



满分：100分

限时：10分钟 完成

答题纸上要写上：班级， 姓名， 学号 （不用抄题）

1. 名词解释

(1) 指令延迟槽：

2. 现有16个处理器，编号分别为0, 1, ..., 15, 用一个N=16的互连网络互连。处理器i的输出通道连接互连网络的输入端i, 处理器i的输入通道连接互连网络的输出端i。当该互连网络实现的互连函数分别为：

$$(1) \text{Cube}_2 \quad 0011$$

写出3号处理器连接的其它处理器的编号

3. Alpha AXP机器中，假设Cache不命中开销为30个时钟周期，当不考虑存储器停顿时，所有指令的执行时间都是2.2个时钟周期，访问Cache不命中率为3%，平均每条指令访存1.35次。试分析有Cache与无cache对系统性能的影响。要给出计算过程。

解(1)

1. 指令延迟槽

解释：指令延迟槽（Instruction Delay Slot）是指在流水线处理器中，当一条指令被执行时，紧接在其后的一条指令被称为延迟槽指令。延迟槽的引入是为了优化流水线操作，减少由于跳转指令或其他控制转移指令引起的性能损失。跳转指令的结果通常会影响到后续指令的执行，但处理器可以在延迟槽中安排其他不依赖于跳转指令结果的指令，从而更有效地利用处理器的资源，避免因流水线停顿而产生的空闲周期。

$$12) \text{Cube}_2(x_3x_2x_1x_0) = x_3\bar{x}_2x_1x_0$$

$$\text{Cube}_2(0011) = 0111$$

$$13) \text{Cube}_2(x_3x_2x_1x_0) = 0011$$

$$\text{if } x_3x_2x_1x_0 = 0111$$

与7号双向互连

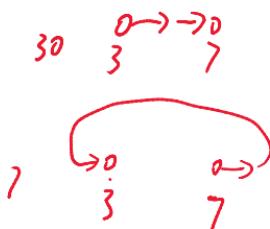
解：(1) 由 $\text{Cube}_3(x_3x_2x_1x_0) = \bar{x}_3x_2x_1x_0$ ，

相连

得 $\text{Cube}_3(1101) = 0101$, 即处理器13连接到处理器5。
13号与5号相连。

令 $\text{Cube}_3(x_3x_2x_1x_0) = 1101$, 得 $x_3x_2x_1x_0 = 0101$, 故与处理器13相连的是处理器5。

所以处理器13与处理器5双向互连。





3. Alpha AXP机器中，假设Cache不命中开销为30个时钟周期，当不考虑存储器停顿时，所有指令的执行时间都是2.2个时钟周期，访问Cache不命中率为3%，平均每条指令访存1.35次。试分析有Cache与无Cache对系统性能的影响。要给出计算过程。

3. Alpha AXP机器中，假设Cache不命中开销为30个时钟周期，当不考虑存储器停顿时，所有指令的执行时间都是2.2个时钟周期，访问Cache不命中率为3%，平均每条指令访存1.35次。试分析有Cache与无cache对系统性能的影响。要给出计算过程。

解：

CPU时间_{有 cache}

$$IC \times (2.2 + 1.35 \times 3\% \times 30) \times \text{~}$$

$$1.2K + 2.2 = 3.4K$$

实际CPI：3.4K

$$3.4K / 2.2 \approx 1.55(27)$$

若没有 cache

$$CPI = 2.2 + 1.35 \times 100\% \times 30$$

$$42.7$$

CPI_{无 cache} = CPI execution + 每条指令的平均访存次数 × 不命中率 × 不命中开销

考虑Cache的不命中后，有Cache时的性能为：

$$\text{CPU时间}_{\text{有 cache}} = IC \times (2.0 + 1.33 \times 2 \% \times 50) \times \text{时钟周期时间}$$

$$= IC \times 3.33 \times \text{时钟周期时间}$$

实际CPI：3.33

$$3.33 / 2.0 = 1.67(\text{倍})$$

CPU时间也增加为原来的1.67倍。

若没有Cache，则：

$$CPI = 2.0 + 1.33 \times 100 \% \times 50 = 2.0 + 1.33 \times 50 = 68.5$$

CPU时间_{无 cache} = IC × 68.5 × 时钟周期时间

CPU时间_{无 cache} > CPU时间_{有 cache}

CPI_{无 cache} = CPI execution + 每条指令的平均访存次数 × 不命中率 × 不命中开销

$$\frac{42.7}{3.4K} \approx 12.50366$$