1. (40 points) Quickly testing the basics.
Note: Only write answers in the boxes, and use the extra sheet at the end of the question for any scratch work.

(a) (4 points) The 7-bit 2's complement number representation of -11 is:

1110101

(b) (4 points) 6'b110101, when treated as a 6-bit signed magnitude number, has a decimal representation of:

−21

(c) (4 points) 8'b10011101 has a hexadecimal representation of:

unsigned : 9D / signed : −1D

(d) (4 points) If 8'b10011101 is a fixed point 1001.1101, the corresponding number is:

unsigned : 9.8125 / signed : −1.8125

(e) (4 points) If 8'b10011101 was a 4E3 floating point number (IEEE format), bias is:

3

(f) (4 points) If 8'b10011101 was a 4E3 floating point number (IEEE format), corresponding real number is:

$-\frac{31}{64}$

(g) (4 points) True or false: A boolean function of N variables with greater than $2^N - 1$ product terms can always be simplified to an expression using fewer product terms.

True

(h) (4 points) What is the maximum number of product terms in a minimal sum-of-products expression with three variables?

4

(i) (4 points) What is the minimum number of 2-input NAND gates that would suffice for you to be able to build an implementation of any arbitrary 2-input boolean function?

5

(j) (4 points) You are treating the 8-bit numbers A[7:0] and B[7:0] as unsigned numbers. If you set B[3:0]=A[7:4] and B[7:4]=0, what is the numeric value of B as a function of A?

B=A/16 if / is viewed as integer division
or floor (A/16) if / is viewed as real division

2. (10 points) Reduce the following expression. The simplified expression should have the minimum number of gates. Show the intermediate steps.
$f(a, b, c, d) = \neg(\neg(a \wedge c \wedge d) \wedge (\neg a \vee \neg b \vee \neg d) \wedge (\neg(a \wedge d) \vee c)) \vee \neg(a \wedge \neg b) \wedge (\neg a \vee (\neg b \wedge c) \vee (\neg b \wedge \neg c))$

$$f(a,b,c,d) = \overline{\overline{acd}(\bar{a}+\bar{b}+\bar{d})(\overline{ad}+c)} + \overline{a\bar{b}}(\bar{a}+\bar{b}c+\bar{b}\bar{c})$$

$$= acd + \overline{(\bar{a}+\bar{b}+\bar{d})} + \overline{(\overline{ad}+c)} + (\bar{a}+b)(\bar{a}+\bar{b}(c+\bar{c}))$$

$$= acd + abd + a\bar{c}d + (\bar{a}+b)(\bar{a}+\bar{b})$$

$$= ad(c+b+\bar{c}) + \bar{a}$$

$$= ad + \bar{a}$$

$$= ad + \bar{a}d + \bar{a}d + \bar{a}\bar{d}$$

$$= d + \bar{a}$$

3. (20 points) Consider the following function
$f(a,b,c,d) = (\neg a \wedge \neg b \wedge c) \vee (b \wedge \neg c \wedge \neg d) \vee (\neg a \wedge c \wedge \neg d) \vee (a \wedge b \wedge \neg d) \vee (a \wedge \neg b \wedge d)$

(a) (5 points) Complete the Karnaugh Map shown below, circuit the prime implicants.

cd \ ab	00	01	11	10
00	0	1	1	0
01	0	0	0	1
11	1	0	0	1
10	1	1	1	0

How many prime implicants are there?

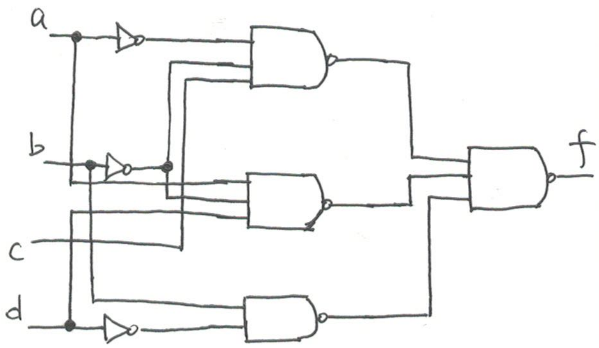
5

(b) (5 points) Show a combinational circuit that implements f using minimum number of inverters and NAND gates only. The maximum number of inputs per gate is less than 4.

According to K-map, using sum-of-products

$f = \bar{a}\bar{b}c + a\bar{b}d + b\bar{d}$

$= \overline{\overline{(\bar{a}\bar{b}c)} \cdot \overline{(a\bar{b}d)} \cdot \overline{(b\bar{d})}}$

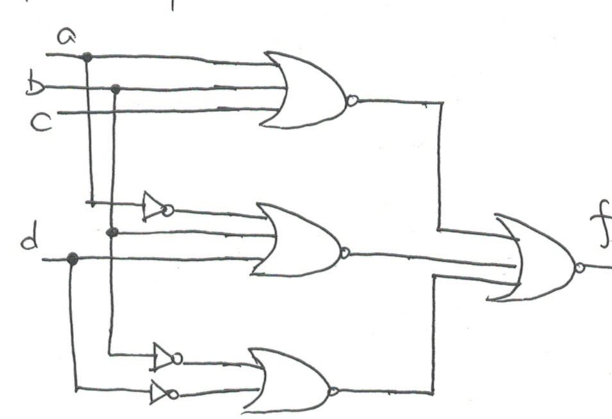


(c) (5 points) Show a combinational circuit that implements f using minimum number of inverters and NOR gates only. The maximum number of inputs per gate is less than 4.

According to K-map, using product-of-sums

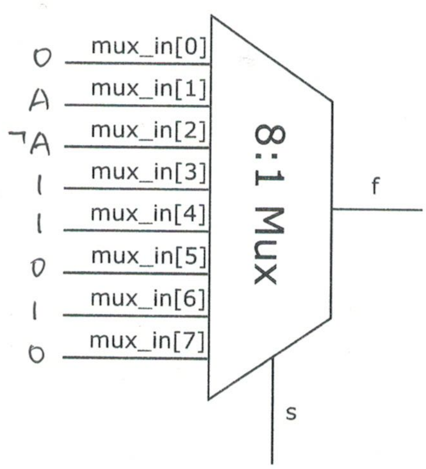
$f = \overline{\bar{a}\bar{b}\bar{c} + a\bar{b}\bar{d} + bd}$

$= \overline{\overline{a+b+c} + \overline{\bar{a}+b+d} + \overline{\bar{b}+\bar{d}}}$



(d) (5 points) Show an 8-input Multiplexer that implements f. The select signal is $s[2:0] = \{B, C, D\}$ where $s = 3'b100$ is $B = 1$ and $C = D = 0$ selecting the input $mux\_in[4]$. Constants, A or $\neg A$ are permissible as inputs, $mux\_in[7:0]$. Write the desired inputs on the figure below.

BCD \ A	0	1
000	0	0
001	0	1
010	1	0
011	1	1
100	1	1
101	0	0
110	1	1
111	0	0

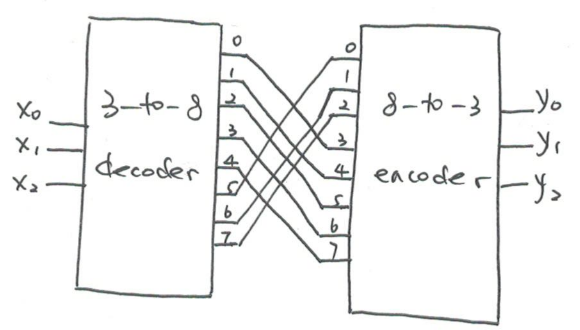


4. (10 points) Design a combinational network that has a three-bit input x representing the digits 0 to 7, and a three-bit output y representing the same set of integers. The function of the system is $y = (x+3) \mod 8$. You may use any building blocks and gates you have learnt.

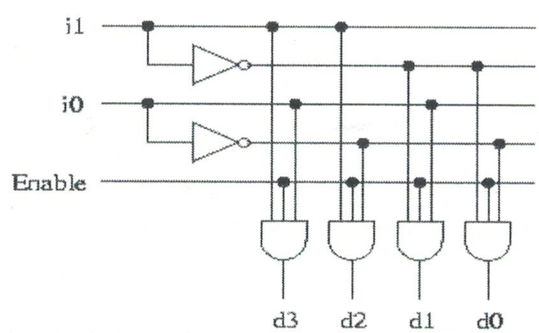
input (x)	output (y)
0	3
1	4
2	5
3	6
4	7
5	0
6	1
7	2

First, create an input-output table. Note that each input has a unique output.

We use a binary 3-to-8 decoder to obtain a 1-hot representation, and then connect the output of the decoder to the input of a 8-to-3 encoder, as shown below.



5. (20 points) A standard decoder typically has an additional input pin E called Enable. The rough idea of using E in a decoder design is shown in the following figure. A 0 value



on E turns decoder off, setting all d_is to 0. Value 1 on E turns decoder on. Design a combinational circuit that converts a 3-bit sign-and-magnitude number, a, into a 3-bit one's complement number, b. You are allowed to use any number and any combination of the following building blocks:

- Decoders with enable pins: $1 \rightarrow 2$ and $2 \rightarrow 4$ decoders
- Encoders: $2 \rightarrow 1$ and $4 \rightarrow 2$ encoders
- Logic gates: Use either OR gates or AND gates, but not both

Every block and wire must be clearly labeled.

decimal	sign-and-magnitude	One's complement
-3	111	100
-2	110	101
-1	101	110
-0	100	111
0	000	000
1	001	001
2	010	010
3	011	011

Observe that if the input is positive, then output b is the same as a. If input is negative, then the bits ~~the b~~ are flipped.