第8章 CPU 的结构和功能 习题与解答

> 说明：课件 ch08（Slide 413–465）无独立编号的例题/习题页，原题以课件中嵌入的**思考题/设问**形式出现（p.431、p.447、p.464），已全部收录并给出完整解答；其余为按考点补充题。共 15 题（原题 3 + 补充 12）。

一、课件原题

题1（课件 p.431，思考题）

影响指令流水效率加倍的因素有哪些？相应的解决办法是什么？

解答： 因素（课件 p.431）： 1. **执行时间 > 取指时间**：二级流水中取指段与执行段不等长，无法完全重叠，效率达不到加倍。 2. **条件转移指令**：必须等上条指令执行结束才能确定下条指令的地址，造成时间损失（断流）。

解决办法： - 针对因素 1：把执行阶段进一步**细分段**，使各段时间尽量一致（发展为六级流水，课件 p.432；分段原则"每段操作时间尽量一致"见 p.445）。 - 针对因素 2：设置**指令部件缓冲区**（指令预取缓冲）+采用**猜测法**（课件 p.431），提前预取/猜测转移方向以减少断流损失。

评分要点：两因素各 1 分；缓冲区+猜测法 1 分；细分段/等长 1 分。

题2（课件 p.447，简答/设问）

中断系统需要解决哪些问题？靠什么手段解决？

解答（课件 p.447，共 7 问）： 1. 各中断源**如何**向 CPU 提出请求？（中断请求标记 INTR） 2. 各中断源**同时**提出请求怎么办？（中断判优：排队器/程序查询） 3. CPU **什么条件、什么时间、以什么方式**响应中断？（EINT=1；执行周期结束；隐指令） 4. **如何保护现场**？（断点由隐指令、寄存器由 PUSH） 5. **如何寻找入口地址**？（硬件向量法/软件查询法） 6. **如何恢复现场、如何返回**？（POP、IRET） 7. 处理中断的过程中**又出现新的中断**怎么办？（多重中断+屏蔽技术）

手段：**硬件＋软件**共同解决（课件 p.447）。 评分要点：7 问每问 0.5~1 分；"硬件+软件"1 分。

题3（课件 p.464，思考题）

多重中断时若把程序断点存入"0"地址，三次中断的三个断点都写入 0 地址会相互覆盖。如何保证断点不丢失？

解答： 问题根源（课件 p.464）：0 地址是固定单元，每层中断都把断点写进去，后一次覆盖前一次→返回时找不到正确断点。 保证不丢失的办法（课件 p.465）： 1. **改用断点进栈**：每层断点压入栈中不同单元，天然不覆盖（课件 p.464 方式(1)）。 2. **仍用 0 地址时，先转存再开中断**：服务程序入口处先把 0 地址内容转存到专用单元

RETURN（`LDA 0` / `STA RETURN`），腾出 0 地址供下一层中断使用；然后才 `ENI` 开中断允许嵌套；返回时用间址 `JMP @ RETURN` 跳回本层断点。配合"置屏蔽字"控制哪些源可嵌套。 关键顺序：**保护现场 → 转存 0 地址断点 → 置屏蔽字 → 开中断 → 服务 → … → 恢复现场 → JMP @ RETURN**（课件 p.465）。 评分要点：指出覆盖原因 1 分；进栈法 1 分；0 地址法"先转存后开中断"+间址返回 2 分。

二、考点补充题

题4【补充•考点：寄存器分类】（选择，基础）

下列寄存器中，用户（程序员）不可见的是（ ）。 A. PC　B. 通用寄存器　C. MAR、MDR、IR　D. 条件码寄存器

解答：选 **C**。 逐项排除：A 错，PC 虽属控制寄存器但**用户可见**（课件 p.418）；B 错，通用寄存器是典型用户可见寄存器（p.417）；D 错，条件码寄存器用户可见、作分支依据（p.417）；C 对，MAR/MDR/IR 为访存/取指接口，对程序员透明、用户不可见（p.418）。

题5【补充•考点：指令周期/机器周期/时钟周期】（填空，基础）

指令周期由若干_____周期组成，后者又由若干_____周期组成；CPU 的四个工作周期是_____、、、，**其中任何指令都必须有的是_____和**。

解答：机器（工作）；时钟（节拍）；取指周期、间址周期、执行周期、中断周期；取指周期、执行周期。 依据：课件 p.420–424 给出四种工作周期；"指令周期⊃机器周期⊃时钟周期"的层级为补充常识（课件用"工作周期"即机器周期）。注意 NOP 无执行周期（p.421），故"必须有"指一般指令；间址仅间接寻址指令有、中断仅响应中断时插入（p.422）。

题6【补充•考点：取指周期数据流】（简答/默写，基础）

写出取指周期的数据流（微操作序列），并说明 PC 何时加 1。

解答（课件 p.425）： 1. (PC)→MAR，送地址总线； 2. CU 经控制总线发**读**命令； 3. 存储器将该单元内容经数据总线→MDR，即 M(MAR)→MDR； 4. (MDR)→IR； 5. (PC)+1→PC。 PC 加 1 发生在**取指周期内**（第 5 步），由 CU 控制，使 PC 指向下一条指令；不是在执行周期。 评分要点：五步各 1 分；PC+1 时机 1 分。

题7【补充•考点：中断隐指令】（简答，基础）

什么是中断隐指令？它完成哪三件事？"保护断点"和"保护现场"分别由谁完成？

解答： 中断隐指令（课件 p.454）：中断周期内由**硬件自动完成**、指令系统中没有对应机器指令的一组操作。三件事： 1. **保护程序断点**：断点存 0 号地址或进栈； 2. **寻找服务程序入口地址**：硬件向量法→向量地址→PC；软件查询法→识别程序入口 M→PC； 3. **硬件关中断**：置 EINT=0。 主体区分：**保护断点**由中断隐指令（硬件）完成；**保**

护现场（寄存器内容）由中断服务程序用 PUSH（软件）完成（课件 p.454、455）。 评分要点：定义 1 分；三件事各 1 分；主体区分 1 分。

题8【补充•考点：流水线数据相关】（识别/选择，综合）

判断下列指令对的相关类型（RAW/WAR/WAW），并说明可用什么技术缓解。 (1) SUB R1,R2,R3；ADD R4,R5,R1 (2) STA M,R2；ADD R2,R4,R5 (3) MUL R3,R2,R1；SUB R3,R4,R5

解答（语义与课件 p.434 一致：SUB Rd,Ra,Rb 表示 (Ra)−(Rb)→Rd；ADD 同理；STA M,Rs 表示 (Rs)→M；MUL Rd,Ra,Rb 表示 (Ra)×(Rb)→Rd）： (1) 前指令**写 R1**，后指令**读 R1→RAW（写后读）**。 (2) 前指令**读 R2**，后指令**写 R2→WAR（读后写）**。 (3) 前指令**写 R3**，后指令**写 R3→WAW（写后写）**。 缓解技术：**后推法**（推迟相关指令）或**旁路技术**（forwarding，把前指令结果直接旁路给后指令）（课件 p.434）。其中旁路对 RAW 最有效。 评分要点：每型 1 分；技术 1 分。

题9【补充•考点：流水线性能计算】（计算，基础）

某六级流水线各段时间均为 Δt=100 ns，连续执行 9 条指令。求：总时间 T、串行时间 T′、加速比 Sp、效率 E、实际吞吐率 Tp 与最大吞吐率 Tpmax。

解答（公式见课件 p.437–440；m=6、n=9）： 1. $T=(m+n-1)\cdot\Delta t=(6+9-1)\times100=14\times100=$**1400 ns**（与课件 p.432 的 14 个时间单位一致）。 2. $T'=n\cdot m\cdot\Delta t=9\times6\times100=54\times100=$**5400 ns**（与课件 54 一致）。 3. $Sp=T'/T=5400/1400=54/14=27/7\approx$**3.86**。 4. $E=n/(m+n-1)=9/14\approx$**0.643（64.3%）**。 5. $Tp=n/T=9/1400$ ns≈**6.43×10^6 条/秒**。 6. $Tpmax=1/\Delta t=1/100$ ns=**1×10^7 条/秒**。 自检：$E=Sp/m=(27/7)/6=9/14$ ✓；$Tp=E/\Delta t=0.643/100$ ns≈6.43×10^6 ✓。 评分要点：每问 1 分；自检关系 1 分。

题10【补充•考点：流水线多发技术】（简答/对比，基础）

比较超标量、超流水线、超长指令字（VLIW）三种多发技术的核心思想、能否调整指令执行顺序、各靠什么实现优化。

解答（课件 p.441–443）：

技术	核心思想	能否调序	优化依赖
超标量	每时钟周期并发多条独立指令；配置多个功能部件	不能	编译优化搭配可并行指令
超流水线	一个时钟周期内再分段（3 段）、一个功能部件使用多次（3 次）；速度 3 倍	不能	编译程序优化
VLIW	编译挖掘指令间潜在并行性，把多条能并行的指令组合成一条含多个操作码字段的超长指令字（可达几百位）；多个处理部件	由编译静态组合	编译程序

要点：超标量=空间并行（多部件）；超流水线=时间细分（部件复用）；VLIW=编译打包。三者均依赖编译优化；超标量/超流水线不能调整指令执行顺序。 评分要点：每行核心 1 分；调序结论 1 分；编译依赖 1 分。

题11【补充•考点：中断响应条件与时间】（简答，基础）

CPU 响应中断的条件和时间是什么？为什么响应后要硬件关中断？

解答： 条件（课件 p.453）：**允许中断触发器 EINT=1**（且存在中断请求）。 时间（课件 p.453）：**指令执行周期结束时刻**由 CPU 发查询信号（保证当前指令完整执行完）。 硬件关中断原因（课件 p.454）：中断隐指令要保护断点、寻找入口，这些操作若被新的中断打断会破坏断点/入口的正确性，故用硬件置 EINT=0 暂时屏蔽，待服务程序保护现场后再由软件开中断。 评分要点：条件 1 分；时间 1 分；关中断原因 2 分。

题12【补充•考点：中断屏蔽字与处理优先级】（综合设计）

四中断源 A、B、C、D 的响应优先级为 A→B→C→D（降序，硬件定死）。现希望**处理优先级**改为 A→C→B→D。设屏蔽字位序为 A B C D、1=屏蔽、0=开放，写出各源的新屏蔽字，并说明响应优先级是否改变。

解答： 原则：某源运行时，屏蔽字中对其**自身**和**处理优先级低于它**的源置 1，对**高于它**的源置 0。 处理优先级秩：A=1、C=2、B=3、D=4。 - A（最高）：屏蔽自身+所有更低（C,B,D）→**1 1 1 1**。 - C（第 2）：高于它的 A 置 0；自身 C 及更低 B,D 置 1→**0 1 1 1**。 - B（第 3）：高于它的 A,C 置 0；自身 B 及更低 D 置 1→**0 1 0 1**。 - D（最低）：高于它的 A,C,B 置 0；自身 D 置 1→**0 0 0 1**。

中断源	新屏蔽字
A	1 1 1 1
B	0 1 0 1
C	0 1 1 1
D	0 0 0 1

验证：A 运行屏蔽所有→不被打断（最高）；C 运行只允许 A 打断→第 2；B 运行允许 A,C 打断→第 3；D 运行允许 A,C,B 打断→最低。故处理优先级 A→C→B→D ✓。 **响应优先级不变**（仍 A→B→C→D），因它由硬件排队器决定、不可改变；屏蔽字只改处理优先级（课件 p.460）。 评分要点：每源屏蔽字 1 分；验证 1 分；响应优先级不变 1 分。

题13【补充•考点：0 地址断点保护程序】（综合分析）

阅读课件 p.465 的 0 地址断点保护程序，说明 `STA SAVE`、`LDA 0 / STA RETURN`、`ENI`、`JMP @ RETURN` 各条指令的作用，以及为什么 `LDA 0 / STA RETURN` 必须在 `ENI` 之前。

解答（课件 p.465）： - `STA SAVE`：保护现场，把 ACC 内容存到 SAVE 单元（服务程序要用 ACC）。 - `LDA 0 / STA RETURN`：把 0 地址中的**程序断点**转存到 RETURN 单元，腾出 0 地址供下一层中断写断点。 - `ENI`：开中断，允许更高优先级中断嵌套（多重中断）。 - `JMP @ RETURN`：**间址返回**，RETURN 单元里存的是本层断点地址，间接跳回断点继续主程序/上层服务。 为什么转存必须在 ENI 之前：若先开中断，可能在转存完成前就发生新一层中断，新断点写入 0 地址**覆盖**本层尚未转存的断点→本层返回地址丢失。故必须"先转存断点、后开中断"。 评分要点：四条指令作用各 1 分；顺序原因 2 分。

题14【补充•考点：指令周期流程与数据流综合】（综合）

某指令采用间接寻址，且执行结束后 CPU 响应了一个中断。列出该指令经历的各工作周期顺序，并写出间址周期与中断周期的数据流。

解答： 周期顺序（课件 p.422、423）：**取指周期 → 间址周期 → 执行周期 → 中断周期**（然后回取指周期取服务程序首指令）。 间址周期数据流（课件 p.426，"取地址"）： 1. Ad(IR)→MAR（形式地址送 MAR）； 2. CU 发读命令； 3. M(MAR)→MDR（经数据总线）； 4. (MDR)→Ad(IR)（有效地址写回 IR 地址字段）。 中断周期数据流（课件 p.427，"存断点"）： 1. CU 形成断点存放地址→MAR（0 地址或栈指针）； 2. (PC)→MDR； 3. CU 发写命令，(MDR)→M(MAR)（保存断点）； 4. 向量地址（或识别程序入口 M）→PC。 评分要点：周期顺序 2 分；间址四步 2 分；中断四步 2 分；读/写性质正确 1 分。

题15【补充•考点：分支断流下的实际吞吐率】（计算，拔高）

六级流水线（m=6）各段 Δt=100 ns。连续执行 n=100 条指令，其中比例 p=20% 为转移指令，每条转移引起 k=3 个时钟周期的断流损失（转移损失，课件 p.436）。求实际总时间 T 与实际吞吐率 Tp，并与无断流理想值比较。

解答： 1. 无断流基础时间 $T0=(m+n-1)\cdot\Delta t=(6+100-1)\times100=105\times100=10500$ ns。 2. 断流附加时间 = 转移条数×每条损失×Δt = $(n\cdot p)\cdot k\cdot\Delta t = (100\times0.2)\times3\times100 = 20\times3\times100 = 6000$ ns。 3. 实际总时间 T = T0 + 6000 = 10500+6000 = **16500 ns**（即 $(m+n-1+n\cdot p\cdot k)\cdot\Delta t = (105+60)\times100$）。 4. 实际吞吐率 Tp = n/T = 100/16500 ns ≈ **6.06×10^6 条/秒**。 5. 理想（无断流）Tp0 = n/T0 = 100/10500 ns ≈ **9.52×10^6 条/秒**。 比较：断流使吞吐率由 9.52×10^6 降到 6.06×10^6（约降 36%），说明控制相关（转移）是流水线效率的重要损失源（课件 p.435、436）。 通用公式：$Tp = n / [(m+n-1+n\cdot p\cdot k)\cdot\Delta t]$。 评分要点：T0 1 分；附加损失 2 分；T 1 分；Tp 1 分；比较/公式 1 分。