

1. 串行总线优点:

- (1) 传输距离更远, 信号的干扰更小;
- (2) 需要的电缆数量较少;
- (3) 传输速率更高, 且可以在传输的过程中进行校验, 因此更可靠;

并行总线优点: 传输速率虽然有限, 但其在传输小数据时具有很高的效率。

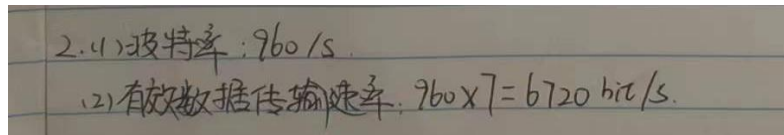
串行总线缺点:

- (1) 传输速率取决于其传输的频率, 因此需要较高的频率才能达到快速传输;
- (2) 对于大数据的传输, 串行总线的速率会受到限制;

并行总线缺点:

- (1) 连接需要使用更多的电缆;
- (2) 需要更多的处理器和内存。

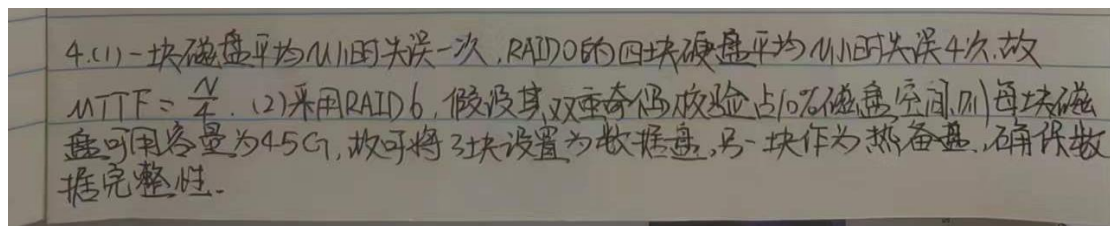
造成它们接口速率不同的原因是因为串行总线在传输数据的时候需要逐个传输, 而并行总线则可以同时传输多个数据位。串行总线的速率受限于其传输的频率, 而并行总线的速率取决于它的宽度和吞吐量。因此, 在小数据传输方面, 并行总线是更快的, 而在大数据传输方面, 串行总线是更快的。



3. (1) I2C 的数据包由起始位、从机地址、读写位、数据位和停止位组成。当主机想要与 I2C 从机进行通信时, 它会发送一个起始位信号, 然后发送从机地址和读写位, 以指示其希望读取还是写入数据。从机在收到其地址时响应, 然后, 如果需要, 主机将发送一串数据位。最后, 主机向从机发送停止位以结束数据传输。

(2) I2C 是半双工的, 这意味着数据只能在一个方向上传输。在 I2C 通信期间, 主机和从机之间会依次进行发送和接收数据的操作, 而不是同时进行。这是因为 I2C 只使用两根线进行通信: 一个是时钟线, 另一个是数据线。数据线的状态取决于时钟线的状态, 在时钟线低电平时, 数据线传输数据, 而在时钟线高电平时, 数据线上的数据被读取。因此, 由于只使用了一根数据线, 因此 I2C 是半双工协议。

(3) I2C 的起始条件是主机在时钟线为高电平时将数据线的电平从高电平拉低。而停止条件是当主机在时钟线为高电平时将数据线的电平从低电平拉高。从机在检测到起始条件后会开始应答主机的命令, 在传输数据完成后, 主机将发送停止条件。



5. 寻道时间: 磁头从当前位置到目标磁道并消除抖动所需要的时间。

旋转时间: 磁头移动到目标磁道后, 目标扇区随着盘片转动而经过磁头下方所需的时间。

数据传输时间: 磁头完成读出或写入所需的时间。

影响因素:

寻道时间: 寻道距离, 磁头定位精度, 磁头移动速度, 磁盘旋转速度。

旋转时间: 磁盘转速, 数据所在位置。

数据传输时间: 数据块大小, 数据传输数据, 磁盘缓存, 磁盘控制性能。

6. (1) 总容量: $6 \times 240 \times 12 \text{ KB} = 17280 \text{ KB}$ 。

(2) 数据传输速率: $6 \times 12 \times \frac{5400}{60} = 6480 \text{ KB/s}$

(3) 平均旋转时间: $\frac{60}{5400} \times \frac{1}{2} = 5.56 \text{ ms}$ 。

7. 磁盘控制电路可以通过对请求的调度来优化磁盘的访问次序。如果磁盘控制电路将访问请求调度成最优的顺序, 可以最大限度地最小化磁盘的寻道、旋转和传输时间, 从而减少磁盘访问用时。常见的磁盘调度算法包括先来先服务 (First Come, First Serve)、最短寻找时间优先 (Shortest Seek Time First)、电梯算法 (Elevator Algorithm) 等。这些算法可以根据不同的场景和需求来进行选择, 以达到最小化磁盘访问用时的目的。

8. 对于 RAID4 中的写入优化, 它主要是针对数据块和奇偶校验块的存储位置做出优化。RAID4 中采用串行写入方式, 即将要写入的数据块写入到同时位于不同物理磁盘的位置上, 而奇偶校验块则需要与数据块一起写入到其中一个磁盘的位置上。因此, 通过将奇偶校验块写入到不同的位置, 可以避免多个写操作同时对同一磁盘进行操作, 从而提高写入的效率。

9. 由 $W = \mu - \lambda$, 当磁盘 I/O 请求减少时, 即 I/O 请求到达率 λ 减小, 而平均服务率 μ 不变时, 平均响应时间减小并趋于 $\frac{1}{\mu}$ 。

1. (1) 集中式仲裁: 由一个专门的仲裁器来控制整个总线的使用权, 其他设备在需要访问总线时需要向仲裁器发出请求。优点是实现简单, 但是会造成总线瓶颈和单点故障, 并且在多个设备同时请求时效率低下。

(2) 分布式仲裁: 每个设备都会有一个仲裁电路, 当需要访问总线时会向总线发送仲裁请求信号, 并进行仲裁。优点是不需要专门的仲裁器, 可以提高高效性, 但是实现复杂, 需要保证所有仲裁电路的同步性, 否则可能会导致死锁。

(3) 抢占式仲裁: 设备在访问总线时会检查总线上是否有正在传输的数据, 如果没有则可以直接访问; 如果有, 则会抢占总线访问权, 将当前正在访问的设备中断, 自己获得访问权。优点是能够尽快完成访问, 但是可能会导致更长的延迟, 并且可能会丢失数据。

(4) 时间片仲裁: 将总线时间分成若干个时间片, 每个设备在固定的时间片内获得总线访问权。优点是可以保证公平性, 并且不会出现死锁, 但是可能会浪费一定的时间片, 降低访

问效率。

不同的总线仲裁机制适用于不同的场景。集中式仲裁适用于设备数量小，访问请求不密集的场景。分布式仲裁适用于设备数量较多，并且需要高效利用总线资源的场景。抢占式仲裁适用于延迟要求较高，但访问请求稀疏的场景。时间片仲裁适用于设备数量较多，需要保证公平性和稳定性的场景。

2.APB 总线协议：Advanced Peripheral Bus (APB)，是在系统中连接芯片外设的简单总线。APB 总线采用了简单的、低功耗的点对点串行结构，适用于连接低速外设（如低速 ADC 等）。

AHB 总线协议：Advanced High-performance Bus (AHB)，是 AMBA 总线族中第二代总线协议。AHB 总线采用了点对点的总线连接方式，实现了高带宽、低延时的片上总线通信，适用于连接高性能的外设与内存。

AXI 总线协议：Advanced eXtensible Interface (AXI)，是在 AMBA 总线中最高级别的通用总线协议。AXI 总线可以连接多种类型的主处理器和外设，在高速数据传输和低延迟的同时，实现了高度的可定制性。因此，AXI 总线常用于高性能计算和嵌入式系统中。

ACE 总线协议：AXI Coherency Extensions (ACE)，是在 AXI 总线协议基础上增加了缓存互联和数据一致性的特色协议，使得多处理器系统可以共享一致的缓存数据和不同节点之间的数据传输。因此，ACE 总线协议适用于多处理器高性能计算系统。

CHI 总线协议：Coherent Hub Interface (CHI)，一种高性能、低功耗、多个主处理器和多个从设备之间进行通信的总线协议。CHI 总线协议适用于需要支持高种类和速率的芯片设计，例如专业网络、高速计算机视觉和高速数据通信应用。

3. (1) AXI 总线包含五个独立的事务通道，分别为读通道、写通道、读写同步通道、写唤醒通道和全局通道。由于读响应与读请求在同一通道上进行，响应与请求采用不同的标识进行区分。通过在响应信号中返回相应的标识，使得读响应可以和读请求对应起来。

(2) 在读/写传输事务中，通道的握手信号时序需要满足以下依赖关系：先发送请求信号，然后是地址和有效数据信号，最后是读取或写入完成信号。这是因为只有在读/写请求信号到达后，才会发送地址和有效数据信号，而读写完成信号只会在地址和有效数据信号被全部传输之后才会发送。

(3) AXI 的突发传输是指一次传输中可以包含多个地址的读写操作，以减少传输时的延迟。突发传输类型包括固定突发 (Fixed Burst)、递增突发 (Incrementing Burst) 和 Wrap 突发 (Wrapping Burst)。固定突发是在读/写开始时同时传输多个数据，递增突发是在传输多次时地址递增，而 Wrap 突发则是在地址递增到最高位时回卷到最低位，并在此基础上继续递增。通过使用突发传输模式，可以提高总线性能并减少总线开销。