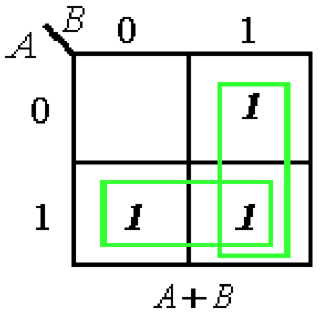
Implementation of Karnaugh Map

1. Start with the truth table or Boolean expression, if you have one.
2. If starting from the truth table, write the Boolean expression for each truth table term that is 1 (if SOP) or 0 (if POS).
3. Develop the Boolean expression, if necessary, by OR-ing the AND terms together if SOP or AND-ing OR terms if POS.
4. On the K-Map, plot 1’s (for SOP) or 0’s (for POS).
5. Group implicants together to get the largest set of prime implicants possible.
6. Prime implicants may overlap each other. They will always be square or rectangular groups of cells that are powers of 2.(1, 2, 4, 8).
7. The variables that make up the term(s) of the new expression will be those which do not vary in value over the extent of each prime implicant.
8. Write the Boolean expression for each prime implicant and then OR (for SOP) or AND (for POS) terms together to get the new expression.

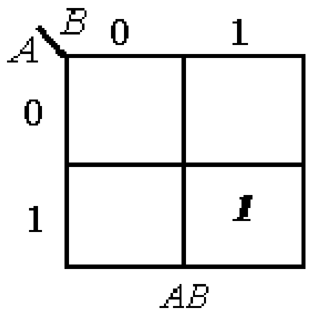
Karnaugh Maps

- The Karnaugh map is a graphical method of simplifying Boolean algebra expressions.
- A diagram consisting of a rectangular array of squares each representing a different combination of the variables of a Boolean function.
- With two input variables there are four line in the truth table, there are eight lines for three variables and 16 lines for four input variables.

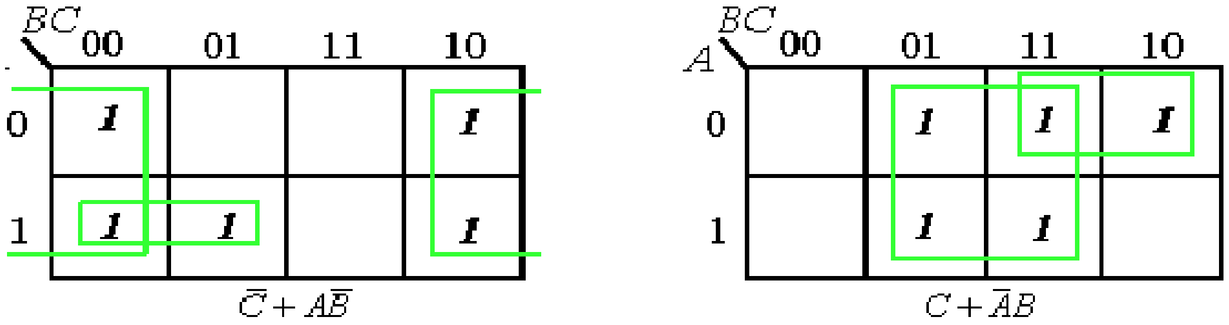
A two-input OR gate



A two -input AND gate



Examples of 3 - variable expressions



The 4 variable K-Map

AB \ CD	00	01	11	10
00	A’B’C’D’	A’B’C’D	A’B’CD	A’B’CD’
01	A’BC’D’	A’BC’D	A’BCD	A’BCD’
11	ABC’D’	ABC’D	ABCD	ABCD’
10	AB’C’D’	AB’C’D	AB’CD	AB’CD’

Continued……

EX.1

	B’	B
A’		
A	1	

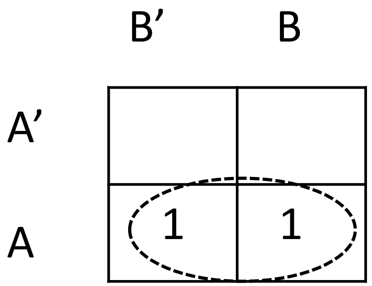
A	B	Output	Products
0	0	0	A’.B’
0	1	0	A’.B
1	0	1	A.B’
1	1	0	A.B

- The expression given as, Output=A.B’
- 0’s represented by complemented variable and 1’s by uncomplemented variable

- To implement the above function, Whenever the output function comes as AB’ considered as 1, others as 0.

EX.2

A	B	Output	Products
0	0	0	A’.B’
0	1	0	A’.B
1	0	1	A.B’
1	1	1	A.B



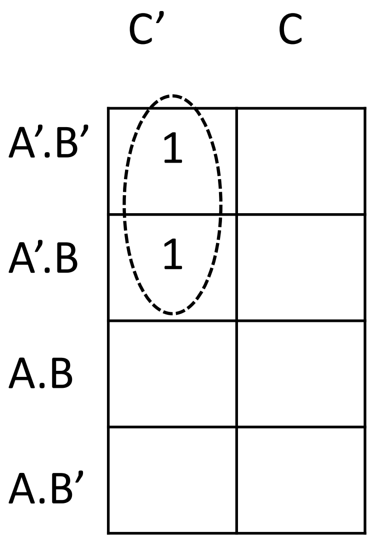
	B’	B
A’		
A	1	1

Output = A.B’+A.B

Can be simplified to

A.B’+A.B =A.(B’+B) = A

Three variable and Four Variable map

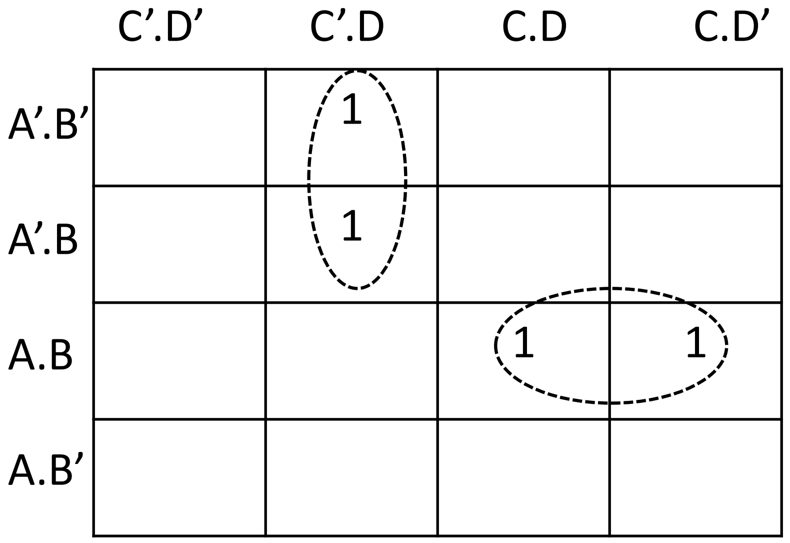


	C’	C
A’.B’	1	
A’.B	1	
A.B		
A.B’		

A	B	C	O/P	Product
0	0	0	1	A’.B’.C’
0	0	1	0	A’.B’.C
0	1	0	1	A’.B.C’
0	1	1	0	A’.B.C
1	0	0	0	A.B’.C’
1	0	1	0	A.B’.C
1	1	0	0	A.B.C’
1	1	1	0	A.B.C’

Four variable Map

Output = A’C’D+ABC



	C’.D’	C’.D	C.D	C.D’
A’.B’		1		
A’.B		1		
A.B			1	1
A.B’				

Output= A’B’C’+A’BC’

= A’C’(B+B’) since, B+B’=1(according to Boolean axiom)

= A’C’

Real time Applications of Boolean Algebra

Mainly used in automated machines

Step:1

Input and output variables to be specified.
C- Coffee button (1-pressed, 0-not pressed
T-tea button, M-Milk button , X- choice verifier

Step:2

Construct a truth table giving the output desired for each input. X is 1 when exactly one of C , T and M is 1.

C	T	M	X
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

EX.1 Automated Cafeteria orders a machine to dispense Coffee, Tea and Milk. Design the machine so that it has a button for each choice and so that a customer can have at most one of the 3-choices.

Design the circuit to ensure that the “at most one” condition is met.

Step:3

Write a Boolean expression with a term for each ‘1’ output row of the truth table

X=C’T’M+C’TM’+CT’M’

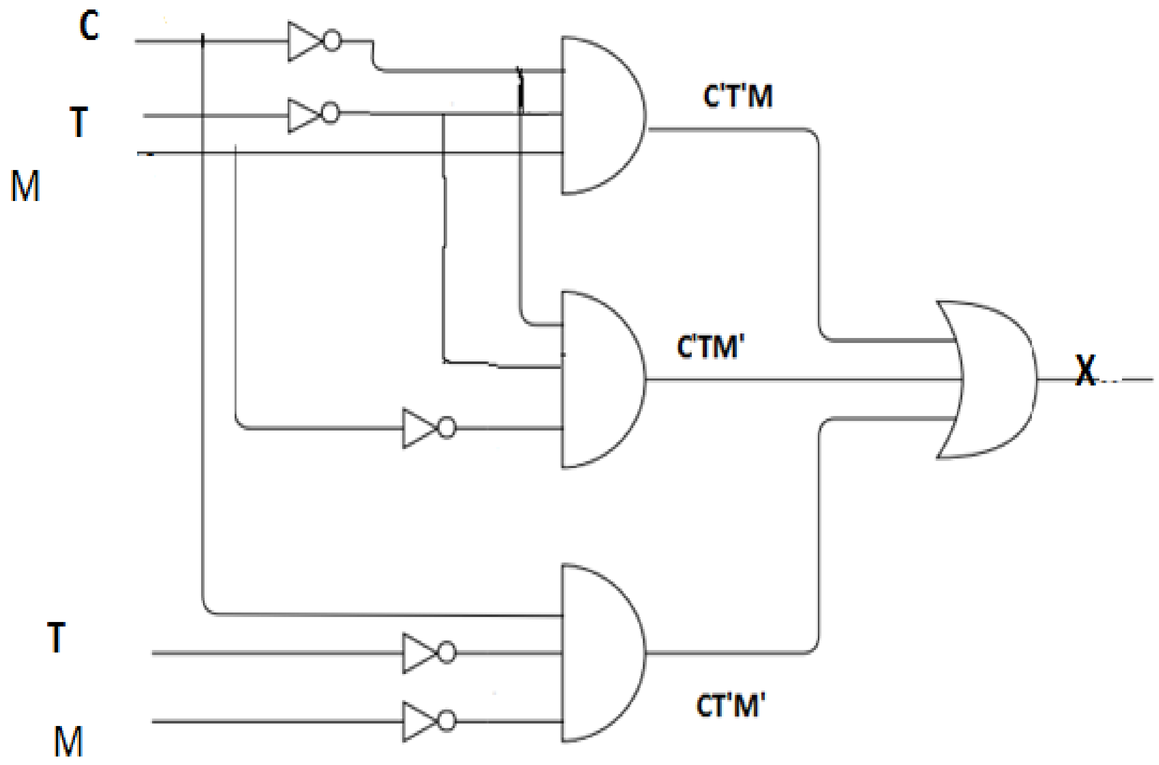
Step:4

Try to simplify using Karnaugh map

	T’M’	T’M	TM	TM’
C’		1		1
C	1			

Not possible to simplify using K-MAP

Logic Circuit implementation



C- Coffee button (1-pressed, 0-not pressed)
T-tea button
M-Milk button
X- choice verifier

Output X= C’T’M+C’TM’+CT’M’

Exercise: Revise the above example , so that the machine offers at most one of four choices (add Green tea/ hot Chocolate with the existing one)

Practice Problems

1.How many minimum number of two input NAND gates that will be required to implement the given function

$Y=ab+cd$

2.The Boolean expression for X is $X=A'B+CD+(A+B)'(ACD+BE)$. Draw the logic circuit of the Boolean expression.

3. If chimney is not blocked and the house is cold and the pilot light is lit, then open the main fuel valve to start boiler. b = chimney blocked c = house is cold p = pilot light lit v = open fuel valve

4. Simplify. $p= x.y + y'.z + x.z + x.y.z$ and draw the logic circuit.

5. Simplify $ab' + a.(b + c)' + b(b + c)'$

6. Simplify using Karnaugh map technique.

1. $F1=\Sigma m(0,1,3)$

2. $F2=\Sigma M(0,1,4,5,7,10,11)$

7. Design a Half Subtractor, Full Subtractor logic circuit using a suitable gates.

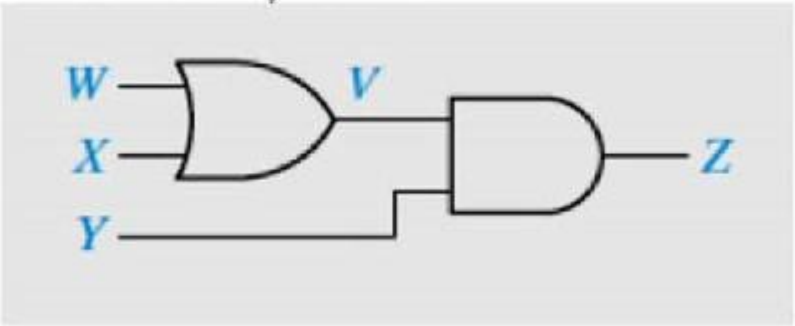
8.Implement a EX-OR gate using NAND gate.

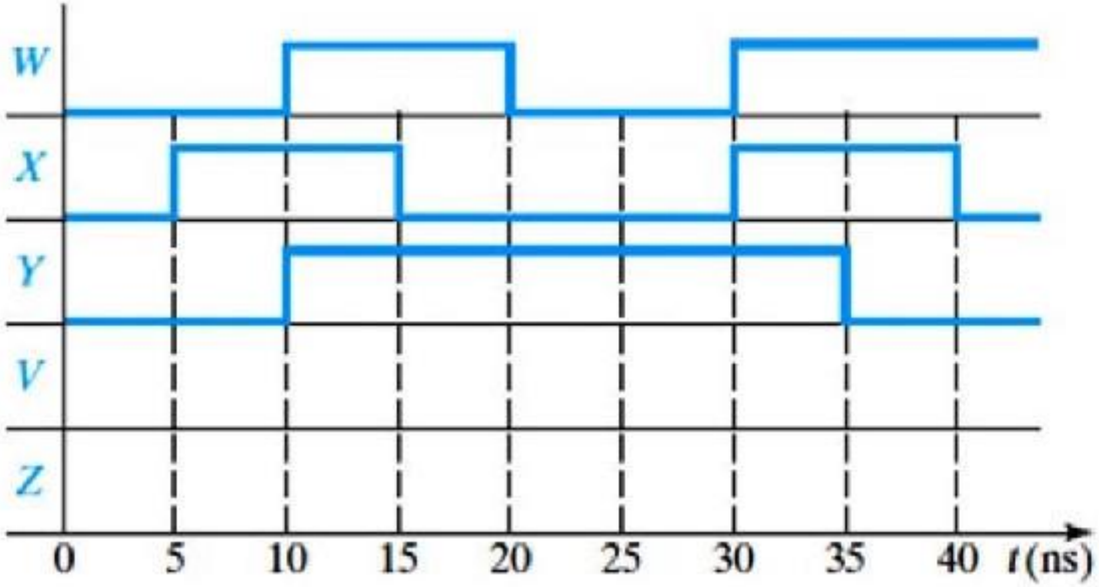
9. How many gates would be required to implement the following Boolean expression before simplification and after simplification? $F= XY + X(X + Z) + Y(X + Z)$.

10. Design a 4-bit even parity generator using a suitable logic.

Continued….

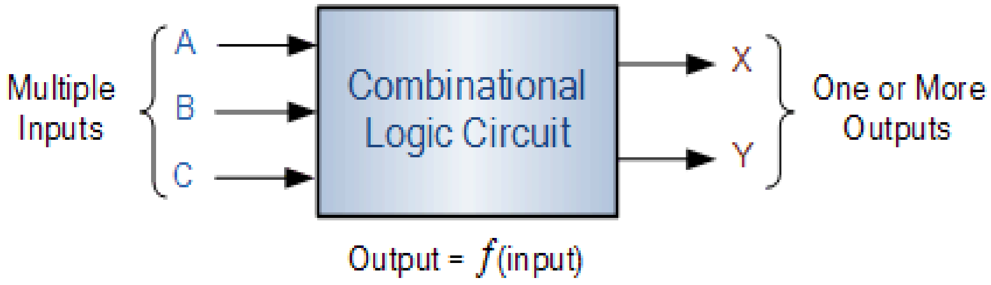
« 3 » A) Draw the timing diagram of *V* and *Z* for the circuit. Assume that the logic gates are ideal and delay is zero:

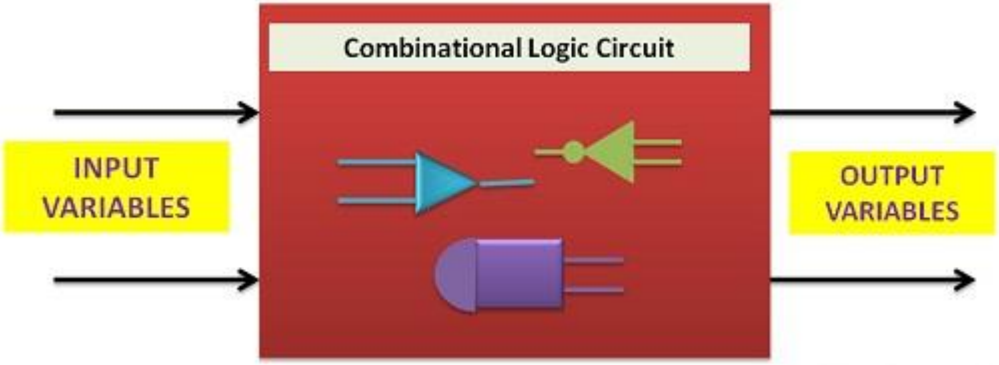




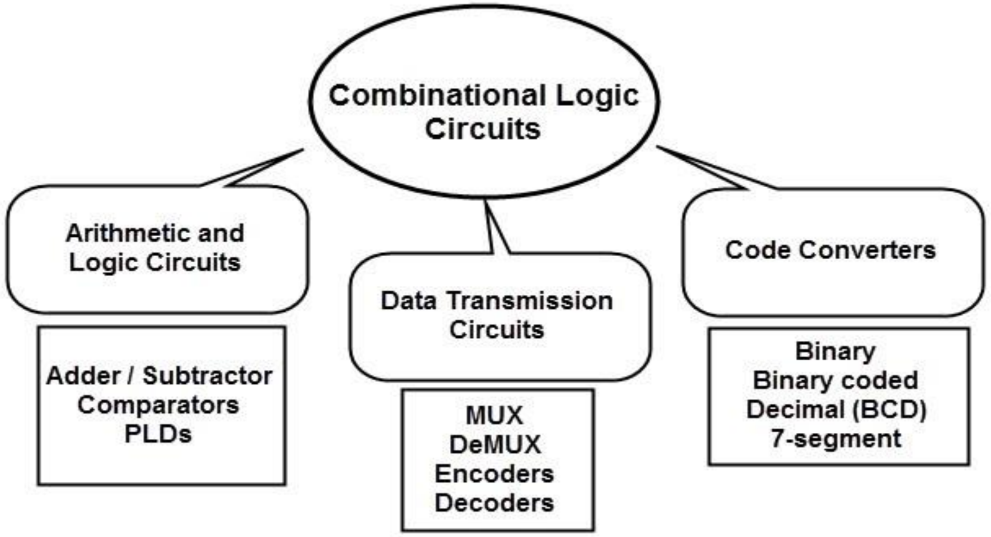
Comparison Of Combinational and Sequential logic Circuit

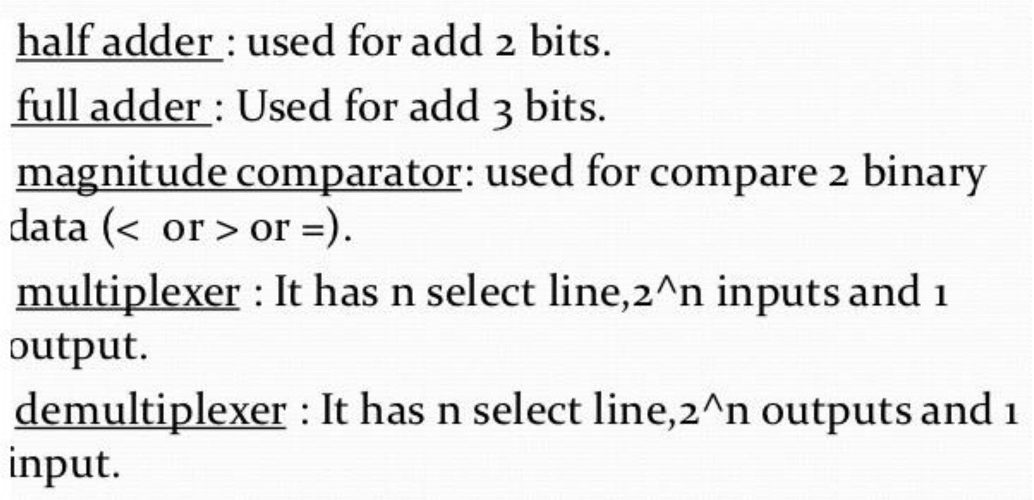
Logic circuits for digital systems may be Combinational or Sequential Circuits.





- Output is a function of only the present inputs
- Does not have state information
- Does not require any memory.
- The outputs of **Combinational Logic Circuits** are only determined by the logical function of their current input state, logic “0” or logic “1”, at any given instant in time.


Combinational Logic Circuits
Arithmetic and Logic Circuits
Adder / Subtractor
Comparators
PLDs
Data Transmission Circuits
MUX
DeMUX
Encoders
Decoders
Code Converters
Binary
Binary coded Decimal (BCD)
7-segment



half adder : used for add 2 bits.

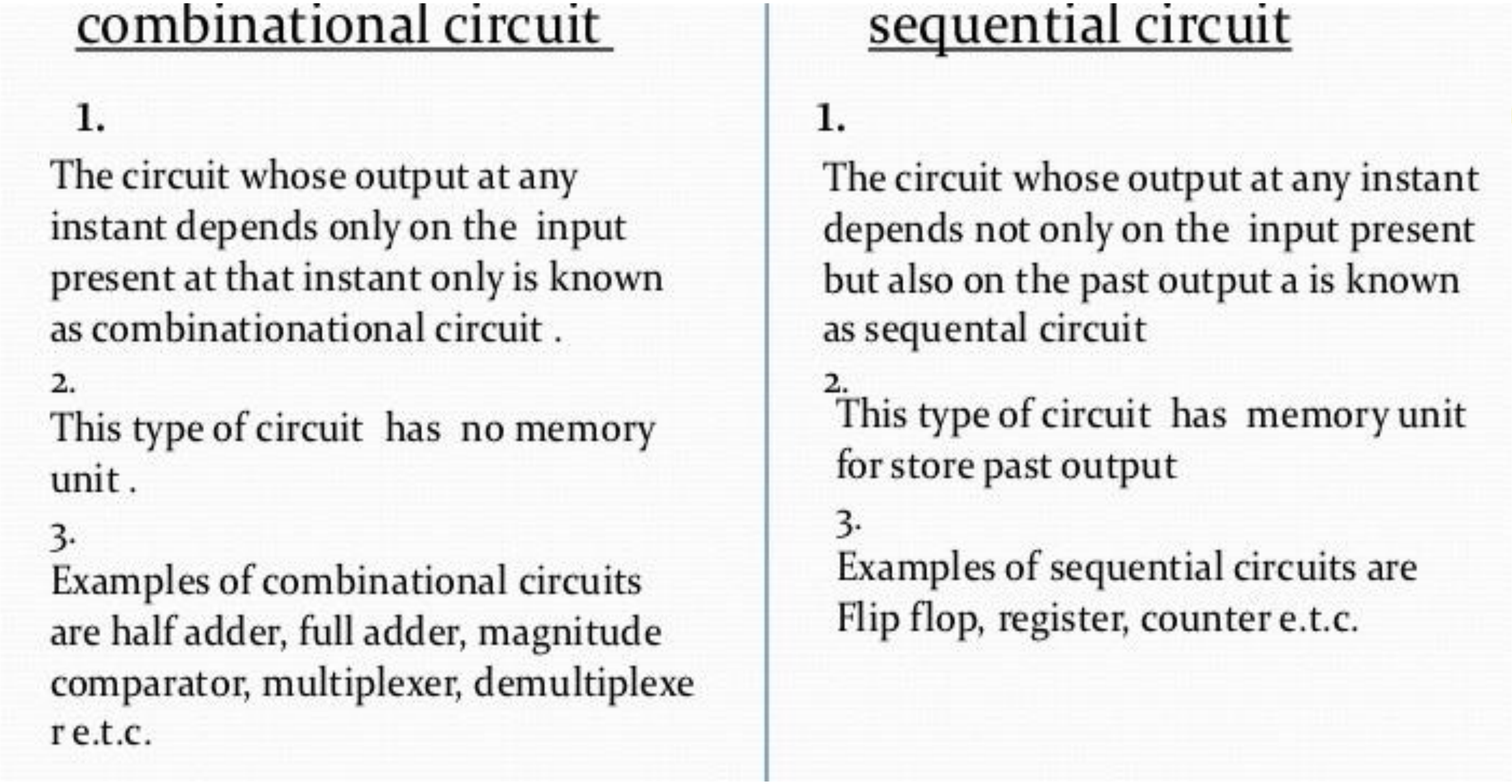
full adder : Used for add 3 bits.

magnitude comparator: used for compare 2 binary data (< or > or =).

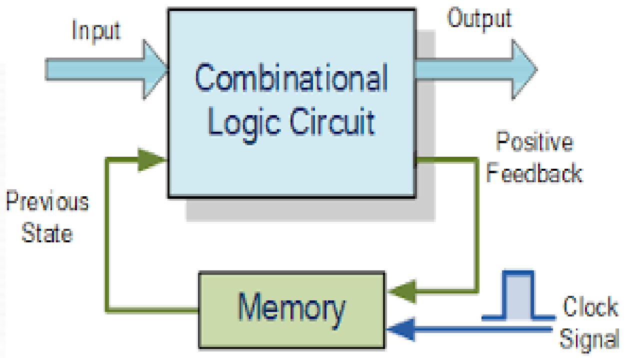
multiplexer : It has n select line,2^n inputs and 1 output.

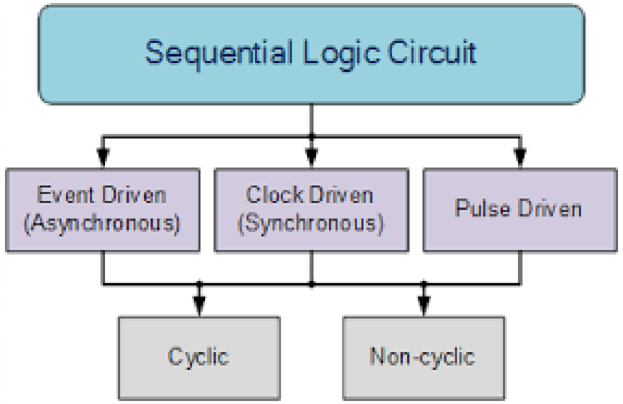
demultiplexer : It has n select line,2^n outputs and 1 input.

Continued....

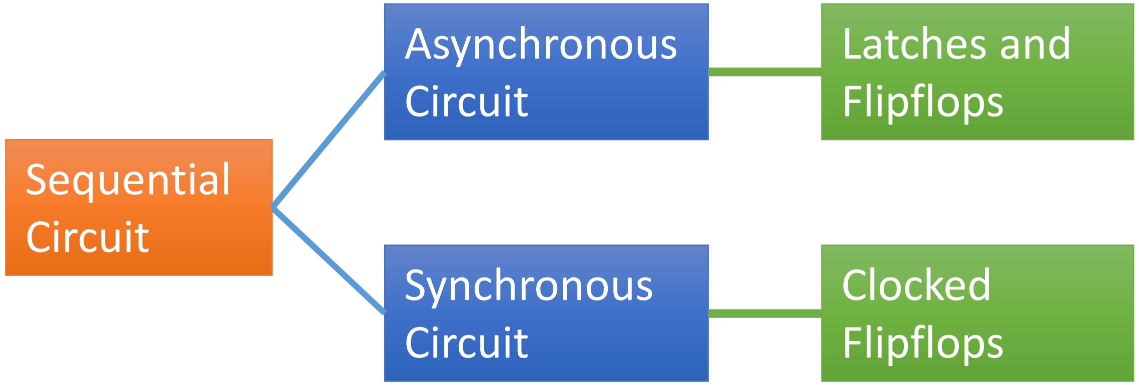


combinational circuit	sequential circuit
1. The circuit whose output at any instant depends only on the input present at that instant only is known as combinational circuit .	1. The circuit whose output at any instant depends not only on the input present but also on the past output a is known as sequental circuit
2. This type of circuit has no memory unit .	2. This type of circuit has memory unit for store past output
3. Examples of combinational circuits are half adder, full adder, magnitude comparator, multiplexer, demultiplexer e.t.c.	3. Examples of sequential circuits are Flip flop, register, counter e.t.c.





- **Synchronous**: The behavior of the circuit depends on the input signal at discrete instances of time *(also called clocked)*
- **Asynchronous**: The behavior of the circuit depends on the input signals at any instance of time and the order of the inputs change



Some of the Combinational Circuits…

1.Parity generator

It is a Combinational circuit that generates the parity

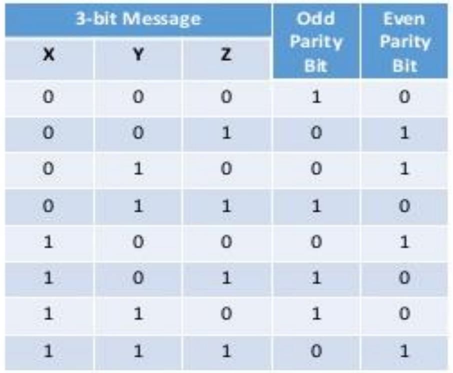
Bit in the transmitter.

Used for the purpose of detecting errors during transmission

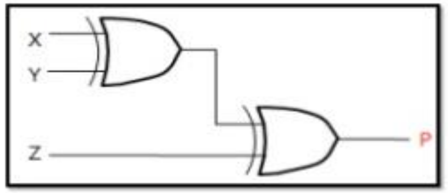
Of binary information

Parity bit is an extra bit included with a binary message

To make the number of 1's either odd or even



3-bit Message			Odd Parity Bit	Even Parity Bit
X	Y	Z		
0	0	0	1	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	0	1



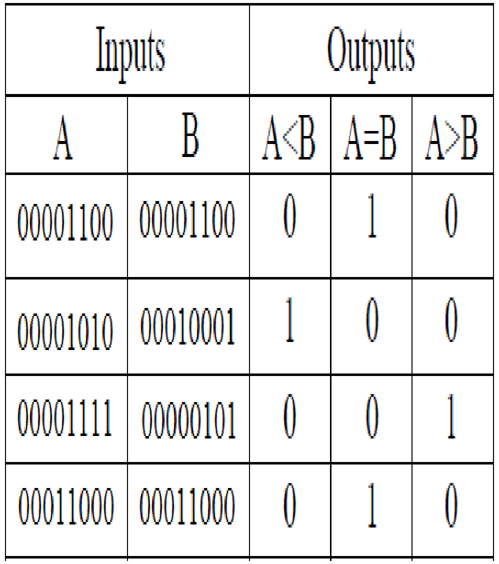
2. Digital Comparator

It compares two digits or binary numbers

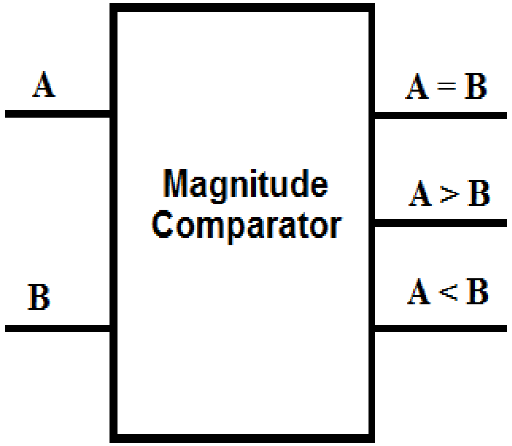
It determines their relative magnitudes in order

to find out whether one number is equal, less than

or greater than the other digital number.



Inputs		Outputs		
A	B	A<B	A=B	A>B
00001100	00001100	0	1	0
00001010	00010001	1	0	0
00001111	00000101	0	0	1
00011000	00011000	0	1	0

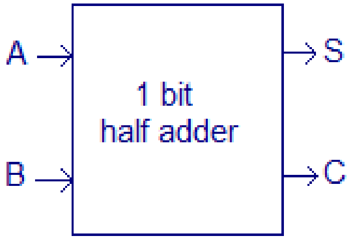


Note:https://www.tutorialspoint.com/computer_logical_organization/pdf/error_codes.pdf
https://www.electronicshub.org/parity-generator-and-parity-check/

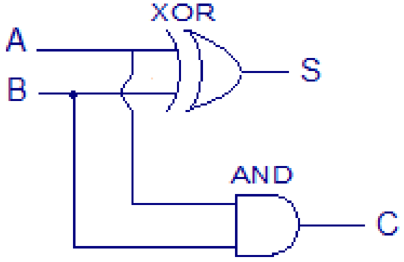
Continued….

Inputs		Outputs	
A	B	S	C
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

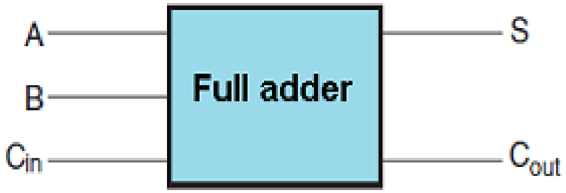
Truth table



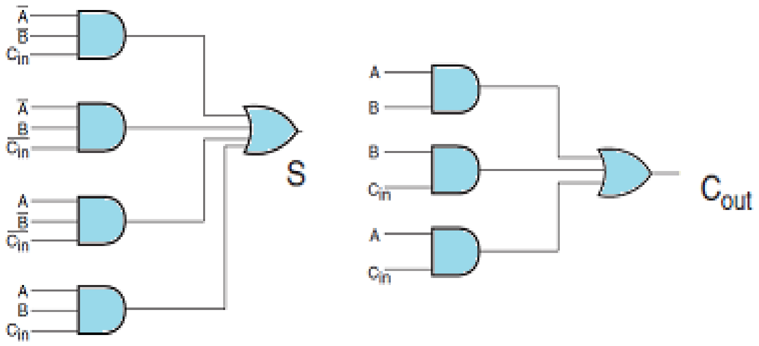
Schematic

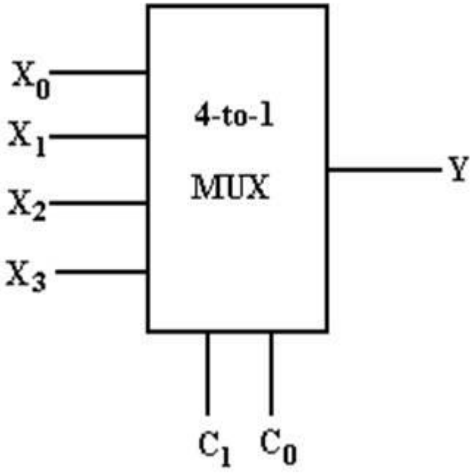


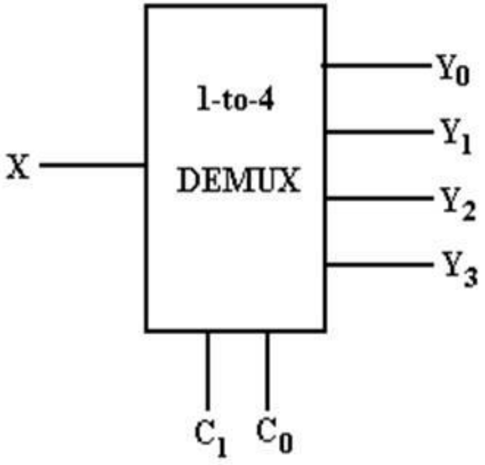
Realization

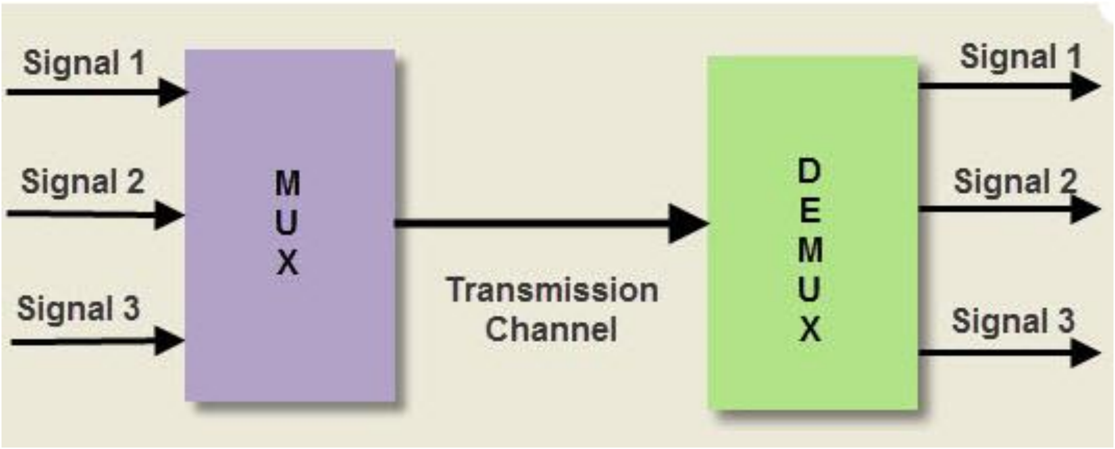


A	B	C_{in}	SUM (S)	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1





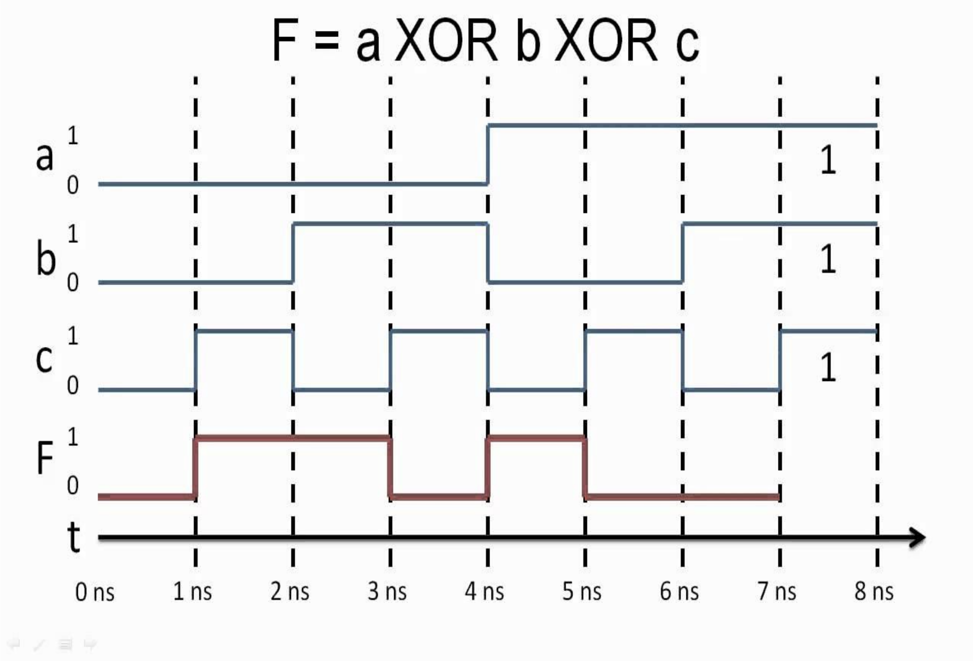


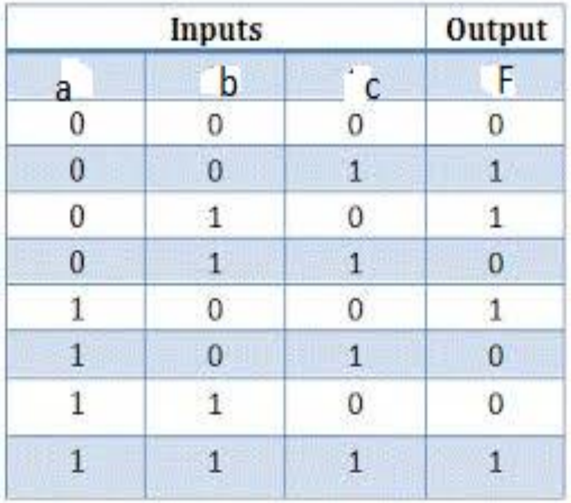


Timing diagram of digital Circuits

Timing Diagram

- It is the representation of a set of signals in the time domain.
- It illustrates the logical behaviour of signals as a function of time.
- A signal timing diagram may contain many different delay specifications.
- The delay depends on the internal circuit structure, Logic family type, Source Voltage and temperature.
- Tool used in Digital Electronics, hardware debugging and Digital Communications.





Inputs			Output
a	b	c	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Thank you