

Lab 1: Logic Gate

**ITI 1100A- Digital Systems
Winter 2024**

**School of Electrical Engineering and Computer Science
University of Ottawa**

Course Coordinator: Dr. Hussein T.Mouftah

Teaching Assistants: Siddhant Tiwari
Emilia Zielinska

Group 29 :

Layth Eleissawi 300330109
Sameed Ahmed 300366534

Experiment Date:02/02/2024
Submission Date:12/02/2024

Objectives

- Construct simple logic circuits and experimentally determine the function of the circuits
- Understand and experimentally test combinational logic
- Predict and analyze the output of combinational logic

Equipment

- Quartus sp13.0 64bit
- Altera DE2-115 Cyclone IV E EP4CE115F28C7

Part I – Combinational Logic Circuits Construction:

5.1.1: One-chip logic circuit:

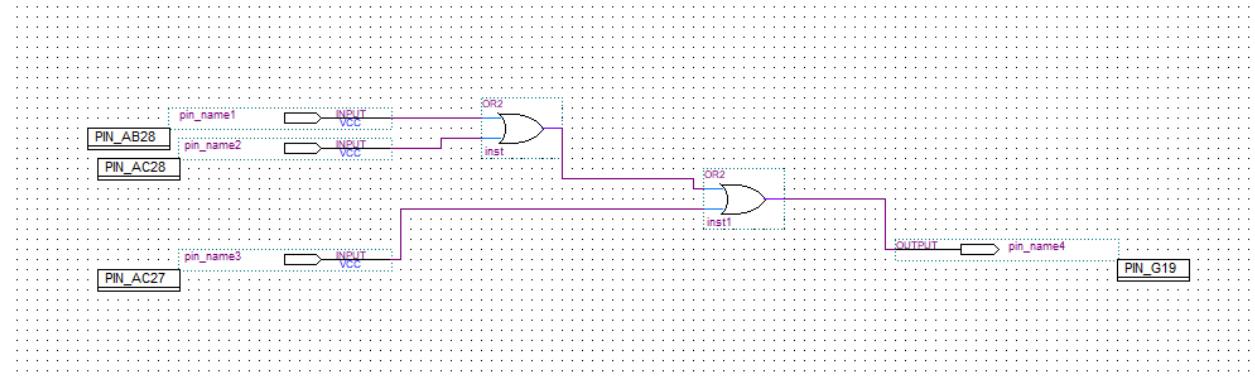


Figure 1: One-chip logic circuit composed of two OR gates.

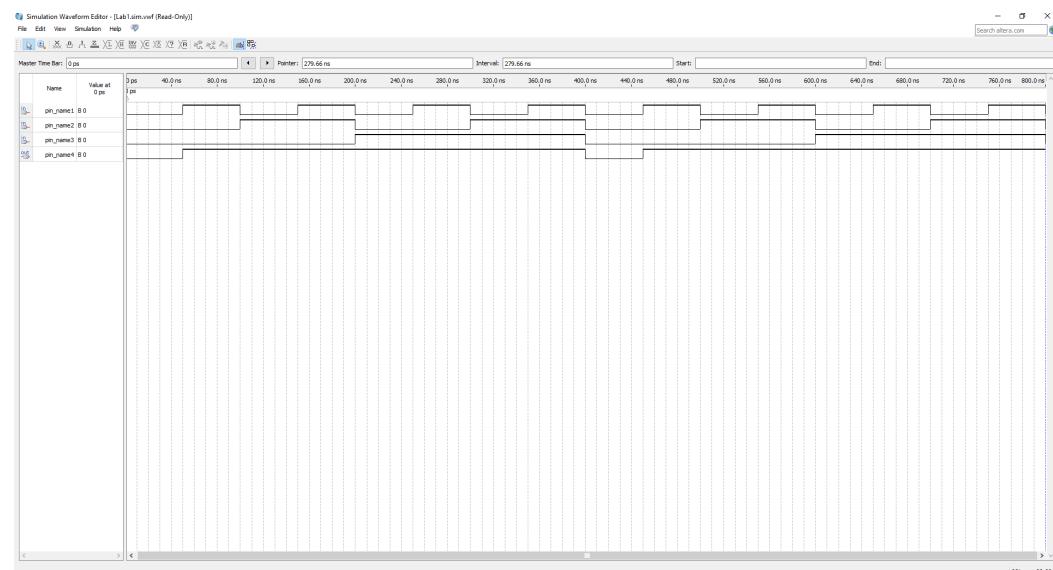


Figure 2: Simulation diagram of a circuit in Figure 1.

Input			Output	
A	B	C	Expected R	Experimental R
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Table 1: Data observed for a One-chip logic circuit

5.1.2: Two-chip logic circuit:

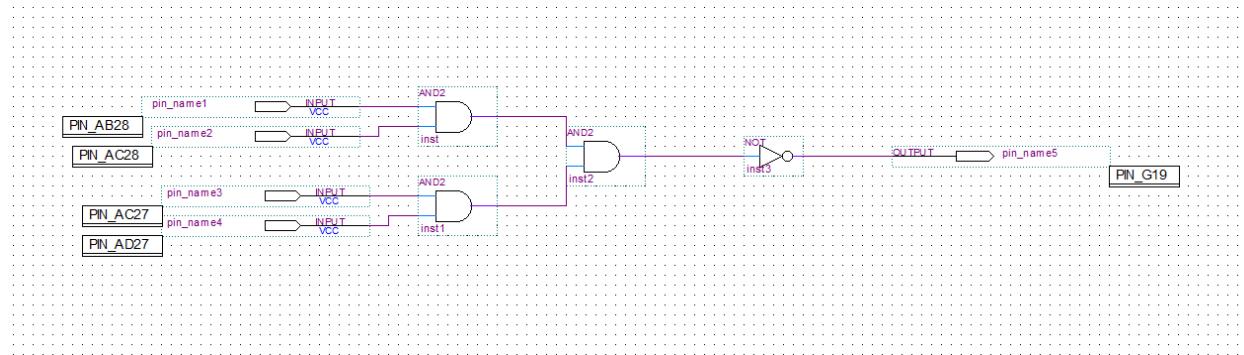


Figure 3 : Two-chip logic circuit composed of three AND gates and one NOT gate.

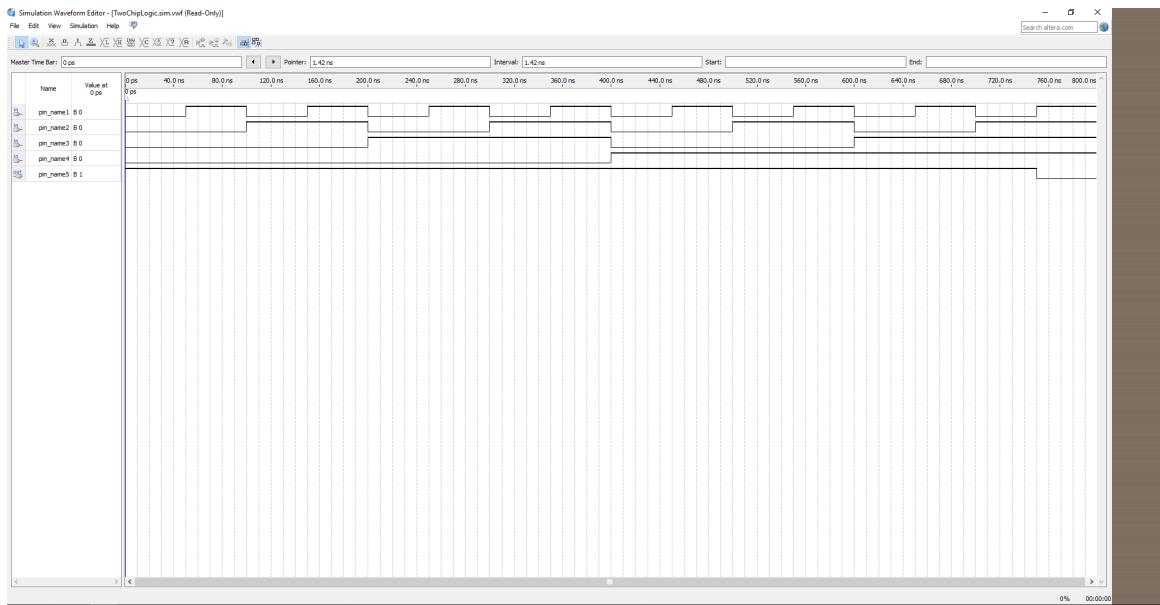


Figure 4 : Simulation circuit of figure 3.

Input				Output	
A	B	C	D	Expected	Actual
0	0	0	0	1	1
0	0	0	1	1	1
0	0	1	0	1	1
0	0	1	1	1	1
0	1	0	0	1	1
0	1	0	1	1	1
0	1	1	0	1	1
0	1	1	1	1	1
1	0	0	0	1	1
1	0	0	1	1	1
1	0	1	0	1	1
1	0	1	1	1	1
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	1
1	1	1	1	0	0

Table 2 : Data observed for a Two-chip logic circuit.

5.1.3: Three-chip logic circuit

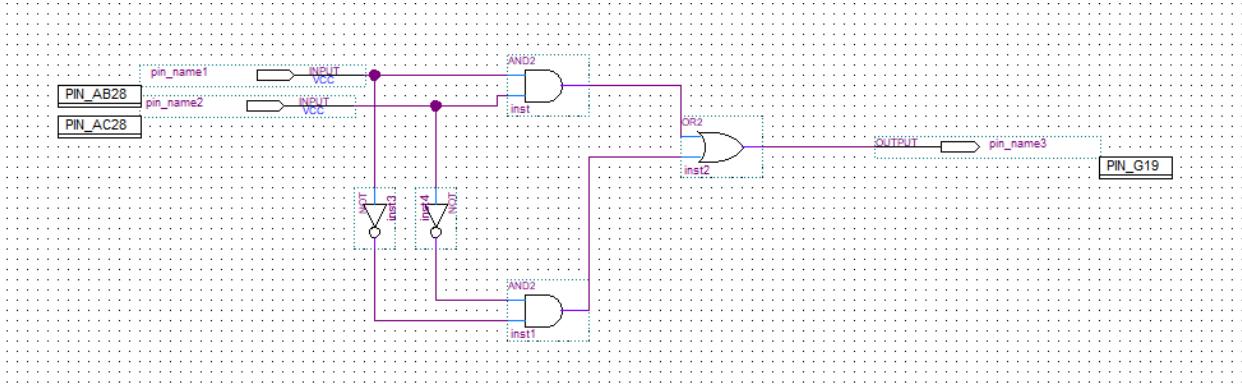


Figure 5: Three-chip logic circuit composed of two NOT gates , two AND gates and one OR gate .

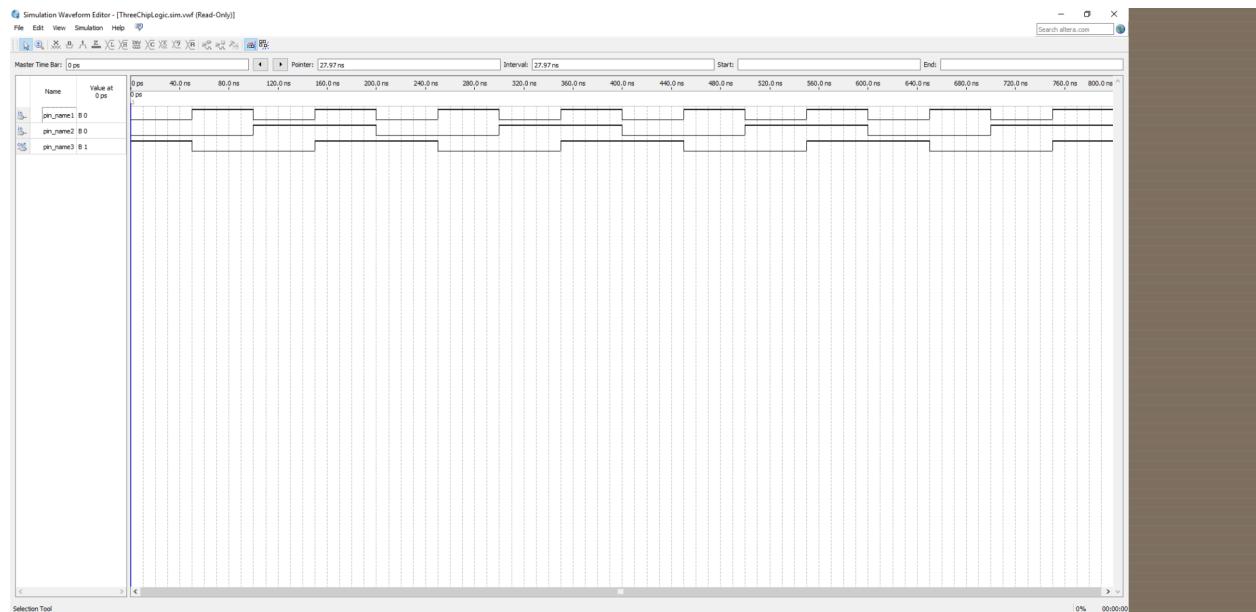


Figure 6: Simulation diagram of figure 5.

Input		Output	
A	B	Expected	Actual
0	0	1	1
0	1	0	0
1	0	0	0
1	1	1	1

Table 3 : Data observed for a Three-chip logic circuit.

Part II - Combinational Logic Circuits Analysis:

5.1.5: Exclusive OR circuit:

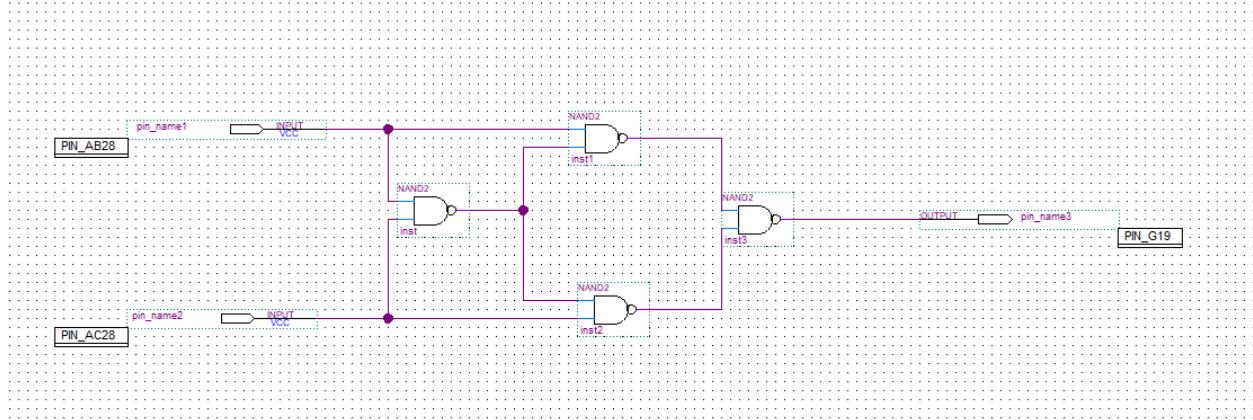


Figure 7: Exclusive OR circuit composed of four NAND gates.

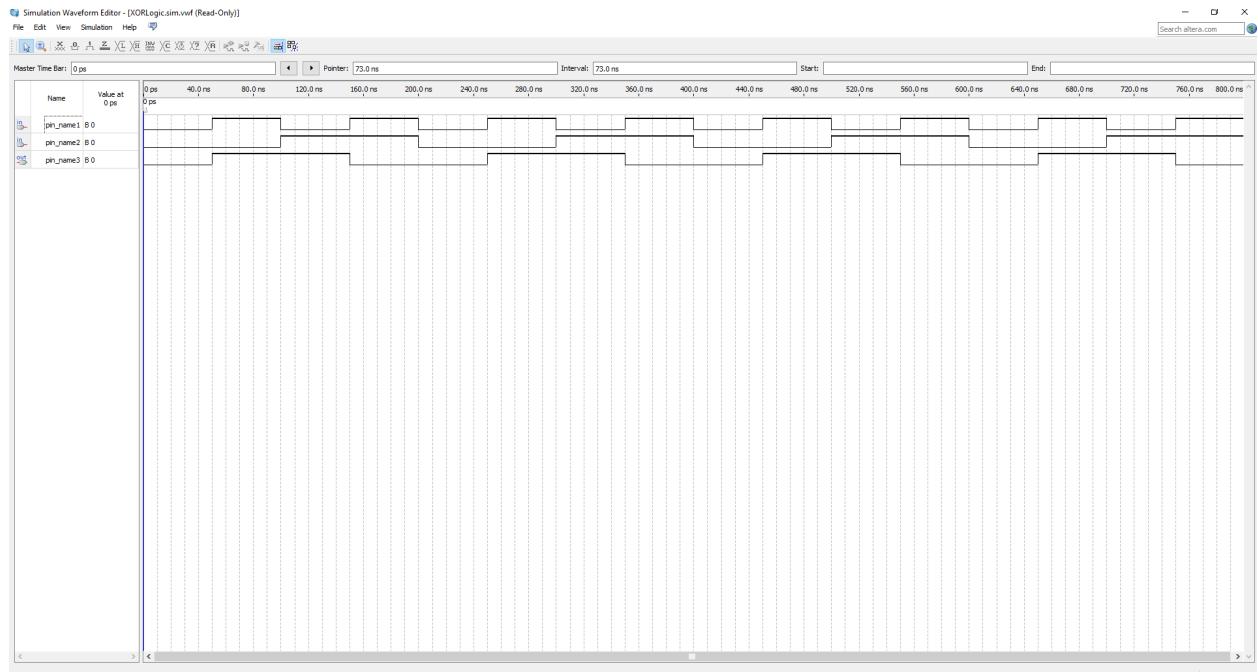


Figure 8: Simulation diagram of circuit in figure 7.

Input		Output	
A	B	Expected L	Experimental L
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

Table 4 : Data observed for an Exclusive OR circuit.

5.1.6: AND Circuit

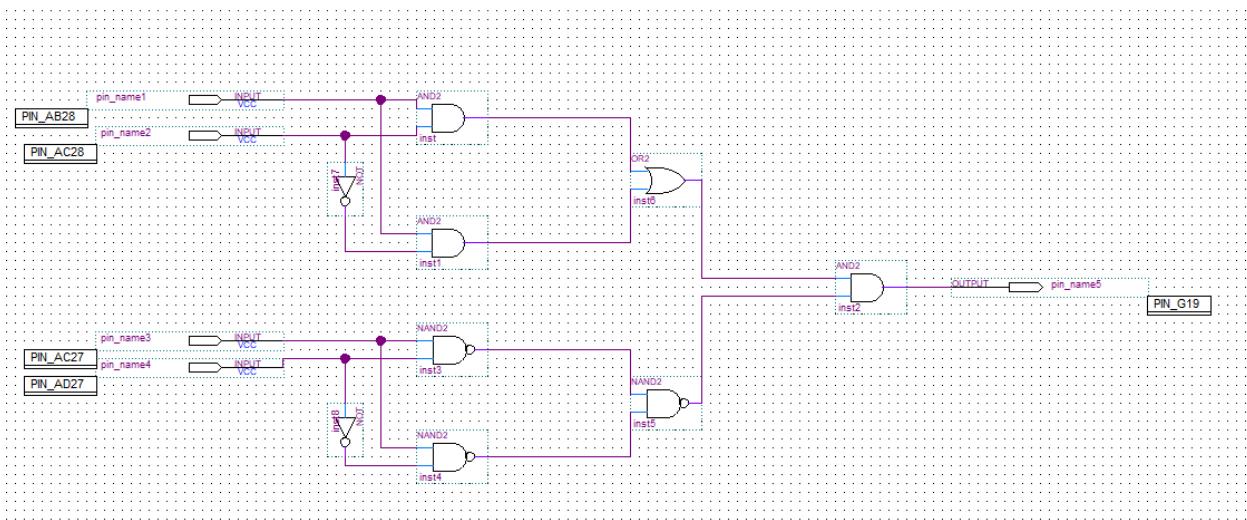


Figure 9 : AND circuit composed of two NOT gates , three NAND gates , three AND gates and one OR gate.

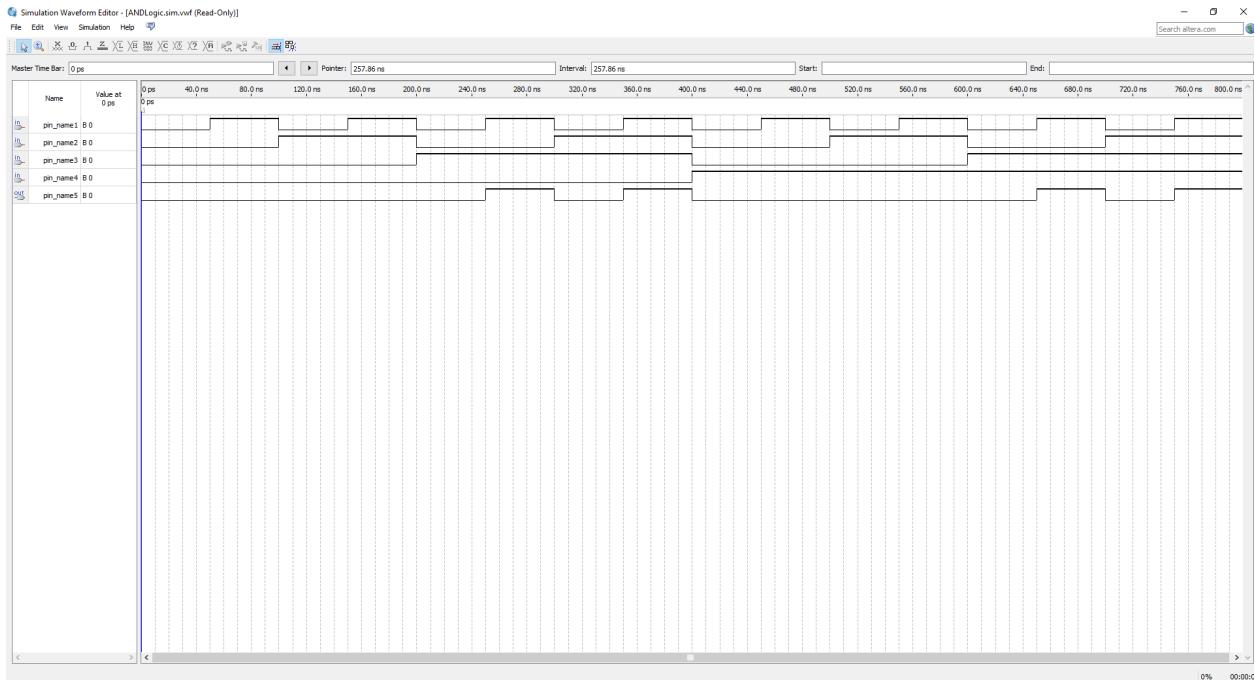


Figure 10: Simulation diagram of circuit in figure 9.

Input				Output	
A	B	C	D	Expected	Actual
0	0	0	0	0	0
0	0	0	0	1	0
0	0	1	0	0	0
0	0	1	1	1	1
0	1	0	0	0	0
0	1	0	1	0	0
0	1	1	0	0	0
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	1	0	0
1	0	1	0	0	0
1	0	1	1	1	1
1	1	0	0	0	0
1	1	0	1	0	0
1	1	1	0	0	0
1	1	1	1	1	1

Ex

Table 5 : Data observed for an AND circuit.

5.1.7: OR Circuit

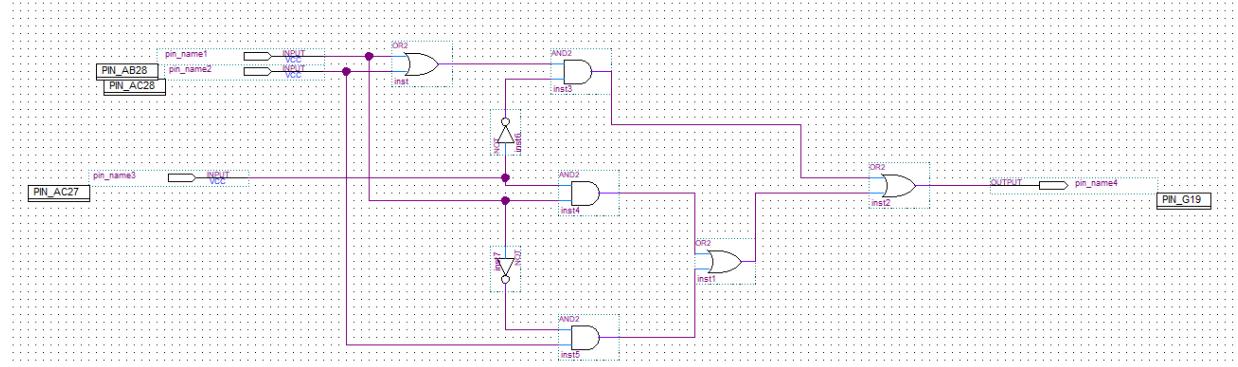


Figure 11 : OR circuit composed of three OR gates , three AND gates and two NOT gates .

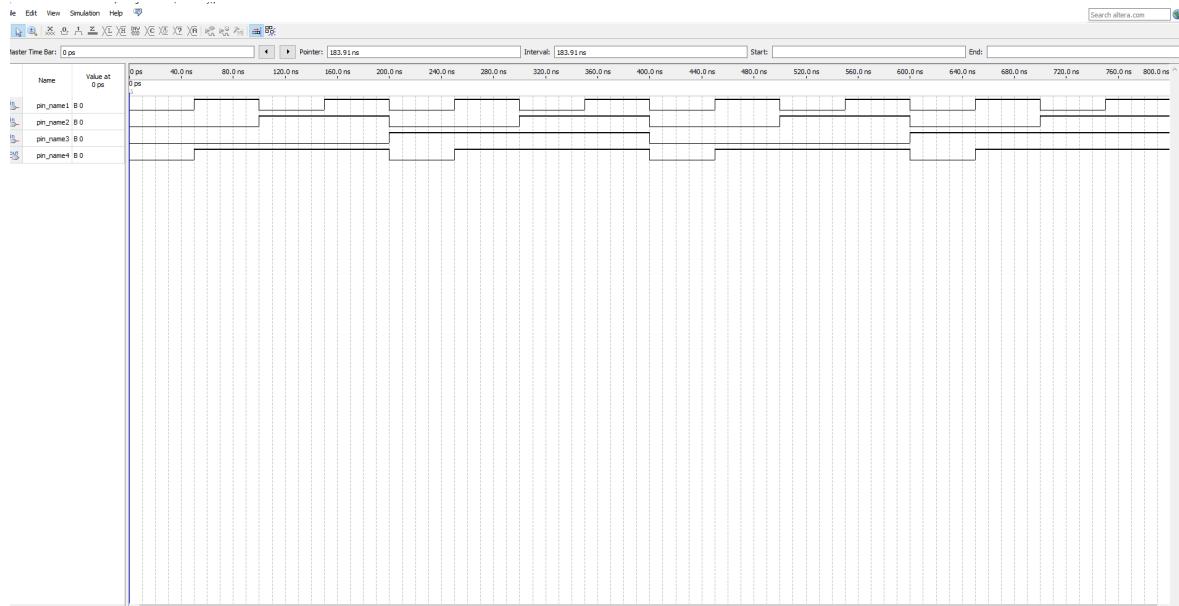


Figure 12: Simulation diagram of circuit in figure 11.

A	B	C	Output	
			Expected	Actual
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Table 6 : Data observed for OR circuit.

5.1.8: Multiple Output Circuit:

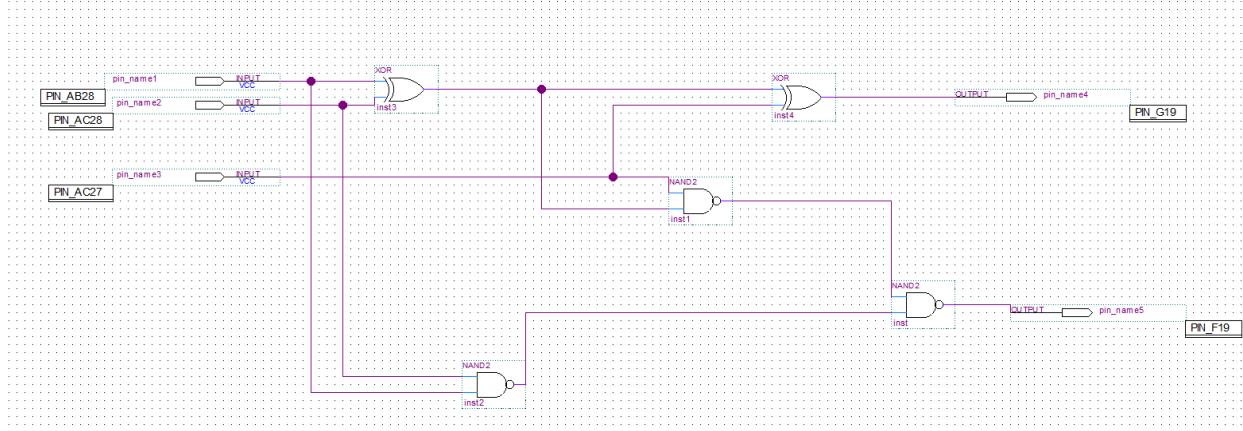


Figure 13: Multiple output circuit composed of three NAND gates and two XOR gates.

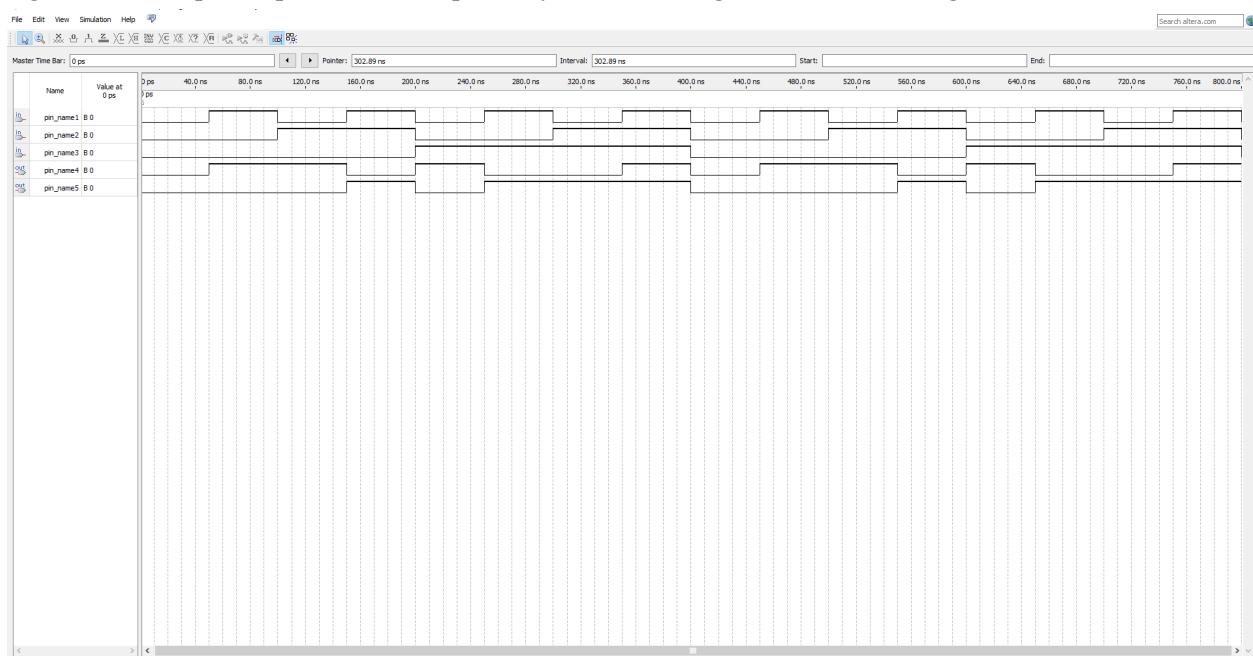


Figure 14 :Simulation diagram of circuit in figure 13.

Input			Output			
A	B	C	Expected C ₀	Expected S	Experimental C ₀	Experimental S
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	1	0	1
0	1	1	1	0	1	0
1	0	0	0	1	0	1
1	0	1	1	0	1	0
1	1	0	1	0	1	0
1	1	1	1	1	1	1

Table 7 : Data observed for Multiple output circuit.

Discussion & Conclusion

Discussion

During this lab, our main aim was to get a better understanding of how digital systems function, both in terms of their software and hardware aspects. To do this, we used Quartus II 13.0 software to create different logic circuits according to the instructions in our lab manual. After setting up the circuits, we ran simulations to see what results we'd get for different combinations of inputs - whether they were true or false. Our next step was comparing these simulation results with the truth tables we made based on our understanding of Boolean expressions and the circuit diagrams. If our simulation results matched the truth tables, it meant we had successfully implemented the logic expressions correctly. But if there were discrepancies, it indicated potential mistakes in our design or understanding. Throughout the lab, we were able to verify essential logical laws like De Morgan's laws and Boolean algebra. By matching expected and experimental data, we confirmed the functionality of our circuits and achieved our main objectives of understanding digital systems better.

However, we did encounter some challenges along the way. For instance, there were times when our simulation results didn't align with what we expected. To address this, we had to carefully review our circuit designs or double-check our input configurations. Through troubleshooting and refining our approaches, we were able to overcome these issues and ensure the reliability of our experimental outcomes.

Conclusion

The lab detailed multiple different kinds of logic combination circuits, and how certain logic operands can be built through a combination circuit, such as Exclusive OR Logic. The lab also highlighted multi-output logic, where different streams would return different values, essentially outputting multiple different truth values simultaneously. Overall, the experiment was performed without any errors.

Appendix (Pre-lab)

5.1.1					
A	B	C	A+B	R: (A+B)+C	
1	1	1	1	1	1
1	1	0	1	1	1
1	0	1	1	1	1
1	0	0	1	1	1
0	1	1	1	1	1
0	1	0	1	1	1
0	0	1	0	1	
0	0	0	0	0	

5.1.2

A	B	C	D	A^*B	C^*D	$(A^*B)^*(A^*C)$	$U = [(A^*B)^*(A^*C)]'$
1	1	1	1	1	1	1	0
1	1	1	0	1	0	0	1
1	1	0	1	1	0	0	1
1	1	0	0	1	0	0	1
1	0	1	1	0	1	0	1
1	0	1	0	0	0	0	1
1	0	0	1	0	0	0	1
1	0	0	0	0	0	0	1
0	1	1	1	0	1	0	1
0	1	1	0	0	0	0	1
0	1	0	1	0	0	0	1
0	1	0	0	0	0	0	1
0	0	1	1	0	1	0	1
0	0	1	0	0	0	0	1
0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	1

5.1.3

A	B	A^*B	A'	B'	A'^*B'	$K = (A^*B) + (A'^*B')$	$K = (A \oplus B)'$
1	1	1	1	0	0	0	1
1	0	0	0	1	0	0	0
0	1	0	1	0	0	0	0
0	0	0	1	1	1	1	1

5.1.5

A	B	C:		D:		$L = A \oplus B$
		$(A^*B)'$	$(A^*(A^*B))'$	$(B^*(A^*B))'$	$(C^*D)'$	
1	1	0	1	1	0	
1	0	1	0	1	1	
0	1	1	1	0	1	
0	0	1	1	1	0	

5.1.6

A	B	C	D	E		F		I		G		H		J		$V=C*D$
				$A*C$	A'^*C	$E+F$	$(D*B)'$	$(B'^*D)'$	$(G^*H)'$	I^*J						
1	1	1	1	1	0	1	0	1	0	1	1	0	1	1	1	1
1	1	1	0	0	1	0	1	1	1	1	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	1	1	1	1	0	0	0
1	1	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
1	0	1	1	1	0	1	1	1	0	0	1	0	1	1	1	1
1	0	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0
1	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
0	1	1	0	1	1	1	1	1	1	1	1	0	0	0	0	0
0	1	0	1	0	0	0	0	0	0	0	1	1	1	1	0	0
0	1	0	0	0	0	0	0	0	1	1	1	0	1	1	0	0
0	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
0	0	0	1	0	0	0	0	0	1	0	1	0	1	0	1	0
0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0

5.1.7

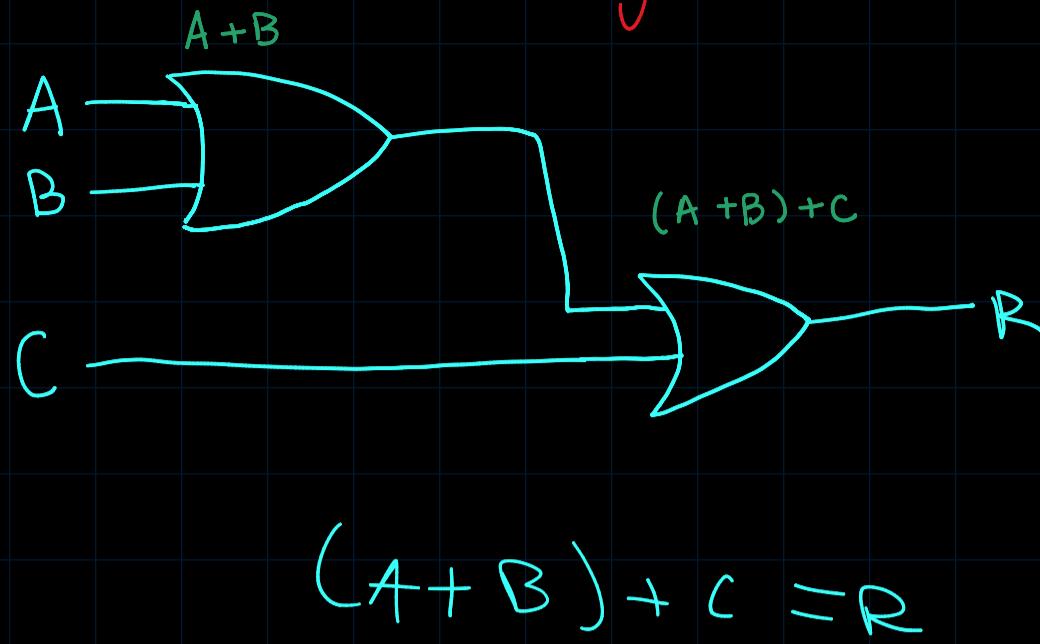
A	B	C	D	E	F	G	X	Y	$P=A+C$	
			$(A+C)$	(B'^*D)	$A*B$	A'^*C	$F+G$	D^*E		
1	1	1	1	1	0	1	0	1	0	1
1	1	0	1	0	1	0	1	0	1	1
1	0	1	1	1	0	0	0	1	1	1
1	0	0	1	1	0	0	0	1	1	1
0	1	1	1	0	0	1	1	0	1	1
0	1	0	0	0	0	0	0	0	0	0
0	0	1	1	1	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0

5.1.8

A	B	C	D	E	F	$S=C \oplus D$	$Co=(E^*F)'$
			$A \oplus B$	$(A^*B)'$	$(C^*D)'$		
1	1	1	1	0	0	1	1
1	1	0	0	0	1	0	1
1	0	1	1	1	0	0	1
1	0	0	1	1	1	1	0
0	1	1	1	1	1	0	1
0	1	0	1	1	1	1	0
0	0	1	0	1	1	1	0
0	0	0	0	0	1	1	0

5.1.1: One-Chip Logic

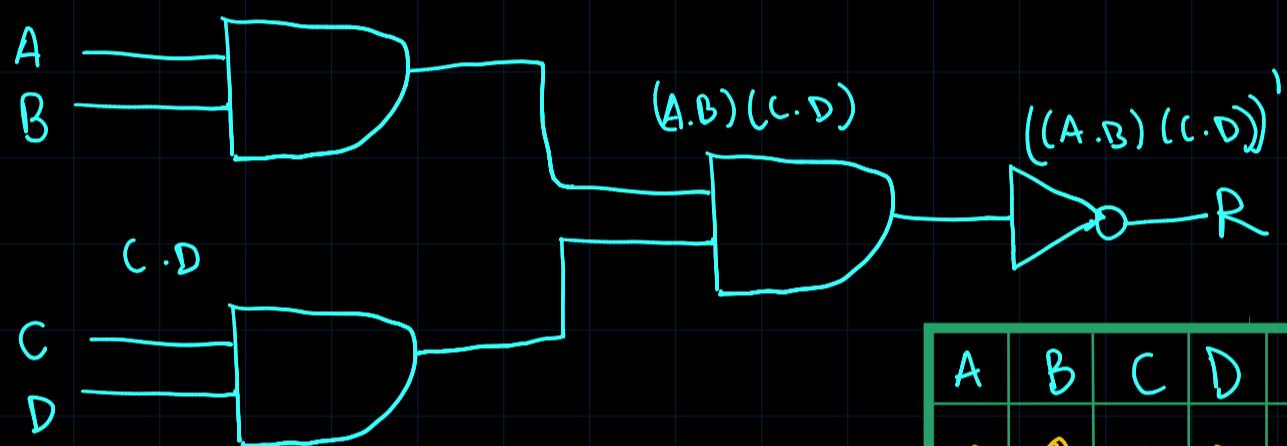
PRELAB #1



A	B	C	$A+B$	$(A+B)+C$
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

5.1.2 Two Chip logic

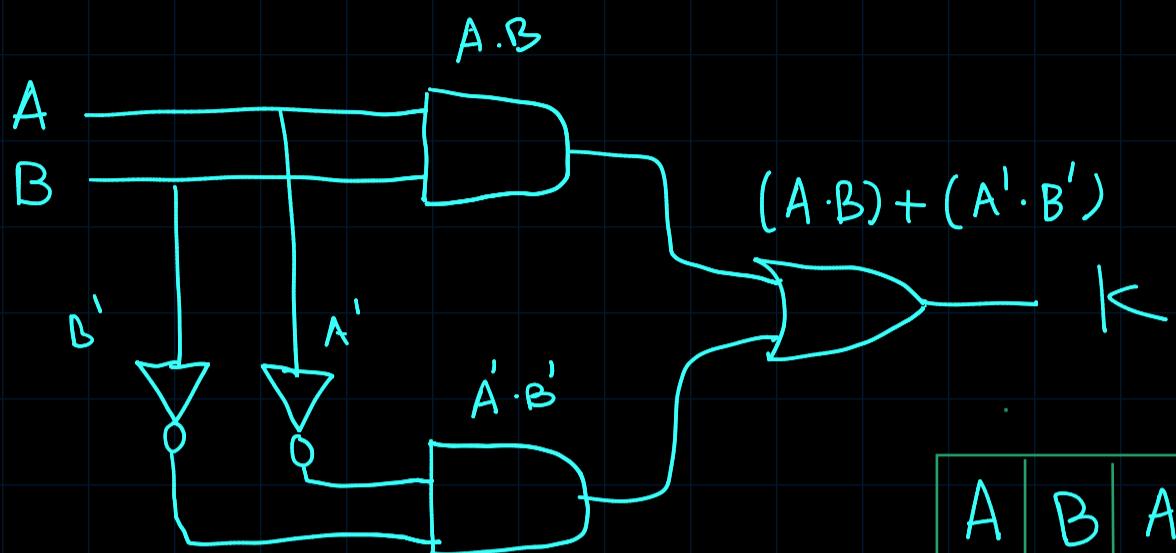
A.B



$$((A \cdot B) \cdot (C \cdot D))' = R$$

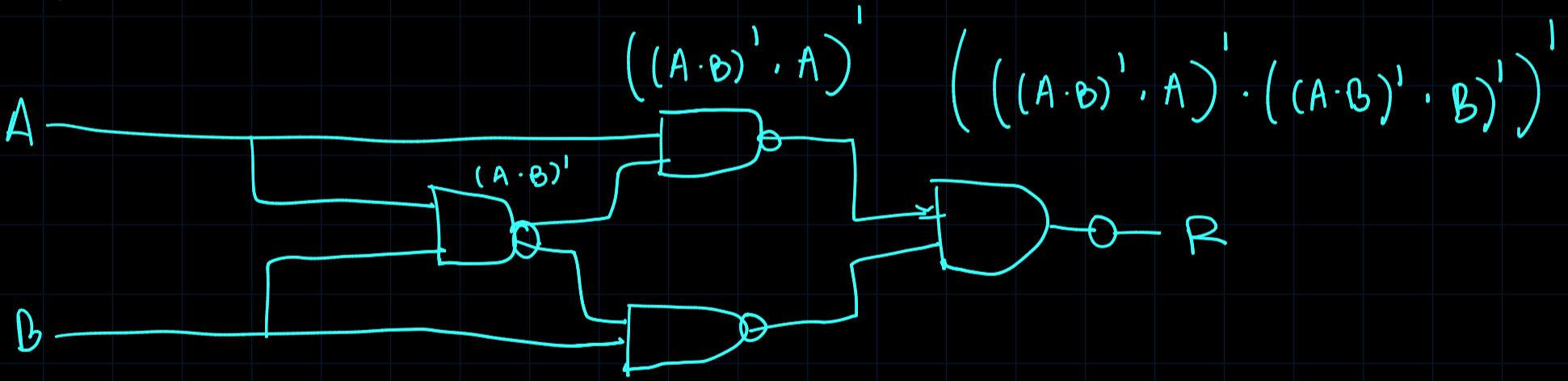
A	B	C	D	A.B	C.D	$(A \cdot B) \cdot (C \cdot D)$	$((A \cdot B) \cdot (C \cdot D))'$
0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1
0	0	1	0	0	0	0	1
0	0	1	1	0	0	0	1
0	1	0	0	0	0	0	1
0	1	0	1	0	0	0	1
0	1	1	0	0	0	0	1
0	1	1	1	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	1	0	0	0	1
1	0	1	0	0	0	0	1
1	0	1	1	0	0	0	1
1	1	0	0	1	0	0	1
1	1	0	1	1	0	0	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	1

5.1.3 Three-Chip Logic



A	B	A'	B'	$A \cdot B$	$A' \cdot B'$	$(A \cdot B) + (A' \cdot B')$
0	0	1	1	0	0	1
0	1	1	0	0	0	0
1	0	0	1	0	0	0
1	1	0	0	1	0	1

5.1.5 Ex. OR

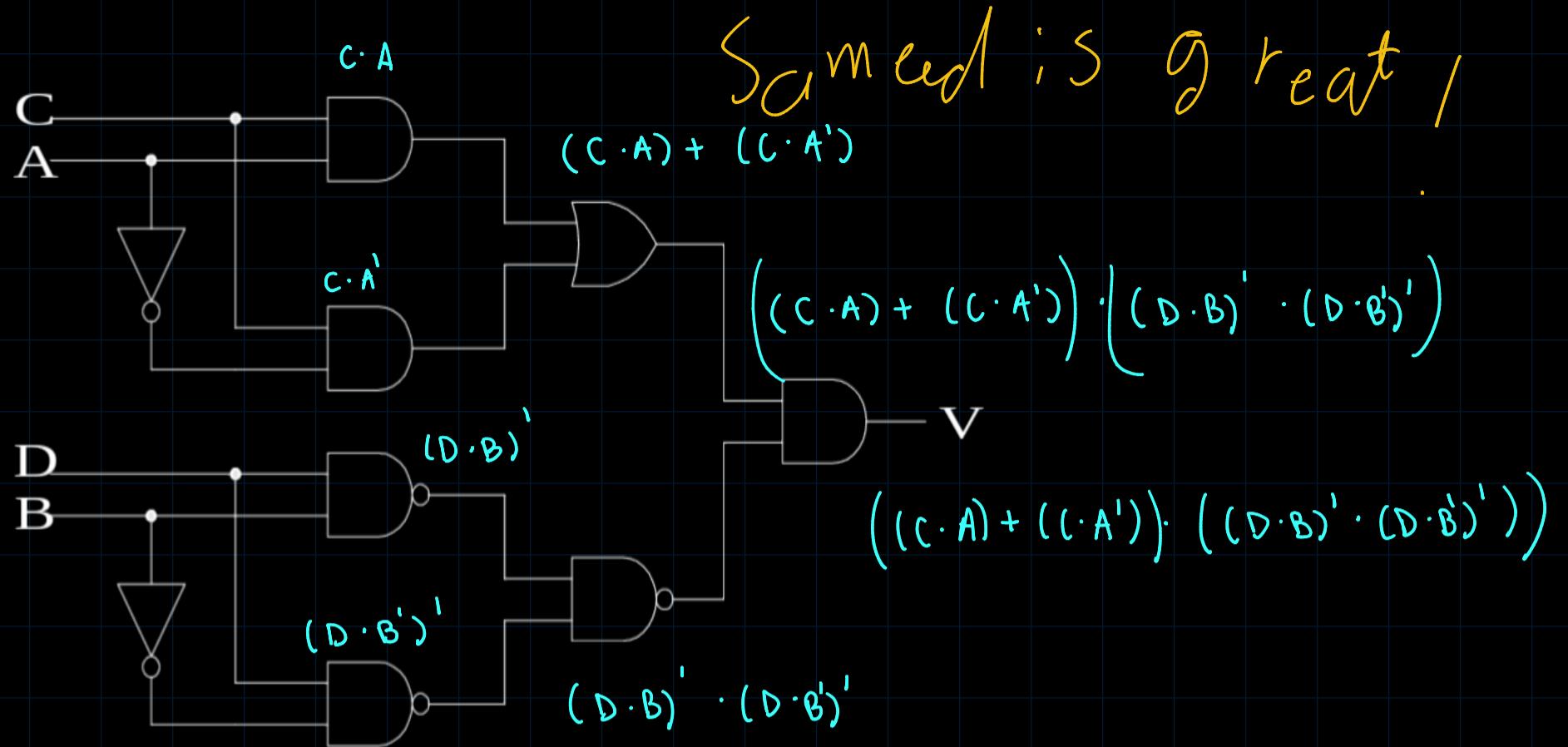


$$((A \cdot B)' \cdot A)' \cdot ((A \cdot B)' \cdot A)' = R$$

A	B	$(A \cdot B)'$	$((A \cdot B)' \cdot A)'$	$((A \cdot B)' \cdot B)'$	R
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	0

U

5.1.6 AND Circuit



C	A	D	B	$C \cdot A$	$C \cdot A'$	$(D \cdot B)'$	$(D \cdot B')'$	$(C \cdot A) + (C \cdot A')$	$(D \cdot B)' \cdot (D \cdot B')'$	R
0	0	0	0	0	0	1	1	0	0	0
0	0	0	1	0	0	1	1	0	0	0
0	0	1	0	0	0	1	0	0	1	0
0	0	1	1	0	0	0	1	0	1	1
0	1	0	0	0	1	0	1	0	0	0
0	1	0	1	0	0	1	1	0	0	1
0	1	1	0	0	0	1	0	0	0	0
0	1	1	1	0	0	0	1	0	0	0
1	0	0	0	0	1	1	1	1	1	0
1	0	0	1	0	1	1	1	1	1	1
1	0	1	0	0	1	1	0	1	1	0
1	0	1	1	0	1	0	1	1	1	0
1	1	0	0	1	0	1	1	1	1	1
1	1	0	1	1	0	1	1	1	1	1
1	1	1	0	1	0	0	1	1	1	1
1	1	1	1	1	0	0	0	1	1	1

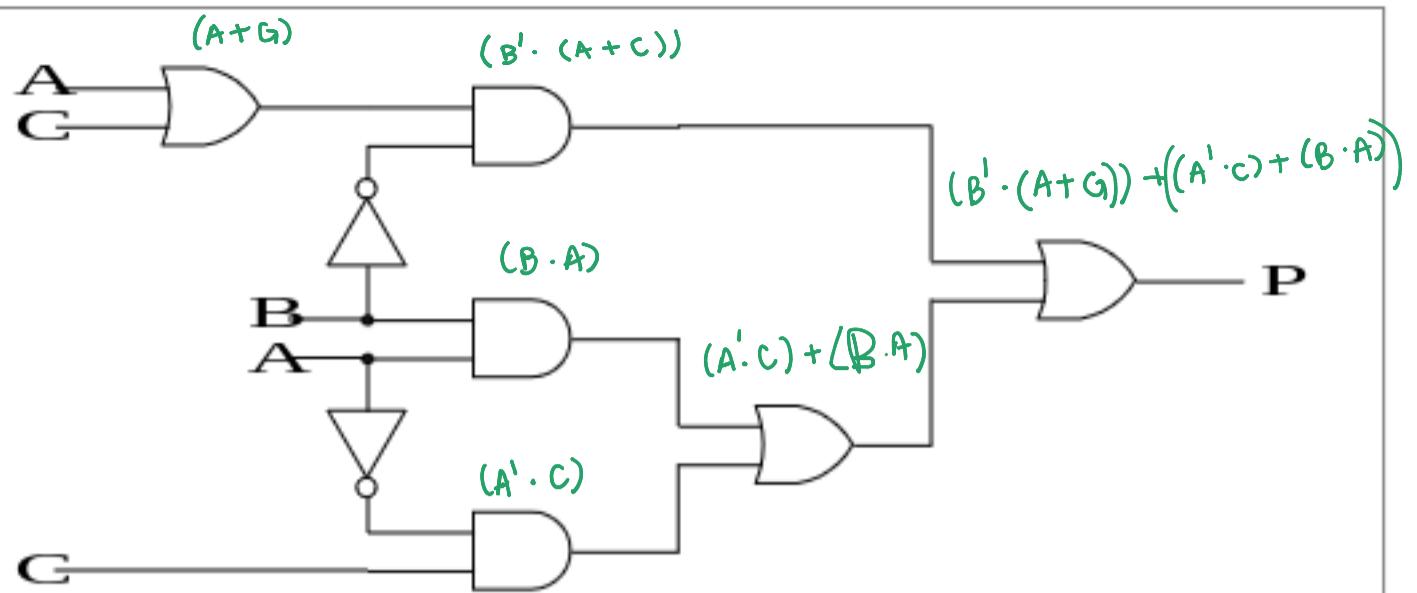


Figure 5.1.7: OR circuit

5.1.7 OR Circuit								
A	B	C	(A+C)	(B.A)	(A'.C)	(B'.(A+C))	((A'.C)+(B.A))	((B'.(A+C)) + ((A'.C)+(C.A))) = R
0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	1	1
0	1	0	1	0	0	0	0	0
0	1	1	1	1	1	0	1	1
1	0	0	1	0	0	1	0	1
1	0	1	1	0	0	1	0	1
1	1	0	1	0	0	0	0	0
1	1	1	1	1	0	0	1	1

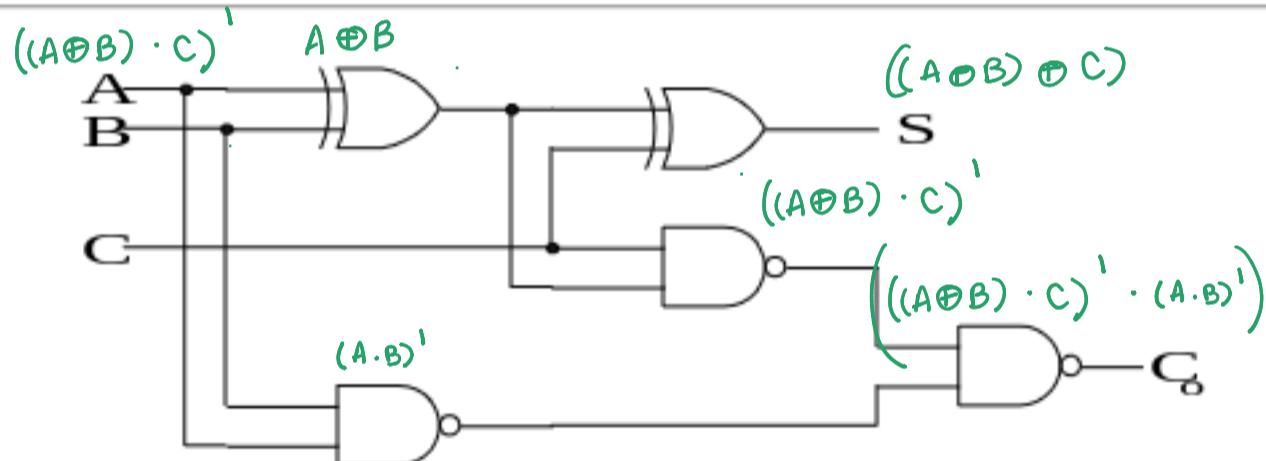


Figure 5.1.8: Multiple output circuit

5.1.8 Multiple Output Circuit			$(((A \oplus B).C)'.(A.B)') = C_0$			$((A \oplus B) \oplus C) = S$		
A	B	C	(A⊕B)	(A.B)'	((A⊕B).C)'	$(((A \oplus B).C)'.(A.B)')$	$((A \oplus B) \oplus C)$	
0	0	0	0	1	1	1	0	
0	0	1	0	1	1	0	0	
0	1	0	1	1	1	1	0	
0	1	1	1	1	0	1	1	
1	0	0	1	1	1	1	0	
1	0	1	1	1	0	1	1	
1	1	0	0	0	1	1	1	
1	1	1	0	0	1	0	1	