《Computer Organization & Design》 2023

1.2 (5%)

A computer system has a task that can be divided into two parts: Part A and Part B. Part A takes up 60% of the total time to complete the task, while Part B takes up the remaining 40%.

(1) If we can speed up Part A by a factor of 4 and Part B by a factor of 2, what is the overall speedup we can achieve using Amdahl's law?

1.3 (5%)

We have eight great ideas in computer design, match them to the following seven examples in real world (hint: one example gets two ideas).

Ideas:

- Design for Moore's Law
- Use Abstraction to Simplify Design
- Make the Common Case Fast
- Performance via Parallelism
- Performance via Pipelining
- Performance via Prediction
- Hierarchy of Memories
- Dependability via Redundancy

Examples:

1. Cooking recipes that only specify the type of sugars, oils and flours without naming the detailed brand.
2. Apply for multiple waitlist when you book your train tickets on 12306.

3. Organize your clothes and pack them in storage boxes while only leave your favorite ones in your closet.
4. Dividing a team of workers into different groups, each team work independently on their own tasks.
5. Dividing a team of workers into different groups, each team is in charge of a specific task and work in series.
6. A supermarket manager decided to open five checkout lanes at 6pm, and 2 lanes at 9pm.
7. Ultralight fiber materials which can make large size backpacks with reasonable weight.

---

2. Arithmetic for Computer (35%)

2.1 (4%)

What is the octal value of `0x20230508`.

A: ________________________

2.2 (4%)

What is the binary value of -503 (using 2-s complement).

A: ________________________

2.3 (4%)

What is the binary bit pattern representation of -11.53125 (using single precision).

A: ________________________

2.4 (4%)

Double precision representation of -11.53125

A: ________________________

2.5 (4%)

Assume that registers in RV32I (Risc-V 32-bit) hold the following values:

- `x1`: 0x7FFFFFFF
- `x2`: 0x00000001
- `x3`: 0x12345678

What is the value of x3 immediate after the execution of the following assembly code?

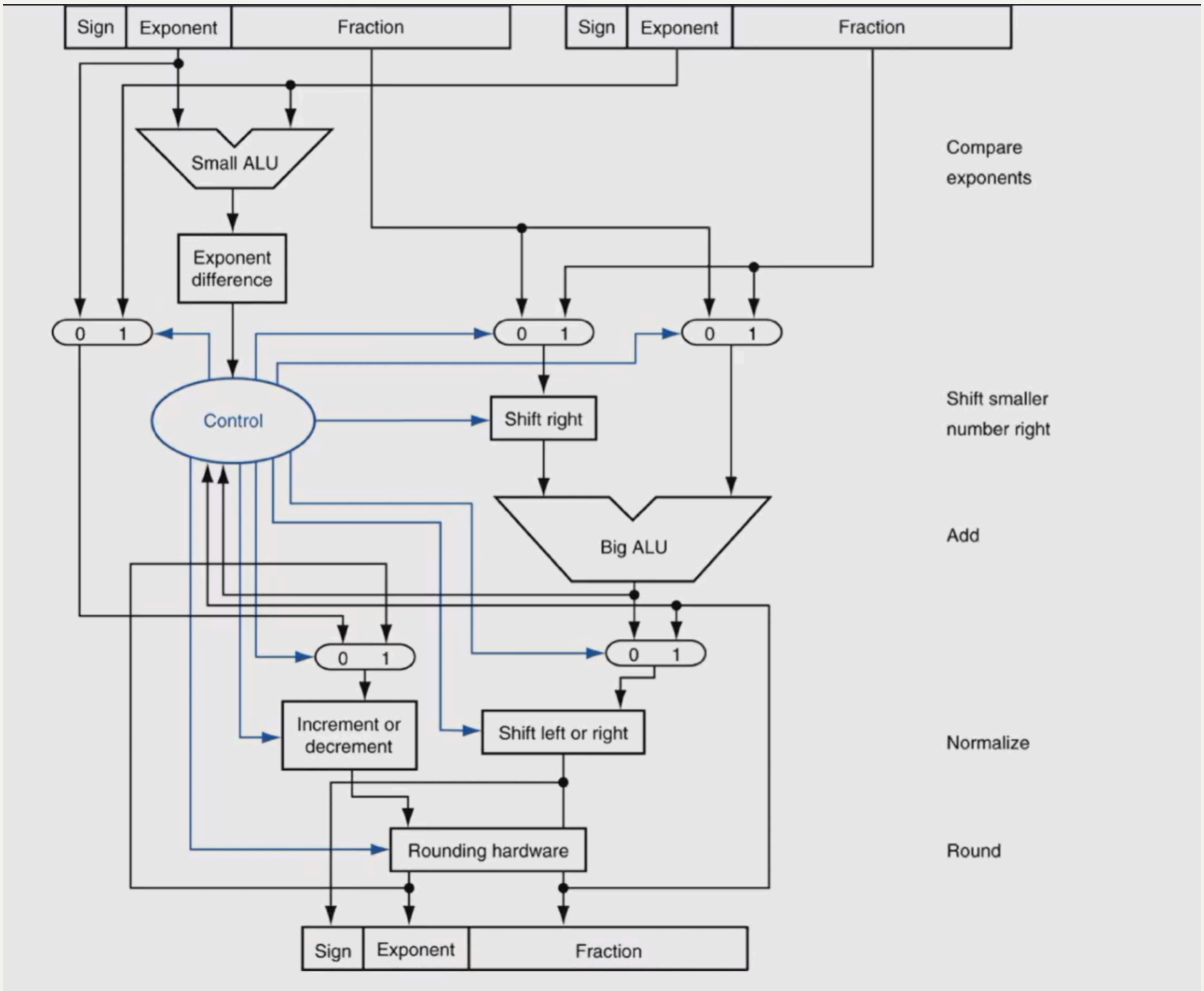
```
add x3, x1, x2
```

A: ________________________

2.6 (15%) Floating arithmetic

(1) Calculate 5.9140625 + 2.34375 × 10^{-2} by hand, assume 1 guard, 1 round bit and 1 sticky bit, and round to the nearest even. Show all the steps, write your answers in single precision floating point format, show all the necessary steps.

(2) What are the control signals for the five MUXs in the diagram below? Suppose 5.9140625 is placed in the left side circuit.



3. Instruction (30%)

3.1 (6%)

Translate the following two RISC-V assembly code into the binary and the hex:

(1) `sw x9, 120(x10)`

A: ____________________________

(2) `beq x7, x8, 100`

A: ____________________________

3.2 (4%)

What is the assembly instruction of the following machine code `0x10D62C23`.

A: ______________________

3.3 (10%)

Suppose you executed the code below on a version of the （单周期） CPU that does not handle data hazards. All registers are initialized to zero.

```
main:   addi zero,zero,0x0
        addi ra,zero,0x1
        addi sp,zero,0x1
        addi gp,zero,0x1
        sw   tp,0x9(tp)
        andi tp,gp,0x8
        ori  gp,sp,0x8
        xor  t0,sp,gp
        lw   t1,0x0(gp)
        bne  t0,t1,flag1
        addi a0,a0,0x1
        slli ra,tp,0x1
        jal  zero,flag2
flag1:  srl  a0,gp,ra
        sub  zero,gp,ra
        jal  ra,main
flag2:  addi sp,sp,0x8
```

After the instructions running, what is the content of the following registers?

- `t1`: ( )
- `gp`: ( )
- `tp`: ( )
- `ra`: ( )
- `a0`: ( )

3.4 (10%)

Implement the following C code in RISCV.

```c
void Sort(int A[], int n)
{
    for (int i = 1; i < n; i++)
    {
        if (A[i] < A[i-1])
        {
            int j = i-1;
            int x = A[i];
            while (j > -1 && x < A[j])
            {
                A[j+1] = A[j];
                j = j - 1;
            }
            A[j+1] = x;
        }
    }
}
```

4. Single cycle processor (20%)

4.1 (10%)

Suppose you are designing a single cycle processor that only support R-type and S-type instructions, draw the most compact processor design below, including both datapath and control logic.

(Space for drawing is provided here)

4.2 (5%)

How many control signals are decoded from the instruction? What are their values for the instruction `add, x1, x2, x3`?

- RegWrite:
- ALUsrc:
- ALUop:
- MemWrite:

4.3 (5%)

Suppose you are designing a single cycle processor that only support `addi` and `ld` instructions. How many control signals are decoded from the instruction? What are their values for the instruction `addi, x1, x2, 5`?

- MemRead:
- MemToReg: