1. (a) Convert $(123.625)_{10}$ to hexadecimal. (Hex = Base 16)

123/16 = 7 (rem 11)
7/16 = 0 (rem 7)
= 7B (11 = B in Hex)

.625 * 16 = 10
= A (10 = A in Hex)

$\mathbf{(123.625)_{10} = (7B.A)_{16}}$

(b) Convert $(123.4)_5$ to decimal.

$(3 \times 5^0) + (2 \times 5^1) + (1 \times 5^2)$
$= 3 + 10 + 25 = 38_{10}$

$4 \times 5^{-1} = 0.8_{10}$

$\mathbf{(123.4)_5 = (38.8)_{10}}$

(c) Convert $(1011)_3$ to binary and $(18)_{10}$ to a number in the number system with radix 5.

$1 * 3^3 + 0 * 3^2 + 1 * 3^1 + 1 * 3^0$
$= 27 + 0 + 3 + 1 = (31)_{10}$

31/2 = 15 (rem 1)
15/2 = 7 (rem 1)
7/2 = 3 (rem 1)
3/2 = 1 (rem 1)
½ = 0 (rem 1)
= $\mathbf{(11111)_2}$

18/5 = 3 (rem 3)
⅗ = 0 (rem 3)
= $\mathbf{(33)_5}$

2. Fixed-point representation is used to represent what type of numbers?
Is it possible to use fixed point representation to represent some fractional numbers.

Fixed point representation is used to represent a number whose binary point is fixed at one position. It is possible to use fixed-point representation to represent fractional numbers. Yes

- - -

3. Three variable-length representations for the character set {A, B, C, D, E} are given below. Indicate if each of them is a correct representation and explain why.

Representation 1		Representation 2		Representation 3	
A	0	A	1	A	0
B	11	B	01	B	10
C	100	C	001	C	11
D	1010	D	000	D	to be determined
E	1011	E	010	E	to be determined

Representation one is a correct representation because it is one to one and prefix free
The prefix is the previous number of the following number.

Rep. 2 is incorrect since the prefix for B and E are the same.

Rep. 3 will have two three bit codes for D and E and can't start with 0, 10, 11 so yea no prefixes.

4. An 9-bit floating-point number representation system has the following specification: Among the 8 bits (B_8 B_7 B_6 B_5 B_4 B_3 B_2 B_1 B_0), bit B8 is the sign bit, and bits B_7 B_6 B_5 B_4 is the exponent with a bias value (0 1 0 0). Bits B_3 B_2 B_1 B_0 consist of the mantissa. The binary number that is represented is $(-1)^{B_8} \times (0.B_3\,B_2\,B_1\,B_0) \times 10^{B_7B_6B_5B_4-0100}$.
(a) What are the largest positive and negative numbers this system can represent?
(b) Find a positive number x which is representable by this system such that 1+x=1.
Find the largest positive number y which is representable by this system such that 1+y=1.

a. $.M \times 10^{expo-bias}$
$(.1111 \times 10^{1111-0100} = .1111 \times 10^{1011})_2 =$
$(.1111 * 10^{0100}) = 1111$
$10^{1011} \times 1/10^{0100} = 10^{1011-0100} = 10^{111}$

$1111 \times 10^{111} = 15 \times 128$

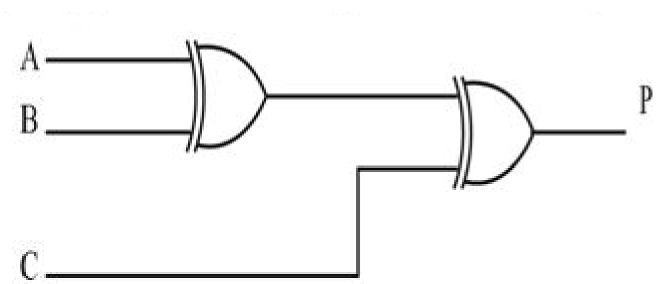
For negative do the same shit as a but just multiply by -1

b. a. **1+x=1** $1_2 + (0.0000\ 0001)_2 = .10000\ 0001 \times 10^{0101-0100}$
$.00001111 * 10^{0101-0100} = (.00001111 * 10^{0100}) \times (10^{0101} / 10^{0100}) = .1111 \times 10^{0001-0100}$
$.1111 \times 10^{-011}$

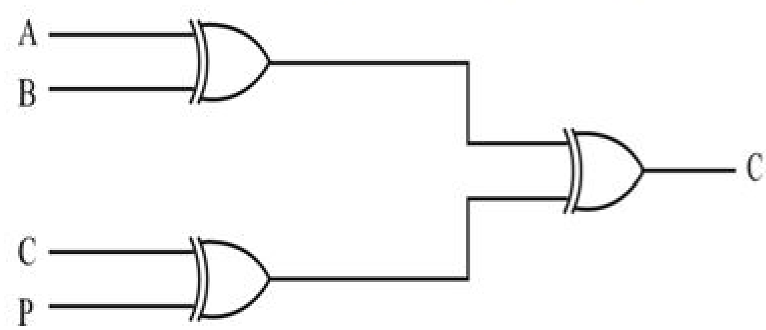
1+y=1.
ANSWER: (0 0000 0001) represents $2^{-4} \times 2^{-4} = 2^{-8} = (0.0000\ 0001)_2$
$1_2 + (0.0000\ 0001)_2 = 1.0000\ 0001 = .1000\ 00001 \times 10^{0101-0100}$
the closest of the above can be represented by (0 0101 1000), which represents 1.

$1+y = .1000\ 1111 \times 10^{0101-0100}$,
means that $y = .0000\ 1111 \times 10^{0101-0100} = .1111 \times 10^{0001-0100} = (1/2+1/4+1/8+1/16) \times 2^{-3}$,
which is represented by *(0 0001 1111)*.

5. Draw the following error detection circuits:
(a) An even parity generator that has 3 input bits and 1 output bit. The output bit is the parity bit.
(b) An even parity checker that has 4 inputs and 1 output bit. The output is an error indicator.



a.



b.

6. Using the technique of even parity bit to design an error correction code for strings of 9-bit, where each string has at most 1 bit error. Give one code that uses as few parity bits as possible.
Sol: $(m + r + 1) <= 2^r$ that is $(10 + r) <= 2^r$. R = 4
(m = bits which would be 9)
b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 (9 bits plus R which is 4) (Everything that is 2^n)
b3 = 1 + 2
b5 = 1 + 4
b6 = 2 + 4

b7 = 1 + 2 + 4
b9 = 1 + 8
10 = 2 + 8
11 = 1 + 2 + 8
12 = 4 + 8
13 = 1 + 4 + 8

On sending, the parity generators generate the following parity bits:
b1 = b3 XOR b5 XOR b7 XOR b9 XOR b11 XOR b13
b2 = b3 XOR b6 XOR b7 XOR b10 XOR b11
b4 = b5 XOR b6 XOR b7 XOR b12 XOR b13
b8 = b9 XOR b10 XOR b11 XOR b12 XOR b13
On the receiving side, the parity checkers check the following:
q1 = b1 xor b3 xor b5 xor b7 xor b9 xor b11 xor b13
q2 = b2 xor b3 xor b6 xor b7 xor b10 xor b11
q3 = b4 xor b5 xor b6 xor b7 xor b12 xor b13
q4 = b8 xor b9 xor b10 xor b11 xor b12 xor b13

Then, if (q4 q3 q2 q1) = (0 0 0 0), then no error.
If (q4 q3 q2 q1) = (0 0 0 1), then bit b1 is flipped.
If (q4 q3 q2 q1) = (0 0 10), then bit b2 is flipped.
If (q4 q3 q2 q1) = (0 0 11), then bit b3 is flipped.
... ...
If (q4 q3 q2 q1) = (1 101), then bit b13 is flipped.

7. Design a combinatorial circuit on the decoding site (i.e. receiver) for 7-bit binary strings equipped with Hamming coding. The input to the circuit is a 7-bit string (I_7 I_6 I_5 I_4 I_3 I_2 I_1), where I_1, I_2 and I_4 are the three even parity bits. Suppose that each 7-bit string (I_7 I_6 I_5 I_4 I_3 I_2 I_1) has at most error on one bit. The output of the circuit has 4 bits B_7, B_6, B_5 and B_3, and for $k = 3, 5, 6, 7$, $B_k = I_k$ if the input bit I_k is correct, otherwise bit $B_k = I_k'$.

Sol:

3 = 1 + 2
5 = 1 + 4
6 = 2 + 4
7 = 1 + 2 + 4

Parity generator:
b1 = b3 xor b5 xor b7
b2 = b3 xor b6 xor b7
b4 = b5 xor b6 xor b7

Receiving side:
q1 = I1 xor I3 xor I5 xor I7
q2 = I2 xor I3 xor I6 xor I7
q3 = I4 xor I5 xor I6 xor I7

(q3 q2 q1) = (0 0 0), no error, Bk = Ik for k = 3,5,6,7

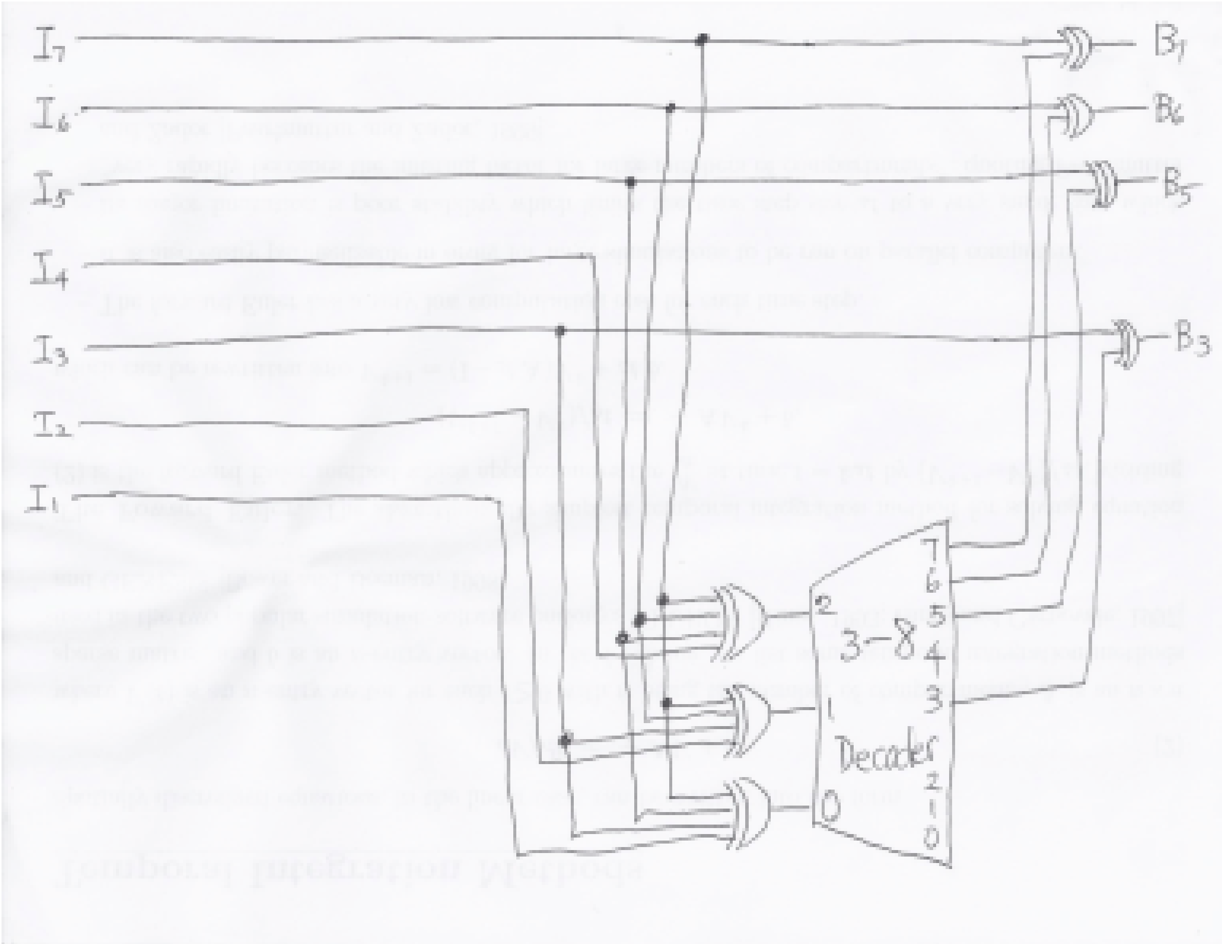
= (0 1 1), error, then b3 = I3’
b5 = I5
b6 = I6
b7 = I7

= (1 0 1), error, then b3 = I3
b5 = I5’
b6 = I6
b7 = I7

= (1 1 1), error, then b3 = I3
b5 = I5
b6 = I6
b7 = I7’

= (1 1 0), error, then b3 = I3
b5 = I5
b6 = I6’
b7 = I7

magic



I7
I6
I5
I4
I3
I2
I1
B7
B6
B5
B3
3-8
Decoder
2
1
0
7
6
5
4
3
2
1
0