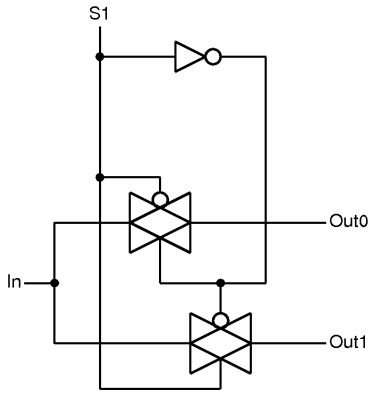
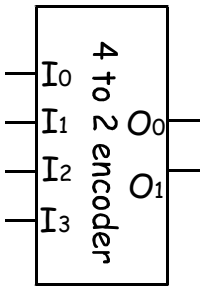
Problem 1 (3 parts, 22 points) Building Blocks

Part A (6 points) Implement a 1 to 2 demultiplexor with pass gates and inverters. Be sure to label all inputs and outputs.



Part B (8 points) Suppose the circuit below has the following input priority: $I_2 > I_3 > I_0 > I_1$. Complete the truth table by filling in the input values that would produce the given outputs and derive a simplified expression for O_1.

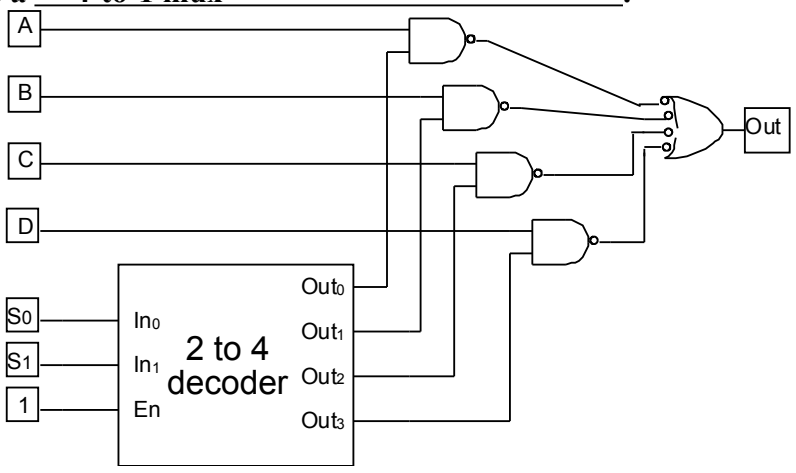


I_3	I_2	I_1	I_0	V	O_1	O_0
0	0	0	0	0	X	X
0	0	x	1	1	0	0
0	0	1	0	1	0	1
x	1	x	x	1	1	0
1	0	x	x	1	1	1

$O_1 =$ $I_2 + I_3$

Part C (8 points) Which building block does the following circuit implement? Label all inputs and outputs by filling in the squares.

This implements a 4-to-1 mux.



Problem 2 (3 parts, 28 points) Number Systems

Part A (10 points) Convert the following notations:

binary notation	decimal notation
1010 1100.	**172**
110.1011	6.6875
101 1111.101	**95.625**
hexadecimal notation	**octal notation**
A3.B	243.54
5F3.DC	**2763.67**

Part B (12 points) For the 32 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 (32 bits) . (0 bits)	**4G**	**1**
signed fixed-point (24 bits) . (8 bits)	**8M**	**1/256**
signed fixed-point (28 bits) . (4 bits)	**128M**	**1/16**
signed fixed-point (16 bits) . (16 bits)	**32K**	**1/64K**

Part C (6 points) A 24 bit floating point representation has a 15 bit mantissa field, a 8 bit exponent field, and one sign bit.

What is the largest value that can be represented (closest to infinity)? 2^{127}

What is the smallest value that can be represented (closest to zero)? 2^{-128}

How many decimal significant figures are supported? 4

Problem 3 (3 parts, 24 points) Adding & Subtracting

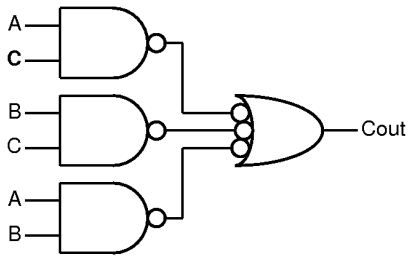
Part A (12 points) For each problem below, compute the operations using the rules of arithmetic, and indicate whether an overflow occurs assuming all numbers are expressed using a **six bit unsigned** and **six bit two's complement** representations.

	101101 + 10111	1110 + 10101	11001 - 111000	110100 - 10010
result	**000100**	**100011**	**100001**	**100010**
unsigned error?	**Yes**	**No**	**Yes**	**No**
signed error?	**No**	**Yes**	**Yes**	**No**

Part B (4 points) Complete the truth table below for a full adder.

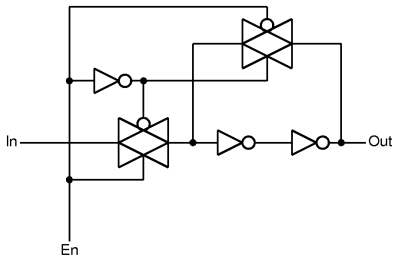
A	**B**	**C_{in}**	**C_{out}**	**Sum**
0	**0**	**0**	**0**	**0**
0	**0**	**1**	**0**	**1**
0	**1**	**0**	**0**	**1**
0	**1**	**1**	**1**	**0**
1	**0**	**0**	**0**	**1**
1	**0**	**1**	**1**	**0**
1	**1**	**0**	**1**	**0**
1	**1**	**1**	**1**	**1**

Part C (8 points) Implement the full adder using only NAND, XOR, and inverter gates. Label inputs **A**, **B**, and **C_{in}**. Label outputs **C_{out}** and **Sum**.

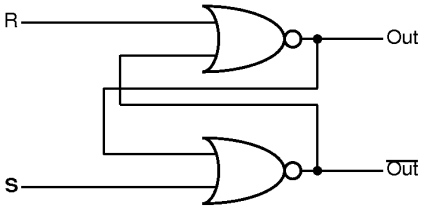


Problem 4 (4 parts, 26 points) Registers and Latches

Part A (6 points) Implement a transparent latch using gates of any type (e.g., AND, inverter, pass gates), but use the minimum number of transistors. Label the inputs **In** and **En**, and output **Out**.

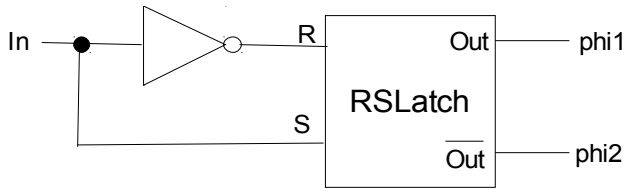


Part B (7 points) Design an RS latch using NOR gates only. Label inputs R and S. Label outputs OUT and $\overline{OUT}$. Do not attempt to employ mixed logic notation. Also complete the truth table.



R	S	OUT	$\overline{OUT}$
0	0	Q0	$\overline{Q0}$
0	1	1	0
1	0	0	1
1	1	0	0

Part C (5 points) Expand the RS latch to an implementation of a two-phase non-overlapping clock, generated from an input signal **In** that is a periodic square wave. Use only an RS latch and basic gates (AND, OR, NAND, NOR, and inverters). Label the input **In** and the outputs F1 and F2.



Part D (8 points) Assume the following signals are applied to a register with write enable. Draw the output signal **Out**. Draw a vertical line where **In** is sampled. *Assume **Out** is initially zero.*