Lab 8

Digital Gates

REFERENCE: Horowitz and Hill Sections 8.01 - 8.13

INTRODUCTION

" Digital devices are basically electronic switches that use two states, or logic levels.

TTL and CMOS are two families of digital circuits. Most commonly, they are powered by a +5 V power supply, although CMOS is sometimes used with +12 V.

These are integrated circuits, i.e., IC's, or "chips."

A digital signal is either ON or OFF:

state	other names for the state	TTL	CMOS
ON	(HI, "1", TRUE)	3.0 -5.5 V	3.5 - 5.5 V or 8.5 - 12.5 V
OFF	(LO, "0", FALSE)	-0.5 - 1.5 V	-0.5 - 1.5 V

Digital ICs called "gates" perform arithmetic and logical operations:
one input: NOT (inverter)
two or more inputs: OR, AND, NOR, NAND, XOR (Exclusive OR).

On 14-pin logic chips, usually pin 14 is $V_{CC} = +5V$, and pin 7 is ground.

We will try out these devices and their truth tables, then combine them to perform more complicated logical operations. We will also test half and full adders, which perform the arithmetic operation of addition, and a multiplexer.

EQUIPMENT

Prototyping board
Power supply
Digital multimeter
Resistors: 1k (4), 220 Ω (4)
LEDs (4)

Digital ICs:	74<u>HC</u>00 <u>and</u> 74<u>LS</u>00	Quad NAND
	74LS02 (or 74HC02)	Quad NOR
	74LS04 (or 74HC04)	Hex inverter
	74LS86 (or 75HC86)	XOR
	74LS151	8 Input Multiplexer
	74LS240	Tristate Octal Buffer

PROCEDURE

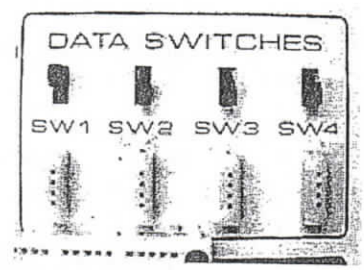
0. Things to know

“ About digital inputs

In making tests, you will need to connect some inputs to your digital circuits.

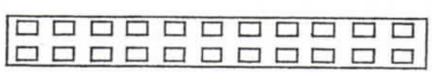
For a LO input, use a wire between the digital input and ground.
For a HI input, use a 1 k resistor between a digital input and +5 V.

An alternative method (more convenient than using resistors as described above) requires using a prototyping board that has built-in switches for logical HI and LO inputs. These provide suitable inputs for logic circuits and don't require resistors.

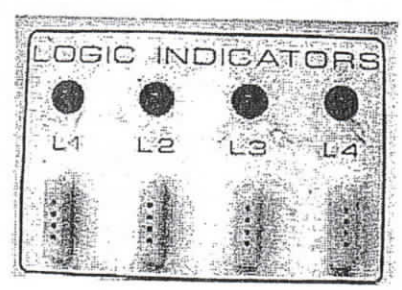


“ On your prototyping board:

Use two wires to connect the +5 V and GND outputs of the power supply to a strip that looks like:



Your prototyping board has built-in LED "Logic Indicators" that you can use for displaying a signal. As an alternative, you may use a discrete LED as an indicator, but be sure to use a series resistor, as described below.



" How to read part numbers:

The part number consists of numbers and letters.

Example: `SM74LS00` -- the `SM` indicates that the manufacturer is Motorola (and this is not important), `7400` indicates that the chip is a quad 2-input NAND, and the LS indicates that it is a variety of TTL that is "low-power Schottky-input." Other chips that perform the same logical function are `7400` (plain old-fashioned TTL) and `74HC00` ("High-speed" variety of CMOS). All of these chips are logically identical, and their pin configurations are the same.

" About CMOS:

CMOS is more common nowadays than TTL because it has a lower power dissipation. Nevertheless, your instructor may prefer TTL for this course because TTL chips, unlike CMOS, are immune to damage by static electricity.

Static electricity can destroy CMOS chips. To avoid damage:

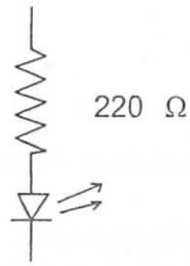
- store unused chips properly, in special plastic tubes or special plastic bags.
- before handling a CMOS chip, ground yourself by touching a ground, such as the metal cover plate on an electrical power outlet on wall.

" About the light-emitting diode (LED)

In building digital circuits, you will often use an LED as an indicator of the state, HI or LO.

ALWAYS USE A RESISTOR IN SERIES WITH THE LED.

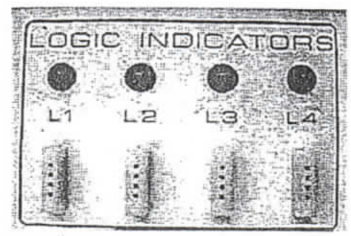
The series resistor (220 Ω typical) is needed to avoid destroying the LED. The LED is a diode, and when it is on, it has very little resistance. If you connected an LED directly between a source and either ground or V_{CC}, it would try to pass an infinite amount of current, which will burn up the LED.



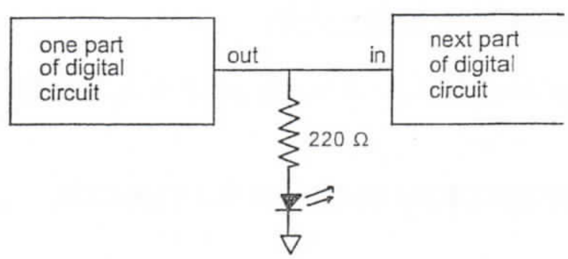
About buffers for the LED

A buffer is often needed to drive an LED. The outputs of many gates and circuits cannot source enough current to drive an LED directly; this is when you must provide a "buffer". The buffer is a digital version of a follower; its output is at the same voltage as the input, but is capable of driving more current while sinking very little at its input.

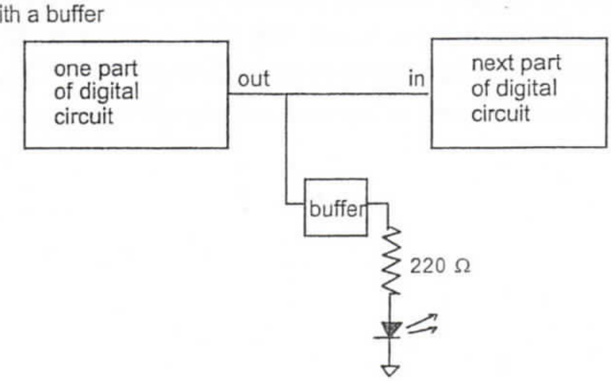
Your prototyping board has LEDs that are already wired so that you can use them as digital indicators, but to acquaint you with buffers, you may build a 4-channel LED indicator, as follows in part 1.



without a buffer



with a buffer



1. <u>The '240 Octal Tristate Buffer used as an LED driver</u>

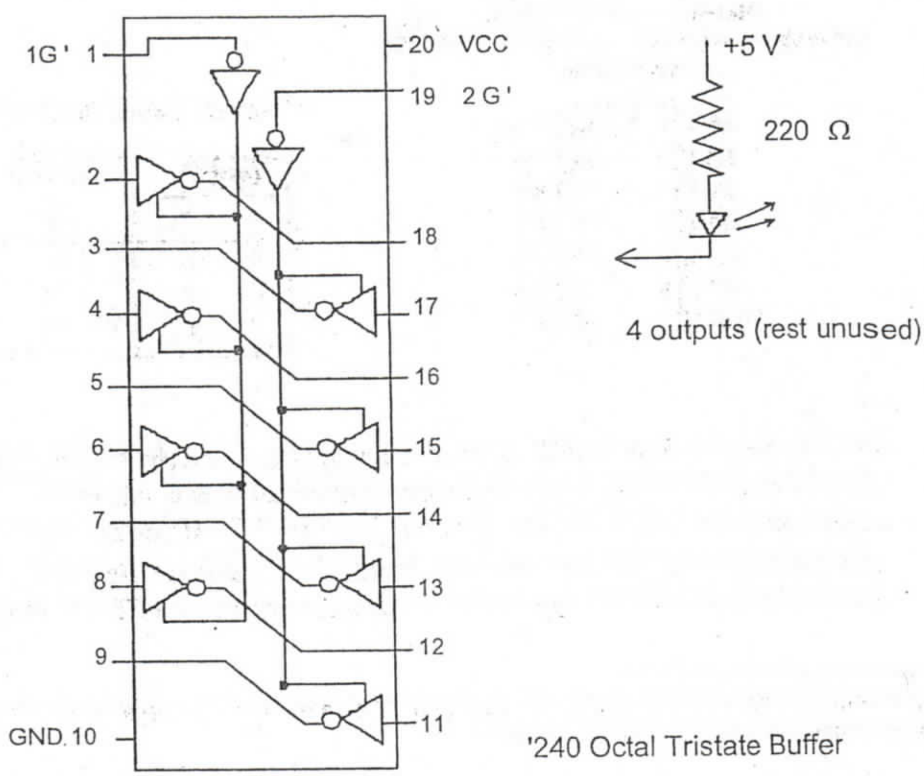
Use the 5V power supply for V_{cc}.

☛ Install the '240 octal tristate buffer on the prototyping board to drive the LED display (see figure below). This will be your digital indicator. Connect three inputs (pins 2,4,6) to switches on your prototyping board (e.g., SW1, etc.). Examine the pin diagram below to identify the output pin corresponding to each of these input pins. Connect the corresponding outputs to LEDs, each in series with a 220 Ω resistor. Connect Enable (pin 1) to ground.

☛ ✍ Verify that a HIGH on the input of one of the 4 buffer channels will cause the corresponding LED to light.

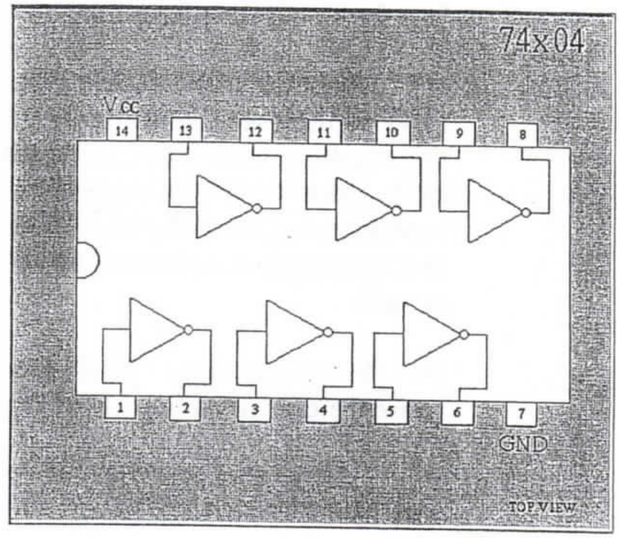
❝ Here you tested the '240 as an LED driver. Your prototyping board may have built-in LED's -- if so, they surely have a buffer and series resistor similar to what you will build here.

If your prototyping board does <u>not</u> have such LED indicators, you will need this buffer for the following circuits to indicate digital states. If your board <u>does</u> have LED indicators, you may use them, and disassemble the '240 circuit now.



'240 Octal Tristate Buffer

2. The '04 Hex inverter



Use the 5V power supply for V_{cc}.

☞ Install the '04 hex inverter on the prototyping board. Examining the pin diagram, choose any of the six inverters. Connect its input to a "data switch" and its output to a Logic Indicator LED.

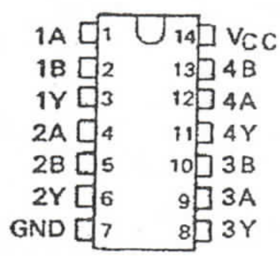
☞ ✍ Verify that a HIGH on the input will cause the corresponding LED to be dark (LOW).

☞ ✍ Verify that a LOW on the input will cause the corresponding LED to be bright (HIGH).

✍ Write the truth table (i.e., function table) for the inverter.

3. Quad 2-input NAND 74HC00 [CMOS] and 74LS00 [TTL]

SN7400 . . . N PACKAGE
SN74LS00, SN74S00 . . . D OR N PACKAGE
(TOP VIEW)



FUNCTION TABLE (each gate)

INPUTS		OUTPUT
A	B	Y
H	H	L
L	X	H
X	L	H

☞ This chip has four NAND gates, each with two inputs (hence the name Quad 2-input NAND). Look up the pin configuration in a data sheet.*** The output pins are 3, 6, 8, and 11. Connect V_{CC} to + 5 V. If you are using a prototyping board with built-in input-switches use these for the inputs, and use the built-in LEDs to display the outputs. At first use the CMOS chip.

*** It is a useful exercise to look up the pin diagram in the data sheet in a data book. This skill is necessary when you design a circuit yourself.

(a) NAND (Figure 8-1a)
(b) Inverter with NAND (Figure 8-1b)
(c) AND (Figure 8-1c)
(d) OR (Figure 8-1d)
(e) NOR (Figure 8-1e)
(f) Mystery NAND circuit (Figure 8-1f)

☛ See the pin diagram, above.

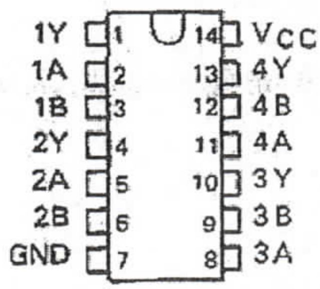
☛ ✍ Write down a Truth Table for each configuration listed above, and check off every state as you test it. Draw the gates.

☛ ✍ Finally, replace the CMOS chip with the TTL chip, and verify that the chip works the same, i.e., that the truth table for (a) is the same as for the CMOS chip.

❝ For the remaining exercises, you may use either TTL or CMOS.

4. Quad 2-Input NOR (74LS02 or 74HC02)

SN7402 . . . N PACKAGE
SN74LS02, SN74S02 . . . D OR N PACKAGE
(TOP VIEW)



FUNCTION TABLE (each gate)

INPUTS		OUTPUT
A	B	Y
H	X	L
X	H	L
L	L	H

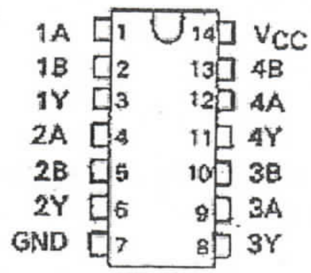
(a) NOR (Figure 8-2a)
(b) Inverter with NOR (Figure 8-2b)
(c) OR (Figure 8-2c)
(d) AND (Figure 8-2d)

☛ See the pin diagram, above.

☛ ✍ Write down a Truth Table and check off every state as you test it. Draw the gates.

4. XOR (74LS86 or 74HC86)

SN7486 . . . N PACKAGE
SN74LS86A, SN74S86 . . . D OR N PACKAGE
(TOP VIEW)



FUNCTION TABLE

INPUTS		OUTPUT
A	B	Y
L	L	L
L	H	H
H	L	H
H	H	L

H = high level, L = low level

(a) XOR from 7486 (Figure 8-3a)
(b) XOR using NAND only (Figure 8-3b)

☞ See the pin diagram, above.

☞ ✍ Write down a Truth Table and check off every state as you test it. Draw the gates.

5. Half Adder

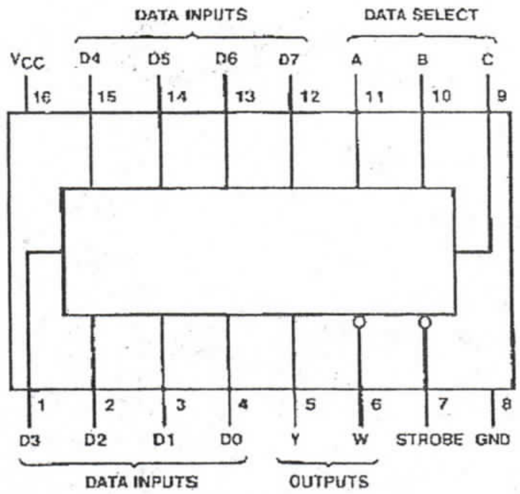
☞ See Figure 8-4. Connect this circuit.

☞ ✍ Write down a Truth Table and check off every state as you test it. Draw the gates.

6. `151 Multiplexer

} A multiplexer is a switch. It has multiple outputs and a single input. The purpose of the multiplexer is to allow the user to determine which of the inputs should be connected to the output.

Dual-In-Line Package



TL/F.

Inputs				Outputs	
Select			Strobe	Y	W
C	B	A	S		
X	X	X	H	L	H
L	L	L	L	D0	$\overline{D0}$
L	L	H	L	D1	$\overline{D1}$
L	H	L	L	D2	$\overline{D2}$
L	H	H	L	D3	$\overline{D3}$
H	L	L	L	D4	$\overline{D4}$
H	L	H	L	D5	$\overline{D5}$
H	H	L	L	D6	$\overline{D6}$
H	H	H	L	D7	$\overline{D7}$

H = High Level, L = Low Level, X = Don't Care
D0, D1...D7 = the level of the respective D input

Order Number 54LS151DMQB, 54LS151FMQB, 54LS151LMQB, DM54LS151J, DM54LS151W, DM74LS151M or DM74LS151N

☛ Connect the 74LS151 as shown below.

	DATA SELECTS			OUTPUT
binary	2^2	2^1	2^0	
state	C	B	A	Y
IC pin	9	10	11	5
board	SW1	SW2	SW3	LED1

☛ Set the following pins on the 74LS151 to LO:

```
Strobe   (pin 1)
All inputs except D1 and D2
```

☛ Set D_2 and D_1 to HI.

(a) Verification

☛✍ Verify that the output is LO unless D1 or D2 is selected as the address (i.e., the output Y is HI only when the address bits are correctly set).

(b) XOR

☛✍ Use the multiplexer chip to generate the truth table for XOR see **text** p. 495).

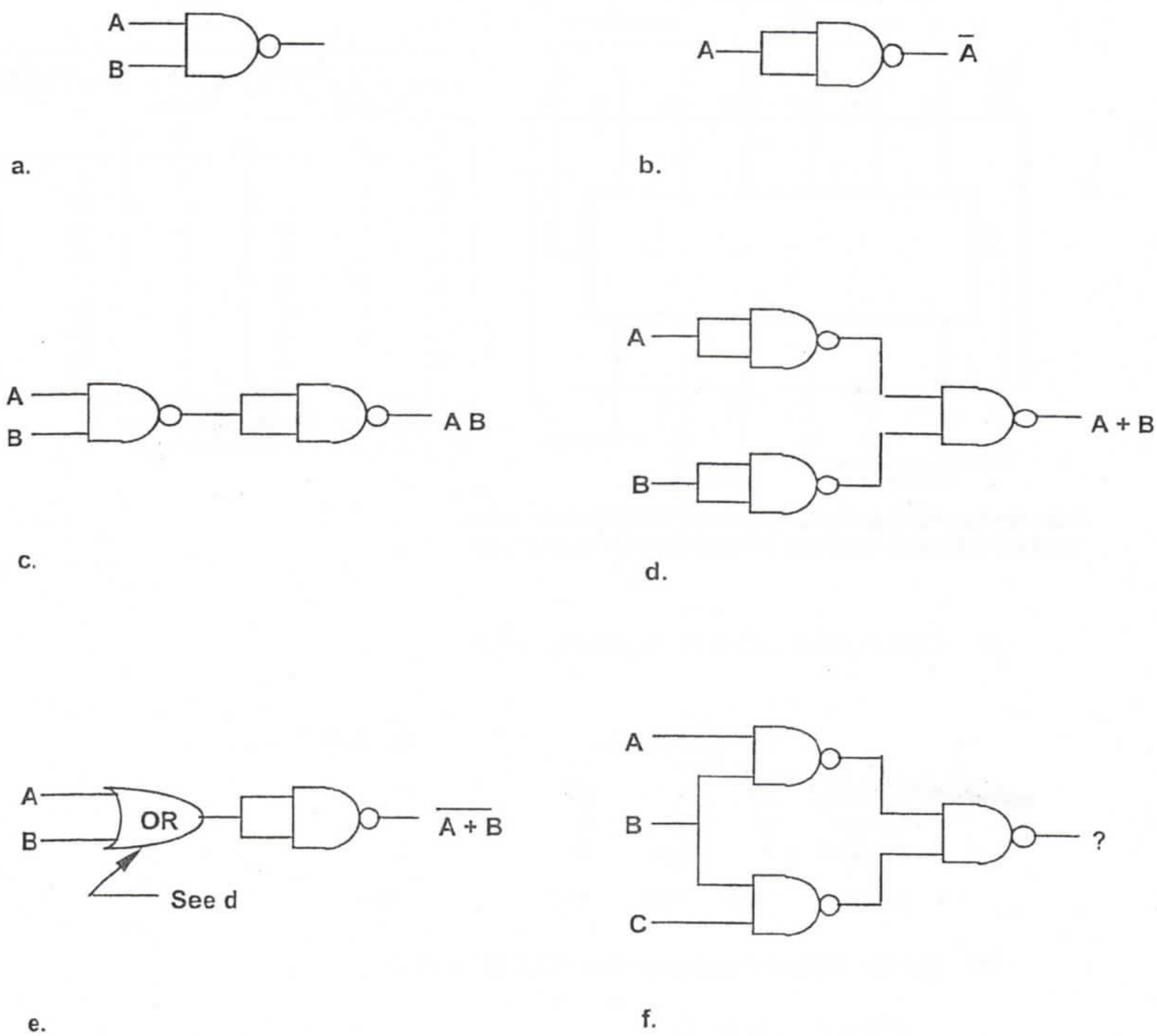


Figure 8-1 Nand gate circuits

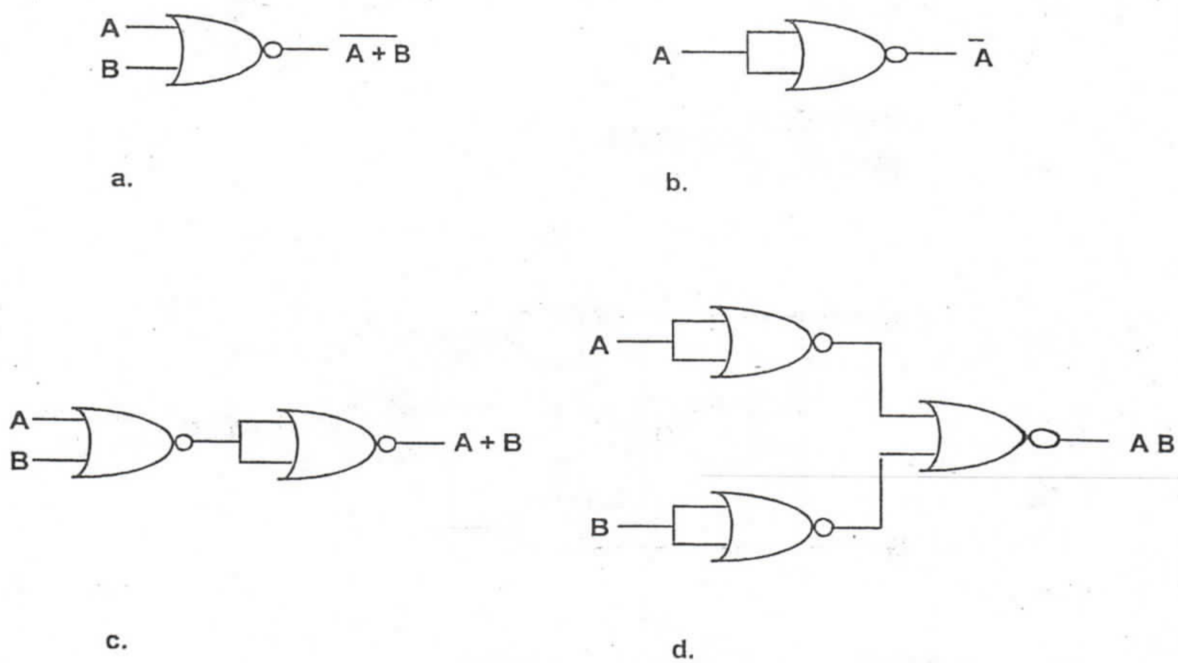
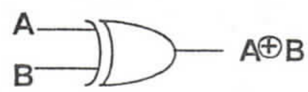


Figure 8-2 Circuits using NOR gates

a.



b.

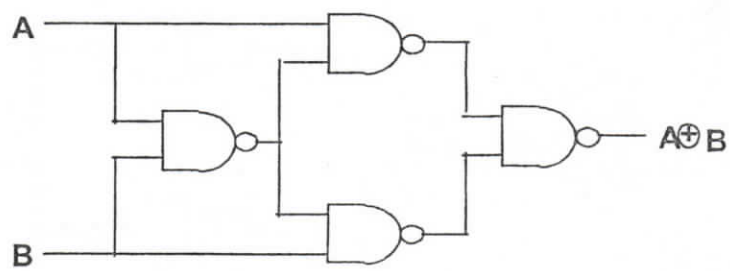


Figure 8-3 Different realizations of XOR

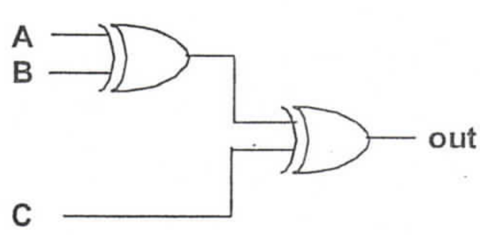


Figure 8-4 Half adder