PAMANTASAN NG LUNGSOD NG MAYNILA
(University of City of Manila)
College of Information Systems and Technology Management

COMPUTATIONAL SCIENCE MIDTERM

BS COMPUTER SCIENCE 3-4

MEMBERS:

Flores, Franchezka

Merino, Krizzia Ydel M.

Monzon, Trisha Ann N.

Ong, Ma. Alexandra M

.

Prof. Joel H. Cruz

April 6, 2024

Detecting Malfunctions in a Regular Traffic Light

A regular traffic light consists of three colors (signals): red, yellow, and green. The red signal indicates to stop before the (stop) line. The yellow signal shows that the signal is changing. While the green signal means to go. A malfunction in a traffic light happens when two or more signals are on or when there is no signal (color). This paper shows the truth table, boolean expression, and the circuit design determining if a traditional traffic light is malfunctioning.

Truth Table

A, B, and C were used to represent each color in a traffic light, where A represents the color red, B for yellow, and C for green. The possible inputs are 0 and 1, where 0 indicates that the color is off, while 1 indicates that the color is on. The expected output shows when the malfunction happens from the combinations of inputs.

A	B	C	Expected Output	Expression
0	0	0	1	A'B'C'
0	0	1	0	
0	1	0	0	
0	1	1	1	A'BC
1	0	0	0	
1	0	1	1	AB'C
1	1	0	1	ABC'
1	1	1	1	ABC

Initial Truth Table

Input			Output						
A	B	C	AB	BC	AC	A+B+C	(A+B+C)'	(A+B+C)' + AB + BC + AC	Expected Output
0	0	0	0	0	0	0	1	1	1
0	0	1	0	0	0	1	0	0	0
0	1	0	0	0	0	1	0	0	0
0	1	1	0	1	0	1	0	1	1
1	0	0	0	0	0	1	0	0	0
1	0	1	0	0	1	1	0	1	1
1	1	0	1	0	0	1	0	1	1
1	1	1	1	1	1	1	0	1	1

Final Truth Table

Boolean Expression

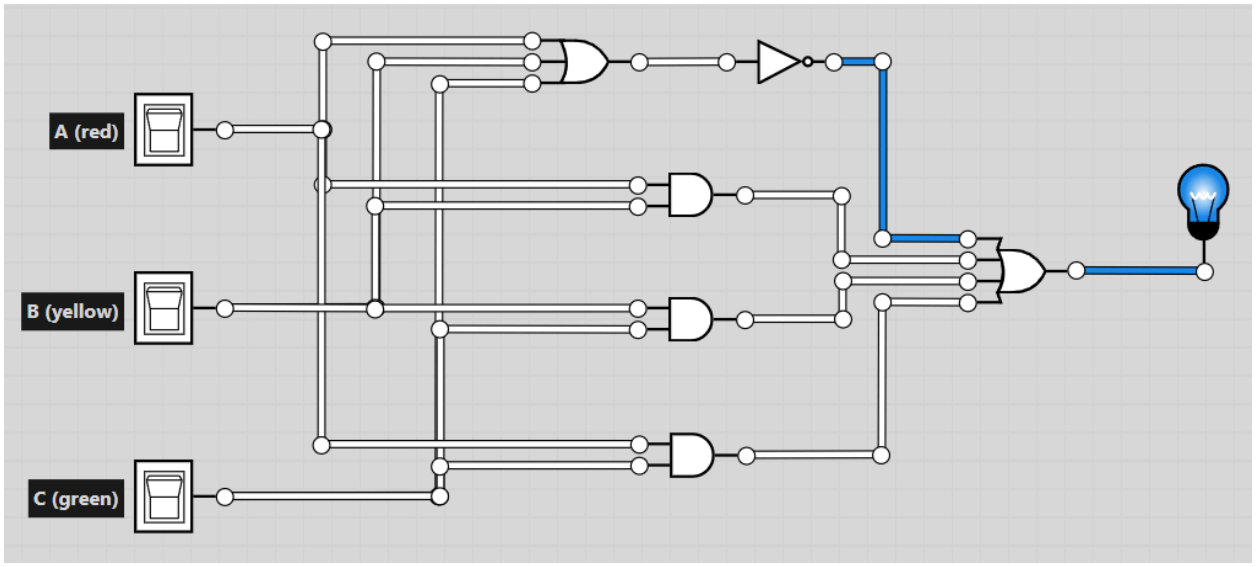
(A+B+C)' + AB + BC + AC

Simplifying Expression

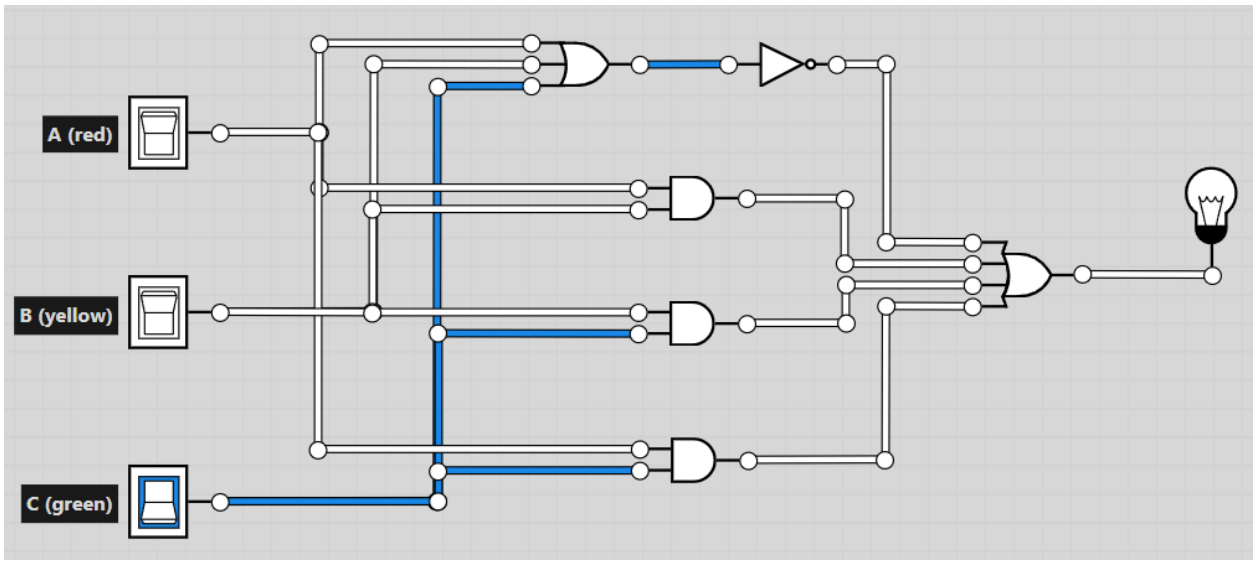
A'B'C' + A'BC + AB'C + ABC' + ABC	
A'B'C' + A'BC + ABC' + AC(B' + B)	Distributive Law
A'B'C' + A'BC + ABC' + AC(1)	Complement Law
A'B'C' + A'BC + ABC' + AC	
A'B'C' + A'BC + A(BC' + C)	Distributive Law
A'B'C' + A'BC + A[(B+C)(C+C')]	**Distributive Law*
A'B'C' + A'BC + A[(B+C)(1)]	**Complement Law*
A'B'C' + A'BC + A(B + C)	*Redundancy Law
A'B'C' + A'BC + AB + AC	
A'B'C' + B(A'C + A) + AC	Distributive Law
A'B'C' + B(C + A) + AC	Redundancy Law
A'B'C' + BC + AB + AC	
(A+B+C)' + BC + AB + AC	De Morgan's Law *(to avoid inverting the inputs, and limits the use of NOT gate to 1)*

Circuit Design and Testing

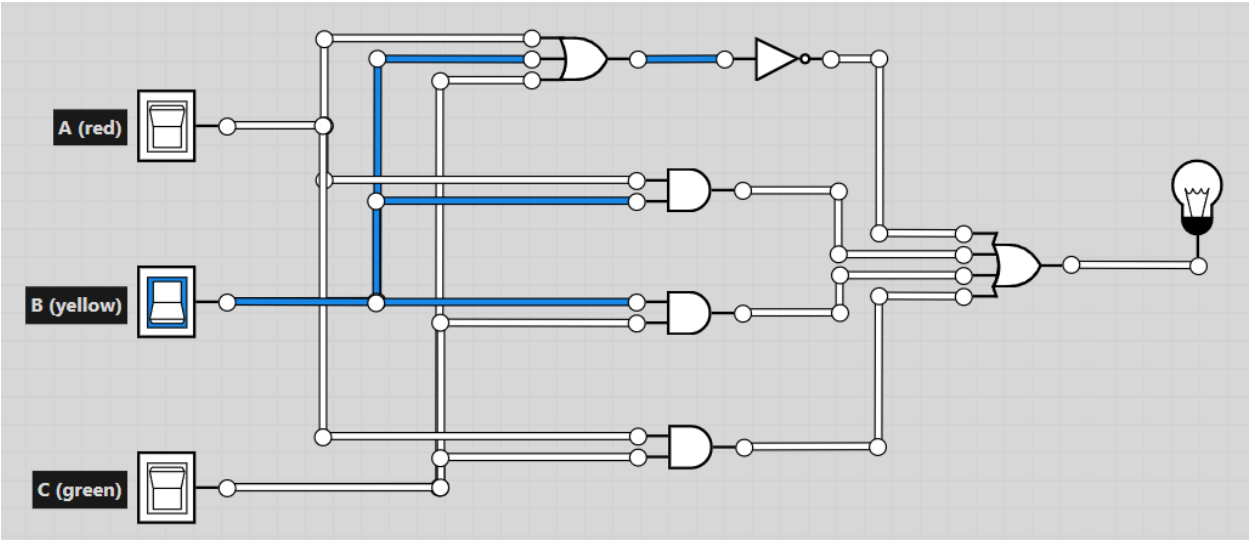
The following pictures show the circuit design and the output from various input combinations.



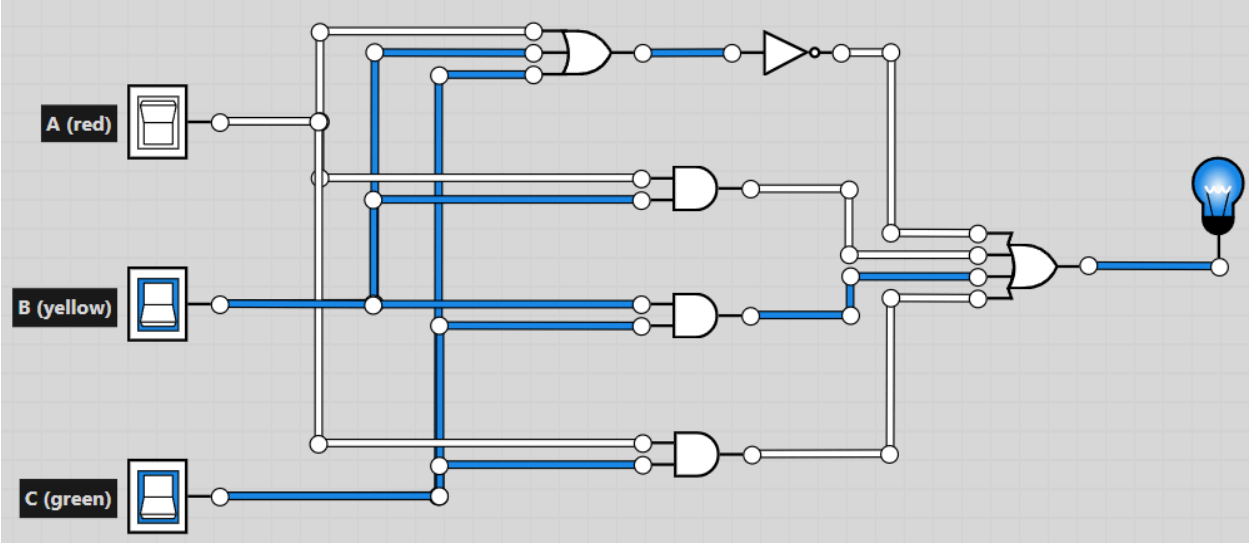
Input (000)



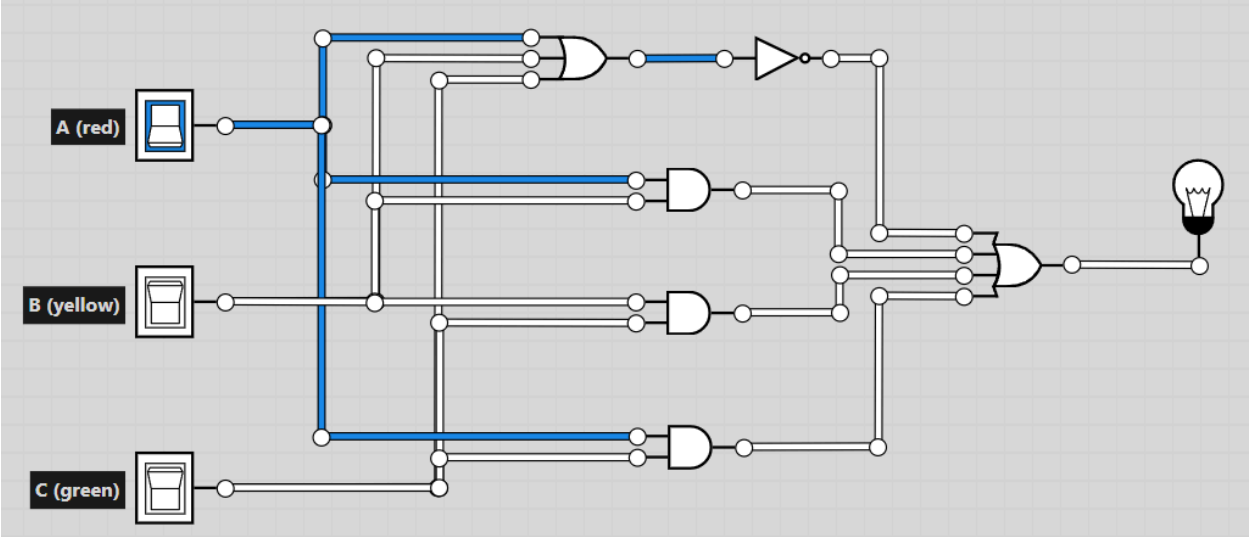
Input (001)



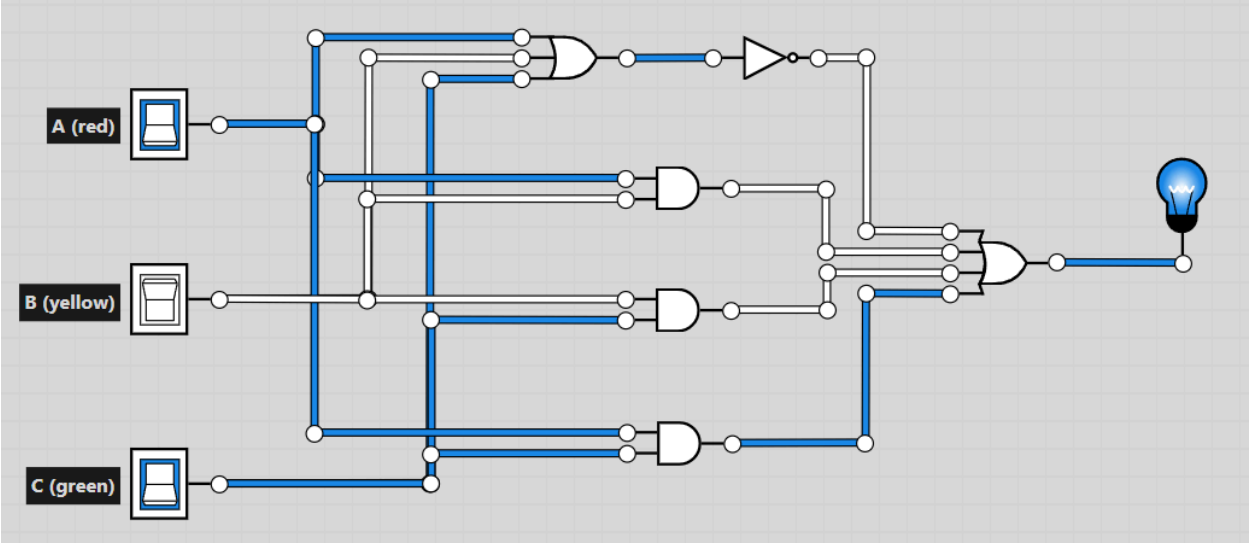
Input (010)



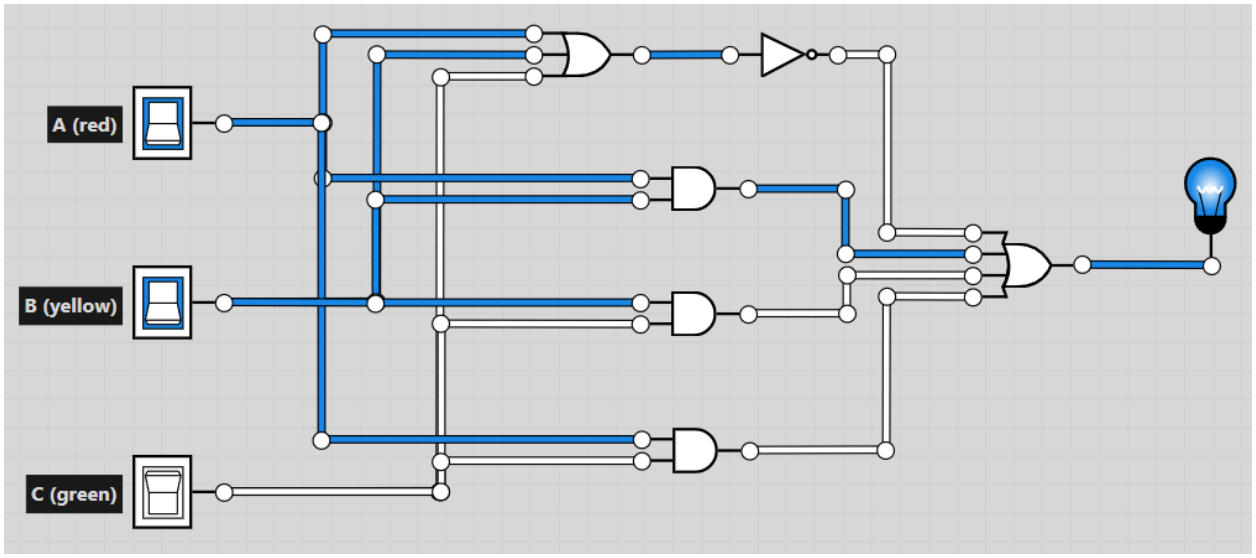
Input (011)



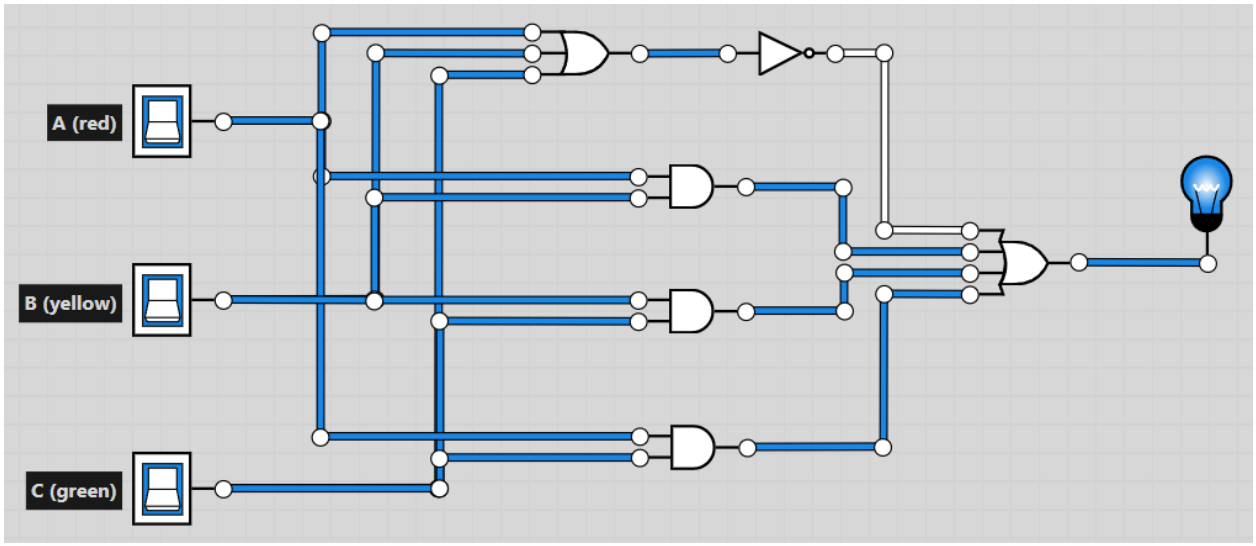
Input (100)



Input (101)



Input (110)



Input (111)