DIGITAL LOGIC AND DESIGN LAB

PROJECT :1

(Section: B5)

Submitted to:

Sir Umair

Submitted by:

Zain L1S23BSCS0247

Qaraib L1S23BSCS0248

Zarmeen Fatima
L1S23BSCS0139

Date of submission:

JAN.23,2024

Scanned with CamScanner

Logic gates: Logic gates are used to carry out logical operations on single or multiple binary inputs and give one binary output. In simple terms, logic gates are the electronic circuits in a digital system.

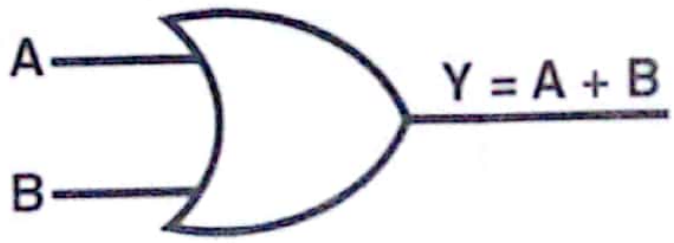
Types of Logic Gates:

There are several basic logic gates used in performing operations in digital systems. The common ones are

- OR Gate
- AND Gate
- NOT Gate
- XOR Gate

Additionally, these gates can also be found in a combination of one or two. Therefore, we get other gates, such as NAND Gate, NOR Gate, EXOR Gate and EXNOR Gate.

OR Gate:



Working:

The OR gate produces a HIGH output (1) if any of its inputs are HIGH (1). The output is LOW (0) only when all inputs are LOW

Boolean Expression:

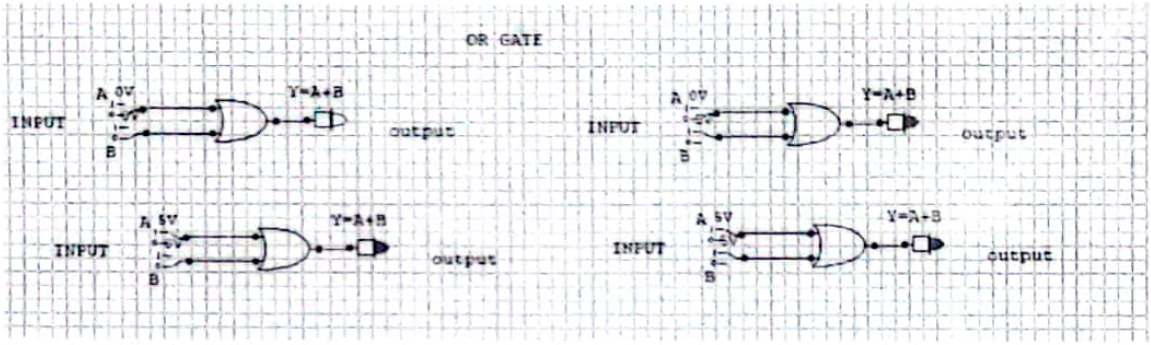
Y = A + B

Scanned with CamScanner

Truth Table:

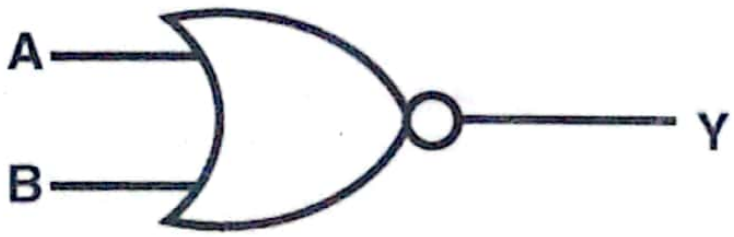
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

VERIFICATION



Nor Gate:

This gate is the combination of OR and NOT gates.



:

Boolean expression:

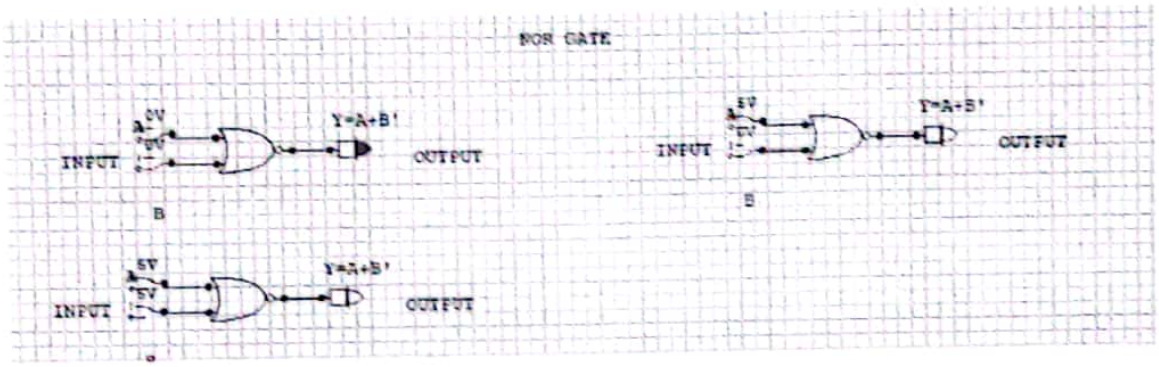
$$Y = \overline{A + B}$$

Scanned with CamScanner

Truth Table:

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

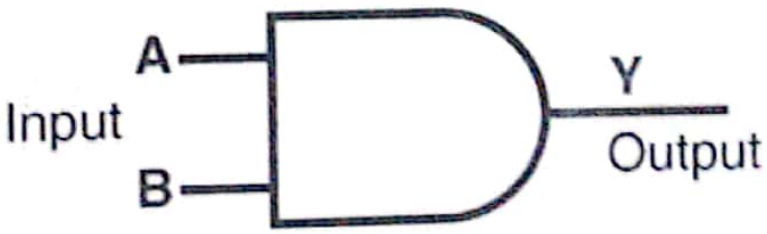
VERIFICATION



Working:

The NOR gate is the inverse of the OR gate. It produces a HIGH output (1) only when all of its inputs are LOW (0). Otherwise, it produces a LOW output (0).

AND GATE:



Scanned with CamScanner

Boolean Expression:

$Y = A.B$

Truth Table:

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

VERIFICATION

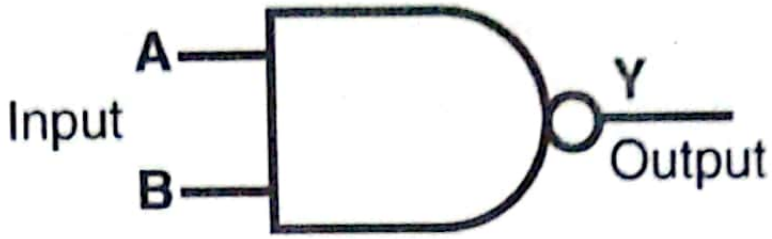
Working:

The AND gate produces a HIGH output (1) only when all of its inputs are HIGH (1). If any input is LOW (0), the output is LOW (0).

NAND GATE:

This basic logic gate is the combination of AND and NOT gates.

Scanned with CamScanner



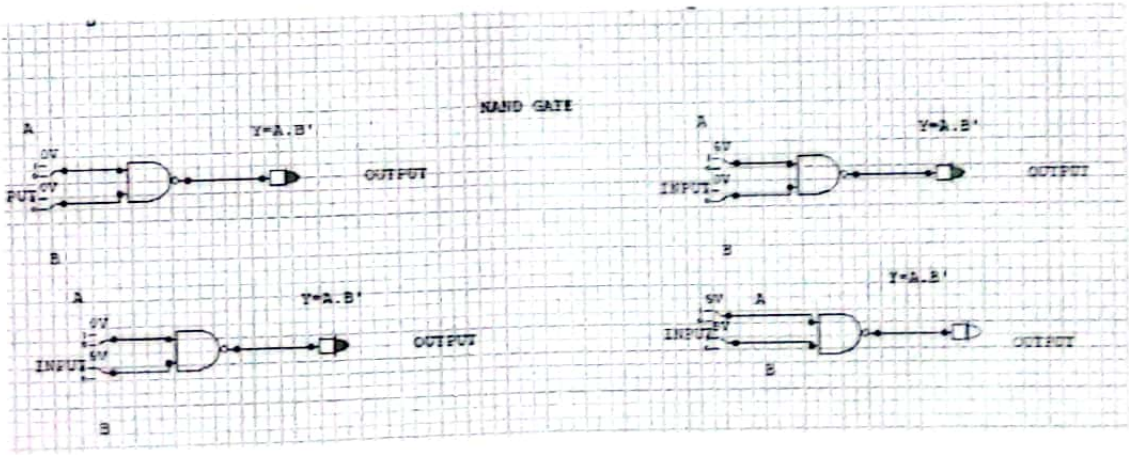
Boolean Expression:

$$Y = \overline{A.B}$$

Truth Table:

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

VERIFICATION

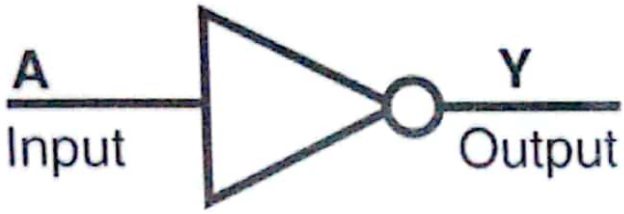


Scanned with CamScanner

Working:

The NAND gate is the inverse of the AND gate. It produces a LOW output (0) only when all of its inputs are HIGH (1). Otherwise, it produces a HIGH output (1).

NOT GATE:



Boolean Expression:

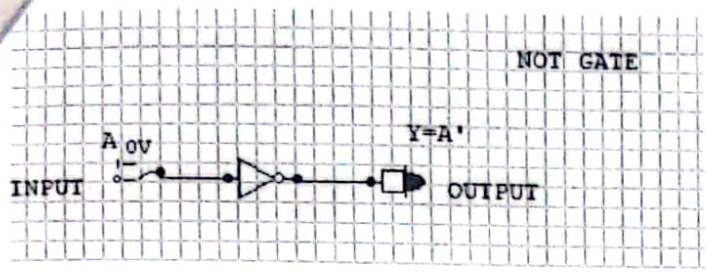
$$Y = \bar{A}$$

Truth Table:

A	Y
0	1
1	0

VERIFICATION

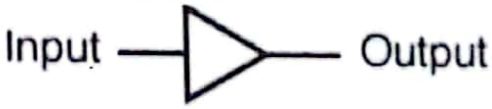
Scanned with CamScanner



Working:

The NOT gate has a single input and inverts its value. It produces a HIGH output (1) when the input is LOW (0) and vice versa.

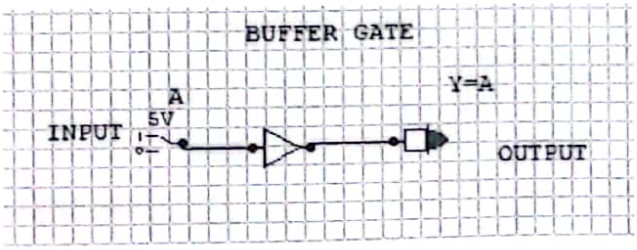
Buffer GATE:



Truth Table:

Input	Output
0	0
1	1

VERIFICATION



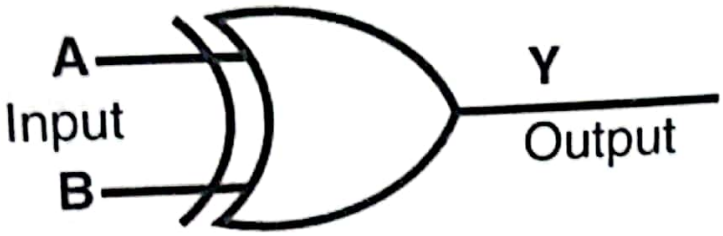
Working:

- A buffer gate has a single input and a single output.

Scanned with CamScanner

- It simply duplicates the input signal to the output without any change.
- Its main purpose is to isolate or strengthen a signal.

Exclusive-OR gate (XOR Gate):



Boolean Expression:

$A.\bar{B} + \bar{A}.B$

or

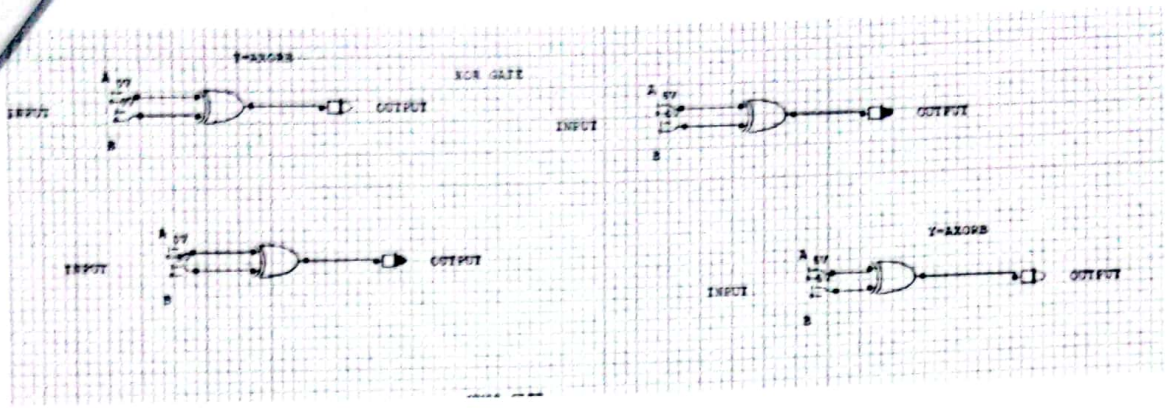
$Y = A \oplus B$

Truth Table:

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

VERIFICATION

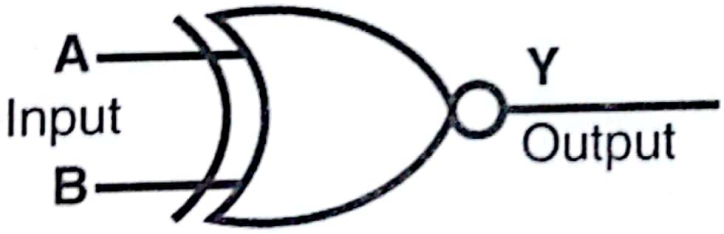
Scanned with CamScanner



Working:

The XOR gate produces a HIGH output (1) when its inputs are different. It produces a LOW output (0) when its inputs are the same.

Exclusive-NOR Gate (XNOR Gate):



Boolean Expression:

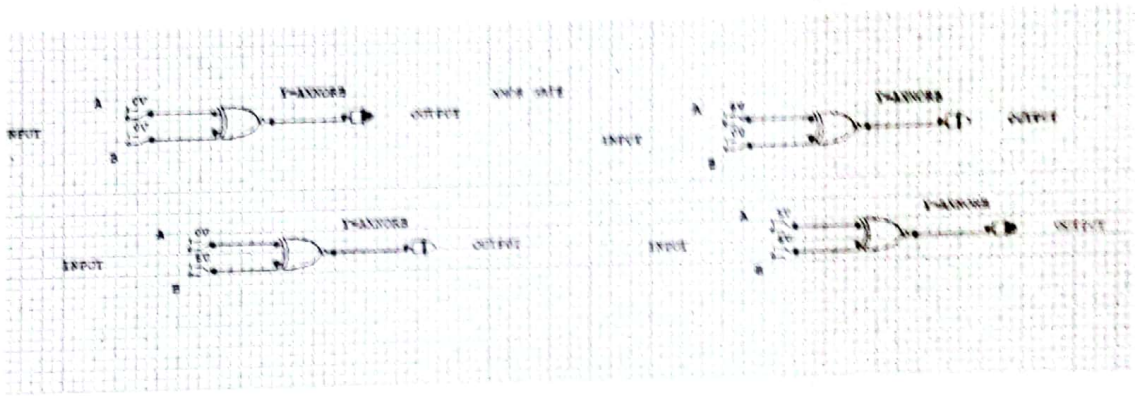
$$Y = (\overline{A \oplus B}) = (A.B + \overline{A}.\overline{B})$$

Scanned with CamScanner

Truth Table:

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

VERIFICATION



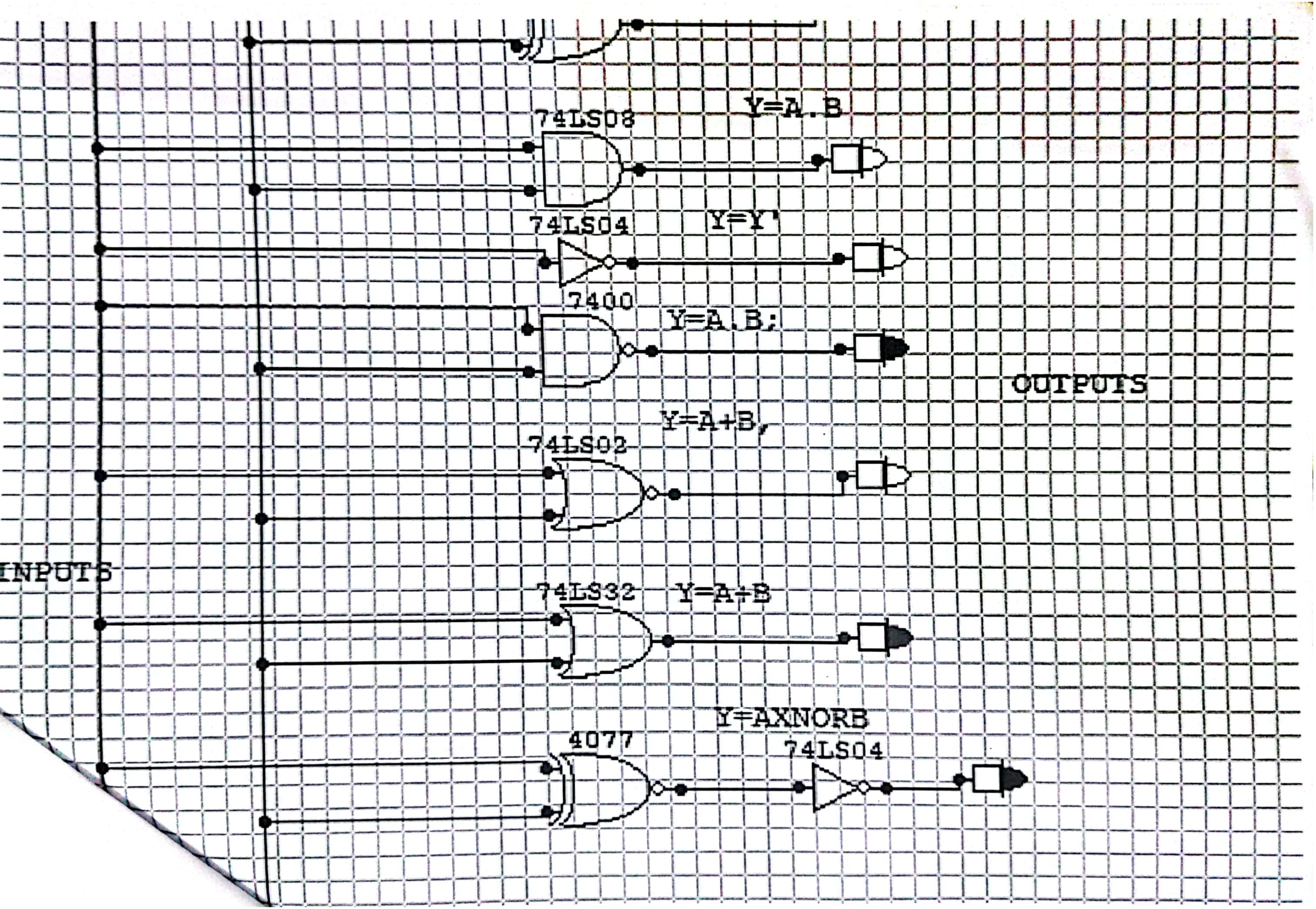
Working:

The XNOR gate is the inverse of the XOR gate. It produces a HIGH output (1) when its inputs are the same. It produces a LOW output (0) when its inputs are different.

Application of Logic Gates:

Logic gates have a lot of applications, but they are mainly based on their mode of operations or their truth table. Basic logic gates are often found in circuits such as safety thermostats, push-button locks, automatic watering systems, light-activated burglar alarms and many other electronic devices.

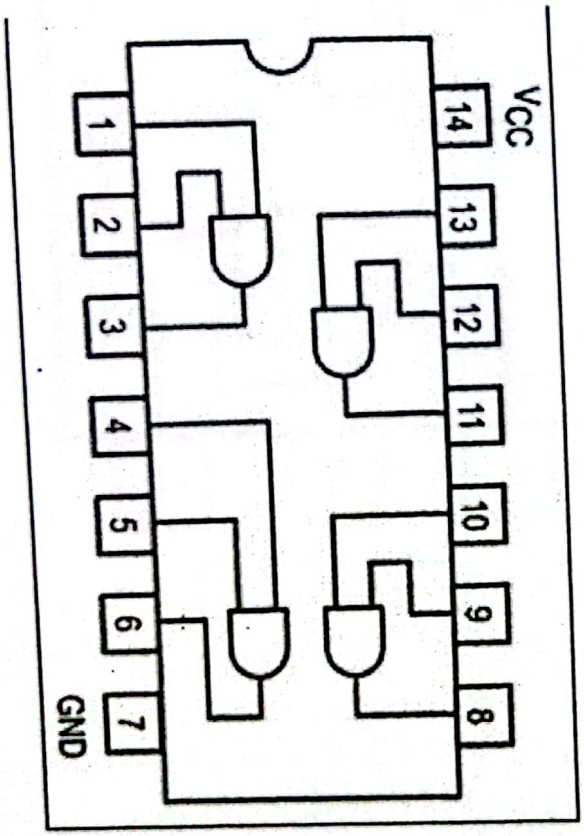
Scanned with CamScanner



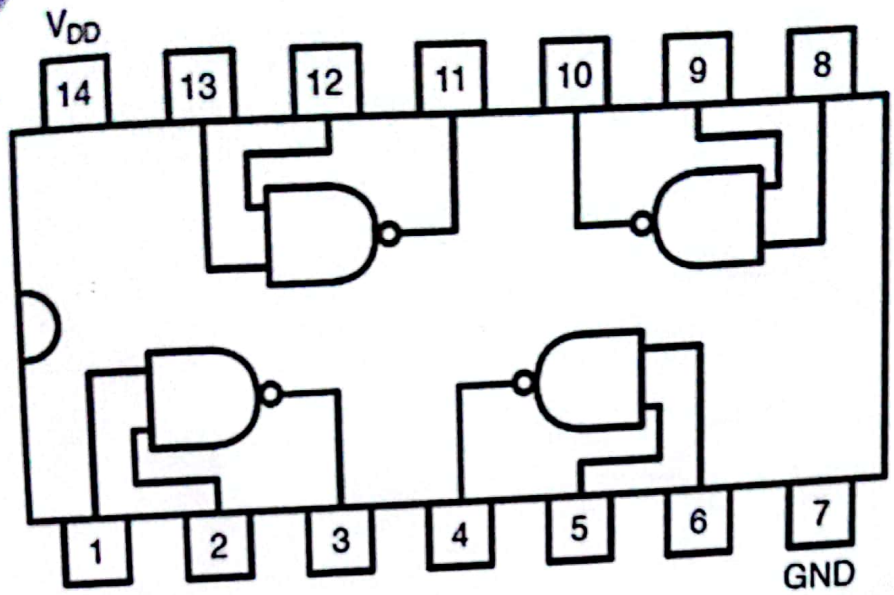
Scanned with CamScanner

IC DIAGRAMS

AND GATE

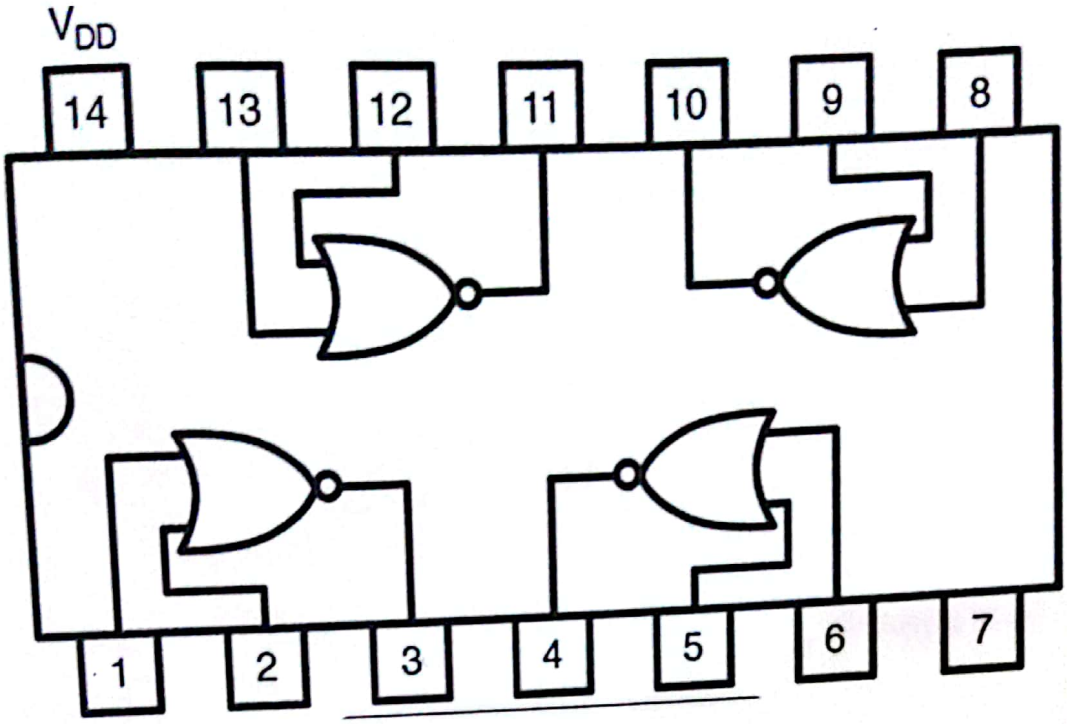


Scanned with CamScanner

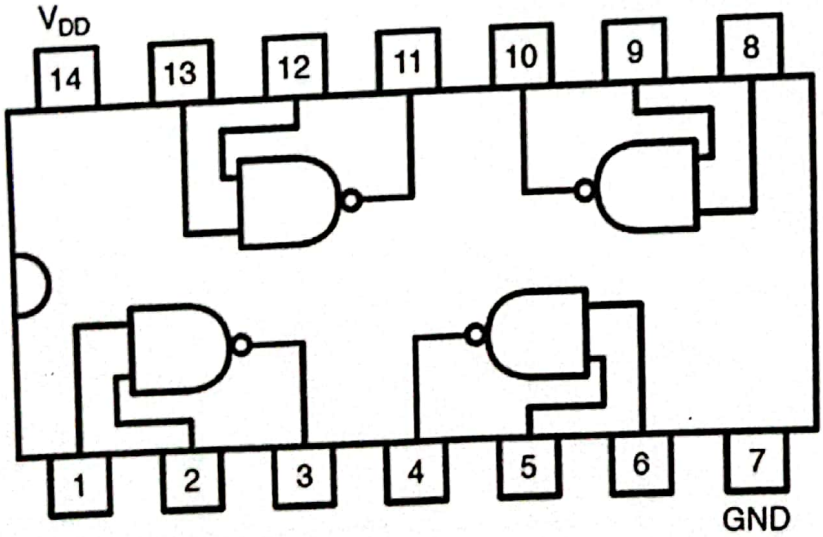


NOR GATE

Scanned with CamScanner

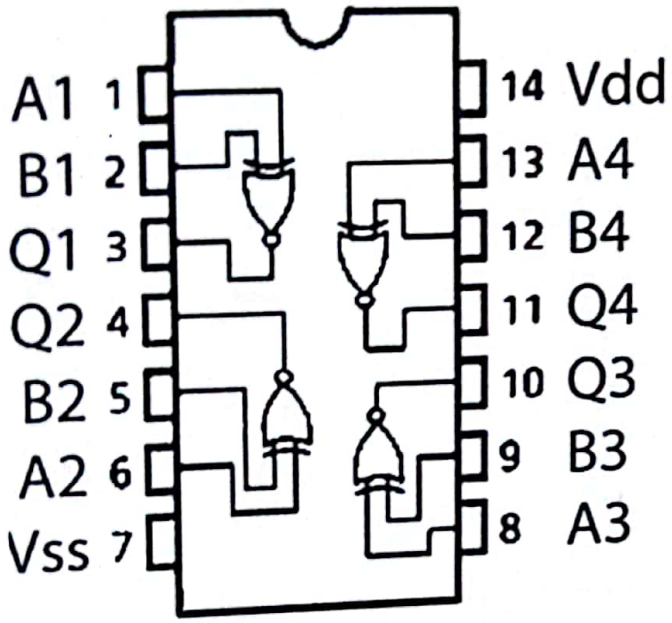


XOR GATE

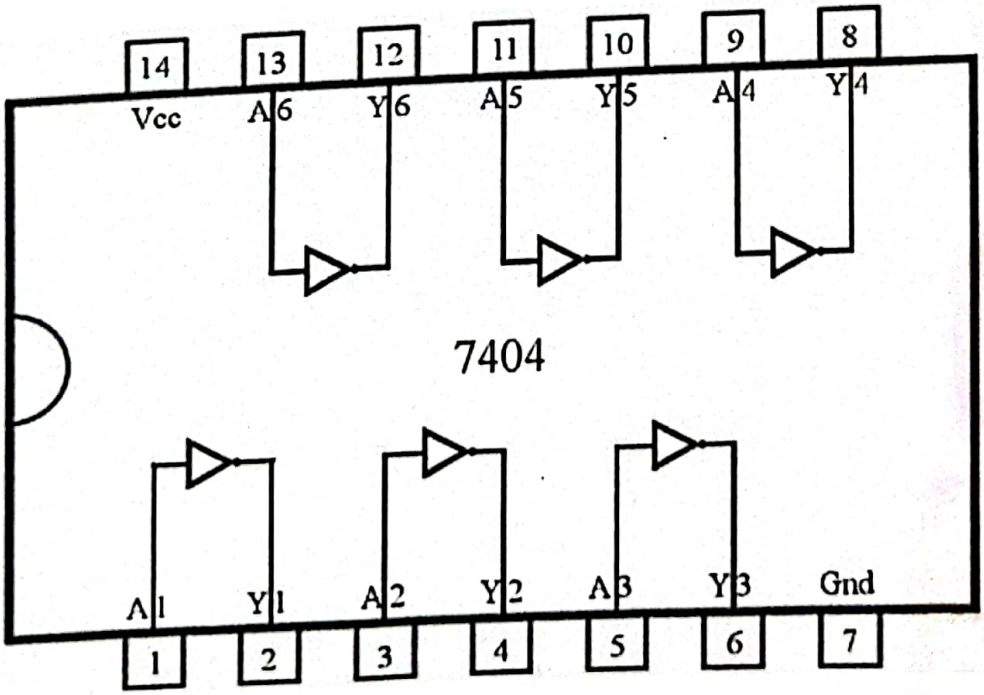


Scanned with CamScanner

XNOR GATE



NOT GATE



Scanned with CamScanner