

Review Sheet for Exam 1

ENEE 244-020x

Fall 2026

Base Conversion:

1. Use the polynomial method to convert the number 2102 from ternary (base 3) to decimal.

$$342_{(10)} = 101010110_{(2)}$$

$$.1_{(10)} = .00011_{(2)}$$

2. Use the iterative method to convert the number 342.1 from decimal to binary.

$342/2 = 171 \text{ R } 0$	$21/2 = 10 \text{ R } 1$	$.1 \times 2 = 0 + .2$	$.6 \times 2 = 1 + .2$
$171/2 = 85 \text{ R } 1$	$10/2 = 5 \text{ R } 0$	$.2 \times 2 = 0 + .4$	
$85/2 = 42 \text{ R } 1$	$5/2 = 2 \text{ R } 1$	$.4 \times 2 = 0 + .8$	
$42/2 = 21 \text{ R } 0$	$2/2 = 1 \text{ R } 0$	$.8 \times 2 = 1 + .6$	
	$1/2 = 0 \text{ R } 1$		

3. Use the special base conversion technique to convert the number A5E3 from hexadecimal to binary.

final answer: 101010110.00011

1's and 2's Complement:

For the following problems, assume $\ell = 8$ bits of storage.

1. Use 2's complement to perform the subtraction: $01011010 - 1000111$

$$\begin{array}{r} 1011010 \\ + 10111001 \\ \hline 100010011 \end{array}$$

2. Use 1's complement to perform the subtraction: $01000111 - 10111010$

$$\begin{array}{r} 1000111 \\ + 10100101 \\ \hline 11101100 \end{array}$$

& if there is a carryout = 1's complement it gets added back to the least significant bit.

3. Give an example where overflow occurs in 2's complement during (a) addition and (b) subtraction.

$[-128, 127]$

(a) $01111111 + 00000001$

(b) $10000000 - 00000001$

10000000 01111111

Error Correcting Codes:

$$x_1 \dots x_n \quad y_1 \dots y_n$$

$$\sum_{i=1}^n x_i y_i \pmod{2}$$

Consider the following parity check matrix for the Hamming Code:

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

$$\rightarrow \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

1. Encode message $\vec{m} = 1001$

Codeword: 1001001

2. Apply error correction and decode $\vec{s} = 0101101$

Corrected codeword: 0100101

message: 0100

Boolean Algebra:

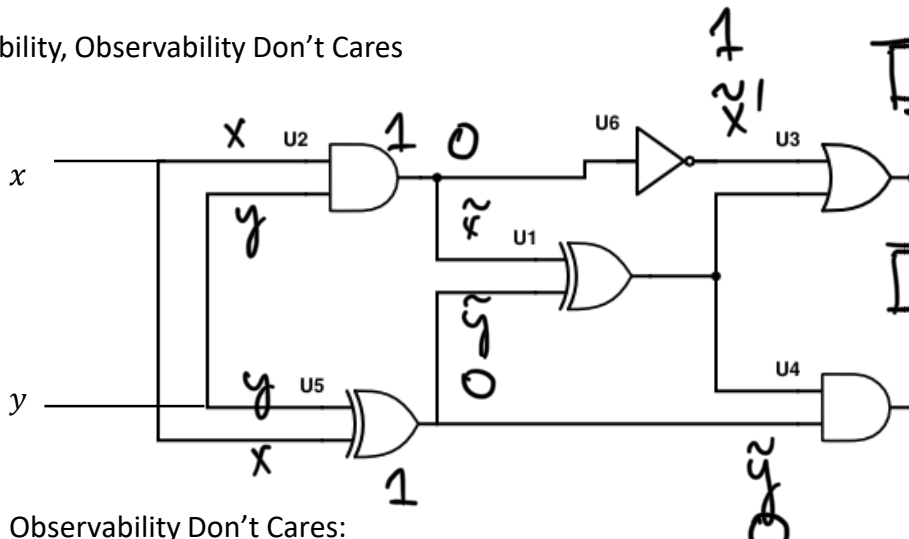
1. Prove: $y'z' + x'z + xy = x'y' + yz + xz'$

$$\begin{aligned} \text{LHS} &= y'z' \cdot 1 + x'z \cdot 1 + xy \cdot 1 \quad \text{Identity} \\ &= y'z'(x+x') + x'z(y+y') + xy(z+z') \quad \text{Complement} \\ &= \underline{y'z'x} + \underline{y'z'x'} + \underline{x'zy} + \underline{x'zy'} + \underline{xyz} + \underline{xyz'} \quad \text{Distributive} \end{aligned}$$

$$\begin{aligned} &\underline{x'y'z'} + \underline{x'y'z} + \underline{x'yz} + \underline{xyz} + \underline{y'xz'} + \underline{y'xz} \quad \text{Commutative} \\ &x'y'(z'+z) + (x'+x)yz + (y'+y)xz' \quad \text{Distributive} \\ &x'y' \cdot 1 + 1 \cdot yz + 1 \cdot xz' \quad \text{Complement} \\ &= \text{RHS} \quad \text{Identity} \end{aligned}$$

3. Prove: $(x+z)(y+z)(y'+z') = y'z + xyz'$

Satisfiability, Observability Don't Cares



$\tilde{x}$	$\tilde{y}$	XOR	
0	0	0	1
0	1	1	
1	0	1	
1	1	0	1

1. Observability Don't Cares:
Due to observability don't cares, Gate U1 can be replaced with which type of gate (e.g. AND, OR, NAND, NOR, XNOR)? Justify your answer.

NAND

NAND

$\tilde{x}$	$\tilde{y}$	~~XOR~~	
0	0	~~0~~	1
0	1	1	
1	0	1	
1	1	0	

2. Satisfiability Don't Cares:
Due to satisfiability don't cares, Gate U1 can be replaced with which type of gate (e.g. AND, OR, NAND, NOR, XNOR)? Justify your answer.

OR

Universal Gates

Prove that the 3-input gate $f(x, y, z) = y'z + xy'z' + xz$ is universal.

OR

$\tilde{x}$	$\tilde{y}$	~~XOR~~	
0	0	0	
0	1	1	
1	0	1	
1	1	~~0~~	1

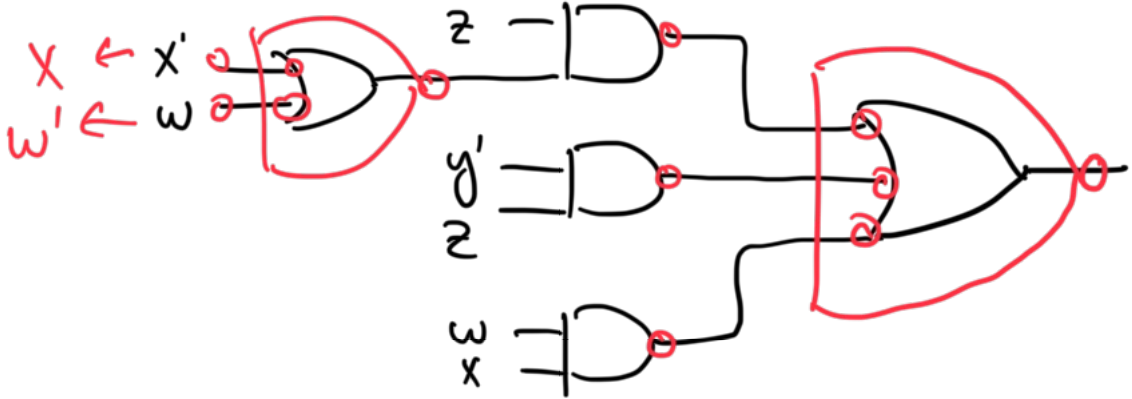
De Morgan's Law:

$$\overline{x + y} = \overline{x} \cdot \overline{y}$$

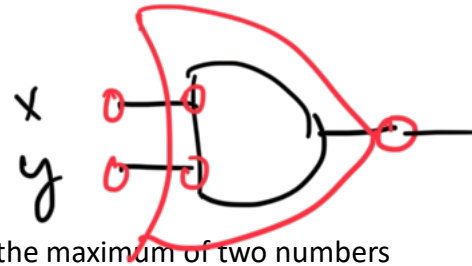
$$\overline{\overline{x + y + z}} = \overline{\overline{x} \cdot \overline{y} \cdot \overline{z}}$$

NAND-Gate Realizations

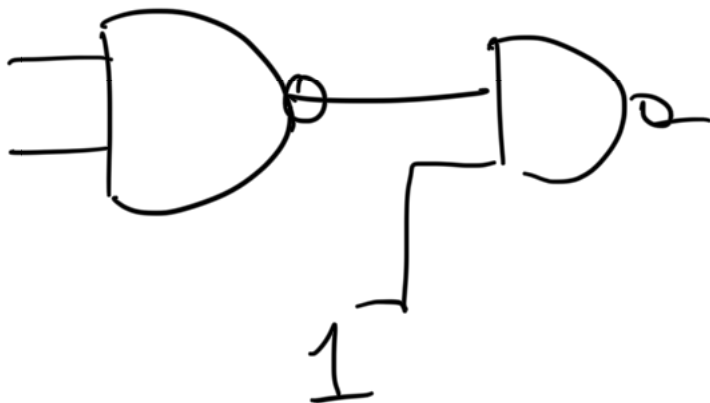
1. Draw the logic diagram with AND and OR gates directly from the following expression:
 $f(w, x, y, z) = [z(x' + w)] + y'z + wx$. You can assume that double-rail logic is used. Then convert the (AND, OR) circuit to one with only NAND gates.



$$\text{NOR} : \overline{x + y} = \overline{x} \cdot \overline{y}$$



Synthesis Problem: You are asked to design a system that outputs the maximum of two numbers x and y , where $x, y \in \{4, 5, 6\}$. Give the system specification at a high level and binary level. For each of the output bits, represent it as a Boolean function in the canonical minterm format (use the m-notation).



K-Maps. Using K-Maps, determine a minimal sum for each of the following Boolean functions:

1. $f(w, x, y, z) = \sum m(1, 4, 6, 8, 9, 10, 11, 12, 13) + dc(3, 15)$

2. $f(v, w, x, y, z) = \sum m(1, 3, 5, 7, 9, 10, 11, 13, 17, 19, 21, 23, 24, 25, 26, 27, 28, 29)$