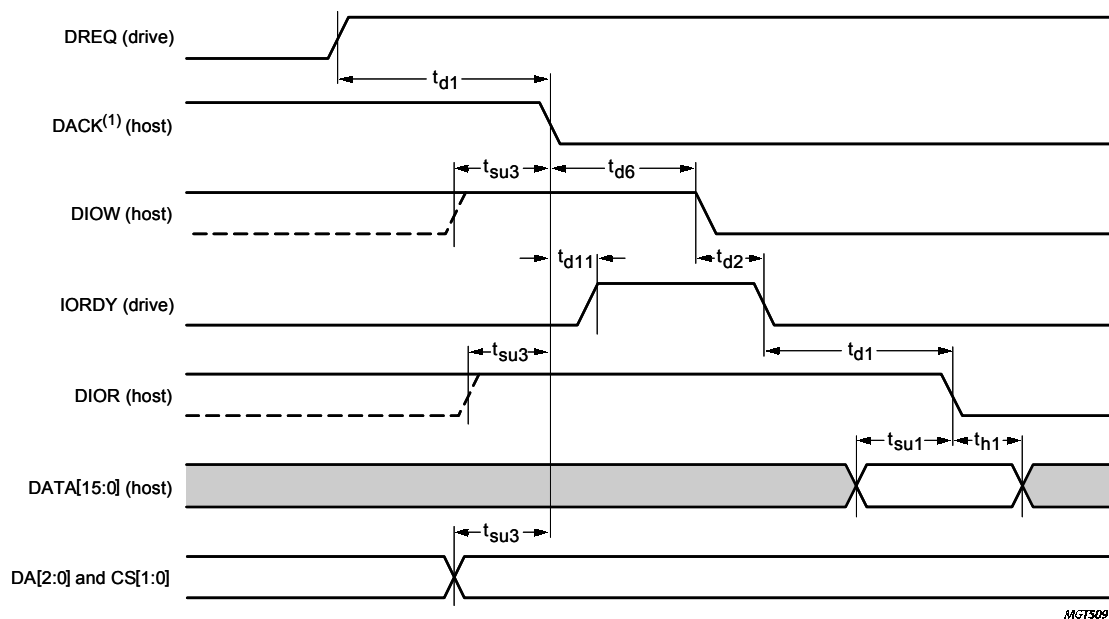
(1) 由于信号在电缆上的停留时间和传输延时的存在, DATA[15:0]和接收器的选通信号需要一段稳定时间。

图 28 UDMA 时序: 不间断的同步脉冲



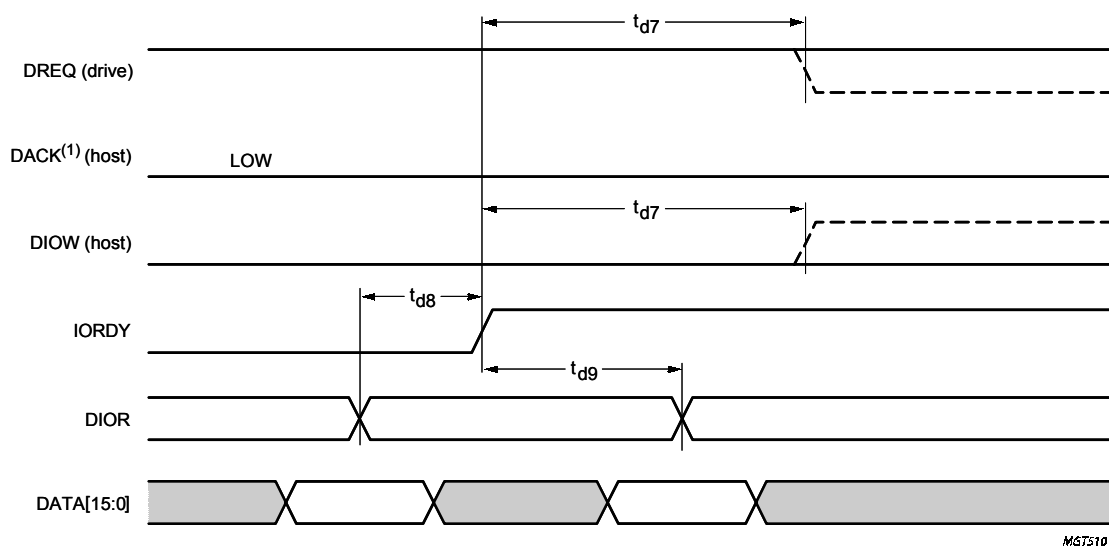
(1) 信号的触发极性可编程: 图中所示为低电平有效。

图 29 UDMA 时序: 驱动器读命令脉冲的产生



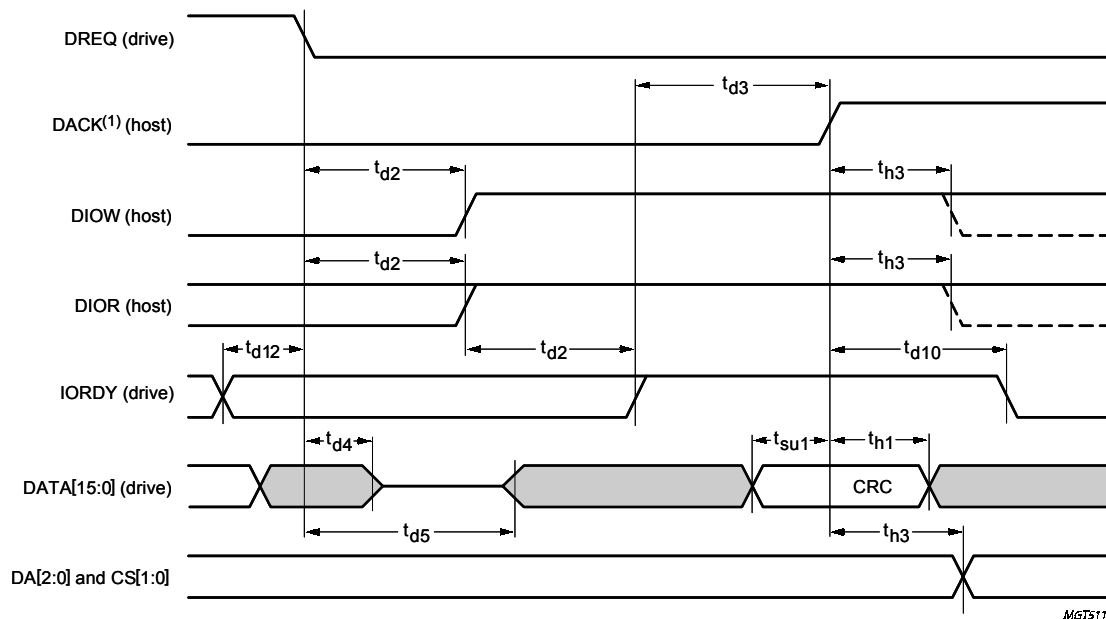
(1) 信号的触发极性可编程：图中所示为低电平有效。

图 30 UDMA 时序：驱动器写命令脉冲的产生



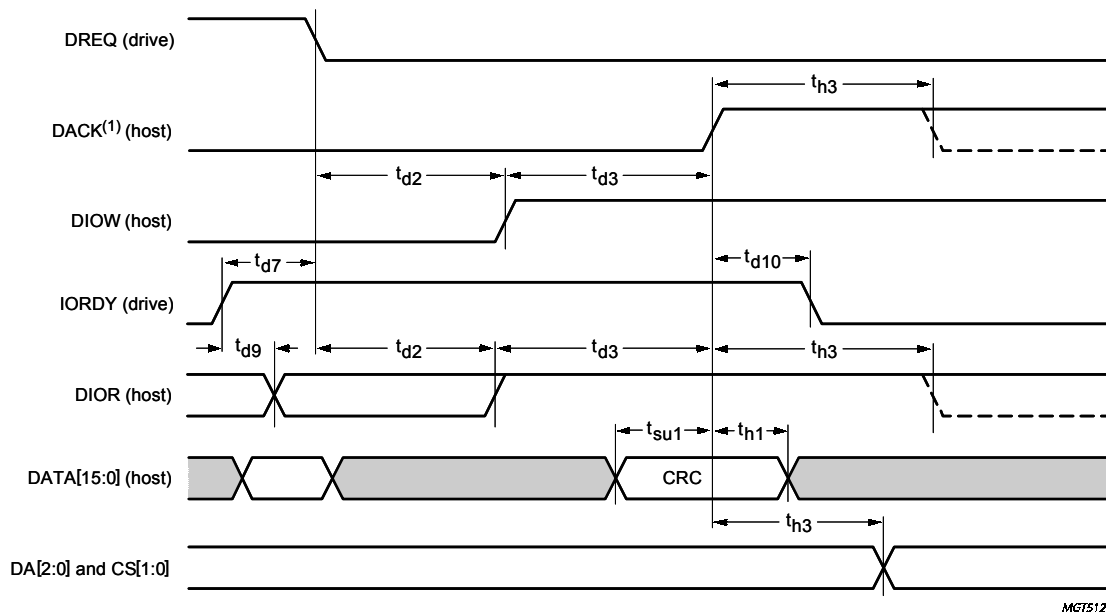
(1) 信号的触发极性可编程：图中所示为低电平有效。

图 31 UDMA 时序：接收器暂停脉冲产生



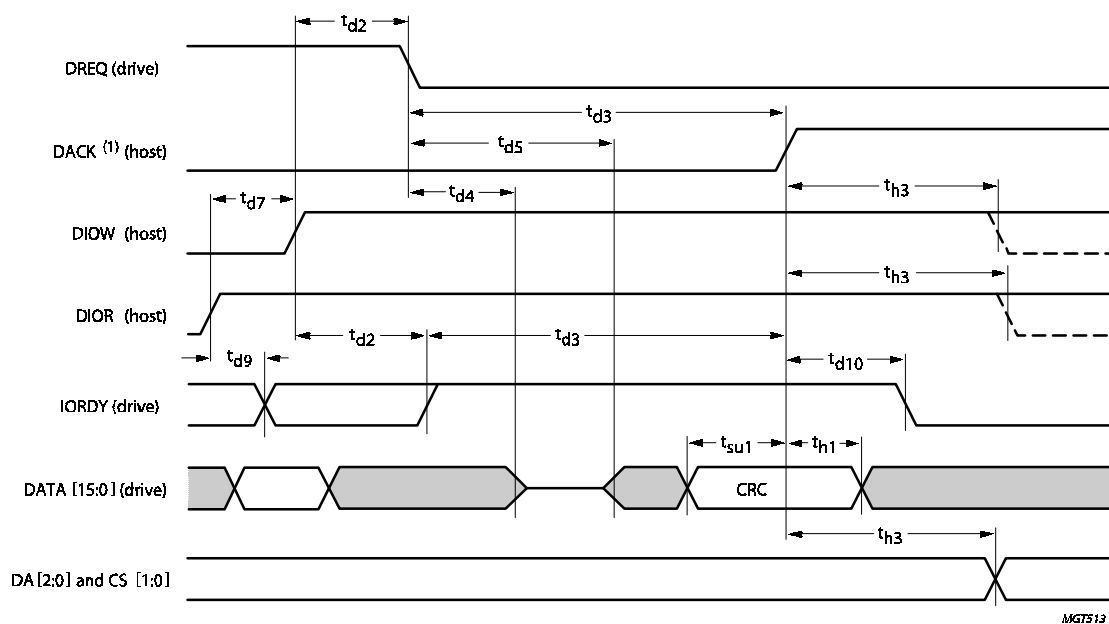
(1) 信号的触发极性可编程：图中所示为低电平有效。

图 32 UDMA 时序：驱动器读命令终止脉冲的产生



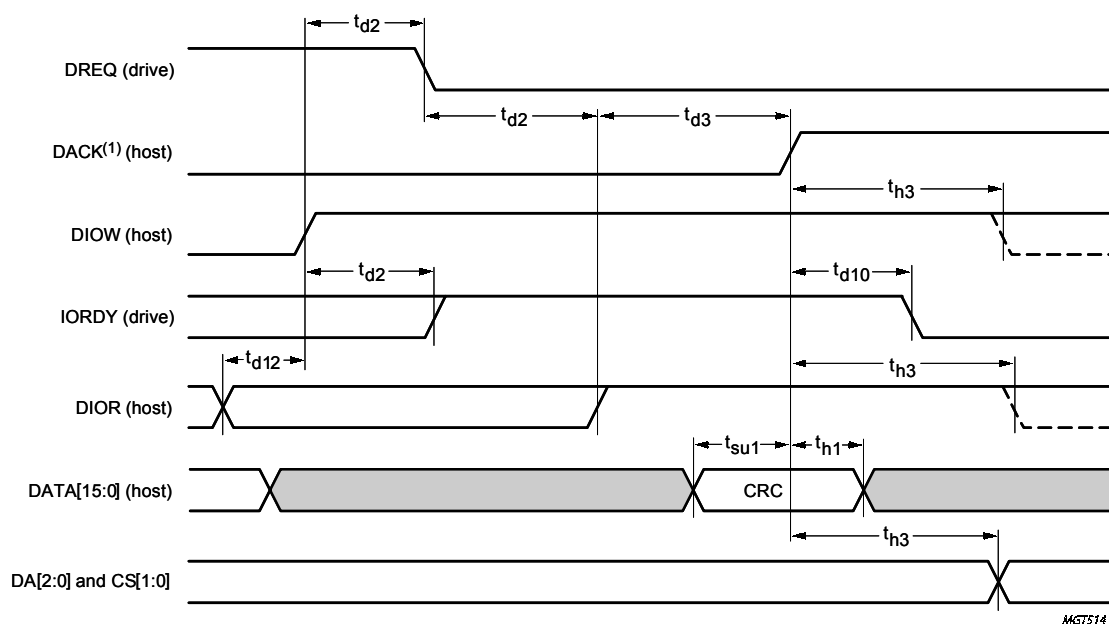
(1) 信号的触发极性可编程：图中所示为低电平有效。

图 33 UDMA 时序：驱动器写命令终止脉冲的产生



(1) 信号的触发极性可编程：图中所示为低电平有效。

图 34 UDMA 时序：主机读命令终止脉冲的产生



(1) 信号的触发极性可编程：图中所示为低电平有效。

图 35 UDMA 时序：主机写命令终止脉冲的产生

表 81: UDMA 模式时序参数

$V_{CC}=4.0$ 到 $5.5V$; $V_{GND}=0V$; $T_{amb}=-40$ 到 $+85^{\circ}C$ 。

符号	参数	模式 0		模式 1		模式 2		单位
		最小	最大	最小	最大	最小	最大	
T_{cy1}	读/写周期 (信号的一个选通边沿到另一个选通边沿)	114	-	75	-	55	-	ns
t_{su2}	接收器数据建立时间	15	-	10	-	7	-	ns
t_{h2}	接收器数据保持时间	5	-	5	-	5	-	ns
t_{su1}	发送器数据建立时间	70	-	48	-	34	-	ns
t_{h1}	发送器数据保持时间	6	-	6	-	6	-	ns
t_{d1}	不受限的内部时钟时间 ^[1]	0	-	0	-	0	-	ns
t_{d2}	受限的内部时钟时间 ^[1]	0	150	0	150	0	150	ns
t_{d3}	受限内部时钟时间 ^[1] 的最小值	20	-	20	-	20	-	ns
t_{d4}	数据线驱动器断开延时	-	10	-	10	-	10	ns
t_{d5}	数据线驱动器接通延时 (主机)	20	-	20	-	20	-	ns
	数据线驱动器接通延时 (驱动器)	0	-	0	-	0	-	ns
t_{su3}	DACK 有效前的控制信号建立时间	20	-	20	-	20	-	ns
t_{h3}	DACK 无效后的控制信号保持时间	20	-	20	-	20	-	ns
t_{d6}	DACK 有效到控制信号转换的时间	20	70	20	70	20	70	ns
t_{d7}	准备好到暂停开始的延时	160	-	125	-	100	-	ns
t_{d8}	选通到准备好产生一个同步暂停信号的时间	-	50	-	30	-	20	ns
t_{d9}	准备好到后选通沿的时间	-	75	-	60	-	50	ns
t_{d10}	DACK 无效到 IORDY 高阻态的时间	-	20	-	20	-	20	ns
t_{d11}	DACK 有效到 IORDY 高的时间	0	-	0	-	0	-	ns
t_{d12}	后选通边沿到 DREQ 无效或 DIOW 有效的的时间	50	-	50	-	50	-	ns
t_{d13}	控制信号有效到第一个选通沿的时间	0	230	0	200	0	170	ns

[1]互锁时间就是一个动作者的动作到在其它动作者的作用下发生的下一个动作之间的时间间隔。动作者可以是发送者或接收者。只有得到其它动作者的响应,互锁才可得以继续进行。即互锁只有在至少有两者参与的情况下才能发生。

15. 应用

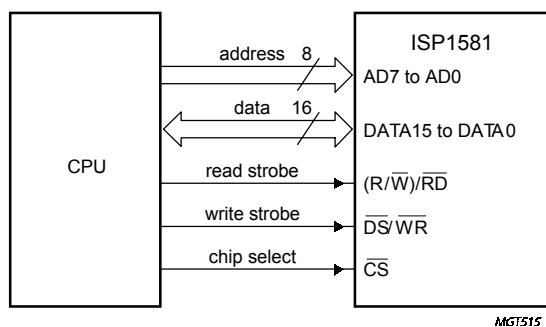


图 36 通用处理器模式的典型接口电路

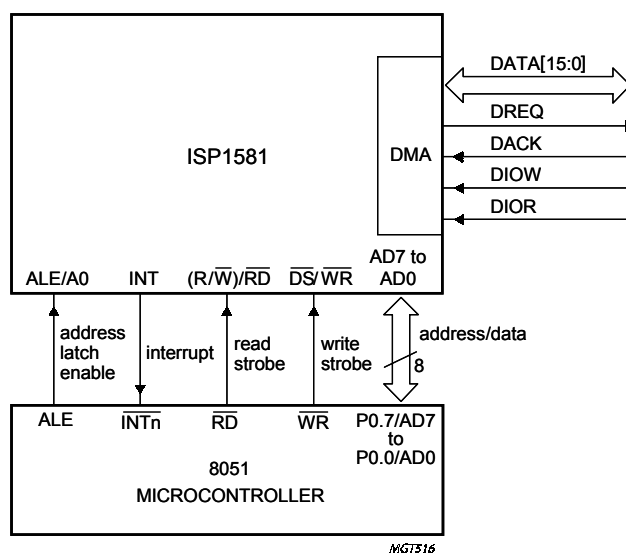
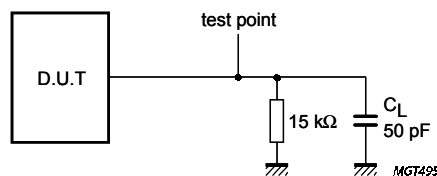


图 37 断开总线模式的典型接口电路（从机模式）

16. 测试

模拟 I/O 端口 (D+,D₋) 的动态特性在表 75 中已列出，它是通过图 37 的电路测试得到的。

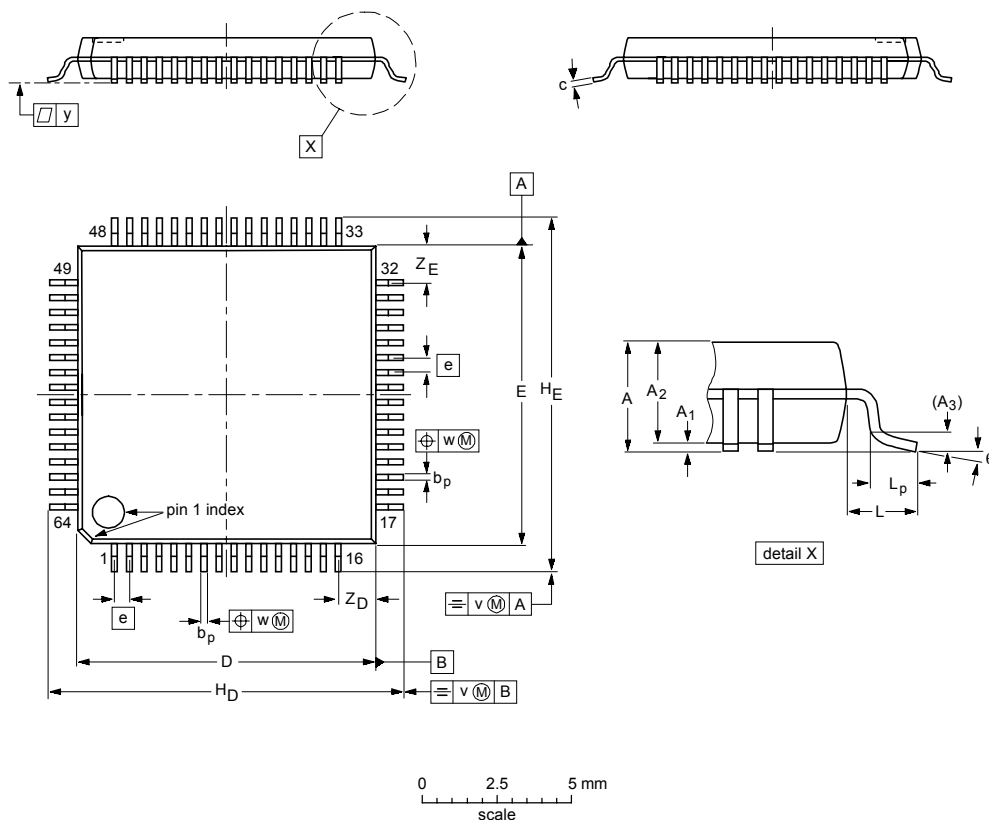


在全速模式中，内部有一个 1.5KΩ 的上拉电阻与管脚 D+ 相连。

图 38 D+和 D₋管脚的负载阻抗（全速模式）

17. 封装

LQFP64: 小塑料方形扁平封装; 64 脚;



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A ₁	A ₂	A ₃	b _p	c	D ⁽¹⁾	E ⁽¹⁾	e	H _D	H _E	L	L _p	v	w	y	Z _D ⁽¹⁾	Z _E ⁽¹⁾	θ
mm	1.60	0.20 0.05	1.45 1.35	0.25	0.27 0.17	0.18 0.12	10.1 9.9	10.1 9.9	0.5	12.15 11.85	12.15 11.85	1.0	0.75 0.45	0.2	0.12	0.1	1.45 1.05	1.45 1.05	7° 0°

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT314-2	136E10	MS-026				99-12-27 00-01-19