Combinational Circuits

Aims

- to show how logic gates are combined on a chip
- to show how logic gates can be combined to produce useful circuits such as :- ***multiplexors*** and ***comparators***

2.0 Introduction

Previously, we saw how to implement simple circuits using individual logic gates, and how to specify the behaviour of the circuits using truth tables.

Gates are not manufactured individually but in units called ***integrated circuits*** (ICs or chips).

Chips can be divided into rough classes based on the number of gates they contain:

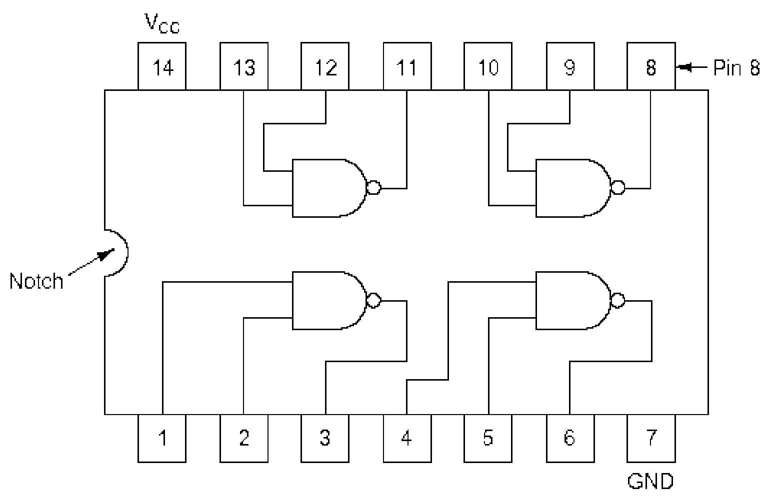
SSI	(Small Scale Integrated)	circuit	1 to 10 gates
MSI	(Medium Scale Integrated)	circuit	10 to 100 gates
LSI	(Large Scale Integrated)	circuit	100 to 100,000 gates
VLSI	(Very Large Scale Integrated)	circuit	> 100,000 gates

An ***SSI*** chip typically contains between two and six gates, each of which can be used individually.

Below is a diagram of the 7400 SSI chip containing four NAND gates.

Each of these gates has two inputs and one output.

Additionally the chip needs power (Vcc) and ground (GND) ; shared by all the gates.



Notice, the chip has a notch near pin one to identify the orientation.

In the 1970s, computers were constructed out of large numbers of these chips, but technological advances have allowed an entire CPU to be etched onto a single chip.

The current state of the art allows nearly 10 million gates on a chip.

2.1 Combinational Circuits

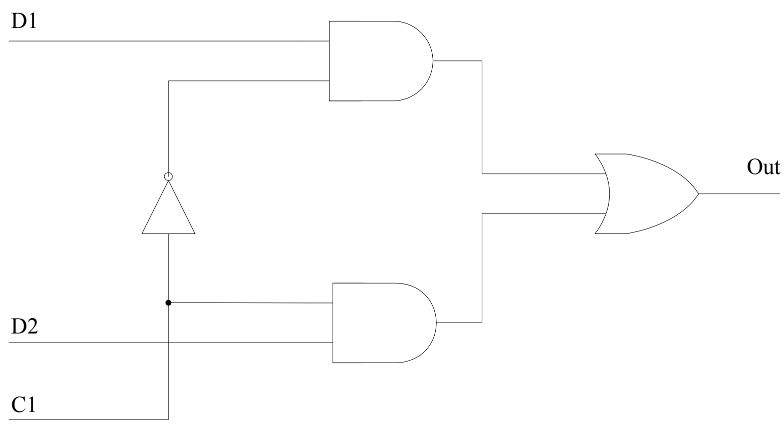
Many circuits have the property of their outputs being wholly dependent on their inputs.

Such circuits are called ***combinational circuits***.

Over the next few weeks we will examine some frequently used combinational circuits.

2.1.1 Multiplexors

The circuit below is a ***multiplexor*** with two inputs and one output. (a 2-to-1 multiplexor)



D1	D2	C1	$\overline{C1}$	D2.C1	D1.$\overline{C1}$	Out
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

If you carefully examine the truth table you can see that the
output is either D1 or D2, depending on the value of C1.

Therefore, the truth table reduces to:

C1	Out
0	D1
1	D2

Therefore, we can see that the
function of a <u>multiplexor</u> is to <u>route</u> one of many data inputs to a single output.

The ***control input (C1) determines which of the data inputs is passed through to the output***.
A **2^n**-to-1 multiplexor has **2^n** data inputs and ***n*** control inputs.

The bit combination on the control inputs determines which data input is selected for output.

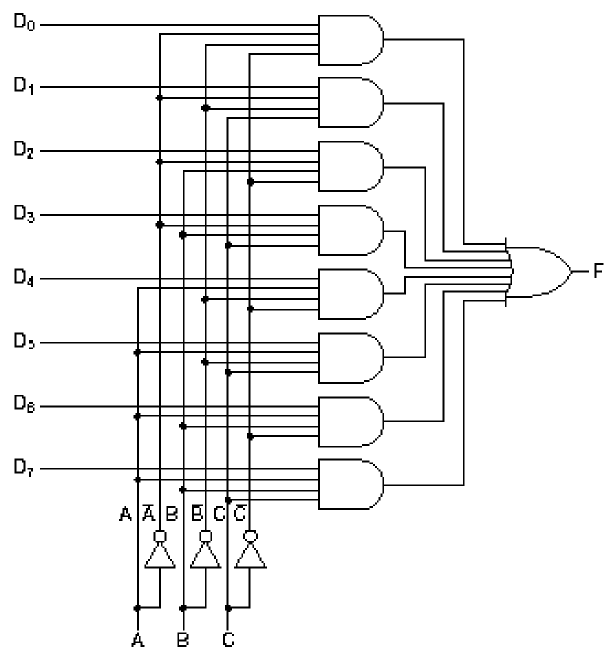
An eight-input multiplexor is shown below.

- Each of the eight data inputs (D_0 to D_7) are applied to one input of an **AND** gate.
- The three control inputs (**A**, **B**, **C**) are decoded to select a particular **AND** gate.
- The outputs of the **AND** gates are applied to a single **OR** gate to provide the single output.

Consider the case when **ABC = 101** (**A** = 1, **B** = 0, **C** = 1).

- In this case the **AND** gate associated with D_5 will have three inputs equal to **1**.
- The fourth input of the gate is connected to D_5.
- The other seven **AND** gates will have at least one input equal to **0**, which makes their outputs equal to **0**.
- The **OR** gate output is now equal to the value of D_5, thus providing a path from the selected input to the output.

A	B	C	Output
0	0	0	D0
0	0	1	D1
0	1	0	D2
0	1	1	D3
1	0	0	D4
1	0	1	D5
1	1	0	D6
1	1	1	D7

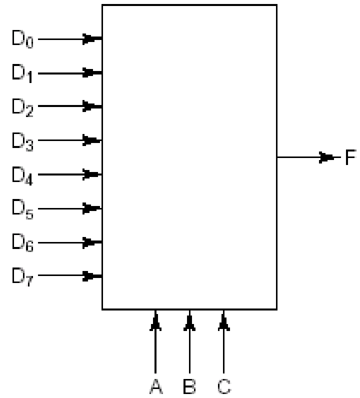


SAQ 2.1

1. What is the function of a multiplexor?
2. How many outputs does a 4-input multiplexor have?
3. How many control inputs does a 4-input multiplexor have?
4. How many outputs does an 8-input multiplexor have?
5. How many control inputs does an 8-input multiplexor have?

answer

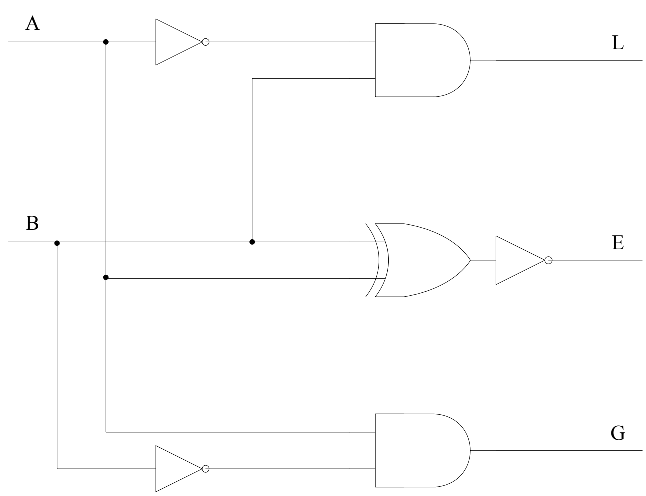
A simpler way to diagrammatically represent an ***eight-input multiplexor*** is shown below.



2.2 Comparators

A ***comparator*** *compares two inputs and outputs* :

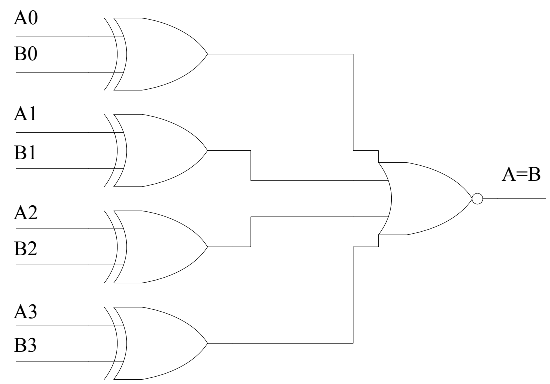
a **1** appears on the **E** output if the inputs are equal.



A	B	$\overline{A}$	$\overline{B}$	L	E	G
0	0					
0	1					
1	0					
1	1					

The ***4-bit comparator*** takes two inputs, **A** and **B**, each of length **4 bits**, and outputs a **1** if they are equal and a **0** *if they are not equal*.

- The circuit is based on the **XOR** gate, which outputs a **0** if its inputs are the same and a **1** if they are not the same.
- If the two inputs are the same, all four of the **XOR** gates will output **0**.
- Finally, the four signals are **NOR**ed together.
- If the result is **1**, the inputs are equal. If the result is **0**, the inputs are not equal.

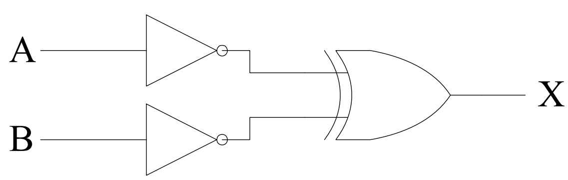


A0	B0	A1	B1	A2	B2	A3	B3	A0 ⊕ B0	A1 ⊕ B1	A2 ⊕ B2	A3 ⊕ B3	NOR
0	0	0	0	0	0	0	0					
0	1	0	1	0	1	0	1					
1	0	1	0	1	0	1	0					
1	1	1	1	1	1	1	1					

SAQ 2.2

Below is a simple combinational circuit.

Under what circumstances does the circuit output a **1**?



answer

References

http://worldclassprogramme.com/Arithmetic-Circuits.php