Our Example ISA

This is the same ISA used in HW03 and HW04, but presented to fit onto one printed page.

Each instruction is one or two bytes, with the meaning of those bytes being:

7	6	5	4	3	2	1	0
`0`	`icode`			`a`		`b`	

byte at `pc`

7	6	5	4	3	2	1	0
immediate							

byte at `pc + 1`

Not all instructions have the second byte; those that do describe it below as the byte "at `pc + 1`".

In the table below `rA` means "the value stored in register number `a`" and `rB` means "the value stored in register number `b`."

`icode`	`b`	Behavior	add to `pc`
0		`rA = rB`	1
1		`rA += rB`	1
2		`rA &= rB`	1
3		`rA` = read from memory at address `rB`	1
4		write `rA` to memory at address `rB`	1
5	0	`rA = ~rA`	1
5	1	`rA = -rA`	1
5	2	`rA = !rA`	1
5	3	`rA = pc`	1
6	0	`rA` = read from memory at `pc + 1`	2
6	1	`rA` += read from memory at `pc + 1`	2
6	2	`rA` &= read from memory at `pc + 1`	2
6	3	`rA` = read from memory at the address stored at `pc + 1`	2
7		if `rA <= 0`, set `pc = rB`	N/A
7		if `rA > 0`, do nothing	1

If the first bit of the byte at `pc` is `1` instead of `0`, the above text does not define what the instruction means, but some other source (such as a question on this exam) might. If it has no defined meaning either here or elsewhere, leave the `pc` and all other registers and memory values unchanged.

CSO1
Fall 2023
Midterm 1
2023-02-24
Time Limit: 50 minutes

Name: ______________________

Computing ID ______________________

Instructions:

1. This exam contains 10 pages (including this cover page) and 21 questions.
2. You have **50 minutes** to complete the examination. As a courtesy to your classmates, we ask that you not leave during the last fifteen minutes.
3. Write your answers in this booklet. We scan this into GradeScope, so **please try to avoid writing on the backs of pages**.
4. If a question presents several options in a list, mark the bubble next to the one correct answer. All such questions on this test are single-select.
5. You may not use a calculator or notes.
6. Because this assessment is being given in several places, we cannot fairly answer questions during it. If you find a question ambiguous or unclear, please explain that on the page by the question itself and we will consider your explanation during grading.
7. Please sign the below Honor Code statement.

I have neither given nor received aid on this exam.

Signature: ______________________________________

1 Binary and Hex

1. (2 points) Convert 45_{10} (base 10) to binary. (Remember to write your answer in **1s and 0s** in the box so that the grader can see it.)

0b

2. (2 points) What is the result in **hex** when we XOR `0x9FDF` with `0xBEEF`? (Remember to write your answer in the box so that the grader can read it.)

```
  0x9FDF 1001111111011111
⊕ 0xBEEF 1011111011101111
  -----------------------
```

0x

3. (2 points) Write 132_{10} in base 5. Use the space below for rough work, but remember to write your final answer in the box.

$(\qquad\qquad\qquad)_5$

1.1 Signed Representation.

4. (2 points) How would the number -19 be represented on an 8-bit machine using two's complement? Remember to write your answer in 8 bits and pay close attention to the sign.

0b

5. (2 points) How would -0.825 be represented on a 16-bit machine? Let's assume that we are using IEEE 754 Half-precision float, which uses a 5-bit biased exponent. Write your value in hex when you are done

0x

6. (2 points) When adding -2 and -3 on our 5-bit machine using two's complement numbers, did it result in overflow?

   ○ Yes
   ○ No
   ○ It depends

2 Gates

7. (2 points) Consider the following push-pull network. Which gate does it implement? (Hint: write out the truth table.)

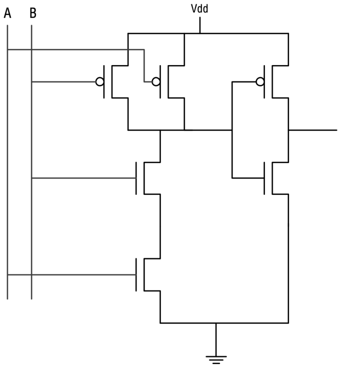

Figure 1: Push-Pull Network, output is the unlabeled wire on the right.

   ○ AND
   ○ OR
   ○ NAND
   ○ XOR
   ○ NOR
   ○ None of the above

3 Bit Wise Operations and Endianess

8. (2 points) Assuming an 8-bit machine with signed integers, what is the value of "-3 » 2"? Assume that shifts are signed extended and the result is interpreted as two's complement. Answer as a signed base-10 number, like "+3" or "-14".

9. (2 points) Which of the following constructs the one-byte mask "01010101"? Select all that apply.

   ○ `0xAA`

   ○ `0xA0`

   ○ `0x0A`

   ○ `0101 << 4`

   ○ `1010 >> 4`

   ○ `~0x33`

   ○ `0x55`

   ○ None of the above

10. (2 points) If the 32-bit integer `0x0A0B0C0D` is to be stored in a **big** endian machine at address 0x21, what bytes are at 0x23?

0x

11. (2 points) If the 32-bit integer `0x0A0B0C0D` is to be stored in a **litte** endian machine at address 0x22, what bytes are at 0x23?

0x

4 Simple Machines: Adders and Simple Machines

12. (2 points) Let's design a simple machine that is able to run the following programming language.

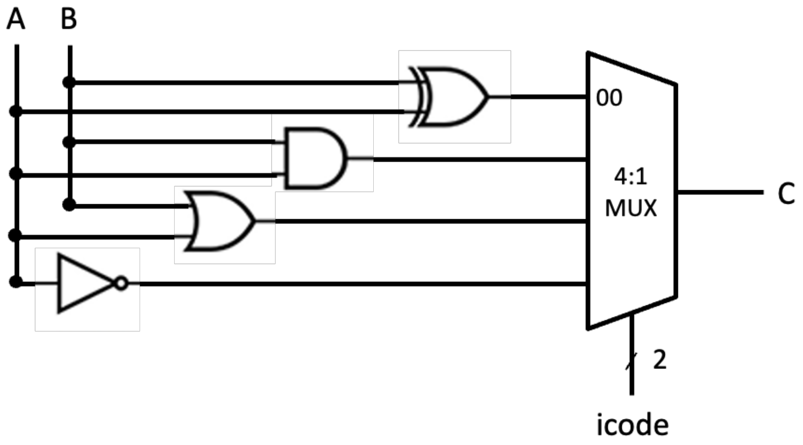

Figure 2: Simple machine

Assuming that the machine instruction layout is as follows:

3	2	1	0
icode		a	b

What is the machine code to execute the following program:

```
A = 0
B = 1
C = A | B
```

Write your answer in hexadecimal.

0x

13. (2 points) What is the result (the value of C in bits) after executing the instruction `0xF` on our machine?

0b

14. (2 points) Is the result (the value of C) the same if the instruction is `0xE`?

○ Yes

○ No

5 Toy Instruction Set Architecture

15. (2 points) In our toy ISA, Break the first instruction in the byte sequence `0x6F 0x20` into its parts by filling in the following:

    If some box is not part of the instruction, write "N/A" in that box.

7	6 5 4	3 2	1 0
R	icode	a	b

7 6 5 4 3 2 1 0
immediate

16. (2 points) A call instruction is stored at address 0x5F. What address gets pushed to the stack when the call instruction is executed?

    0x

17. (5 points) A quine is a program that copies its own code. Quines are interesting because they are self replicating programs. Rearrange the hex instructions below so that it implements a quine that copies it's own code from address `0x00` to address `0xc0`. In other words once the program executes address `0xC0` through `0xCF` should contain exact copy of the program.

    Rearrange the following hex values:

    F0 00 C0 39 01 04 01 7C 6D 60 65 6C 64 11 48 60

    and fill in the cells below. We have populated some cells for you.

		64		60			11		65		6D				

6 Toy Single Cycle Machine

18. (4 points) Here are all the components required to construct a single-cycle machine that implements our Toy ISA. Connect these components to execute the following instruction: `rA += rB`. Ensure that your machine functions correctly in all stages, including fetch, decode, execute, memory, and write-back. If a stage is not required for this instruction, you can skip wiring it. The controller will set C_4 to 2 for this instruction.

    To complete the diagram, indicate which wire each given wire should be connected to. For example, if wire A should be connected to wire E, write "E" in the box next to "A". You do not have to use all the wires. Leaving the appropriate boxes blank is considered the correct answer; filling a box unnecessarily will be considered incorrect.

Function codes for ALU are

code	function
001	and
010	addition
011	multiplication
100	division

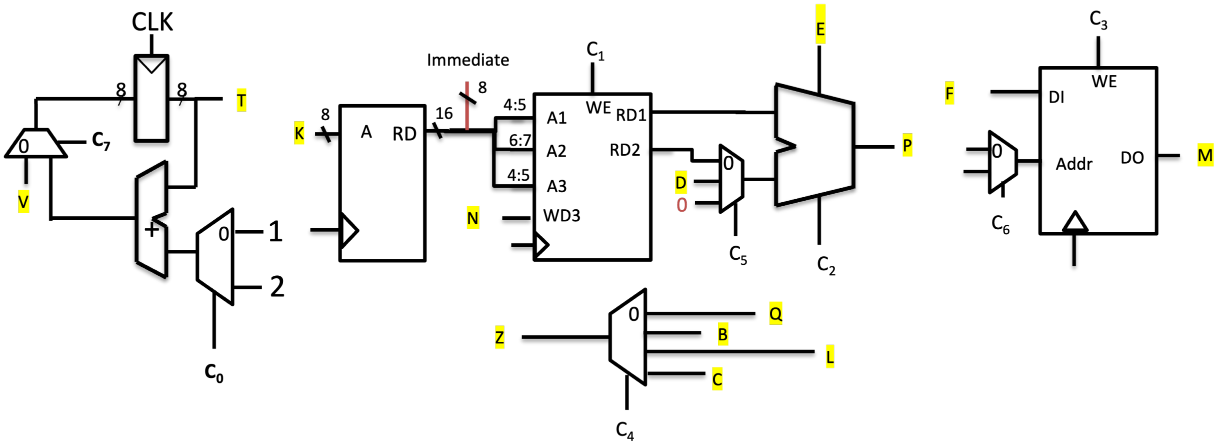

Figure 3: Single Cycle Machine Single Design

T	→		E	→	
M	→		V	→	
Z	→		D	→	
P	→		Q	→	

19. (3 points) What are the values (in decimal) for the other control signals? Write an *X* if we don't care about the value.

C_0	→	
C_1	→	
C_2	→	
C_3	→	
C_4	→	2
C_5	→	
C_6	→	

20. (1 point) Assuming that PC register is made of positive edge triggered D flip flops. When does the PC update?

○ rising edge

○ failing edge

○ dual edge

○ none of the above

21. (2 points) How difficult was this midterm?

○ too easy

○ easy but fair

○ fair

○ difficult but fair

○ too difficult

. . . .

You may use the space below as scratch paper