Experiment (3)

NAND & NOR Implementation

5.1 Objectives:

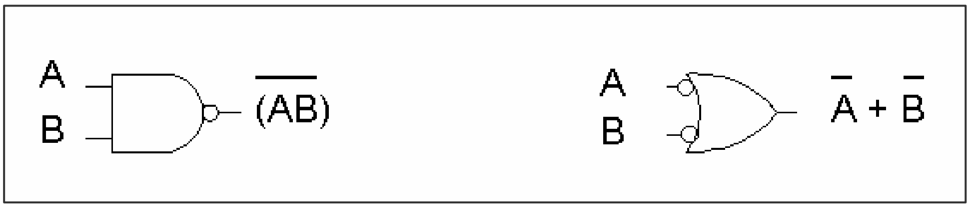
- To demonstrate the implementation of a digital system sing NAND and NOR gates.

5.2 Background Information :

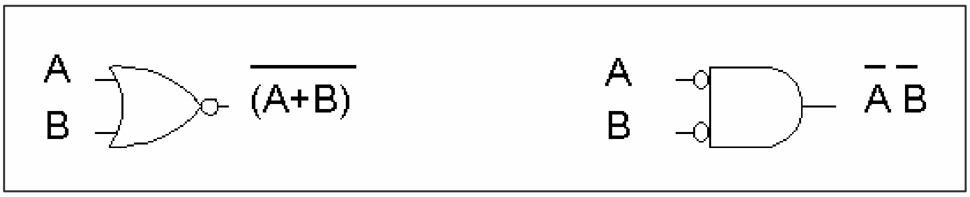
NAND and NOR gates is said to be universal gates because any digital system can be implemented using only one of these gates.

Digital circuits are frequently constructed with only NAND or NOR gates; because these gates are easier to fabricate with electronic components. Because of the importance of NAND and NOR in the design of digital circuits, rules and procedures have been developed for the conversion from Boolean functions in terms of AND, OR and NOT into equivalent NAND or NOR logic diagrams. NAND and NOR are called universal gates because any digital system or Boolean function can be implemented with only these gates.

From DeMorgan theorem, we can see other representation for NAND and NOR gates as follows:



Two symbols for NAND



Two symbols for NOR

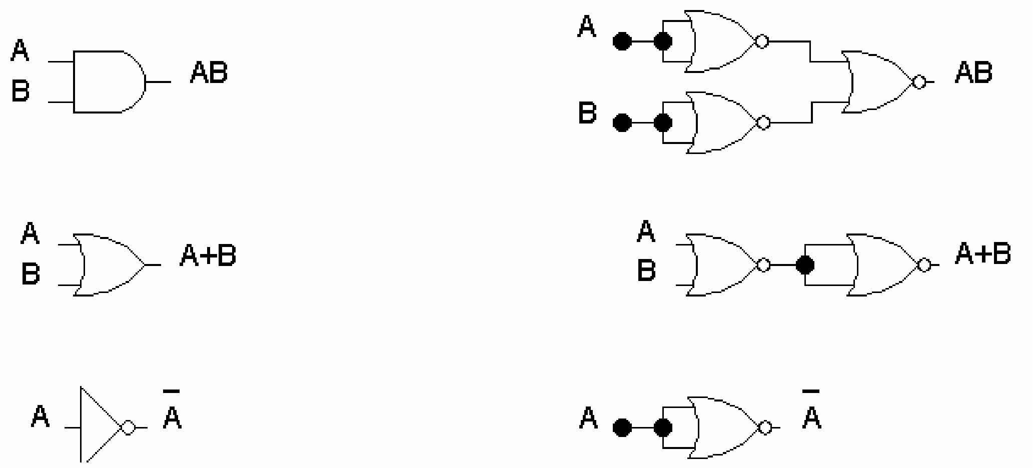
5.3 Equipment Requires :

The following equipments are needed to perform all the procedures :

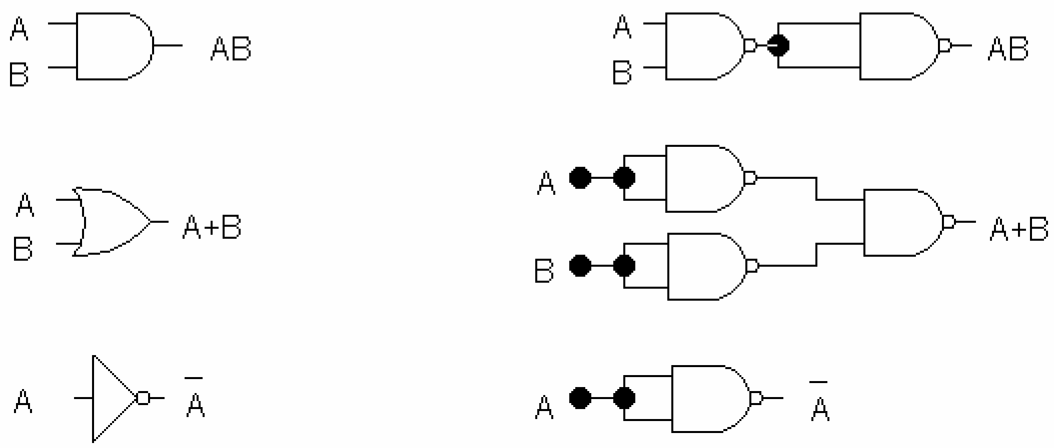
Universal Breadboard
Jumper wire kit
(1) 7400 TTL QUAD 2-INPUT NAND GATE
(1) 7402 TTL QUAD 2-INPUT NOR GATE
(1) 7410 TTL TRIPLE 3-INPUT NAND GATE
(1) 7427 TTL TRIPLE 3-INPUT NOR GATE
2x Toggle Switches
1x Carbon-film Resistor (470Ω)
1x LED

5.4 Procedure :

1. Using only NAND and NOR gates to implement all basic logic gates and then verify your results.



Implementation of AND, OR , NOT using NAND



Implementation of AND, OR, NOT using NOR

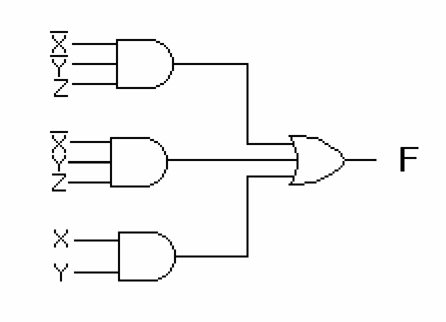
2. NAND Implementation of Boolean functions:

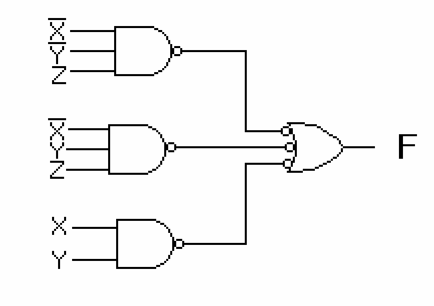
- Assume that all variables and their complements are available as inputs.
- Find the minimum SOP expression of the function.
- Draw the corresponding two-level AND-OR circuit diagram.
- Replace all gates with NAND gates; learning the gates interconnections are kept unchanged.

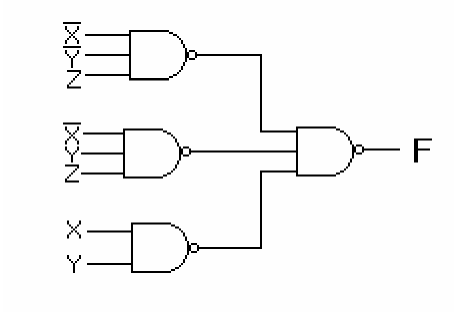
Example:

Try to implement the function F, from the previous experiment, using only NAND gates, and then verify your results.

$$F = x'\ y'\ z + x'\ y\ z + x\ y'$$







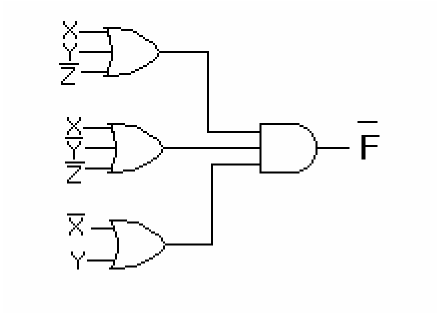
3. NOR Implementation of Boolean functions:

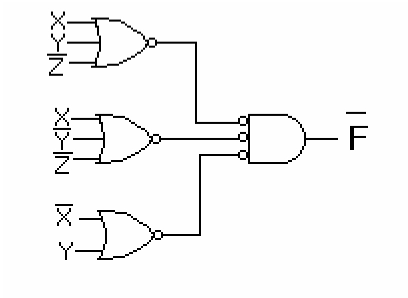
- Assume that all variables and their complements are available as inputs.
- Find the minimum POS expression of the function.
- Draw the corresponding two-level OR-AND circuit diagram.
- Replace all gates with NOR gates, learning the gates interconnections are kept unchanged.

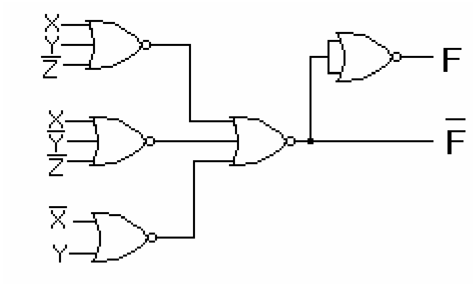
Example:

Try to implement the same function F, using only NOR gates, then verify your results.

$F = x' y' z + x' y z + x y'$
$F' = (x' y' z + x' y z + x y')'$
$F' = (x' y' z)' (x' y z)' (x y')'$
$F' = (x + y + z') (x + y' + z') (x' + y)$







X	Y	Z	Result from the previous exp.	NAND implementation	NOR implementation
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Questions:

1) Implement a NOR gate using NAND gates, then implement A NAND gate using NOR gates?
2) Give a summary of the points you learned from the experiment.