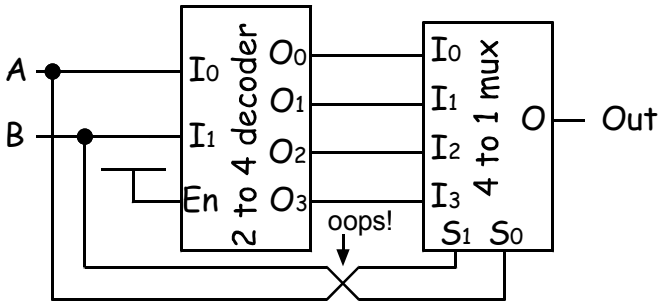
Problem 1 (4 parts, 24 points) Building Blocks

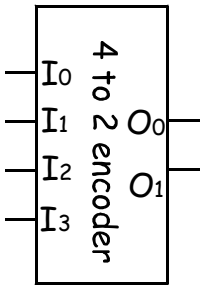
Part A (8 points) Consider the circuit below. Complete the truth table. Then state what logical function this circuit implements.



A	B	Out
0	0	1
1	0	0
0	1	0
1	1	1

This wacky circuit is Even Parity (XNOR)

Part B (6 points) Consider the following circuit below. Determine its input priority.



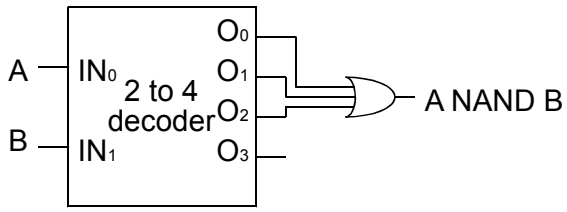
I_0	I_1	I_2	I_3	V	O_1	O_0
0	0	0	0	0	X	X
1	0	0	X	1	0	0
X	1	0	X	1	0	1
X	X	1	X	1	1	0
0	0	0	1	1	1	1

I_2 > I_1 > I_0 > I_3

Part C (4 points) Derive a simplified expression for O_1.

$O_1 = I_2 + \overline{I_0} \cdot \overline{I_1}$

Part D (6 points) Implement a 2-input NAND using a 2-to-4 decoder and a single OR gate (which may have any number of inputs). Label the inputs **A** and **B**, and output **A NAND B.**



Problem 2 (3 parts, 28 points) Number Systems

Part A (10 points) Convert the following notations:

binary notation	**decimal notation**
1100 1010.	128+64+8+2 = **202**
1010 0100.1000 1	128+32+4+0.5+0.03125 = **164.53125**
111 1100.11	64+32+16+8+4+0.5+0.25 = **124.75**
octal notation	**hexadecimal notation**
4733.6	1001 1101 1011.1100 = **0x9DB.C**
24.32	01 0100.0110 1000 = **0x14.68**

Part B (12 points) For the 16 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 integer (16 bits) . (0 bits)	**64K-1**	**1**
signed fixed-point (11 bits) . (5 bits)	**1023 31/32**	**1/32**
signed fixed-point (9 bits) . (7 bits)	**255 127/128**	**1/128**
signed fixed-point (14 bits) . (2 bits)	**8K-1/4**	**1/4**

Part C (6 points) A 34 bit floating point representation has a 24 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}

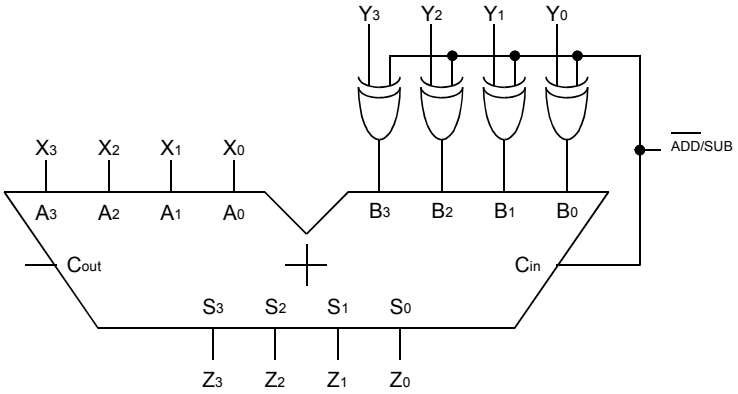
How many decimal significant figures are supported? 7

Problem 3 (3 parts, 24 points) "Go Figure"

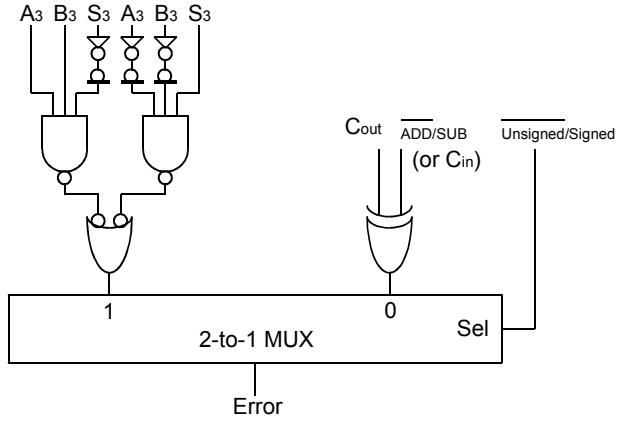
Part A (12 points) For each problem below, compute the operations using the rules of arithmetic, and indicate whether an overflow occurs assuming all numbers are expressed using a **six bit unsigned fixed-point** and **six bit two's complement fixed-point** representations.

	111.101 + 011.100	1.010 + 101.011	101.011 - 10.110	1.010 - 101.011
result	011.001	110.101	010.101	011.111
unsigned error?	**yes**	**no**	**no**	**yes**
signed error?	**no**	**no**	**yes**	**no**

Part B (6 points) The adder below adds two four bit numbers A and B and produces a four bit result S. Add extra digital logic to support subtraction as well as addition. Label inputs X_3, X_2, X_1, X_0, Y_3, Y_2, Y_1, Y_0, $\overline{ADD}/SUB$ and outputs Z_3, Z_2, Z_1, Z_0.

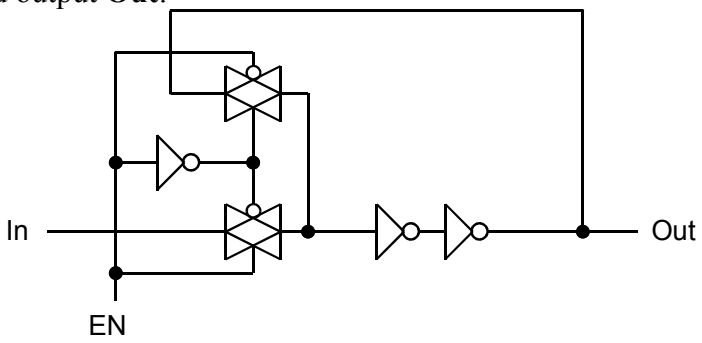


Part C (6 points) Below is a partial implementation of an overflow error detector for the adder/subtractor in part B. It detects errors in addition or subtraction of signed and unsigned numbers. Complete the implementation by filling in the dashed boxes with the appropriate error detector circuitry. For full credit, label all the inputs using signals from **part B** (e.g., $\overline{ADD}/SUB$ and S_3) and the additional input signal $\overline{Unsigned}/Signed$ which is 1 if the numbers being added/subtracted are two's complement numbers and which is 0 if the numbers are unsigned.



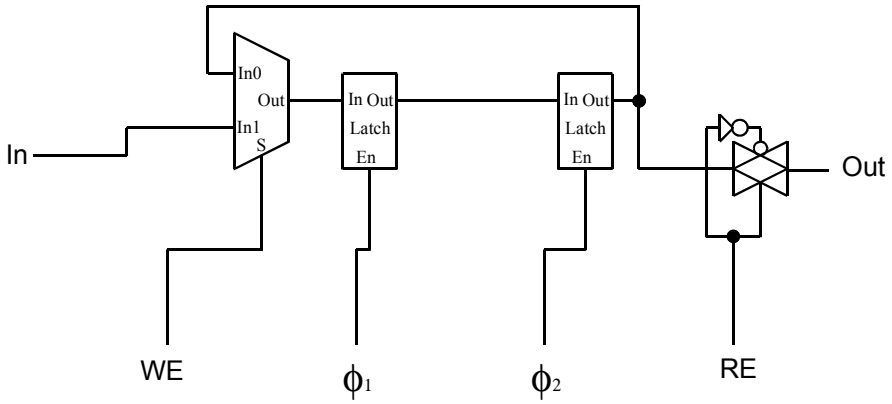
Problem 4 (3 parts, 24 points) Registers and Latches

Part A (8 points) Implement a transparent latch using only inverters and pass gates. Label the inputs **In** and **En**, and output **Out**.



Part B (10 points) Implement a register below using needed muxes, latches, pass gates, and inverters (all in icon form). Complete the behavior table at right. Recall that the CLK signal indicates a full Φ_1 Φ_2 cycle; so the output should be the value at the end of a cycle (with the given inputs).

In	WE	RE	Clk	Out
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 your register. Draw the output signal **Out**. Draw a vertical line where **In** is sampled. *Assume **Out** is initially zero.*