

武汉大学计算机学院

2024-2025 学年第一学期 2023 级《计算机组成与体系结构 A》

期末考试试题 A 卷

注：本电子版试卷为利用 LLM 提取照片版试卷中的题目文本后精调排版制作，虽然已经过人工校对，但仍可能存在模型幻觉和人工因素导致的疏漏，仅供复习参考。

一、单项选择题（每小题 2 分，共 14 分）

1. 不改变处理器型号，只增加处理器数量，则有（ ）。
A. 提高吞吐率
B. 减少响应时间
C. A 和 B
D. 没有影响
2. （ ）能被 GPU 和所有线程块共享。
A. 寄存器
B. 私有存储器
C. 局部存储器
D. GPU 存储器
3. 下列给出的指令系统特点中，有利于实现指令流水线的是（ ）。
I. 指令格式规整且长度一致
II. 算数逻辑指令的数据来自于寄存器
III. 只有 Load/Store 指令才能对操作数进行存储访问
A. 仅 I、II
B. 仅 II、III
C. 仅 I、III
D. I、II、III
4. 在 RISC-V 指令执行过程中，若某转移指令所在主存地址为 0x2000，相对位移量字段的内容为 0x08，则该转移指令成功转移后的目标地址是（ ）。
A. 0x2008
B. 0x200C
C. 0x2010
D. 0x2014
5. 假定有 4 个整数用 8 位补码分别表示为 $r1=0xF9$ ， $r2=0xFB$ ， $r3=0x91$ ， $r4=0xEE$ 。若将运算结果存放在一个 8 位寄存器中，则下列运算中会发生溢出的是（ ）。
A. $r1 \times r2$
B. $r2 \times r4$
C. $r1 \times r4$
D. $r3 \times r4$
6. 下列（ ）功能部件是每个指令都需要用到的。
A. PC、I-Mem
B. I-Mem、D-Mem
C. D-Mem、ALU
D. ALU、ImmGen
7. 指令 `addi x10, x10, -11` 在 ID 级会读的寄存器是：（ ）。
A. 10 号寄存器
B. 10 号寄存器和 11 号寄存器
C. 11 号寄存器
D. 10 号寄存器和 21 号寄存器

二、性能计算（共 15 分）

1. （8 分）现有系统在 cache（I-cache 和 D-cache）全命中的情况下 CPI 为 1，且指令中只有 load 和 store 指令可以访存，它们在程序中占比为 50%。若 cache 缺失率为 2%，缺失代价为 200 个时钟周期，那么，与全命中的情况下性能相比，系统的性能慢多少？
2. （7 分）假设有一个任务由四个步骤 S1-S4 组成，执行时间分别占总时间的 11%、18%、23%和 48%；除了 S1 不能进行优化外，其他三个步骤都可以优化：S2 可以加速 5 倍，S3 可以加速 20 倍，S4 可以加速 1.6 倍。
 - （1）计算优化后执行任务的总体加速比；（4 分）
 - （2）请对形成这个总体加速比的原因进行简要说明。（3 分）

三、指令系统（共 10 分）

假设需要为一个嵌入式应用系统开发新芯片，并希望创建新的 ISA。我们受够了不同的 RISC-V 指令类型，决定只包括一个通用类型-X 型指令。

假设我们只包括以下指令：

0. add rd1, rs1, rs2
1. and rd1, rs1, rs2
2. lw rd1, offset1 (rs1)
3. sw rs2, offset1 (rs1)
4. addi rd1, rs1, imm1
5. beq rs1, rs2, imm1
6. lui rd1, imm1
7. jal rd1, imm
8. stw rs3, imm1, imm2 (rs1)

stw 指令是将寄存器 rs3 的内容存入地址 $rs1 + offset1$ 和 $rs1 + offset2$ ，含义如下：

$R[rs3] \rightarrow Mem(R[rs1] + offset1)$ AND $R[rs3] \rightarrow Mem(R[rs1] + offset2)$

(1) 假设取消 RISC-V 指令中的 funct3 和 funct7 字段，只使用操作码。如果只为了支持上面列出的指令，操作码字段的最小位数是多少？（2 分）

(2) 希望 jal 指令跳转范围达到 -64KiB - 64KiB，请问立即数字段至少是多少位？

假设 jal 指令可以跳转到任意字节地址。（3 分）

(3) 假设指令格式如下：

imm2	imm1	rs3	rs2	rs1	rd2	rd1	opcode
------	------	-----	-----	-----	-----	-----	--------

指令长度是 64 位，imm1 和 imm2 分别都是 11 位，opcode 是 7 位，请问该处理器中最多有多少个寄存器？（2 分）

(4) 将 (3) 中指令长度缩短成 32 位，每个立即数字段 4 位。每个寄存器 4 位。将指令 stw x8, 0, 4(x5) 转换成机器码，操作码从 0 开始顺序编码，用二进制表示。如果未使用字段，请用“x”填充字段。（3 分）

四、数据表示与运算（共 10 分）

已知：

$$f(n) = \sum_{i=0}^n 2^i = 2^{n+1} - 1 = \underbrace{11\dots 1}_n 2$$

计算 $f(n)$ 的 C 语言函数 f1 如下：

```
int f1(unsigned n) {
    int sum=1, power=1;
    for (unsigned i=0; i<=n-1; i++) {
        power*=2;
        sum+=power;
    }
    return sum;
}
```

将 f1 中的 int 都改为 float，可得到计算 $f(n)$ 的另一个函数 f2。假设：

unsigned 和 int 型数据都占 32 位；

float 采用 IEEE754 单精度标准（符号位：1，指数位：8，小数位：23）。

请回答下列问题：

- （1）当 $n=0$ 时，f1 是否会出现死循环？若将 f1 中的变量 i 和 n 都定义为 int 型，则 f1 是否还会出现死循环？请分别解释原因。（2 分）
- （2）f1(24) 和 f2(24) 的返回值是否相等？请分别计算值并回答。（4 分）
- （3）机器数为 0x7F800000 对应的浮点数值是什么？若使 f2(n) 的结果不溢出，则最大的 n 是多少？（4 分）

五、单周期 CPU（共 10 分）

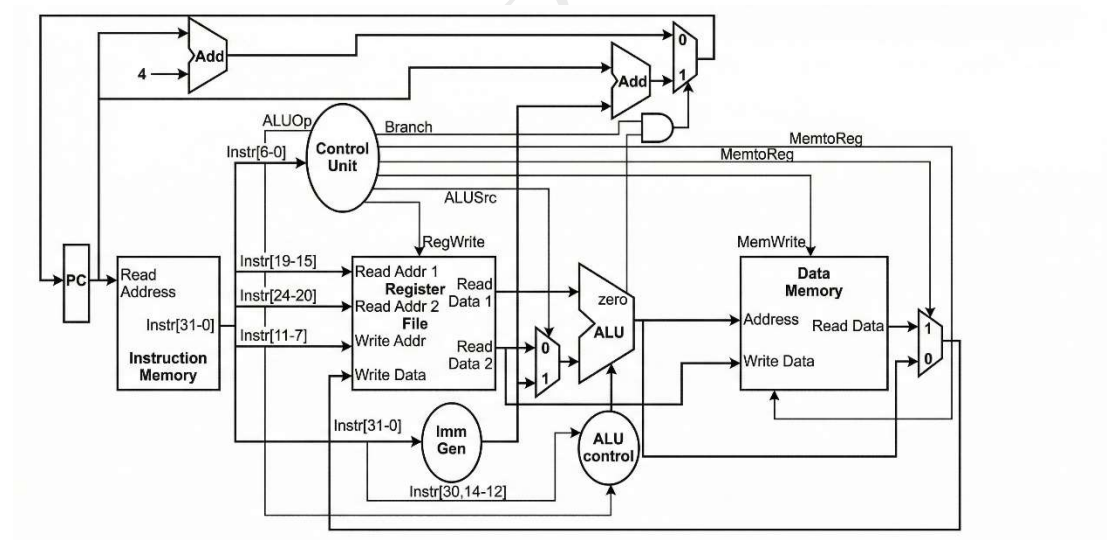
观察 I-MEM 中的如下指令，回答下列问题：

```

addi x5, x0, 100    // 指令地址 0x00000010
addi x6, x0, 1000   // 指令地址 0x00000014
addi x19, x0, 0     // 指令地址 0x00000018
addi x22, x0, 10    // 指令地址 0x0000001C
FOR: bge x19, x22, EXIT // 指令地址 0x00000020
lw x7, 0(x5)        // 指令地址 0x00000024
sw x7, 0(x6)        // 指令地址 0x00000028
addi x5, x5, 4       // 指令地址 0x0000002C
addi x6, x6, 4       // 指令地址 0x00000030
addi x19, x19, 1     // 指令地址 0x00000034
jal x0, FOR          // 指令地址 0x00000038
.....              // .....
EXIT: .....         // 指令地址 0x00001040
  
```

- (1) 该段代码能否正确执行？为什么？（2 分）
- (2) 如果修改数据通路中的功能部件可以使该段代码正确执行，最简单的修改方案是什么？为什么可以这样修改？（4 分）
- (3) 如果通过重构 SB 型指令，可以实现分支范围扩大到 2^{32} 字节寻址，则需要修改和增加哪类功能部件和控制信号？请画出新 SB 指令的数据通路图。（4 分）

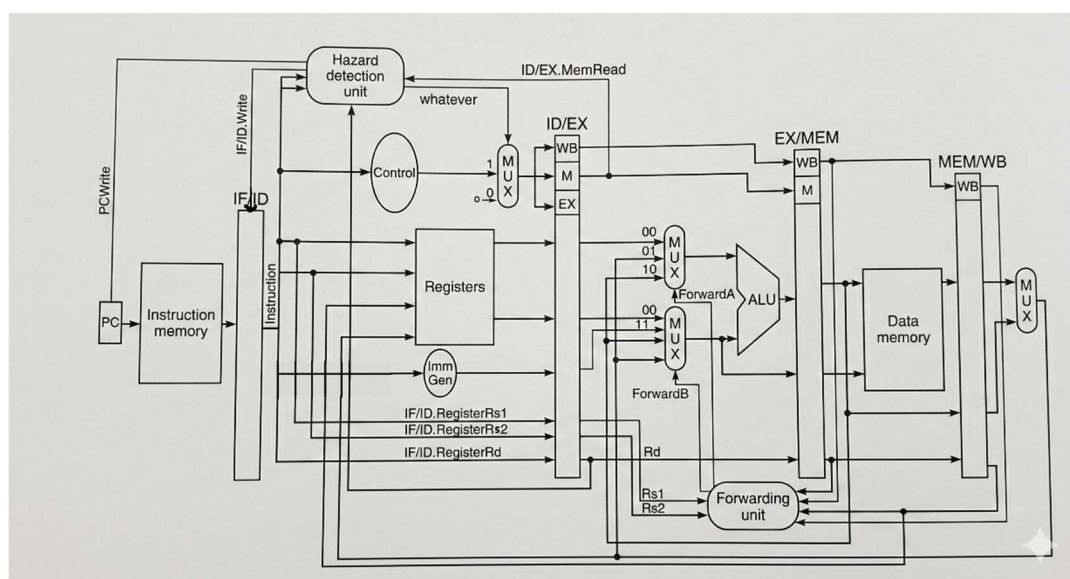
参考如下数据通路图



注：该图像由 Nano Banana Pro 参考原图高清重绘，可能存在错误

六、流水线 CPU（共 15 分）

为处理流水线数据冒险，可以给 RISC-V 五级流水线（IF、ID、EX、MEM、WB）添加前递单元和阻塞单元，如下图所示。



注：该图像由 Nano Banana Pro 参考原图高清重绘，可能存在错误

前递单元的输入： ID/EX.rs1、ID/EX.rs2；EX/MA.RegWrite、EX/MA.rd；
MA/WB.RegWrite、MA/WB.rd

前递单元的输出： ForwardA、ForwardB

ForwardA 的定义： 为 00 时选择从寄存器读出的数据；为 10 时选择上条指令的运算结果；为 01 时选择上上条指令的运算结果。

ForwardB 的定义： 为 00 时选择从寄存器读出的数据；为 10 时选择上条指令的运算结果；为 01 时选择上上条指令的运算结果。

阻塞单元的输入： IF/ID.rs1、IF/ID.rs2；ID/EX.MemRead、ID/EX.rd

阻塞单元的输出： PCWrite、IF/IDWrite、whatever

PCWrite 的定义： 为 1 时正常修改 PC；为 0 时阻止修改 PC。

IF/IDWrite 的定义： 为 1 时正常修改 IF/ID；为 0 时阻止修改 IF/ID。

whatever 的定义： 为 0 时 ID/EX 的控制信号清 0；为 1 时 ID/EX 的控制信号正常。

在上述流水线上执行以下代码：

```
sw x0, 0(x2)
add x10, x11, x12
addi x10, x13, 10
sub x14, x15, x10
```

假设第 1 个时钟周期（C1）取第 1 条指令，请回答下列问题：

(1) 画出多周期流水线状态示意图；(5 分)

(2) 在下面的表格中填入指定周期冒险检测单元的输入、输出：(5 分)

冒险检测单元	C4	C5
IF/ID. rs1		
IF/ID. rs2		
ID/EX. MemRead		
ID/EX. rd		

冒险检测单元	C4	C5
PCWrite		
IF/IDWrite		
ID/EXwhatever		
——		

(3) 在下面的表格中填入指定周期的前递单元的输入、输出：(5 分)

前递单元	C5	C6
IF/ID. rs1		
IF/ID. rs2		
EX/MA. RegWrite		
EX/MA. rd		

前递单元	C5	C6
MA/WB. RegWrite		
MA/WB. rd		
ForwardA		
ForwardB		

七、Cache（共 10 分）

考虑如下代码片段：

```
#include <stdio.h>
#include <time.h>
#define SIZE 50000
#define STRIDE 1
int main() {
    int array[SIZE];
    int sum = 0;
    // 初始化数组
    for (int i = 0; i < SIZE; i++) {
        array[i] = i;
    }
    // 特定步长访问数组并求和
    clock_t start = clock();
    for (int i = 0; i < SIZE; i += STRIDE) {
        sum += array[i];
    }
    clock_t end = clock();

    double time_taken = ((double)(end - start)) / CLOCKS_PER_SEC;
    printf("Sum: %d\n", sum);
    printf("Time taken: %lf seconds\n", time_taken);
    return 0;
}
```

假设计算机系统具有以下 cache 特性：采用 2 路组相联映射方式，cache 容量为 16KiB，块大小为 32 字节，主存按字节编址，且 cache 初始为空。

（1）详细计算在特定步长访问数组并求和阶段的 cache 命中率（需考虑 cache 地址映射关系、块替换策略等，假设采用 LRU 替换策略）。（4 分）

（2）若访问 cache 的时间为 $t_c = 4ns$ ，访问主存的时间为 $t_m = 40ns$ ，根据计算出的命中率计算该阶段的平均访存时间。（3 分）

（3）如果将代码中的 STRIDE 值修改为 100，分析该阶段的 cache 命中率和平均访存时间会出现什么情况，并说明原因。（3 分）

八、虚拟存储器（共 10 分）

在一个高性能计算机系统中，采用了页式虚拟存储器管理机制，并结合了快表（TLB）来加速地址转换过程。该系统的虚拟地址空间大小为 256TiB，页面大小为 64KiB。物理内存容量为 64GiB。TLB 采用全相联映射方式，TLB 共有 128 个表项，每个表项大小为 64 位。

系统运行一个大型数据库管理程序，该程序在运行过程中会频繁地对数据库中的数据进行读写操作，并且数据在虚拟地址空间中的分布呈现出一定的局部性，但同时也存在一些跨页面的大数据块访问。

（1）VPN（虚拟页号）、PPN（物理页号）和 offset（页偏移）分别多少位？TLB 表项中必须包含哪些信息？（3 分）

（2）假设在某一时刻，系统为该进程分配了 5 个物理页面，其对应的虚拟页号分别为 10、20、30、40 和 50，且这些页面在物理内存中的物理页号分别为 100、200、300、400 和 500，虚拟地址：0x00122030、0x00286090 和 0x0070E150 对应的物理地址是多少？（3 分）

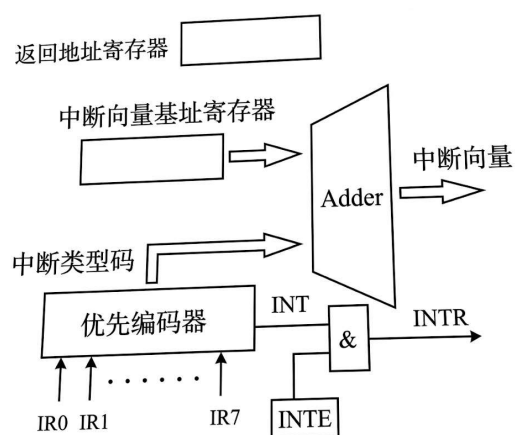
（3）假设数据库程序当前需要访问虚拟地址序列：0x00100000、0x00101000、0x00200000、0x00202000、0x00300000、0x00303000、0x00400000、0x00404000、0x00500000、0x00505000。在初始状态下，TLB 为空，页表中所有页表项均有效且已将相关页面调入物理内存。

假设页面的访问权限均为可读可写，初始页面状态均为未修改。

若 TLB 的命中时间为 $t_{TLB} = 3ns$ ，访问主存中的页表时间为 $t_{PT} = 100ns$ ，访问物理内存数据的时间为 $t_M = 50ns$ 。根据上述访问序列，忽略 cache，计算该序列的平均访存时间。（4 分）

九、输入输出系统（6 分）

下面是一个简单的不允许中断嵌套的向量中断示意图。其中，触发器 INTE 是中断源 IR0-IR7 的总开关，总开关打开时系统才有可能处理中断，否则不予理睬。请说明系统处理中断的基本过程。



注：该图像由 Nano Banana Pro 参考原图高清重绘，可能存在错误