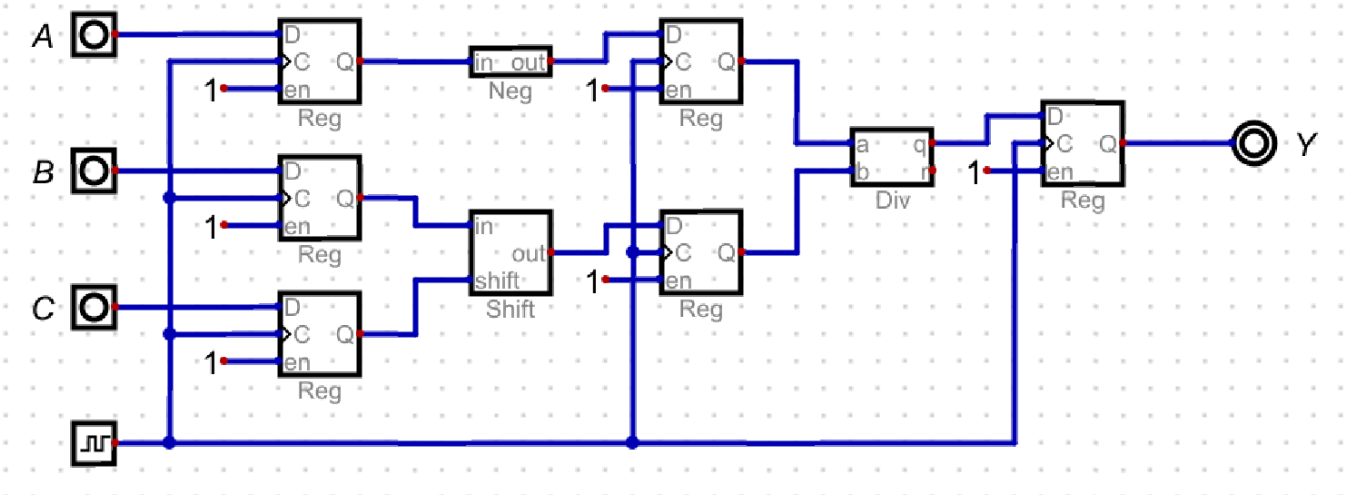
COMP311 Extra Practice 4 Name: _____

Question 1: Two Stage Pipelined Circuit Let the circuit use these timing values:

- Clock-to-Q delay = 22 ps
- Neg propagation delay = 180 ps
- Shift propagation delay = 610 ps
- Div propagation delay = 1100 ps



1.1. How many pipeline stages are in this circuit?

1.2. What is the delay of each pipeline stage? Include the clock-to-Q delay.

1.3. What is the minimum clock perios of this circuit?

1.4. Imagine the circuit no longer has registers. What is the new minimum clock period?

1.5. Briefly explain the tradeoff between the original pipelined design and the design from part 1.4.

Question 2: Load and Store Instructions Assume the following initial state:
Register values: x1 = 0x100, x2 = 200, x3 = 5.
Memory contents: [20, 40, 60] starting at address 0x00000100.

Consider the following RISC-V instructions:

```
 lw x4, 0(x1)
addi x4, x4, 10
 sw x4, 4(x1)
 lw x5, 4(x1)
```

2.1. What value is loaded into x4 after the first instruction?

2.2. What is the value of x4 after the second instruction?

2.3. What value is written to memory, and at what address?

2.4. What is the value of x5 at the end of execution?

2.5. What are the final values stored in memory at addresses 0x100, 0x104, and 0x108?

Question 3: Register Decode II – The Second Transmission Nice work on those R-type transmissions, Agent. But headquarters just intercepted a **second transmission** — and this time the enemy is hiding data in **memory**. Execute the instructions below in order, then decode the ASCII values in `x10`–`x14` to reveal the secret message.

Part I: The Intercepted Instructions

Label	Instruction
I1	`lw x10, 0(x0)`
I2	`lw x15, 0(x8)`
I3	`sw x15, 12(x0)`
I4	`lw x11, 0(x7)`
I5	`add x12, x10, x9`
I6	`sw x12, 32(x0)`
I7	`lw x13, 0(x3)`
I8	`lw x14, 4(x1)`

Initial Register File:

x0	**x1**	**x2**	**x3**	**x4**	**x5**	**x6**	**x7**
0	4	8	20	16	37	55	12
x8	**x9**	**x10**	**x11**	**x12**	**x13**	**x14**	**x15**
24	6	0	0	0	0	0	0
x16	**x17**	**x18**	**x19**	**x20**	**x21**	**x22**	**x23**
91	33	56	7	88	19	64	3
x24	**x25**	**x26**	**x27**	**x28**	**x29**	**x30**	**x31**
45	72	11	99	15	60	21	47

Initial Memory (word-aligned, byte-addressed):

Address	**0**	**4**	**8**	**12**	**16**	**20**	**24**	**28**	**32**	**36**
Value	74	33	78	52	40	65	65	69	48	90

Question 4: Spy Decoder II – Execute and Decode Execute the 8 instructions from the previous question **in order** on the register file and memory above, then decode the result.

4.1. Write the resulting values in registers `x10`–`x14` after execution.

4.2. Write the final contents of memory after all instructions have executed.

4.3. Decode the values in `x10`–`x14` using the ASCII table below. What is the secret message?

ASCII Reference (selected values):

Dec	65	66	67	68	69	70	71	72	73	74	75	76	77	78
Char	A	B	C	D	E	F	G	H	I	J	K	L	M	N

Dec	79	80	81	82	83	84	85	86	87	88	89	90
Char	O	P	Q	R	S	T	U	V	W	X	Y	Z

Question 5: RISC-V Memory Model & Program Execution Consider a RISC-V processor executing the following code snippet. Assume the registers initially contain $x10 = 0x2000$ and $x11 = 0x40$. The memory at address $0x2040$ contains the 32-bit word $0xDEADBEEF$.

```
0x00400000: addi x12, x10, 0x40
0x00400004: lw   x13, 0(x12)
0x00400008: slli x13, x13, 4
0x0040000C: sw   x13, 4(x12)
```

5.1. After the execution of the `lw` instruction, what is the value stored in the Program Counter (PC)?

5.2. Determine the effective memory address accessed by the `sw` instruction and the exact 32-bit value written to that address in hexadecimal.

5.3. If the memory system is Little-Endian, what is the value of the byte stored at memory address $0x2044$ after the code completes?

Question 6: The 3-Stage Pipeline Model In class, we looked at a first-pass pipelined approach where our CPU is broken into three stages: **Fetch (F)**, **Decode (D)**, and **Execute (E)**.

Consider the following instruction sequence:

```
1: lw   t0, 0(a0)
2: addi t1, a1, 4
3: sw   t1, 8(a0)
```

6.1. Briefly describe which specific hardware components (e.g., ALU, Register File, Instruction Memory, PC) belong in each of the three stages based on this model.

6.2. Complete the pipeline execution table for the three instructions provided above.

Stage	**Cycle** i	**Cycle** $i+1$	**Cycle** $i+2$	**Cycle** $i+3$	**Cycle** $i+4$
Fetch	lw				
Decode					
Execute					

6.3. In this 3-stage model, instructions are decoded in both Fetch and Decode, and Register ports are "Read" in Decode but "Written" at the end of Execute. Based on the table above, in which cycle is the value of `t1` actually stored into memory?

To get more practice with assembly writing, check out the **RISC-V Assembly Practice** on the course website here: `https://github.com/kakiryan/comp-311-assembly-practice-problems`!