SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with 'A' Grade by NAAC

Lecture session 1_ UNIT-3

SCSA1201-FUNDAMENTALS OF DIGITAL SYSTEMS

Unit-3-COMBINATIONAL LOGIC

Topic 1: ADDER

By
V.GEETHA
ASSISTANT PROFESSOR/EEE
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
CHENNAI-119

UNIT 3 COMBINATIONAL LOGIC

Introduction to Combinational circuits

TOPIC1:

-Half Adder, Full Adder - Half Subtractor, Full Subtractor

-Parallel binary Adder,-Parallel binary Subtractor

TOPIC 2

-Carry look ahead Adder

-BCD Adder

TOPIC 3

- Decoders

-Encoders

-Priority Encoder

-TOPIC 4

- Multiplexers-

MUX as universal combinational modules

-Demultiplexers

TOPIC 5

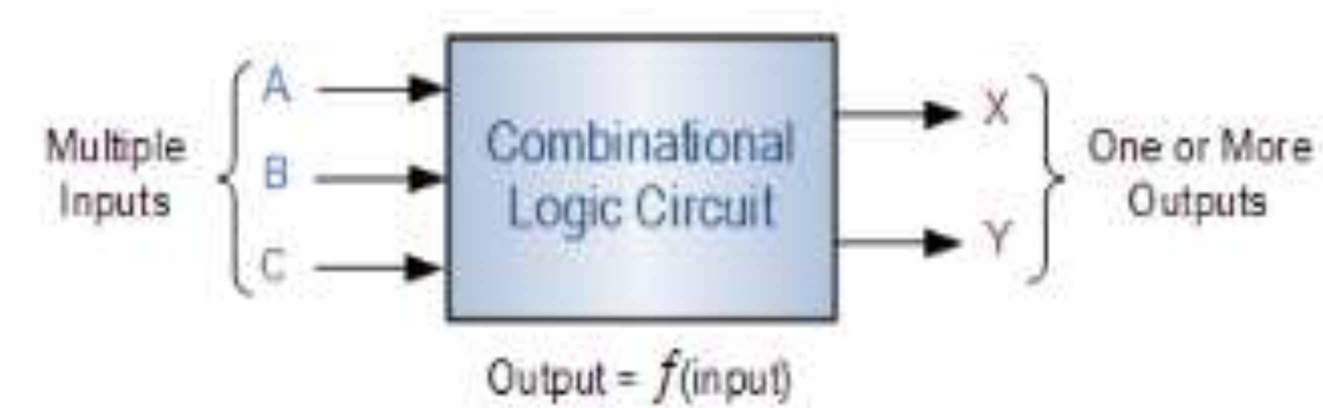
-Code convertors

- Magnitude Comparator.

INTRODUCTION TO COMBINATIONAL CIRCUITS

Definition:

when logic gates are converted together to produce a specified output for a certain specified combinations of input variables, with no storage involved, the resulting circuit is called "COMBINATIONAL CIRCUIT".



Design Procedure

1. Problem definition
2. Determine required number of inputs and outputs from the specifications.
3. Assigning letters and symbols to input and output variable
4. The derivation of truth table indicating the relationship between the input and output variables
5. Obtain the simplified Boolean expressions for each output.
6. Obtain the logical diagram.

EXAMPLE:

Design a combinational logic circuit with three input variables that will produce a logic 1 output when more than one input variables are logic 1.

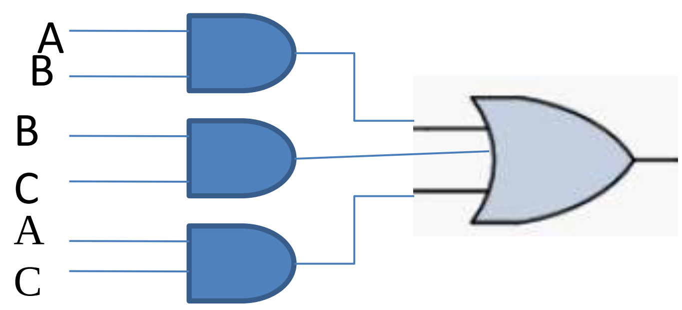
No of inputs and no of outputs
inputs = 3
output = 1

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

K-MAP TO FIND THE RELATION BETWEEN INPUT AND OUTPUT

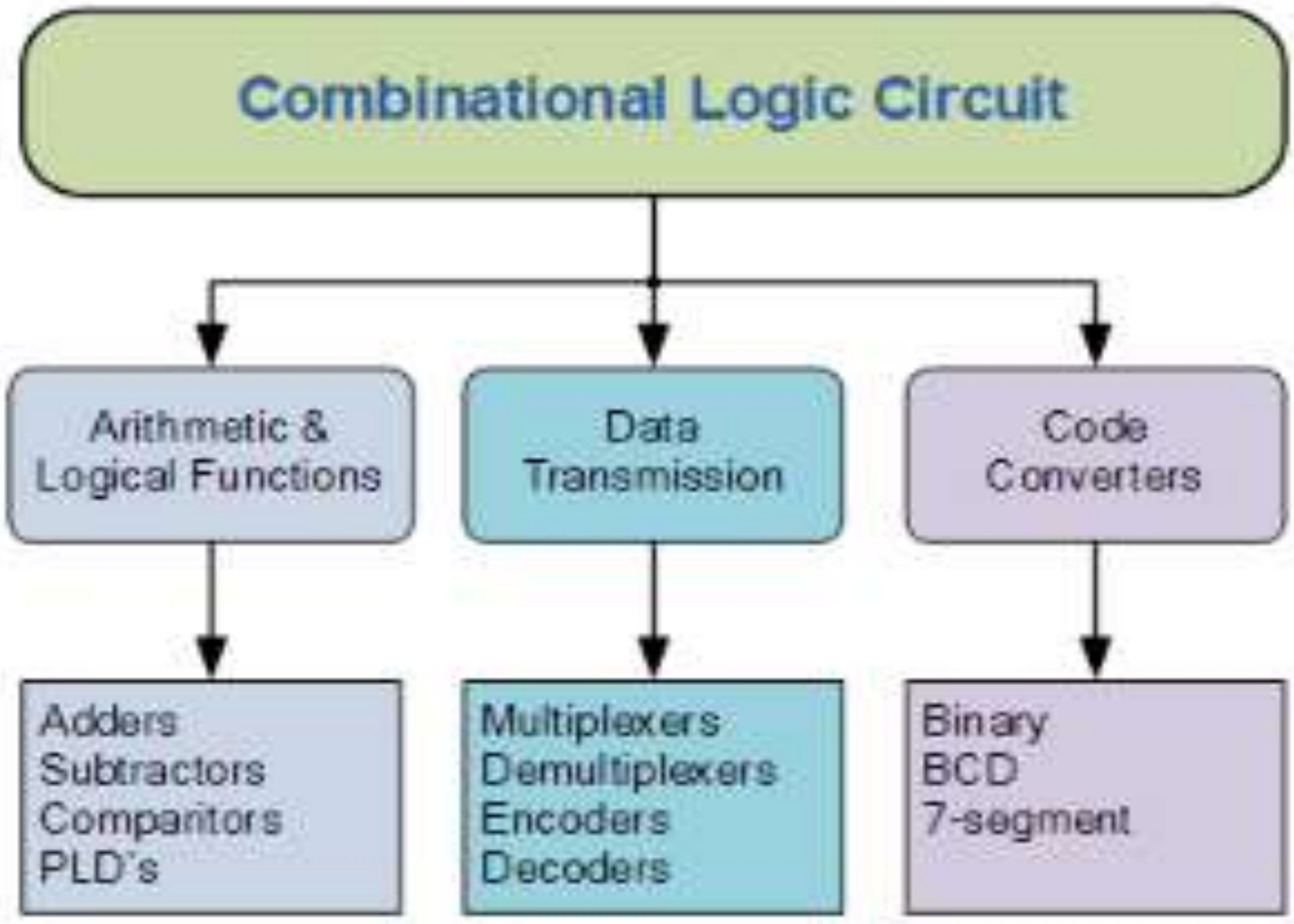
	B'C'	B'C	BC	BC'
A'	0	0	1	0
A	0	1	1	1

Y=AB+BC+AC



COMBINATIONAL CIRCUIT

CLASSIFICATION OF COMBINATIONAL CIRCUITS



What is an Adder?

An adder is a kind of calculator that is used to add two binary numbers. When I say, calculator, I don’t mean one with buttons, this one is a circuit that can be integrated with many other circuits for a wide range of applications.

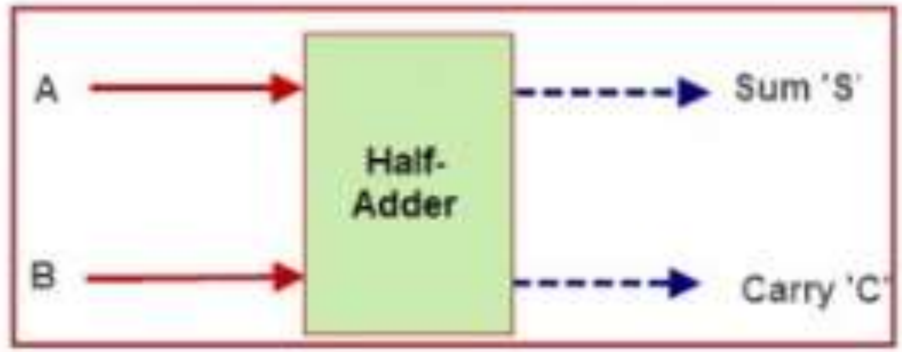
There are two kinds of adders;

Half adder

Full adder

HALF ADDER

0+0 = 0
0+1 = 1
1+0 = 1
1+1 = 10



Half Adder

These are the least possible single-bit combinations. But the result for 1+1 is 10, the sum result must be re-written as a 2-bit output. Thus, the equations can be written as

0+0 = 00
0+1 = 01
1+0 = 01
1+1 = 10

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

B \ A	0	1
0	0	1
1	1	0

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

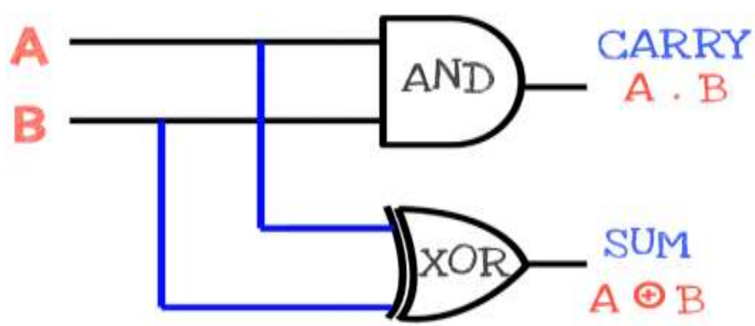
AND

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

B \ A	0	1
0	0	0
1	0	1

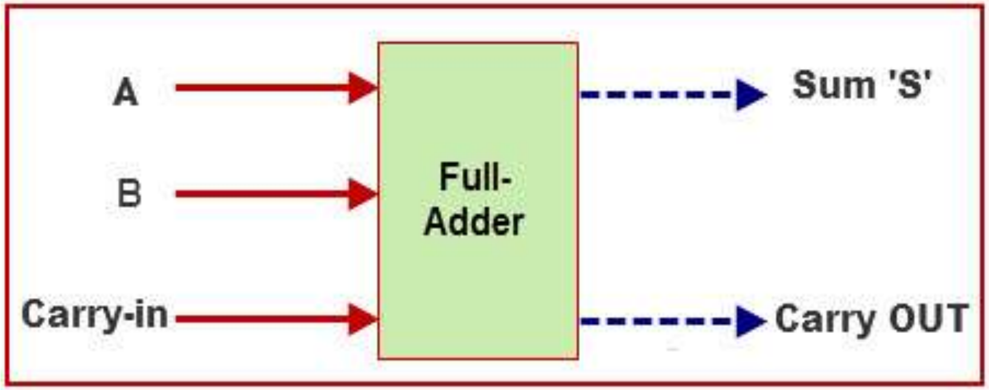
$A.B$

Half Adder



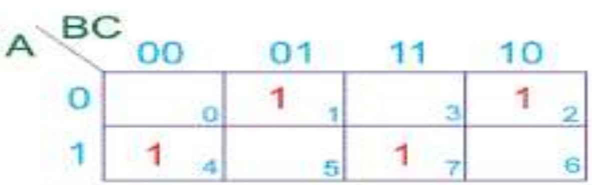
FULL ADDER

The difference between a half-adder and a full-adder is that the full-adder has three inputs and two outputs, whereas half adder has only two inputs and two outputs. The first two inputs are A and B and the third input is an input carry as C-IN. When a full-adder logic is designed, you string eight of them together to create a byte-wide adder and cascade the carry bit from one adder to the next.

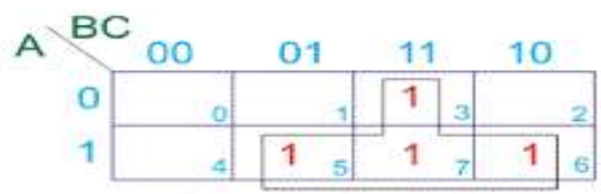


INPUTS			OUTPUT	
A	B	C-IN	C-OUT	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

From the above table, we can draw K-map for sum (s) and final carry (C_{out}).



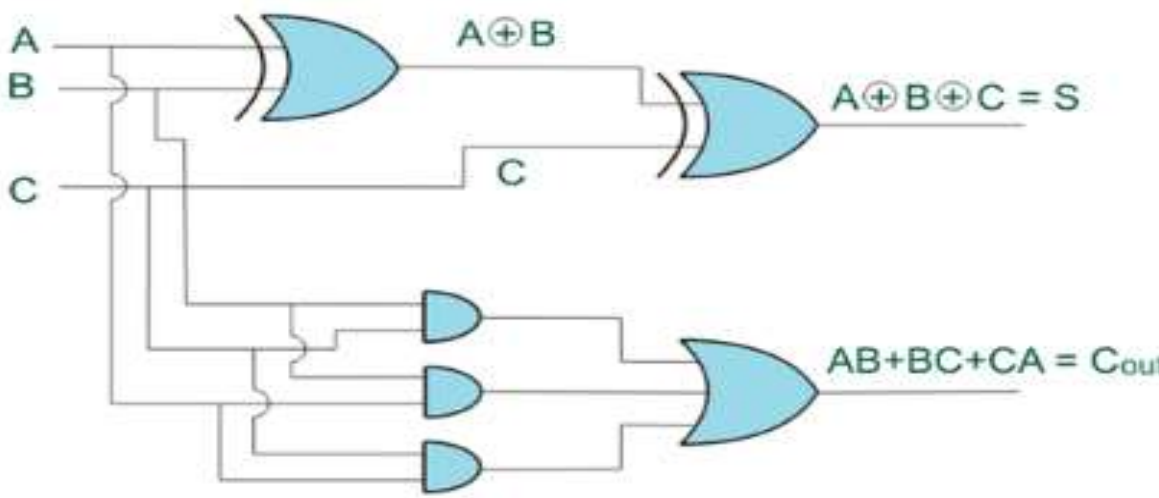
K-map for Sum (S)



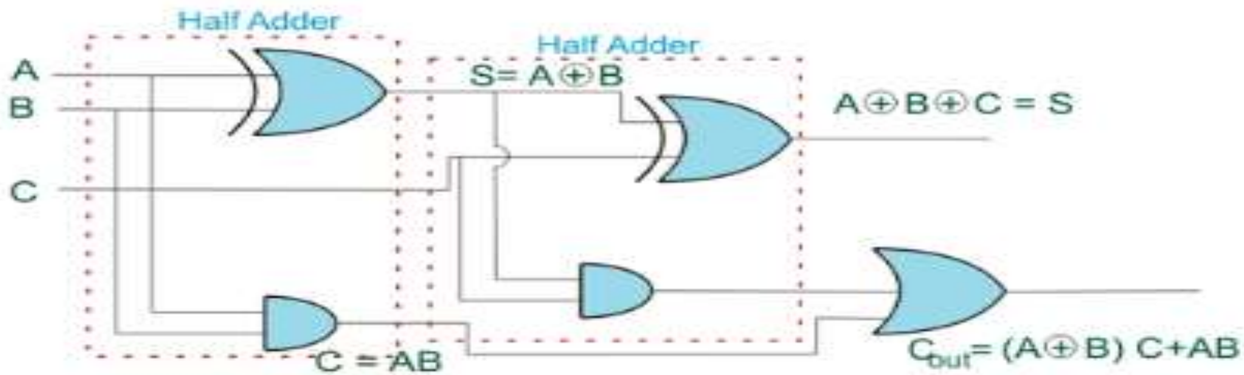
K-map for Carry (C_{out})

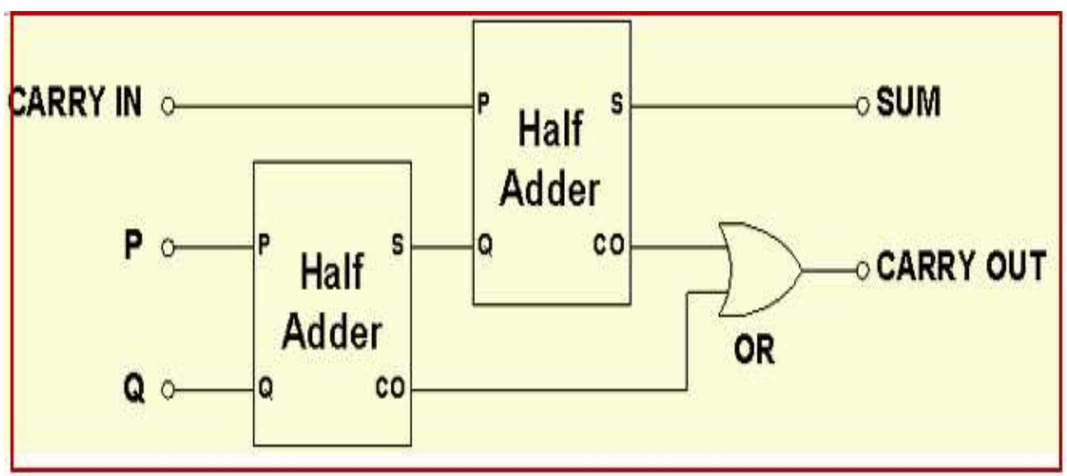
Hence, from K-maps,

$$
\begin{aligned}
S &= A\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + ABC + \overline{A}B\overline{C} \\
&= C\,(AB + \overline{A}\,\overline{B}) + \overline{C}\,(\overline{A}B + A\,\overline{B}) \\
&= C\,(\overline{\overline{A}B + A\,\overline{B}}) + \overline{C}\,(\overline{A}B + A\,\overline{B}) \\
&= C\,(\overline{A \oplus B}) + \overline{C}\,(A \oplus B) \; = A \oplus B \oplus C.
\end{aligned}
$$



$$
\begin{aligned}
C_{out} &= \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC \\
&= (\overline{A}B + A\overline{B})\,C + AB\,(\overline{C} + C) \\
&= (A \oplus B).C + AB.
\end{aligned}
$$





Thank you