JOINT INSTITUTE
交大密西根学院

ECE3700J Introduction to Computer Organization

Homework 2

Assigned: September 29, 2022

Due: 2:00pm on October 13, 2022

Submit a PDF file on Canvas

1. (5 points) Following memory location has address 0x0F000000 and content 0x18D5FE00.

	0	1	2	3
0x0F000000	00	FE	D5	18

Write RISC-V assembly instructions to load the byte FE as a signed number into register x18, then show the content of x18 after the operations.

Answer:

```
lui x18, 0x0F000
lb x18, 1(x18)
```

Content of x18 is: FFFF_FFFE

2. (10 points) The RISC-V assembly program below computes the factorial of a given input n (n!). The integer input is passed through register x12, and the result is returned in register x10. In the assembly code below, there are a few errors. Correct the errors.

```
FACT: addi sp, sp, 8        #addi sp, sp, -8
      sw x1, 4(sp)
      sw x12, 0(sp)
      add x18, x0, x12      //not necessary. Don't deduct points
      addi x5, x0, 2        #addi x5, x0, 1
      bge x12, x5, L1
      mul x10, x18, x10     //x10 not initialized. Don't deduct
                            //points. Mul not discussed.
      addi sp, sp, -8       #addi sp, sp, 8
      jalr x0, 0(x1)
L1:   addi x12, x12, -1
```

```
jal x1, FACT
addi x10, x0, 1
lw x12, 4(sp)      #lw x12, 0(sp)
lw x1, 0(sp)       #lw x1, 4(sp)
addi sp, sp, -8    #addi sp, sp, 8
mul x10, x12, x10 //don't deduct point, M extension was
                  //not covered in class
jalr x0, 0(x1)
```

3. (10 points) Consider a proposed new instruction named `rpt`. This instruction combines a loop's condition check and counter decrement into a single instruction. For example,

```
rpt x29, loop
```

would do the following:

```
if (x29 > 0) {
   x29=x29-1;
   goto loop;
}
```

1) (5 points) If this instruction were to be added to the RISC-V instruction set, what is the most appropriate instruction format?

Answer:

The J format would be most appropriate because it would allow the maximum number of bits possible for the "loop" parameter, thereby maximizing the utility of the instruction.

2) (5 points) What is the shortest sequence of RISC-V instructions that performs the same operation?

Answer:

```
loop: addi x29, x29, -1 # Subtract 1 from x29
      bgt x29, x0, loop # Continue if x29 not negative
      addi x29, x29, 1  # Add back 1 that shouldn't have been
                        # subtracted.
```

4. (7 points) Given a 32-bit RISC-V machine instruction:

```
1_1111111111_0_10100010_10101_1101111
```

1) (6 points) What does the assembly instruction do?

Answer:

jal x21, Target

the immediate number is: 1_10100010_0_1111111111 = -191489

where Target address = current PC + (-382978)

2) (1 point) What type of instruction is it?

Answer:

J/UJ type.

5. (6 points) Given RISC-V assembly instruction:

```
sw x21, -16(sp)
```

1) (5 points) What is the corresponding binary representation?

Answer:

-16 = 1111111_10000

Machine code is: 1111111_10101_00010_010_10000_0100011

2) (1 point) What type of instruction is it?

Answer:

S type

6. (12 points) If the RISC-V processor is modified to have 64 registers rather than 32 registers:
   1) (4 points) show the bit fields of an R-type format instruction assuming opcode and func fields are not changed.

Answer:

funct7	rs2	rs1	funct3	rd	opcode
7 bits	6 bits	6 bits	3 bits	6 bits	7 bits

The total number of bits would be 35 bits.

   2) (4 points) What would happen to the I-type instruction if we want to keep the total number of bits for an instruction unchanged?

Answer:

imm	rs1	funct3	rd	opcode
10 bits	6 bits	3 bits	6 bits	7 bits

If we keep the total number of bits 32, each register field will be 6 bits, opcode and func3 fields have to stay the same to specify different functions, then the immediate field has to be 2 bits shorter (10 bits).

3) (4 points) What is the impact on the range of addresses for a `beq` instruction? Assume all instructions remain 32 bits long and the size of opcode and func fields don't change.

Answer:

imm	rs2	rs1	funct3	imm	opcode
4 bits	6 bits	6 bits	3 bits	6 bits	7 bits

The immediate number will be 2 bits shorter.
Originally, the range of beq was -2^{12} ~ $2^{12}-1$
After the change, the range becomes -2^{10} ~ $2^{10}-1$

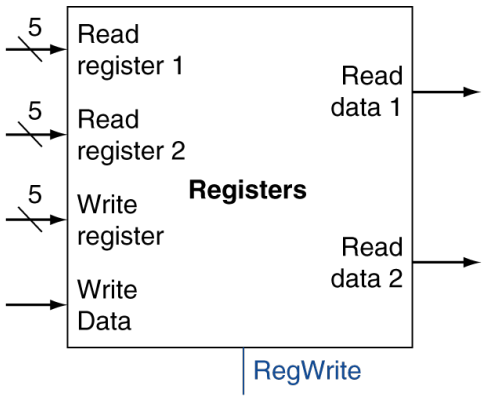
7. (15 points) Convert the following assembly code fragment into machine code, assuming the memory location of the first instruction (LOOP) is 0x1000F400

```
LOOP:    bge x0, x5, ELSE     # B type
         jal x0, DONE         # J type
ELSE:    addi x5, x5, -1      # I type
         add x25, x25, x5     # R type
         jal x0, LOOP         # J type
DONE:    …
```

Instr.	Address	Machine code
LOOP:bge	0x1000F400	0_000000_00101_00000_101_0100_0_1100011
jal DONE	0x1000F404	0_0000001000_0_00000000_00000_1101111
ELSE:addi	0x1000F408	111111111111_00101_000_00101_0010011

add	0x1000F40C	0000000_00101_11001_000_11001_0110011
jal LOOP	0x1000F410	1_1111111000_1_11111111_00000_1101111
DONE:	0x1000F414	-

8. (15 points) Model the Register File component shown below in Verilog HDL. Show source code and screen shots of simulation results.



9. (20 points) Model the following Immediate Generator component in Verilog HDL. Show source code, and simulation results of one instruction for each type involving immediate numbers.