CSE260 Lab Report

Experiment Name: Applications of Boolean Algebra

Submitted byShaiane Prema Baroi
19241019
Section 05
Date: 08 November 2020

4>	Name of Experiment: Applications of Boolean Algebra
. A.) 8	(A-A) ((A-B)) ((A-B)) ((A-B)) (A-(A-B)) (A-(A-B))
	Objective
	to investigate the rules of Boolean algebra
	- to gain experience working with practical circuits
	- to simplify a complex function using Boolean algebra
3>	Required components and equipments
	- AT-100 Portable an Analoguel an traminages sidt at
	Cincuit and obtained results accordingly X TOOP Toget
bпл	- logic states were strup and trubbie soltas
	- NANDologic gatesuntanos I nata bastanabau
ej c	- sgroundituat adt stiam of bad I sustange -
(ton)	the output. Even though it seemed assniwle-
	the Dollight shoot notto, never tonit to
9 ao a	touth table into several parts, it became
iean	convenient to solve the trath toble. The Bot
(8 4 A)	Experimental A Setup tugtuo ant ant maissances
n n	If we had to simplify this expression us
~	Boolean theorems, we could do en by-
	12t A. (A. 8) be x
	v ed (8, 67 8 49)
	Y ('U : 'X) :
VA.	General Control of the Control of th

5>	Inuth ATable 008	40	Applications	*	Experiment	40 9mnM
	32		4 4			

ŀ						. 1
	A	В	(A.B)'	(A. (A.B))	(B. (A.B))	(A (A B)') (B (A B))
						vitagidO <c< th=""></c<>
	0	orean Oaigeb	es DF Bo	Ina 1 024		De la constantina della consta
-		ositonala di:	-		Đ	
-		e noisu r			n9ikomi	V
Contract or Appeal Statement	1	U L	0	l	1	0
				and Address		

3) Required components and equipments

In this experiment, we had constructed the given circuit and obtained results accordingly. The togic entire circuit was quite easy to construct and understand. After I constructed the circuit on Proteus, I had to write the truth table for the output. Even though, it seemed a little lengthy at first, however, after breaking down the truth table into several parts, it became more convenient to soive the truth table. The Boolean expression for the output was (A.(A.B)). (B.(A.B)). If we had to simplify this expression using Boolean theorems, we could do so by-

let B. (A.B)' be x

$$\therefore (x', y')' = Y$$

```
=> y = (x', y')'

=> y = (x')' + (y')' [De Morgan's Law]

=> y = x + y [(x')' = x, (y')' = y]

=> y = A \cdot (A \cdot B)' + B \cdot (A \cdot B)'

=> y = A \cdot (A' \cdot B') + B \cdot (A' \cdot B')

=> y = A \cdot A' + A \cdot B' + B \cdot A' + B \cdot B'

=> y = 0 + A \cdot B' + B \cdot A' + 0

=> y = A \cdot B' + B \cdot A'

... y = A \oplus B
```

After constructing the truth table and obtaining observing the results, we notice that the outputs are exactly identical to that of a XOR gate. The Simplified Boolean expression also shows the final output in its simplest form, which is equivalent to a XOR gate. From this experiment, we can see how we can use multiple NAND gates to construct a circuit which gives the same output as a XOR gate. This is more advantangeous than using one single logic gate such as in terms of expense. This is why both NAND and NOR gates are called universal gates as they can be used to form any circuit.