Problem 1 (4 parts, 40 points) Safety in Numbers

Part A (10 points) Convert the following notations:

binary notation	**decimal notation**
1 0111 1100.	256 + 64 + 32 + 16 + 8 + 4 = 380
1100.1011	8 + 4 + .5 + .125 + .0625 = 12.6875
1000101.0111	69.4375
binary notation	**hexadecimal notation**
101 1001 0101.0101 101	595.5A
1000 1011 0110.0011 11	8B6.3C

Part B (12 points) For the 28 bit representations below, determine the most positive value and the step size (difference between sequential values). **All answers should be expressed in decimal notation**. Fractions (e.g., 3/16ths) may be used. Signed representations are two's complement.

representation	most positive value	step size
unsigned fixed-point (26 bits) . (2 bits)	64M	1/4
signed integer (28 bits) . (0 bits)	128M	1
signed fixed-point (21 bits) . (7 bits)	1M	1/128
signed fixed-point (14 bits) . (14 bits)	8K	1/16K

Part C (6 points) A 40 bit floating point representation has a 30 bit mantissa field, a 9 bit exponent field, and one sign bit.

What is the largest value that can be represented (closest to infinity)? 2^{255}

What is the smallest value that can be represented (closest to zero)? 2^{-256}

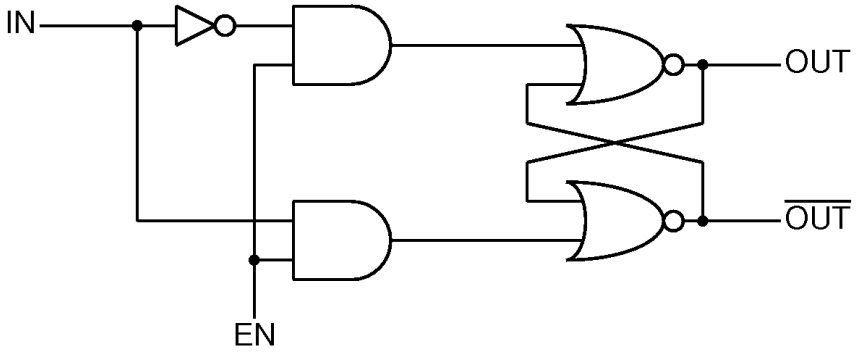
How many decimal significant figures are supported? 9

Part D (12 points) For each problem below, compute the operations using the rules of arithmetic, and indicate whether an overflow occurs assuming all numbers are expressed using a **five bit unsigned fixed-point** and **five bit two's complement fixed-point** representations.

	11.011 + 1.001	1.111 + .001	1.010 - 11.011	10.000 - 0.001
result	00.100	10.000	1.111	1.111
unsigned error?	□ no ■ yes	■ no □ yes	□ no ■ yes	■ no □ yes
signed error?	■ no □ yes	□ no ■ yes	■ no □ yes	□ no ■ yes

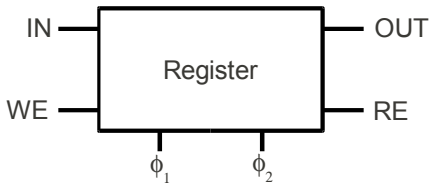
Problem 2 (3 parts, 18 points) State Representatives

Part A (8 points) Implement a transparent latch using only basics gates (AND, OR, NAND, NOR, and NOT). Also complete the behavior table below.



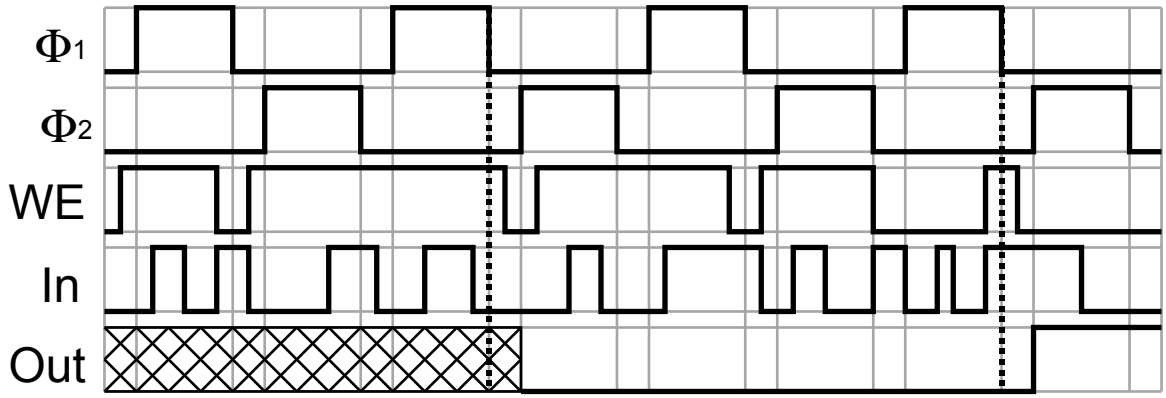
IN	EN	OUT	$\overline{OUT}$
A	0	Q_0	$\overline{Q}_0$
A	1	*A*	$\overline{A}$

Part B (4 points) Consider a register with a selectable *write enable* (WE) and *read enable* (RE). It is implemented with transparent latches, a 2to1 mux, and a pass gate. Describe its behavior by completing the output values. Also indicate when a write and/or a read is being performed.



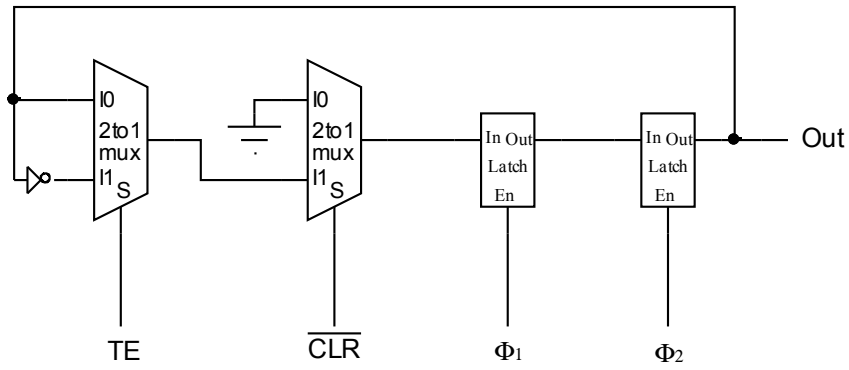
IN	WE	RE	CLK	OUT	write?	read?
A	0	0	↑↓	Z_0		
A	1	0	↑↓	Z_0	✓	
A	0	1	↑↓	Q_0		✓
A	1	1	↑↓	*A*	✓	✓

Part C (6 points) Assume the following signals are applied to a register. Draw the output signal **Out**. Draw a vertical line where **In** is sampled. *Draw crosshatch where **Out** is unknown.*



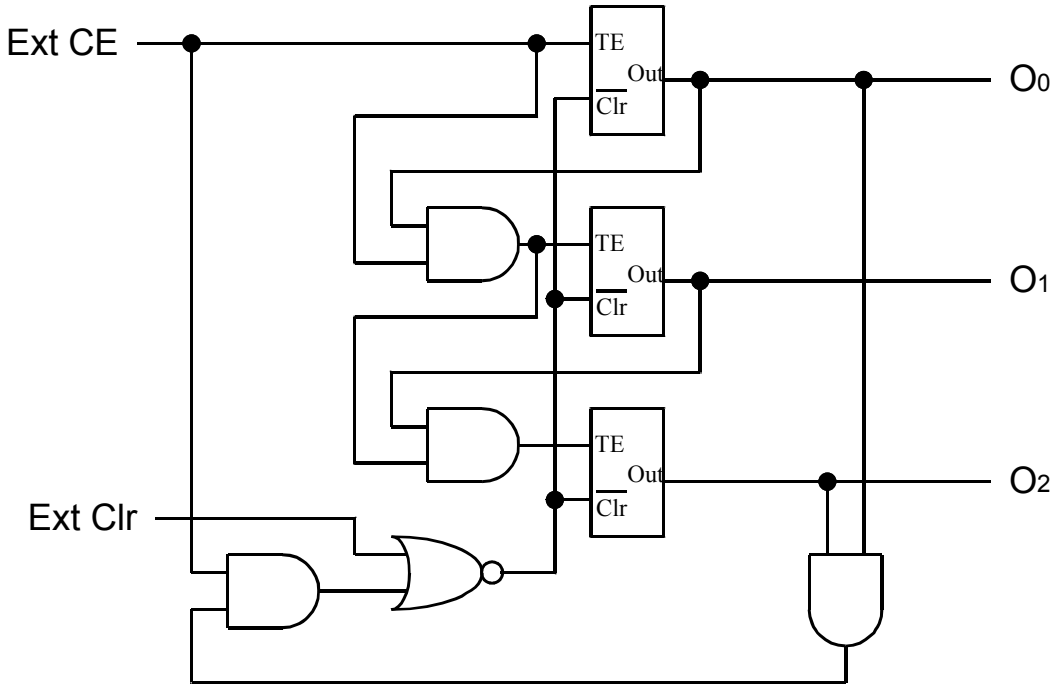
Problem 3 (3 parts, 30 points) Accountable

Part A (10 points) Design a toggle cell using **transparent latches**, **2to1 muxes**, and **inverters** (use icons, **labeling inputs & outputs**). Your toggle cell should have an active high toggle enable input **TE**, and an active low clear input $\overline{\textbf{CLR}}$, clock inputs Φ_1 and Φ_2, and an output **Out**. The $\overline{\textbf{CLR}}$ signal has precedence over **TE**. Also complete the behavior table for the toggle cell.

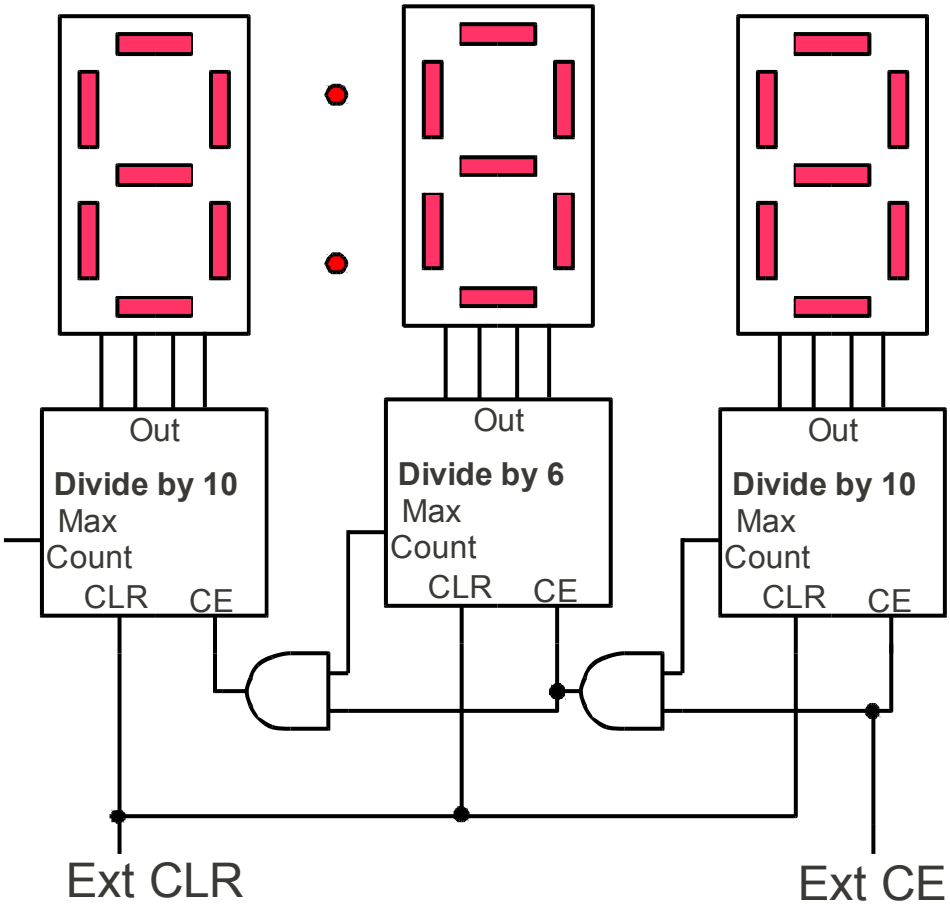


TE	$\overline{CLR}$	CLK	Out
0	0	↑↓	0
1	0	↑↓	0
0	1	↑↓	Q_0
1	1	↑↓	$\overline{Q}_0$

Part B (10 points) Now combine these toggle cells to build a **divide by 6** counter. Your counter should have an external clear, external count enable, and three count outputs O_2, O_1, O_0. Use any basic gates (AND, OR, NAND, NOR, & NOT) you require. Assume clock inputs to the toggle cells are already connected. *Your design should support multi-digit systems*.



Part C (10 points) Build a stopwatch that counts seconds and minutes using divide by N counters drawn below. ***Be sure to fill in the needed divider for seconds, tens of seconds, and minutes.*** Use any basic gates you require. Assume a one hertz clock is already connected.



Problem 4 (1 parts, 12 points) Minimal Packaging

Implement a 4to1 multiplexer using only pass gates and inverters. Label all inputs and outputs.