

计算机组成原理

Homework Solution 5

Made by TA

2023 年 4 月 9 日

题目 1. 计算题：假设能保存5位有效数位并且指数的表示范围与IEEE 754标准单精度浮点数相同

1. 将 1.0625_{10} 和 -0.6875_{10} 用十进制科学计数法表示后计算浮点加法。
2. 将 1.0625_{10} 和 -0.6875_{10} 用二进制科学计数法表示后计算浮点加法。

解答:

1. $1.0625_{10}=1.0625_{10} \times 10^0$ $-0.6875_{10}=-6.8750_{10} \times 10^{-1}$

(1). 对齐小数点 $1.0625_{10} \times 10^0 + -0.6875_{10} \times 10^0$

(2). 将有效数位相加 $1.0625_{10} \times 10^0 + -0.6875_{10} \times 10^0 = 0.3750_{10} \times 10^0$

(3). 标准化结果并检查上溢/下溢 $3.7500_{10} \times 10^{-1}$

(4). 必要时进行舍入和重整 $3.7500_{10} \times 10^{-1}$

2. $1.0625_{10}=1.0001_2 \times 2^0$ $-0.6875_{10}=-1.0110_2 \times 2^{-1}$

(1). 对齐小数点 $1.0001_2 \times 2^0 + -0.1011_2 \times 2^0$

(2). 将有效数位相加 $1.0001_2 \times 2^0 + -0.1011_2 \times 2^0 = 0.0110_2 \times 2^0$

(3). 标准化结果并检查上溢/下溢 $1.1000_2 \times 2^{-2}$

(4). 必要时进行舍入和重整 $1.1000_2 \times 2^{-2}$

题目 2. 假设用来实现处理器数据通路的各功能模块延迟如下所示：

I-Mem / D-Mem	Register File	Mux	ALU	Adder	Single gate	Register Read	Register Setup	Sign extend	Control
250 ps	150 ps	25 ps	200 ps	150 ps	5 ps	30 ps	20 ps	50 ps	50 ps

其中，寄存器读延迟指的是，时钟上升沿到寄存器输出端稳定输出新值所需的时间。该延迟仅针对PC寄存器。寄存器建立时间指的是，寄存器的输入数据稳定到时钟上升沿所需的时间。该数值针对PC寄存器和寄存器堆。

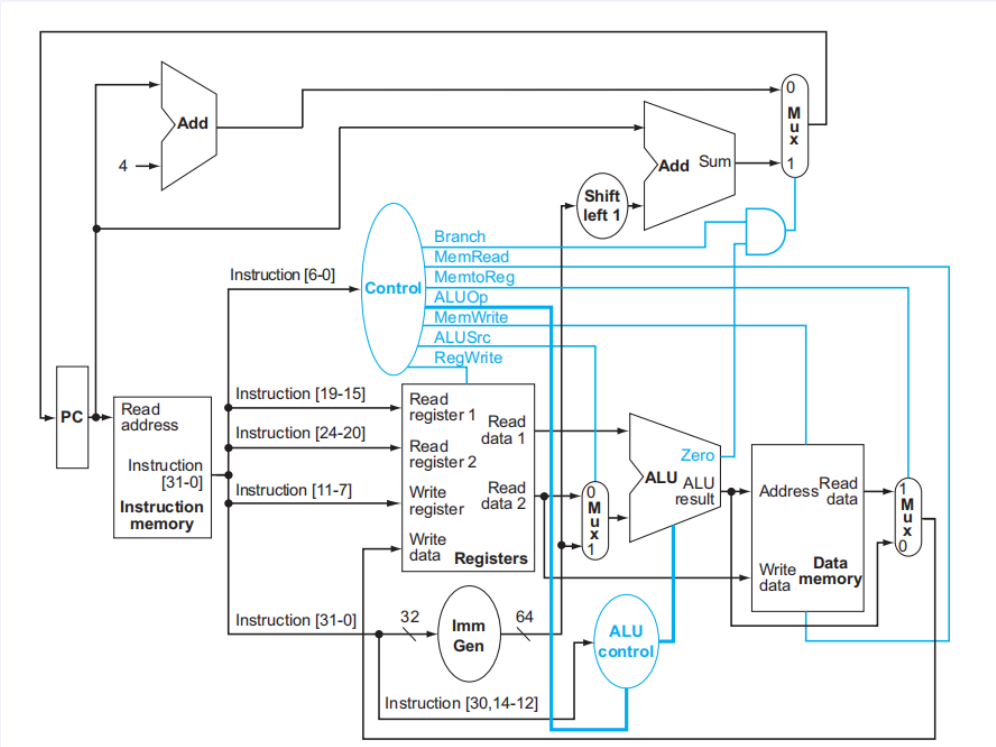


图 1: 数据通路示意图

数据通路参考书中图4-17，即上图；其他延迟默认为0；每条指令从读PC开始，即都包含PC读延迟。

- 1. R型指令的延迟是多少？
- 2. ld指令的延迟是多少？
- 3. sd指令的延迟是多少？
- 4. beq指令的延迟是多少？
- 5. I型指令(不考虑存储器)的延迟是多少？

6. 该CPU的最小时钟周期是多少？

提示： 仔细检查你的答案，许多学生会在关键路径上添加额外的寄存器。

解答：

1. $\text{PCRead (30ps)} + \text{IM (250ps)} + \text{RF (150ps)} + \text{MUX (25ps)} + \text{ALU (200ps)} + \text{MUX (25ps)} + \text{RFSetup (20ps)} = 700\text{ps}$
2. $\text{PCRead (30ps)} + \text{IM (250ps)} + \text{RF (150ps)} + \text{ALU (200ps)} + \text{DM (250ps)} + \text{MUX (25ps)} + \text{RFSetup (20ps)} = 925\text{ps}$
3. $\text{PCRead (30ps)} + \text{IM (250ps)} + \text{RF (150ps)} + \text{ALU (200ps)} + \text{DM (250ps)} = 880\text{ps}$
4. $\text{PCRead (30ps)} + \text{IM (250ps)} + \text{RF (150ps)} + \text{MUX (25ps)} + \text{ALU (200ps)} + \text{Singlegate (5ps)} + \text{MUX (25ps)} + \text{PCsetup (20ps)} = 705\text{ps}$
5. $\text{PCRead (30ps)} + \text{IM (250ps)} + \text{RF (150ps)} + \text{ALU (200ps)} + \text{MUX (25ps)} + \text{RFSetup (20ps)} = 675\text{ps}$
6. CPU 的时钟周期应大于最长指令所需时间。本题中的 CPU 最小时钟周期为 925ps

实验题 1. 假设从t0所保存的地址开始的八个字节存储了一个IEEE 754标准的双精度浮点数，使用RV32I指令读取该双精度浮点数，将其转换为IEEE 754标准的单精度浮点数后存储在t1所指向的地址中。（尾数部分可以直接截断，不用考虑指数部分的溢出，双精度浮点数以小端序存储）

尽可能为你的指令添加注释，使得助教能够更好的理解你的代码。

解答：

```
# Read a double-precision floating-point number
lw x7, 4(t0)
lw x8, 0(t0)

# Extract the sign bit
```

```

srli x9, x7, 31

# Extract the exponent and recalculate the offset
srli x10, x7, 20
andi x10, x10, 0x7ff
addi x10, x10, -1023
addi x10, x10, 127

# Extract the fraction
slli x11, x7, 12
srli x11, x11, 9
srli x12, x8, 29
add x11, x11, x12

# Combine sign bit, exponent and fraction
slli x9, x9, 31
slli x10, x10, 23
add x9, x9, x10
add x9, x9, x11

# Store a single-precision floating-point number
sw x9, 0(t1)

```

思考题 1. 在C语言中在表达式的操作数类型不同时，编译器将尝试隐式类型转换，朝表达数据能力更强的方向进行转换 $\text{int} \rightarrow \text{unsigned int} \rightarrow \text{long} \rightarrow \text{unsigned long} \rightarrow \text{float} \rightarrow \text{double} \rightarrow \text{long double}$:

1. float和double在十进制下的表示精度是多少？
2. 从unsigned long转换到float是否会有精度损失？

解答:

1. float的尾数部分有23位，double的尾数部分有52位， $-\log_{10} 2^{-23} \approx 6.924$ ， $-\log_{10} 2^{-52} \approx 15.654$ ，因此float在十进制下的表示精度为6位-7位，而double为15位-16位。
2. 若认为unsigned long用8个字节存储，由于 $\log_{10} (2^{64} - 1) \approx 19.266$ ，其精度可达20位，在转换为float时会有精度损失。即使是4字节存储的int，由于 $\log_{10} (2^{31} - 1) \approx 9.332$ ，其精度也有10位，因此隐式类型转换时舍弃了部分精度从而取得更大的数据表达范围。

思考题 2. 书中图4-17给出了单周期数据通路：

1. 该数据通路能否执行RV32I的全部指令？
2. 如何修改数据通路以支持更多指令？

解答：

1. 当前的数据通路在不考虑Synch指令（FENCE,FENCE.I），Environment指令（ECALL，EBREAK），CSR指令（CSRRW,CSRRS,CSRRC,CSRRWI,CSRRSI,CSRRCI）时已经能支持大部分RV32I指令，但无法支持JAL、JALR、AUIPC指令，不同位宽的Load、Store指令。
2. 参考Ripes软件中的数据通路

