

第十五周作业

1. 串行总线 { 优点: 物理连线数目少, 消耗硬件资源少, 功耗低, 高频性能好
缺点: 相同频率下传输速率较并行慢

并行总线 { 优点: 可同时传输更多数据, 同频率下传输速率更快
缺点: 较多的物理连线及相应硬件资源, 功耗高, 高频性能差

接口速率不同原因: 随着处理器发展, 工作频率逐渐提高, 串行可以匹配高频率, 而并行受串扰影响, 其工作频率上限较低。故现在大多使用串行总线, 其高频率弥补了其每次传输个数少的缺点, 可以超过并行总线速率。此外, 串行总线还有时钟恢复电路, 差分信号传输, 信道均一化等技术进一步提升速率。

2.

11) UART 异步串行

1个包 $1+7+1+1=10$ 位

$960 \times 10 = 9600$ 位 对应 9600 个周期

波特率为一秒所运行周期数, 即 9600 波特/秒

12) $7 \times 960 = 6720 \text{ b} = 840 \text{ B}$

$\therefore 840 \text{ B/s}$

3. (1) 起始位 + 7~10位地址帧 + 读/写位 + ACK/NACK位 + 8位数据帧1

+ ACK/NACK位 + 8位数据帧2 + ACK/NACK位 + 终止位

(2) 因为I2C简化了硬件接线, 只有一根串行数据线SDA, 所以一个周期只能完成一次单向传输, 但其方向由主到从, 由从到主均可, 故为半双工

(3) 起始条件: 在SCL为高电平时, SDA由高向低跳变

终止条件: 在SCL为高电平时, SDA由低向高跳变

4. (1) **MTTF 失效前平均工作时间**

$$MTTF = \frac{N}{4} \text{ (小时)}$$

(2) 采用RAID-1方式

2块存储, 2块备份

$$MTTF = 2N \text{ (小时)}$$

5. 寻道时间: 磁头臂移动到正确位置并消除抖动所需要的时间

旋转时间: 通过旋转来使得正确的扇区被旋转到磁头的正下方所需时间

数据传输时间: 磁头完成读出或写入所需时间

影响因素: ①寻道时间: 盘大小、控制电路

②旋转时间: 旋转速度、控制电路

③数据传输时间: 旋转速度、端口协议



6. (1) $6 \times 240 \times 12 \text{ KB} = 17280 \text{ KB} = 16.875 \text{ MB}$

(2) **数据传输速率 (没给条件只考虑旋转)**

$$5400 \div 60 = 90 \text{ r/s}$$

$$90 \times 12 \text{ KB} = 1080 \text{ KB/s}$$
 考虑只读1个盘面?

$$(3) 1 \div 90 = 0.01111 \text{ s}$$

$$\text{平均为半圈 } 0.01111 \div 2 = 5.56 \text{ ms}$$

7. 磁盘控制电路可以先缓存一段时间内的磁盘 I/O 请求, 并通过算法使得磁盘访问的地址尽可能接近(重新改变访问次序), 从而达到最短的磁头移动距离, 减小寻道时间和磁盘旋转时间, 从而减少访问用时

8. RAID4 以若干数据块为一个单位, 依次分散在物理磁盘中, 并用额外的一个物理磁盘来存放奇偶校验码。

在应对一系列地址分散的小体积数据读取时, 每个磁盘可独立响应请求, 并行度高, 读取速度快

在面对大块数据读取时, 几个物理磁盘共同工作, 大大提升速度

9. 磁盘 I/O 请求减少 λ 下降

对于 μ 增大到 μ'
其提升为 $\frac{W_{\text{提升前}}}{W_{\text{提升后}}} = \frac{\frac{1}{\mu-\lambda}}{\frac{1}{\mu'-\lambda}} = \frac{\mu'-\lambda}{\mu-\lambda}$ 越大, 提升越多

$$\text{原式} = \frac{\mu-\lambda+\mu'-\mu}{\mu-\lambda} = 1 + \frac{\mu'-\mu}{\mu-\lambda}$$

λ 下降 分母增大 结果减小 提升幅度下降

以下为数学角度分析, 下为工程上分析

磁盘 I/O 请求减少, 其队列长度下降, 等待时间减小

提升性能, 即 μ 增大, 对单个任务来看, 其执行减少时间与队列无关, 为定值

而等待时间长短与队列长度有关, 但当队列长度本身很短时, 其提升也十分有限

故 I/O 请求减少, 性能提升幅度下降

10.6) 会争抢内存带宽资源

(1) 存储器层次设计合理可有效减小此类情况影响

可以使DMA与处理器错峰访问,可使两者在不相冲突情况下访问不同单元

1. 轮询机制: 赋予每个主设备相同优先级, 需要仲裁时, 算法按照轮询的方式依次赋予主设备总线的使用权

优点: 所有设备公平使用总线, 避免“饥饿”现象

缺点: 某些重要操作无法体现其优先性, 等待时间较长

场景: 各个主设备对总线访问需求比较相近

优先级仲裁机制: 赋予每个主设备不同的优先级, 优先级高的主设备在总线仲裁中更容易胜出

优点: 实现简单, 保障重要操作优先执行

缺点: 可能存在操作因优先级过低长期无法执行, 影响后续操作

场景: 存在设备经常访问总线、实时性要求不高

2.

场景

特点

APB: 低速低功率外设, 无复杂传输事务功能, 非流水线操作, 功耗低, 易使用

AHB: 高性能系统模块, 高时钟频率, 可进行总线仲裁, 突发传输, 分离传输, 流水操作

AXI: 高性能系统模块, 分离发送各类信号, 支持读写并行, 乱序, 非对齐操作

ACE: 多核系统, 加入缓存一致性的支持, 多个主设备可共享数据

CHI: 大规模多处理器系统, 更复杂的一致性和缓存管理方案

3. 1) 地址, 读数据, 写数据, 握手信号

读操作响应可通过读数据通道一并返回, 不会互相干扰

2) 握手信号: VALID 和 READY

发送方通过 VALID 表明已在对应通道放置有效信息, 接收方通过 READY 信号表明已准备好接收信息

数据或地址只有在 VALID 和 READY 均为高时才能被接收方接收

只有接收方 READY 信号为高时, 发送方才能更新其 VALID 信号及相应数据或地址

约束原因: 为了实现流控制, 以确保接收方在准备好接收数据时才会接收到数据, 防止因接收方处理速度跟不上而导致数据丢失

3) 突发传输: 主设备发起一次事务请求后, 在事务处理阶段, 数据会按照主设备的要求源源不断地进行传输而无需进行新的事务请求

类型: 固定突发, 增量突发, 回环突发