615

第五章

2. (1) $960\times(1+7+1+1)=9600$ Byte/s

波特率是9600波特.

(2) $960\times7=6720$ Byte/s

有效数据传输速率是6720位/秒.

4. (1) 单个磁盘运行故障会导致数据完全丢失，∴MTTF仍为N小时.

(2) 可以用RAID-5，其中一个磁盘用于奇偶校验位，总容量为50G×3=150G.

5. 寻道时间是指磁头从当前位置移动到目标磁道并消除抖动所需要的时间。

旋转时间是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(上)方所需的时间。

数据传输时间是磁头完成读出或写入所需用的时间。

影响上述时间因素有磁盘性能参数，数据布局，数据大小和传输量，磁盘负载和并发操作。

6. (1) 6×240×12KB = 16.875GB.MB

(2) 5400÷60×16.875GB/s = 1.48GB/s.

(3) 旋转周期 $=\frac{1}{5400\div60}=\frac{1}{90}$ s

平均旋转时间 $=\frac{1}{2}\times\frac{1}{90}=\frac{1}{180}\approx0.0056$ s.

9. $L=\frac{\lambda}{\mu-\lambda}$ 减少，$\frac{\lambda}{\mu-\lambda}$ 减少，若生长率λ增加，则μ-λ差增加，$\frac{1}{\mu-\lambda}=W$ 减少

若生长率λ减少，假设μ不变，μ-λ增加，$W=\frac{1}{\mu-\lambda}$ 减少.

∴性能提升幅度下降.

10. DMA设备可能与处理器争夺内存带宽资源，因为它可以直接与主内存进行数据传输。

存储器层次设计中的缓存、内存带宽和总线设计会有影响。缓存可以缓解DMA与处理器的内存带宽竞争；内存、总线带宽若较小，则会面临带宽限制。

第五章

1.

串行总线的优点包括：

① 简单性：串行总线只需要一根传输线路，相对较简单，易于实现和维护。

② 成本效益：由于只需要较少的传输线路和接口，串行总线在硬件成本上通常比较低。

③ 长距离传输：串行总线对于长距离传输有较好的抗干扰能力，能够稳定传输数据。

④ 可扩展性：串行总线可以通过添加更多的设备来扩展系统的功能。

串行总线的缺点包括：

① 传输速率较低：由于数据是按位序列传输的，所以串行总线的传输速率相对较低。

② 延迟较高：数据需要逐位传输，因此传输的延迟较高。

并行总线的优点包括：

① 传输速率高：并行总线可以同时传输多个位，因此传输速率相对较高。

② 延迟低：由于同时传输多个位，因此并行总线的传输延迟相对较低。

并行总线的缺点包括：

① 复杂性：并行总线需要多个传输线路和对应的接口，相对较复杂，不易实现和维护。

② 成本高昂：由于需要较多的传输线路和接口，因此并行总线在硬件成本上通常较高。

③ 抗干扰能力较差：并行总线对于长距离传输的抗干扰能力相对较差，易受到干扰影响。

造成串行总线和并行总线接口速率不同的原因主要有两个方面：

① 物理约束：并行总线需要同时传输多个位，因此需要相应数量的传输线路和接口。然而，随着线路数量的增加，信号受到干扰的可能性也增加，这限制了并行总线的传输速率。

② 技术限制：串行总线可以通过串行通信协议（如 RS232、USB、Ethernet 等）实现较高的传输速率，利用高速的串行通信技术可以在单条线路上传输多个位。相比之下，并行总线的传输速率受到硬件设计和接口电路的限制，无法像串行总线那样轻松实现高速传输。

3.

1）I2C 的数据包由一系列的数据位组成。一个完整的数据包包括一个起始位（Start bit）、7 位或 10 位的数据位、一个可选的应答位（Acknowledge bit）和一个停止位（Stop bit）。起始位（Start bit）标志着数据传输的开始。它是一个高至低的跳变边缘，将总线上的电平从高电平切换到低电平。数据位是实际的数据传输部分。对于 7 位的数据位，每个位都代表一个数据位；对于 10 位的数据位，前八位是数据位，最后两位是地址扩展位。应答位（Acknowledge bit）是由被接收设备发送给主设备的确认信号，用于确认数据的接收。如果被接收设备成功接收了数据，它将发送一个低电平的应答位；如果出现错误或设备不可用，它将发送一个高电平的非应答位。停止位（Stop bit）标志着数据传输的结束。它是一个低至高的跳变边缘，将总线上的电平从低电平切换到高电平。

2) I2C 是半双工的，这意味着数据的传输是双向的，但同一时间只能进行发送或接收操作。在 I2C 总线上，主设备和从设备共享同一条数据线（SDA）和时钟线（SCL）。

3）传输的起止条件是通过在总线上发送特定的电平跳变边缘来实现的。起始条件（Start condition）是在总线上发送一个高至低的跳变边缘，表示数据传输的开始。起始条件用于告知其他设备总线上将要开始传输数据。停止条件（Stop condition）是在总线上发送一个低至高的跳变边缘，表示数据传输的结束。停止条件用于通知其他设备总线上的数据传输已经完成。通过起始和停止条件的使用，I2C 总线上的设备可以进行数据的传输和通信。主设备在发送完起始条件后，可以发送数据给从设备并接收从设备的应答。然后，主设备可以发送停止条件，结束数据传输。

7.

磁盘控制电路通过决定请求的最优执行次序来减少磁盘访问用时。为了理解这个过程，我们可以考虑以下几个方面：

1) 请求排序：磁盘控制电路可以对接收到的磁盘请求进行排序，以确定最优的执行次序。一种常见的排序算法是最短寻道时间优先（Shortest Seek Time First，SSTF）。这种算法根据当前磁头的位置和请求的位置，选择离当前位置最近的磁道进行访问。这样可以最小化寻道时间，提高访问效率。

2) 请求调度：磁盘控制电路还可以使用请求调度算法来优化执行次序。其中，Elevator 算法（也称为扫描算法或电梯算法）是一种常见的调度算法。该算法将磁头的移动方向设定为一个方向（例如从外向内），并按照该方向顺序执行请求。一旦到达最外或最内磁道，磁头会改变移动方向。这种方式可以减少磁头的反复移动，提高磁盘访问效率。

3) 提前读取（Read-Ahead）：磁盘控制电路还可以通过提前读取来减少访问用时。当磁盘接收到一个读取请求时，它可以预先读取相邻的数据块并存储在缓存中。如果后续的请求需要这些数据块，它们可以直接从缓存中获取，而不需要再次访问磁盘。这样可以减少访问延迟，并提高数据访问的效率。

通过以上这些策略，磁盘控制电路可以优化请求的执行次序，减少寻道时间和磁头移动的次数，从而降低磁盘访问用时，并提高系统的整体性能。这些优化算法和策略可以根据具体的磁盘控制器和应用需求进行调整和优化。

8.

1) 写入性能提升：写入优化可以通过将数据并行写入多个磁盘来提高写入性能。由于数据条带被分布在多个磁盘上，每个磁盘只需要写入自己负责的数据条带，因此并行写入可以加快整体写入速度。这对于需要频繁写入数据的应用场景非常有益。

2) 读取速度影响：尽管写入优化提高了写入性能，但它对读取速度可能有一定的影响。在 RAID 4 中，读取操作需要通过读取多个磁盘上的数据条带并计算奇偶校验位来恢复原始数据。当并发的读取请求发生时，如果多个读取请求涉及同一数据条带所在的磁盘，那么这些读取请求需要等待奇偶校验位计算完成后才能获取数据。这可能会引起读取的延迟，并对读取速度产生一定的影响。

综上所述，写入优化可以提高 RAID 4 的写入性能，但在读取方面可能对速度产生一定的影响。这是因为在读取操作中，需要涉及奇偶校验位的计算和可能的等待。然而，对于某些应用场景来说，写入优化的性能提升可能超过读取速度的影响，使整体系统表现更出色。

第六章

1.

常见的总线仲裁机制有以下几种：

1) 集中式仲裁：集中式仲裁机制使用专门的仲裁器来管理总线访问冲突。在这种机制下，所有设备都必须向仲裁器发送请求，并且仲裁器根据优先级或其他规则来选择下一个获得总线访问权限的设备。优点是实现简单、成本低，适用于设备数量较少的系统。缺点是仲裁器成为了系统的瓶颈，可能引入延迟和吞吐量限制。
2) 分布式仲裁：分布式仲裁机制不依赖于中心化的仲裁器，而是由各个设备自行协调总线访问。常见的分布式仲裁机制包括冲突检测、令牌环和令牌传递等。冲突检测机制中，设备不断检测总线是否被占用，空闲时请求总线访问。令牌环机制中，一个特殊的令牌在总线上循环传递，拥有令牌的设备可以访问总线。令牌传递机制中，设备通过发送请

求消息来申请总线访问，并等待授予的访问权限。分布式仲裁机制具有较好的并行性能和灵活性，适用于设备数量较多且分散的系统。缺点是复杂度较高，需要设备之间的相互通信。

3) 基于时间的仲裁：基于时间的仲裁机制使用时间片或时分多路复用来分配总线访问权。每个设备被分配一个固定的时间段来访问总线，超时后必须释放总线。这种机制具有简单的实现和公平性，但在高负载情况下，可能导致总线资源利用率较低。

选择合适的仲裁机制取决于具体的系统需求和约束条件：

1) 集中式仲裁适用于设备数量较少、优先级规则较简单的系统，如嵌入式系统或低成本系统。
2) 分布式仲裁适用于设备数量较多、需要较好并行性和灵活性的系统，如大型计算机系统或分布式系统。
3) 基于时间的仲裁适用于对公平性较为关注、总线负载相对较轻的系统，如实时系统或轻负载的网络环境。

2.

AMBA（Advanced Microcontroller Bus Architecture）是一种由 ARM 公司提出的总线架构，它定义了一系列总线协议，包括 APB（Advanced Peripheral Bus）、AHB（Advanced High-performance Bus）、AXI（Advanced eXtensible Interface）、ACE（AXI Coherency Extensions）和 CHI（Coherent Hub Interface）。这些协议具有不同的特点和适用场景。

1) APB（Advanced Peripheral Bus）：APB 是一种低功耗、低带宽的总线协议，主要用于连接外设和低复杂性的控制器。它的特点是简单、易实现，适用于功耗敏感和带宽要求较低的系统。
2) AHB（Advanced High-performance Bus）：AHB 是一种高性能的总线协议，具有分段式的地址映射和优先级控制机制。它适用于连接多个高性能 IP 核和存储器等设备，支持多主机、多从设备的并行访问，具有良好的吞吐量和实时性能。
3) AXI（Advanced eXtensible Interface）：AXI 是一种高性能、高扩展性的总线协议，设计用于连接高性能 IP 核、处理器和存储器等设备。它具有多通道、乱序传输和流水线传输等特性，可以提供更高的数据吞吐量和更低的延迟。
4) ACE（AXI Coherency Extensions）：ACE 是在 AXI 基础上增加的一组协议扩展，用于提供缓存一致性支持。它支持多处理器系统中的高速缓存一致性协议，确保多个处理器的缓存内容始终保持一致。
5) CHI（Coherent Hub Interface）：CHI 是一种高性能、高带宽的总线协议，用于连接具有高速缓存一致性的多处理器系统。它提供了更高的带宽、更低的延迟和更好的可扩展性，适用于大型多处理器系统。

根据系统的性能要求、功耗需求和缓存一致性需求等因素，可以选择合适的 AMBA 总线协议。例如，对于低功耗和低复杂性的系统，可以选择 APB；对于高性能和实时性要求较高的系统，可以选择 AHB 或 AXI；对于多处理器系统中需要缓存一致性的系统，可以选择 ACE 或 CHI。选择适合的总线协议有助于提高系统的性能、效率和可靠性。

3.

1)

AXI 总线包含以下独立的事务通道：

① 读（Read）通道：用于从主设备（如处理器）读取数据。读事务的请求和读数据传输分别通过读地址通道（AR）和读数据通道（R）进行。

② 写（Write）通道：用于向主设备写入数据。写事务的请求和写数据传输分别通过写地址通道（AW）和写数据通道（W）进行。

③ 写响应（Write Response）通道：用于主设备接收从从设备返回的写响应。这是唯一的一个单独的响应通道，用于所有写事务的响应。

协议没有设置独立的读响应通道的原因是，读事务的响应可以通过读数据通道（R）传输。当从设备准备好数据时，它可以直接通过读数据通道返回给主设备，省去了额外的响应通道。

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：读事务的响应必须与对应的读事务请求匹配，即读响应必须在读请求之后被发送；写事务的响应必须与对应的写事务请求匹配，即写响应必须在写请求之后被发送。

这样的时序约束是为了确保读和写事务的正确性和一致性。读事务必须在写事务之后进行，以避免读取到过期或无效的数据。同时，读事务的响应必须与对应的请求匹配，以确保读取的数据能够正确地被接收和处理。

3)

AXI 的突发传输是指在一次事务中连续传输多个数据或地址。突发传输可以提高总线带宽的利用率。AXI 总线支持以下几种突发传输类型：

① 固定突发（Fixed Burst）：在一次事务中连续传输固定数量的数据或地址。

② 递增突发（Incrementing Burst）：在一次事务中连续传输递增的地址，每个传输的数据位宽可以不同。

③ 块突发（Wrap Burst）：在一次事务中连续传输固定数量的数据或地址，并在到达末尾后循环到起始位置继续传输。