Faculty of Engineering and Technology

Electrical and Computer Engineering Department

DIGITAL ELECTRONICS AND COMPUTER ORGANIZATION LABORATORY

ENCS2110

Report#1

EXP.No.2.Comparators,Adder,and Subtractors

.....

Student name: Saja Asfour

Student ID: 1210737

Group members name and ID:

-Saja Asfour 1210737

-Baraa Nassar 1210880

-Reda Srour 1201656

Instructor name: Bilal Karaki

Section number:10

Date: 12/11/2022

Place: Masri107

2.1 Abstract

The Aim of the experiment: To understand the principle of the digital comparators,half-adders,full adders,half-subtractor and full-subtractor,and how to implement each one of them.

Equipment Used in the experiment:

*IT-3002 Basic Gates Circuit.

*IT-3003 Adder/Subtractor Circuits.

Contents

List of figures:

List of Tables:

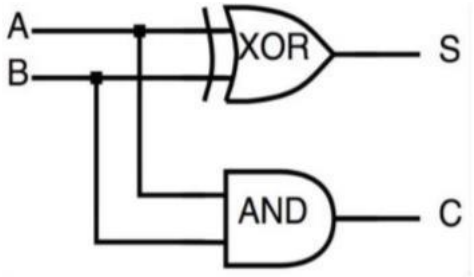
2.2 Theory

2.2.1 Half-adder and full-adder:

2.2.1.1 Half adder:

The half-adder accepts two digits on its inputs and produces two binary digits outputs, a sum bit, and a carry bit. The addition of two bits is done using a combinational circuit Half-adder.

Figure(1) Half-Adder functional Diagram



Figure(2): Half-Adder circuit

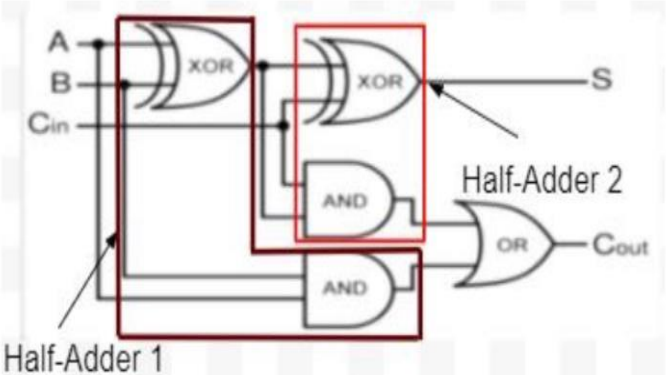
Input		Output	
A	B	Sum	**Carry-out**
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Table (1): Half-Adder truth table.

2.2.1.2 Full-Adder

The full-adder can perform addition or subtraction, it adds together two binary digits and carry in digit which means it has three input and comes up with two outputs.

Figure (3): Full-Adder Functional Diagram



Figure(4): Full-Adder Circuit.

A	B	Carry-In	Sum	Carry-Out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Table (2): Full-Adder truth table

2.2.2 Half-and Full-Subtractor Circuits:

2.2.2.1 Half Subtractor:

The circuit of half-subtractor can be built with NAND and XOR gates, use it if subtraction 1-bit at a time regardless of whether the minuend is greater than the subtrahend .”barrow” from pervious subtraction is not taken into consideration.

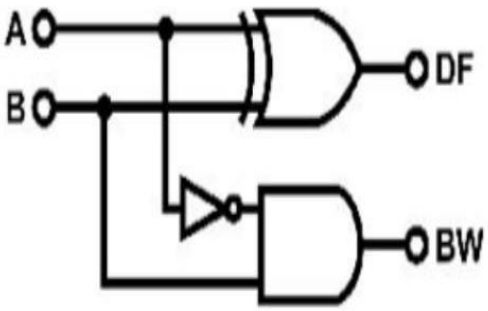


Figure (5): Half-Subtractor circuit

Minuend	Subtrahend	Difference	Borrow
A	B	DF	BW
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Table(3): Half-Subtractor truth table

2.2.2.2 Full Subractor:

A full subtractor is a combinational circuit that performs subtraction of two bits, one is minuend and other is subtrahend, taking into account borrow of the previous adjacent lower minuend bit. This circuit has three inputs and two outputs.

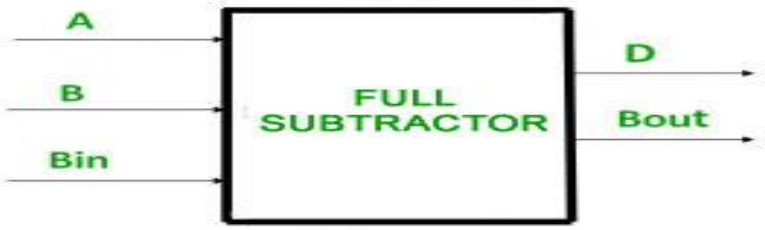


Figure (6): Full-Subtractor circuit Diagram

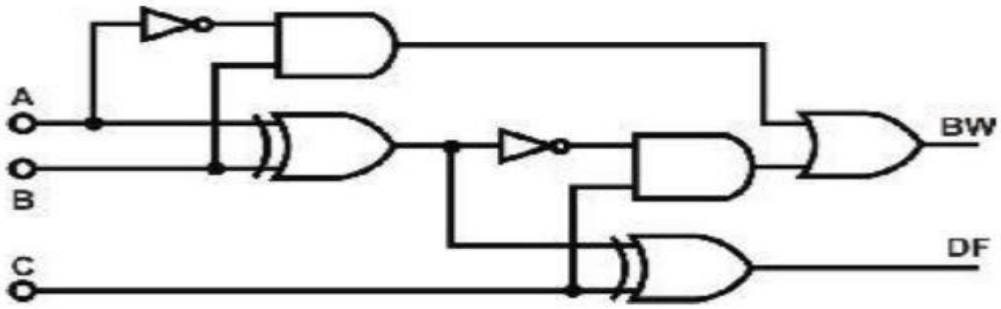


Figure (7): Full-Subtractor circuit

INPUT			OUTPUT	
A	B	Bin	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Table(4):Full-Subtractor Truth table

2.2.3 Comparator Circuit:

2.2.3.1 1-bit Comparator:

The binary or digit comparator can be constructed using standard AND,NOR,NOT gates to compere the digital signals present at their input(A,B) and produce an output depending upon the condition of the inputs, there are three possible outputs:A>B,A<B,A=B.

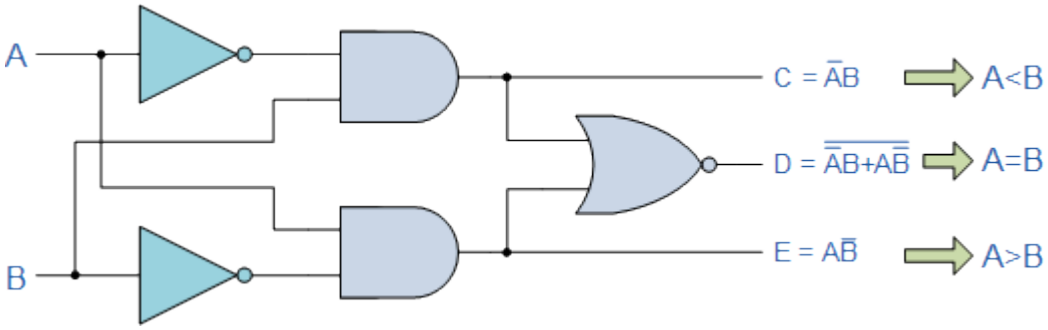


Figure (8): 1-bit Comparator Circuit

Inputs		Outputs		
B	A	A > B	A = B	A < B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

Table (5): 1-bit Comparator truth table

2.2.3.2 4-bit Comparator:

In order to design a 4 – bit comparator, each bit of the 4 – bit numbers should be compared on it’s own, and based on this comparison

-A=B: In order to equal the two inputs, each bit of the first number A should be equal to same bit of the second number B such as:

A0 = B0, A1 = B1, A2 = B2, A3 = B3

-A>B:

There are multiple occasions where this output is true

When A3 > B3

When A3 = B3 AND A2 > B2

When A3 = B3 AND A3 = B3 AND A2 > B2 .

When A3 = B3 AND A3 = B3 AND A2 = B2 AND A1 > B1

When A3 = B3 AND A3 = B3 AND A2 = B2 AND A1 = B1 AND A0 > B0

-A<B:

There are multiple occasions where this output is true

When A3 < B3

When A3 = B3 AND A2 < B2

When A3 = B3 AND A3 = B3 AND A2 < B2

When A3 = B3 AND A3 = B3 AND A2 = B2 AND A1 < B1

When A3 = B3 AND A3 = B3 AND A2 = B2 AND A1 = B1 AND A0 < B0

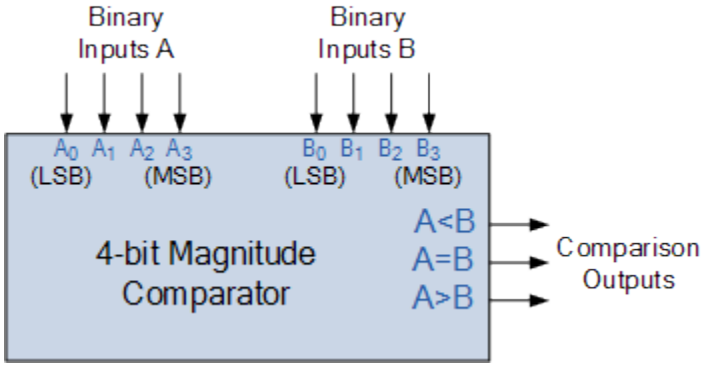


Figure (9): 4-bit comparator circuit diagram

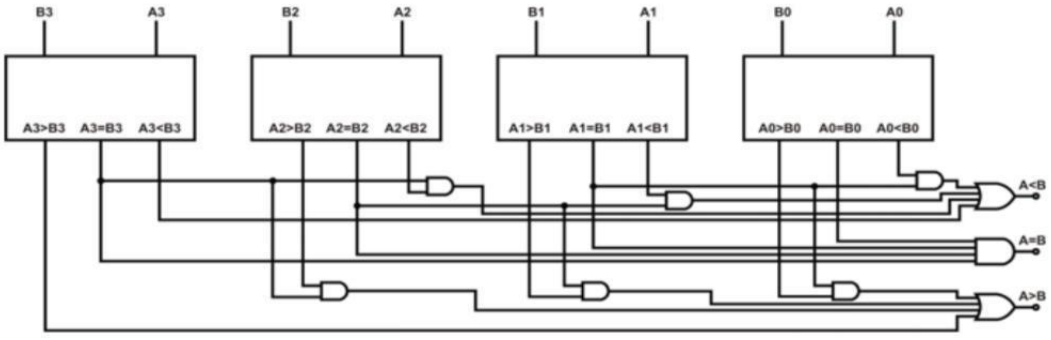


Figure (10): A 4-bit comparator constructed with four 1-bit comparator diagram

2.3 Procedure:

2.3.1 Comparator Circuits:

2.3.1.1 Construction Comparator with Basic Logic Gates:

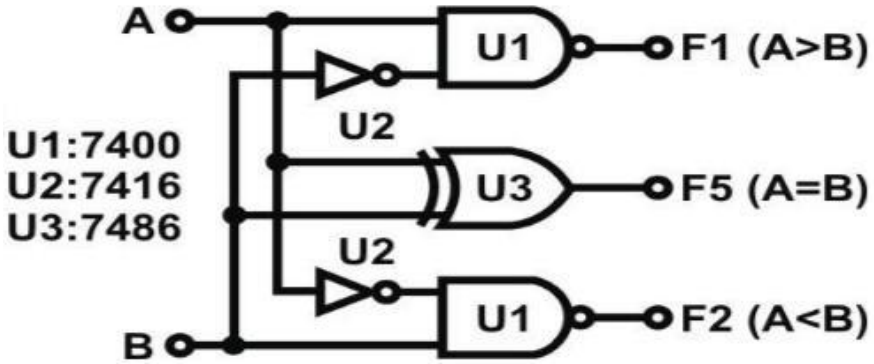


Figure (11): 1-bit comparator Logic diagram

The circuit above was implemented by IT-3002 Comparator 1 block and we recorded the output.

input			**Output**		
Sw2(B)	**Sw1(A)**		**F1**	**F2**	**F3**
0	0	A=B	1	1	0
0	1	A>B	0	1	1
1	0	A<B	1	0	1
1	1	A=B	1	1	0

Table(6):1-bit comparator truth table

After comparing the truth table in the theory with this one, it's clear that the are exactly the opposite since we used NAND and XOR gates instead of NOT ,AND &NOR(low level).

A \ B	0	1
0	0	
1		1

k-map for A=B

(A=B)=AB+A'B'

A \ B	0	1
0		0
1		

k-map for A>B

(A>B)=AB'

A \ B	0	1
0		
1	0	

k-map for A<B

(A<B)=A’B

2.3.1.2 Constructing Comparator with TTL IC:

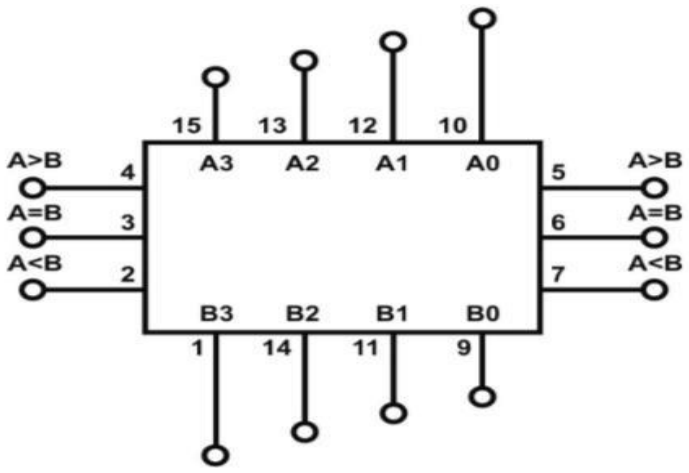


Figure (12): 4-bit Comparator IC(IT-3002 block Comparator 2) diagram

Input			Output		
A>B	**A=B**	**A<B**	**A>B**	**A=B**	**A<B**
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	0	0	0
1	1	1	0	1	0

Table (7):4-bit Comparator IC truth table

2.3.2 Half-and Full-Adder Circuits:

2.3.2.1 Half Adder:

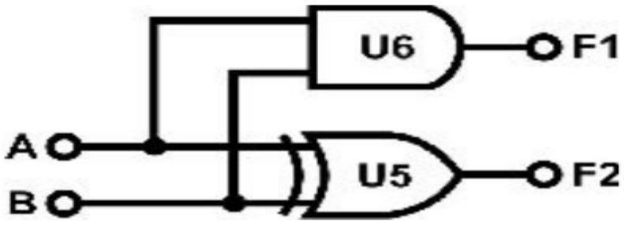


Figure (13):Half Adder implemented

Input		Output	
SW1(B)	**SW0(A)**	**F1(Carry)**	**F2(SUM)**
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Table (8): Half adder truth table

A \ B	0	1
0		1
1	1	

k-map for sum

A \ B	0	1
0		
1		1

k-map for carry

After implementing the k-map on this truth table the functions of the sum and the carry were

Carry = AB = A AND B

Sum = AB’ + A’B = AXOR B

2.3.2.2 Full-Adder with basic gates:

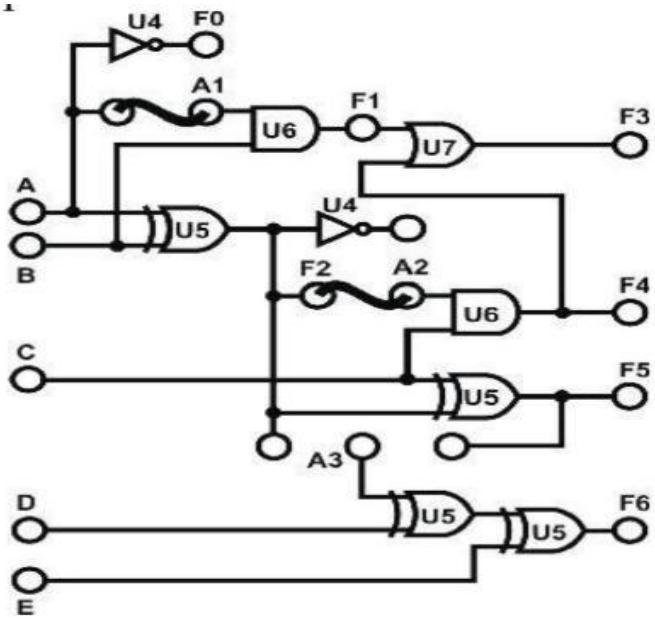


Figure (14): IT-3003 Full-Adder block

Input			Output	
SW3(C)	**SW2(B)**	**SW1(A)**	**F3(CARRY)**	**F5(SUM)**
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Table 9: IT 3003 Full Adder truth table

BA \	00	01	11	10
0		1		1
1	1		1	

K-map for sum

BA / C	00	01	11	10
0			1	
1		1	1	1

k-map for carry

After implementing the k-map on this truth table the functions of the sum and the carry were

Carry = AC + BC + AB

Sum = CA’B’ + C’AB’ + CBA + C’BA’

2.3.2.3 Constructing 4-bit Full-Adder with IC:

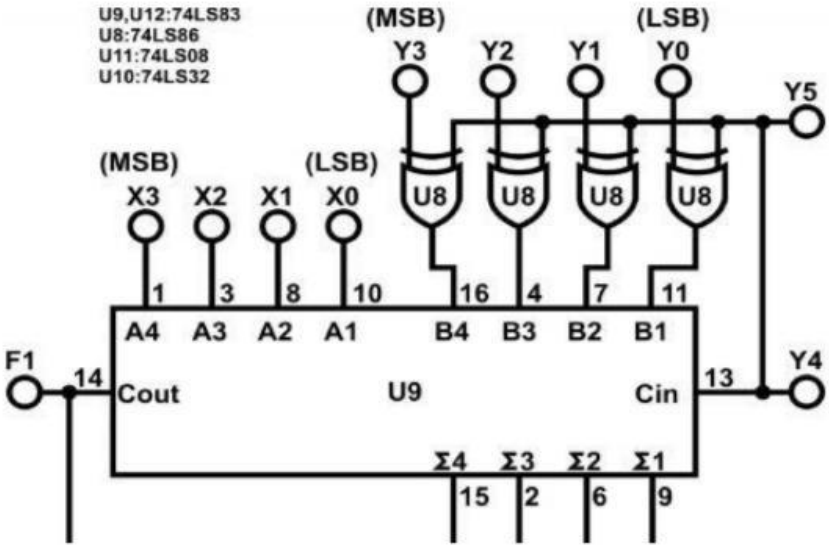


Figure 15 :IT-3003 4-bit Full-adder block diagram

Input								Output				
Y3	**Y2**	**Y1**	**Y0**	**X3**	**X2**	**X1**	**X0**	**{4**	**{3**	**{2**	**{1**	**F1**
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	0	1	1	0	0	1	1	0	0
0	0	0	0	1	0	0	1	1	0	0	1	0
0	0	0	0	1	1	1	1	1	1	1	1	0
0	0	0	1	0	0	1	1	0	1	0	0	0
0	0	0	1	0	1	1	0	0	1	1	1	0
0	0	0	1	1	0	0	0	1	0	0	1	0
0	0	1	1	0	1	1	0	1	0	0	1	0
0	1	0	0	1	0	0	0	1	1	0	0	0
0	1	0	0	1	1	1	1	1	0	1	1	0
1	0	0	0	0	1	1	1	1	1	1	1	0
1	0	0	1	1	0	0	1	0	0	1	0	1
1	0	1	0	1	0	1	1	0	1	0	1	1

Table 10: IT-3003 4-bit Full-Adder truth table

2.3.2.4 Constructing BCD adder

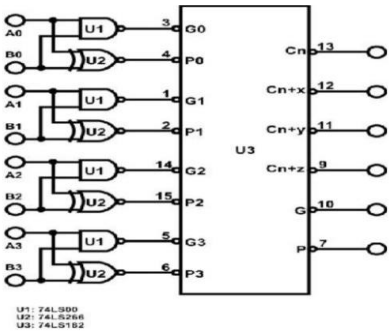


Figure 16:IT-3003 BCD Adder diagram

Inputs								Outputs(U9)					LAST(U12)					
X3	**X2**	**X1**	**X0**	**Y3**	**Y2**	**Y1**	**Y0**	**F1**	**F11**	**F10**	**F9**	**F8**	**F2**	**F3**	**F7**	**F6**	**F5**	**F4**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	1	0	0	1	0	0	0	0	0	1	0	0
0	0	1	1	0	1	0	0	0	0	1	1	1	0	0	0	1	1	1
0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0
0	0	1	0	1	0	0	0	0	0	1	1	0	0	0	0	1	1	0
0	0	1	1	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1
0	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0	1	1	0
0	1	0	0	0	1	0	1	0	1	0	0	1	0	0	1	0	0	1
0	1	0	0	0	1	1	0	0	1	0	1	0	1	1	0	0	0	0
0	1	0	1	0	1	1	0	0	1	0	1	1	1	1	0	0	0	1
0	1	1	0	0	1	1	1	0	1	1	0	1	1	1	0	0	1	1
0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	0	1	0	1
0	1	1	1	1	0	0	1	1	0	0	0	0	1	0	0	1	1	0
1	0	0	0	1	0	0	1	1	0	0	0	1	1	0	0	1	1	1
1	0	0	1	1	0	0	1	1	0	0	1	0	1	0	1	0	0	0
1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0
1	0	1	0	1	0	1	1	1	0	1	0	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0	1	0	1	1	0	1	0	1	1	0	0
1	0	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	0	1	0	0

Table 11: IT-3003 BCD Adder truth table

2.3.3 Half-and Full Subtractor Circuit

2.3.3.1 Constructing Half-/Full Subtractors with basic logic Gates

	INPUTS			OUTPUTS			
	C	**A**	**B**	**F1**	**F2**	**F3**	**F5**
Half-subtractor Half-adder	**0**	**0**	**1**	**1**	**0**	**1**	**0**
	0	**0**	**0**	**0**	**0**	**0**	**0**
	0	**1**	**1**	**0**	**0**	**1**	**0**
	0	**1**	**0**	**0**	**1**	**0**	**1**
Full-subtractor Full-adder	**1**	**0**	**0**	**0**	**0**	**1**	**1**
	1	**0**	**1**	**1**	**1**	**1**	**0**
	1	**1**	**0**	**0**	**1**	**1**	**1**
	1	**1**	**1**	**0**	**0**	**1**	**0**

Table 12: Half and Full subtractor truth table

2.3.3.2 Constructing 4-Bit Full-Subtractor with IC

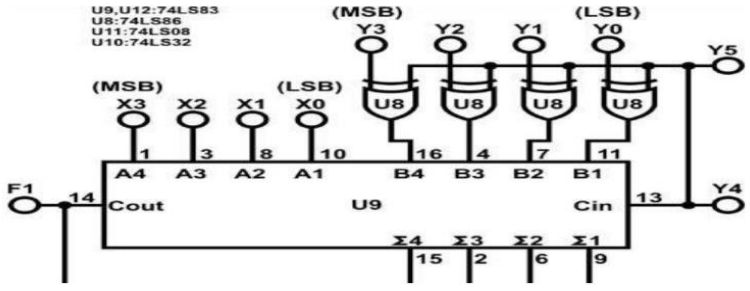


Figure 17: IT-3003 4-bit Full Adder block diagram

INPUTS								OUTPUTS				
X3	**X2**	**X1**	**X0**	**Y3**	**Y2**	**Y1**	**Y0**	**F1**	**F11**	**F10**	**F9**	**F8**
0	1	0	0	0	1	0	0	0	1	1	1	1
0	1	0	0	0	0	1	1	1	0	0	0	0
1	0	0	0	0	0	1	1	1	0	1	0	0
1	0	0	0	0	0	0	1	1	0	1	1	0
1	0	0	1	1	0	0	0	1	0	0	0	0
1	0	0	1	0	1	1	1	1	0	0	0	1
1	0	1	0	0	1	1	0	1	0	0	1	1
1	0	1	0	0	1	0	1	1	0	1	0	0
1	0	1	1	1	0	1	0	1	0	0	0	0
1	1	1	1	1	0	1	0	1	0	1	0	0

Table 13: 4-bit adder with IC truth table

2.4 Conclusion

To conclude , the aim of this experiment is achieved. and the result satisfied the theory part of the report. the experiment helped with understanding more about comparator and adders and subtractors. During the experiment we knew how to construct the circuit using two modules IT-3002 and IT-3003. Comparator,Adder,Subtractor using both basic gates and ICs.

References:

https://www.geeksforgeeks.org/full-adder-in-digital-logic/
https://www.electronics-tutorials.ws/combination/comb_8.html
Digital Lab manual2022