

## 第5章习题

1. 串行总线和并行总线的数据传输方式不同。

串行：一次发送一位，优点是硬件复杂性低，设计简单，传输效率和抗干扰能力较好；缺点是只能连接两个设备，且串行结构的总线延迟更大一些。

并行：<sup>可能</sup>在通信速率较低时，可连接大量外设，且通过中央仲裁机制，可以方便地实现总线设备间通信；缺点是当速率和带宽不断增加时，并行结构的压力增大，设计复杂性增加。

接口速率：

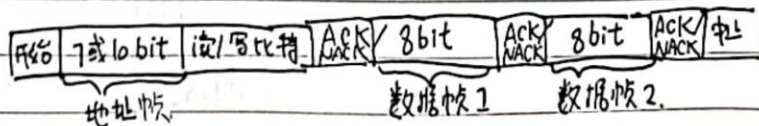
差异主要源于其数据传输方式。并行总线可一次发送多个数据位，但随着时钟频率越高，数据相互干扰严重，有效数据传输速率低于理论；串行总线一次只发送一位，但在高频率下有效工作，还采用了高效编码和调制技术，使接口速率更高。

2. 一个数据包含有 10 个 bit， $\therefore$  每秒传输总 bit 数为  $960 \times 10 = 9600 \text{ bits/s}$

$\therefore$  波特率为 9600 bps

有效传输速率 =  $960 \times 7 = 6720 \text{ bps}$

3. ① 数据包组成：



起始条件：数据开始传输的一个特定信号

地址帧：包括设备地址和读/写位

应答位(ACK/NACK)：表示数据是否被接收设备接收

数据帧：实际传输的数据

停止条件：数据停止传输的一个特定信号。

② I2C是半双工的,在任何时刻,数据只能在一个方向传输(主→从或从→主),因为I2C属于同步通信,传输数据均使用一根线,无法实现全双工方式。

③ 起止条件:

起始:当SCL为高时,SDA由高到低。

终止:当SCL为高时,SDA由低到高。

4. ① MTTF: Mean Time To Failure, 平均无故障时间

$MTTF = N \text{ hours}$ , RAID0情况下:  $MTTF_{RAID0} = \frac{N}{4}$

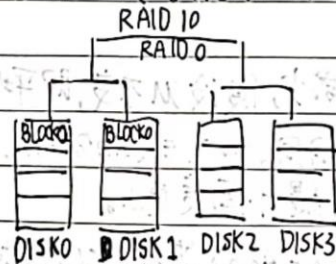
② 总容量必须大于 80G

① 使用 RAID1+RAID0 的情况,即 RAID10, 进行混合配置,此时

总容量为 100G > 80G。一对镜像同时故障的概率为  $(\frac{1}{N})^2$ ,

而且有 2 对镜像,故障概率为  $\frac{2}{N^2}$ , 故此时  $MTTF_{RAID10} = \frac{N^2}{2}$ ,

可以达到尽可能大的 MTTF。



设计的 RAID10 存储方案:

5.  $T = \text{寻道时间} + \text{旋转时间} + \text{数据传输时间}$

寻道时间: 磁头从当前位置移动到目标磁道并消除抖动所需时间

旋转时间: 磁头移动到目标磁道后目标扇区随盘片转动而经过磁头的时间

数据传输时间: 磁头完成读出或写入所需时间

寻道时间和旋转时间受控制电路精度和磁盘物理结构的影响,(例如磁头结构);数据传输时间受数据访问模式、数据传

输时读写速度和技术规模的影响。

Date: / /

6. ① 总容量  $= 6 \times 240 \times 12 \text{ KB} = 17280 \text{ KB} = 16.375 \text{ MB}$

② 每分钟有  $6 \times 5400 = 32400$  个磁道通过磁头, 传输

速率  $= 6 \times 5400 \times 12 \text{ KB/min} \approx 379.7 \text{ MB/min}$

$= 6480 \text{ KB/s}$

③ 平均延迟时间  $T = \frac{1}{2} \cdot \frac{60 \text{ s}}{5400} \approx 5.56 \times 10^{-3} \text{ s}$

7. 磁盘控制器主要是通过执行优化的调度算法解决这个问题, 例如最短寻道时间优先算法 (SSTF), 选择调度处理与当前磁道距离最近的磁头来缩短寻道时间, 或通过电梯算法等。总之, 应选取合适的调度算法来减少磁盘访问时间。

8. RAID4 进行了写入优化, 将当前写入磁盘的数据与旧数据对比, 来计算奇偶校验位改变, 这减少了读取所有磁盘的压力, 使 I/O 操作更加集中, 提升了读取速度。

9.  $W = \frac{1}{\mu - \lambda}$ , 当 I/O 请求减少时,  $\lambda$  会减少, 假设平均请求用时下降相同幅度, 则  $\mu$  改变量相同。但  $\lambda$  较小的系统  $W$  本身较小, 因此磁盘队列处理提升幅度更小。

例如, 若原先  $\mu = 50$ , 改进后  $\mu = 100$ , 对于  $\lambda = 40$  的情况, I/O 请求平均用时从  $100 \text{ ms}$  缩短为  $16.7 \text{ ms}$

若  $\lambda$  降低为  $30$ , 则原先用时  $= 50 \text{ ms}$ , 改进后缩短为  $14.3 \text{ ms}$ , 但提升幅度更低。

10. DMA 会与处理器争夺内存带宽资源, 因为 DMA 会控制 I/O 设备读写内存, 在这一过程中处理器可能需要等待内存的访问, 造成竞争。

优化存储层级可能会减少竞争对系统性能的影响, 例如如果处理器可有效通过多级缓存访问数据, 则可减少访问内存的次数, 从而减少竞争。

## 第6章习题

① 链式查询: 总线上所有部件共用一根总线请求, 当有部件请求时, 需经此线请求信号到总线控制器, 并查询总线是否忙碌。若不忙碌则立即发送响应信号, 依次查询, 直到某部件有总线请求则不再传下去, 直至每个部件。

优点: 优先级固定, 控制和结构简单

缺点: 优先级不能改变, 使低优先级低的部件长期不能使用总线。

应用场景: 适用于设备较少, 资源受限的系统。

② 计数器查询: 采用计数器控制总线使用权, 按预定顺序轮流各设备。

优点: 优先级灵活

缺点: 控制线多, 控制复杂

应用场景: 适用于设备数较多、对公平性要求高的系统。

③ 独立请求: 每个设备都有一个独立请求线和授权线, 可直接向仲裁器请求总线。

优点: 响应速度快, 响应优先级灵活

缺点: 硬件资源要求更高。

应用场景: 适用于设备较多、要求速度和设置优先级的系统。

2. APB: 简单的低带宽接口, 设计简单, 适用于低速, 低复杂度系统。

AHB: 高性能总线接口, 适用于需要高带宽的连接, 例如内存、DMA。

AXI: 高性能、高带宽的总线接口, 支持并行和流水线操作, 适用于与处理器等要求高性能和带宽的连接。

ACE: 在AXI基础上增加了对缓存一致性的支持, 适用于多核系统。

CHI: 高性能、可扩展, 适用于高性能多核系统和大规模集成设备。

## 3. ① AXI 独立事务通道:

读地址、读数据、写地址、写数据、写响应。

在 AXI 协议中, 读响应实际上通过读数据通道返回, 不单独设置, 可简化协议复杂度, 并保持数据和相应状态信息的关联。

② 握手信号时序依赖: 对读操作, 发送读地址和接收读数据的操作时序上不能真颠倒; 对写操作, 发送写地址、发送写数据和接收写响应操作时序不可真颠倒。

这样的约束可以保证数据传输的正确性和一致性, 并简化协议和硬件复杂性。

③ AXI 突发传输: 在地址总线上进行一次地址传输后, 进行多次数据传输。

类型: FIXED burst, 地址在每次传输间保持不变。

INCR (increment) burst: 地址在每次传输间增加, 用于连续数据流传输。

WRAP burst: 地址在传输间增加, 但到达边界后绕回初始地址。