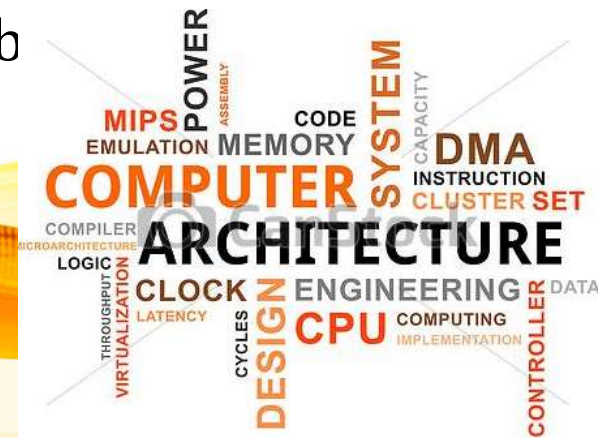


# UNIT II

## ARITHMETIC OPERATIONS

**Addition and subtraction of signed numbers** – Design of fast adders – Multiplication of positive numbers - Signed operand multiplication- fast multiplication – Integer division – Floating point num

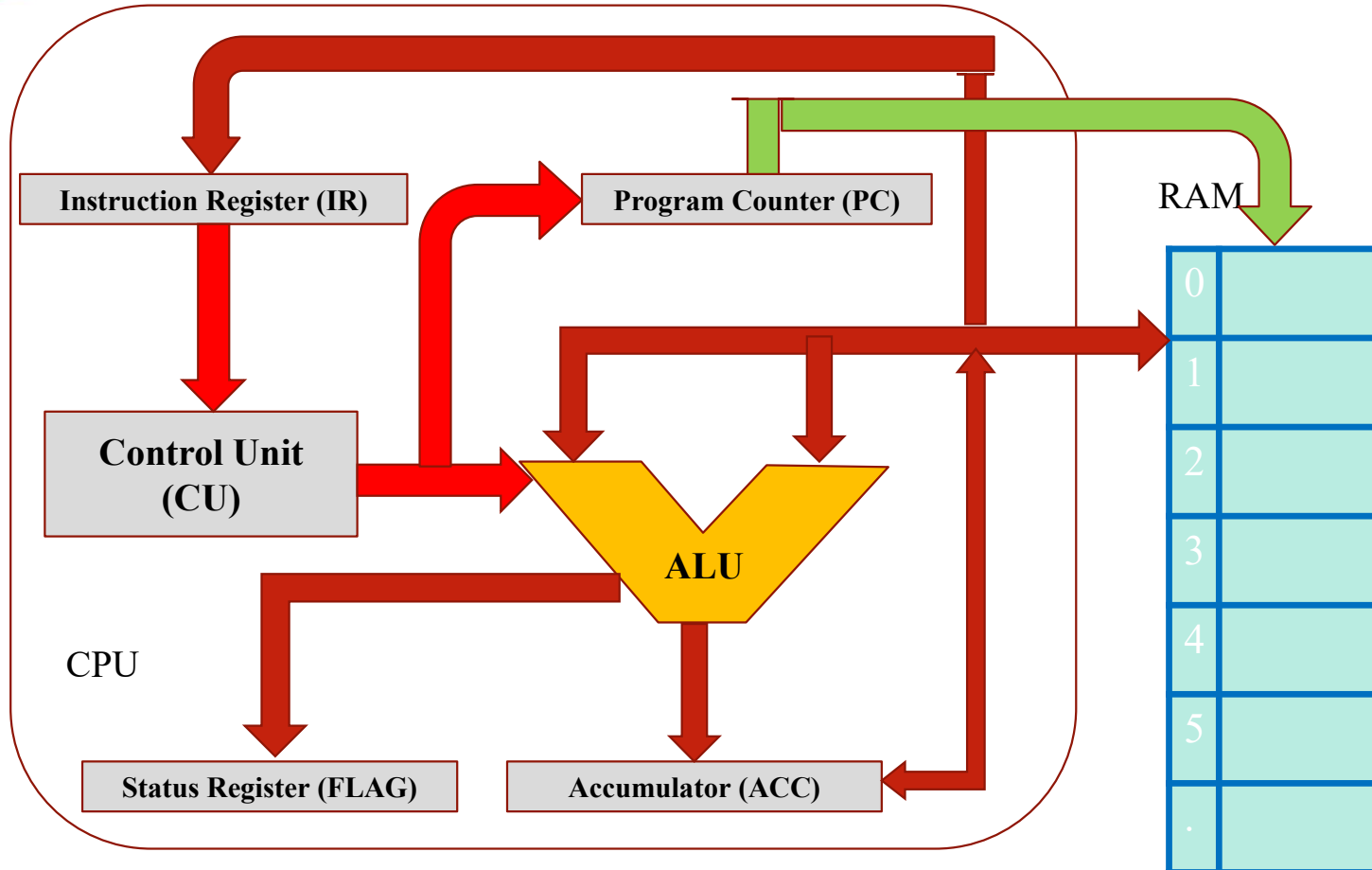


## Recall the Unit I

- Functional units
- Basic operational concepts
- Bus Structures
- Performance
- Memory locations and addresses
- Memory operations
- Instruction and Instruction sequencing
- Addressing modes
- Assembly language



# Recall the Unit I





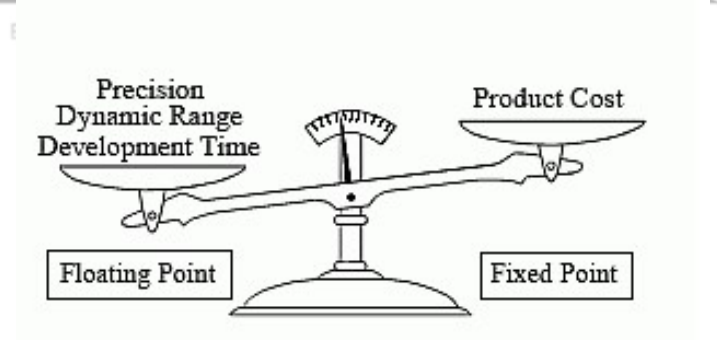
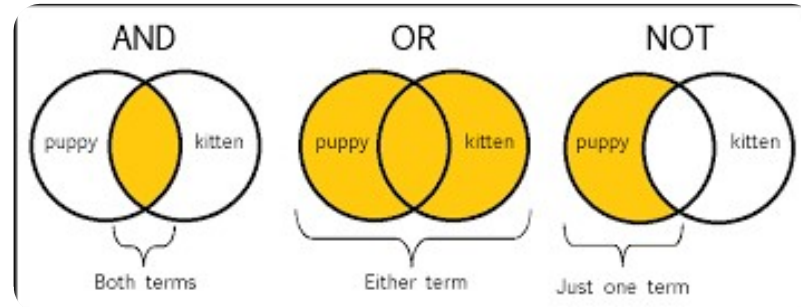
# Introduction

## Basic Arithmetic Operation

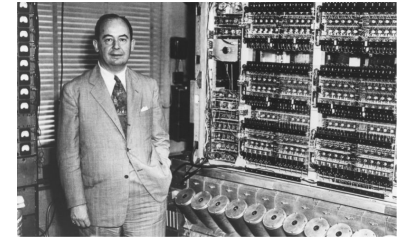


## Representation

## Logical Operation



## What's ALU?

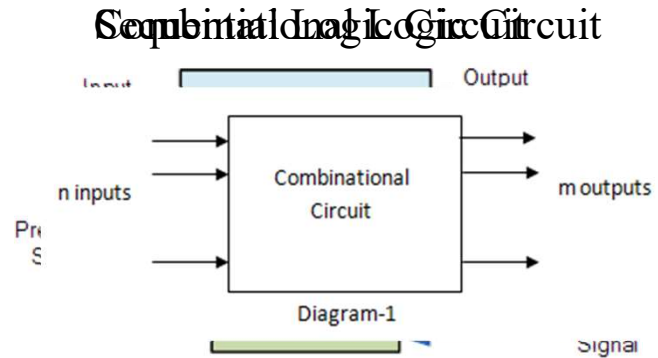
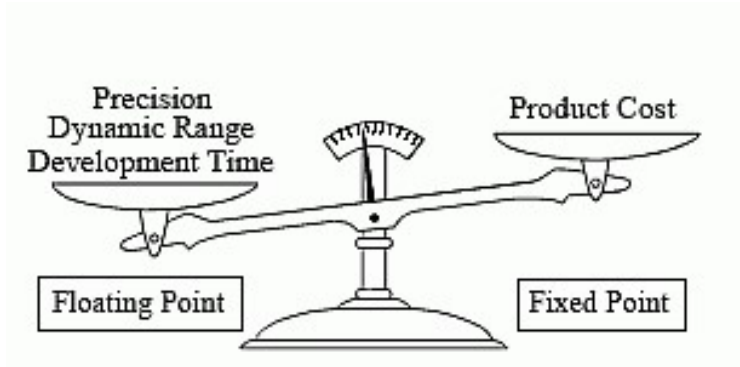


- Stands for **A**rithmetic and **L**ogic **U**nit
- Performs Arithmetic (Add, Sub, . . .) and Logical (AND, OR, NOT) operations.
- John Von Neumann proposed the ALU in 1945 when he was working on EDVAC (electronic discrete variable automatic computer)

# Arithmetic and Logical Unit

Operations

Circuit Design



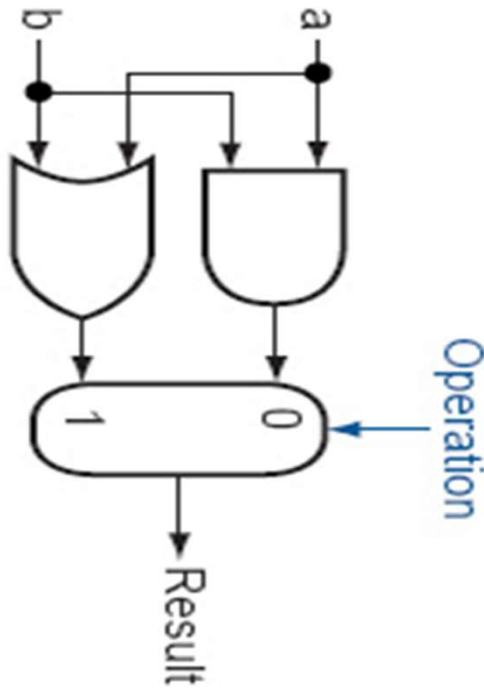
## Typical Schematic Symbol of an ALU

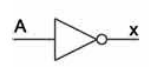
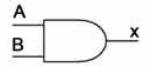


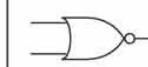


ALU control lines	Function
0000	AND
0001	OR
0010	add
0110	subtract
0111	set on less than
1100	NOR

# 1 Bit ALU

Data line and control Line

## Logic Gates

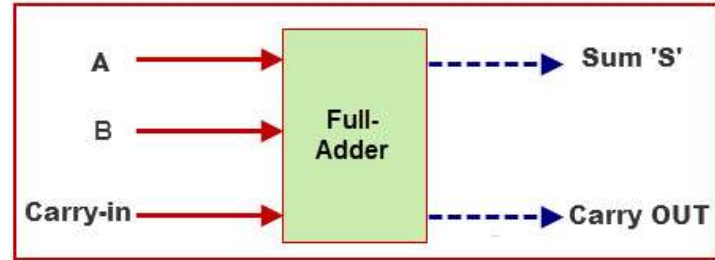
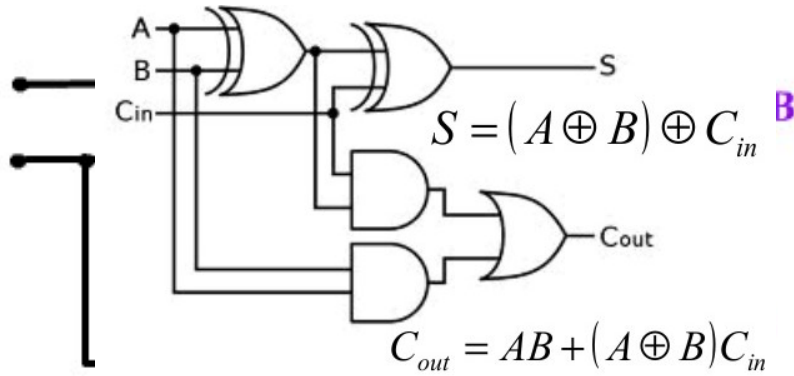


NOT	AND	NAND	OR	NOR	XOR	XNOR																																																																																																
$\bar{A}$	$AB$	$\overline{AB}$	$A+B$	$\overline{A+B}$	$A \oplus B$	$\overline{A \oplus B}$																																																																																																
																																																																																																						
<table border="1"> <thead> <tr> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	X	0	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	B	A	X	0	0	0	0	1	0	1	0	0	1	1	1	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	B	A	X	0	0	1	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	B	A	X	0	0	0	0	1	1	1	0	1	1	1	1	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	B	A	X	0	0	1	0	1	0	1	0	0	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	B	A	X	0	0	0	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>B</th> <th>A</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	B	A	X	0	0	1	0	1	0	1	0	0	1	1	1
A	X																																																																																																					
0	1																																																																																																					
1	0																																																																																																					
B	A	X																																																																																																				
0	0	0																																																																																																				
0	1	0																																																																																																				
1	0	0																																																																																																				
1	1	1																																																																																																				
B	A	X																																																																																																				
0	0	1																																																																																																				
0	1	1																																																																																																				
1	0	1																																																																																																				
1	1	0																																																																																																				
B	A	X																																																																																																				
0	0	0																																																																																																				
0	1	1																																																																																																				
1	0	1																																																																																																				
1	1	1																																																																																																				
B	A	X																																																																																																				
0	0	1																																																																																																				
0	1	0																																																																																																				
1	0	0																																																																																																				
1	1	0																																																																																																				
B	A	X																																																																																																				
0	0	0																																																																																																				
0	1	1																																																																																																				
1	0	1																																																																																																				
1	1	0																																																																																																				
B	A	X																																																																																																				
0	0	1																																																																																																				
0	1	0																																																																																																				
1	0	0																																																																																																				
1	1	1																																																																																																				





# Logic specification for a stage of Binary Addition <sup>9/18</sup>

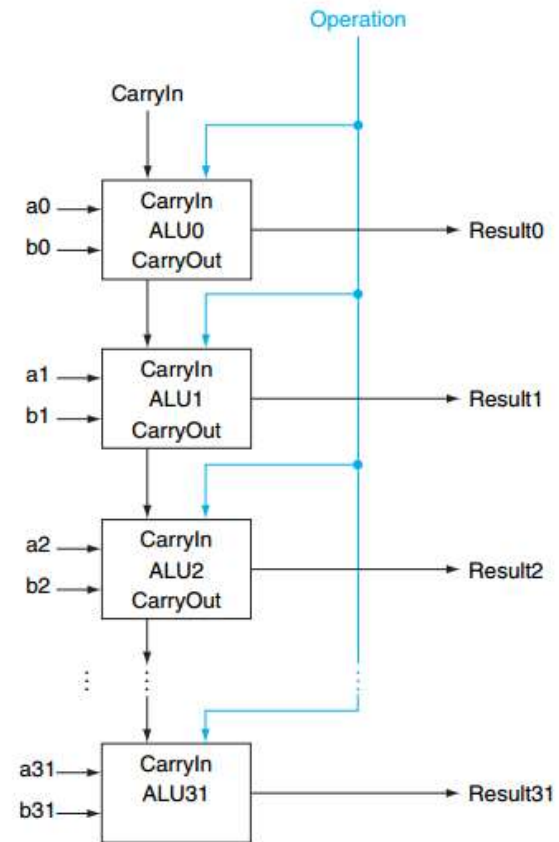
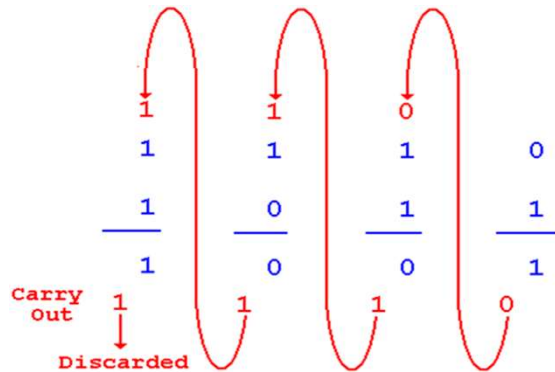
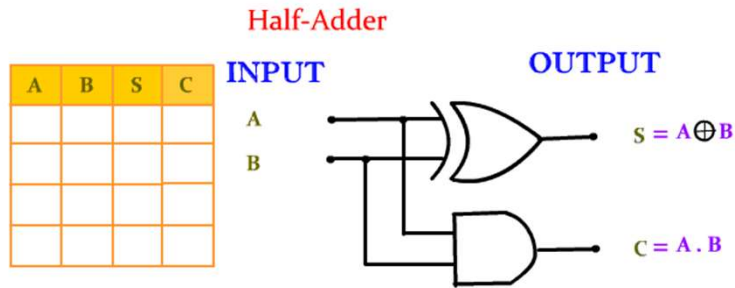


INPUTS			OUTPUT	
A	B	C-IN	C-OUT	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

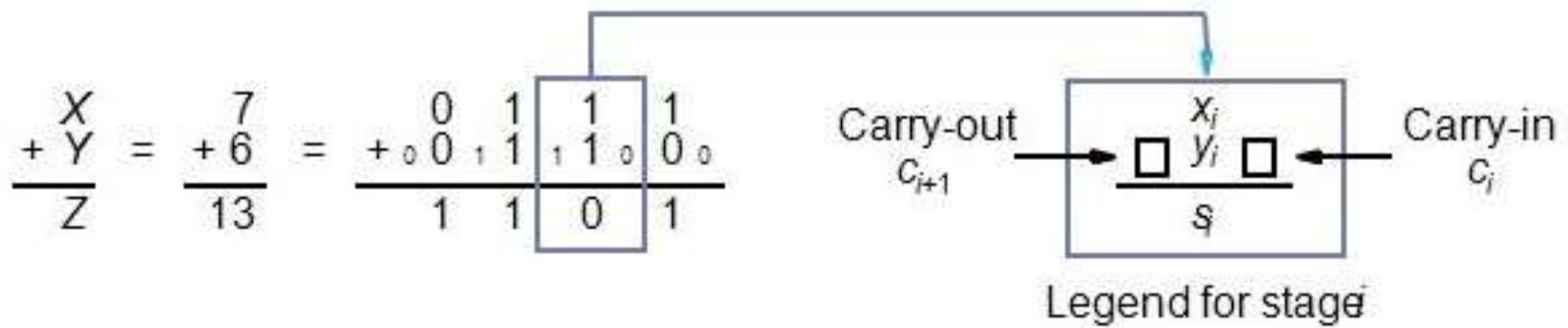
OUTPUTS
CARRY
0
0
0
1



## 32 BIT ALU

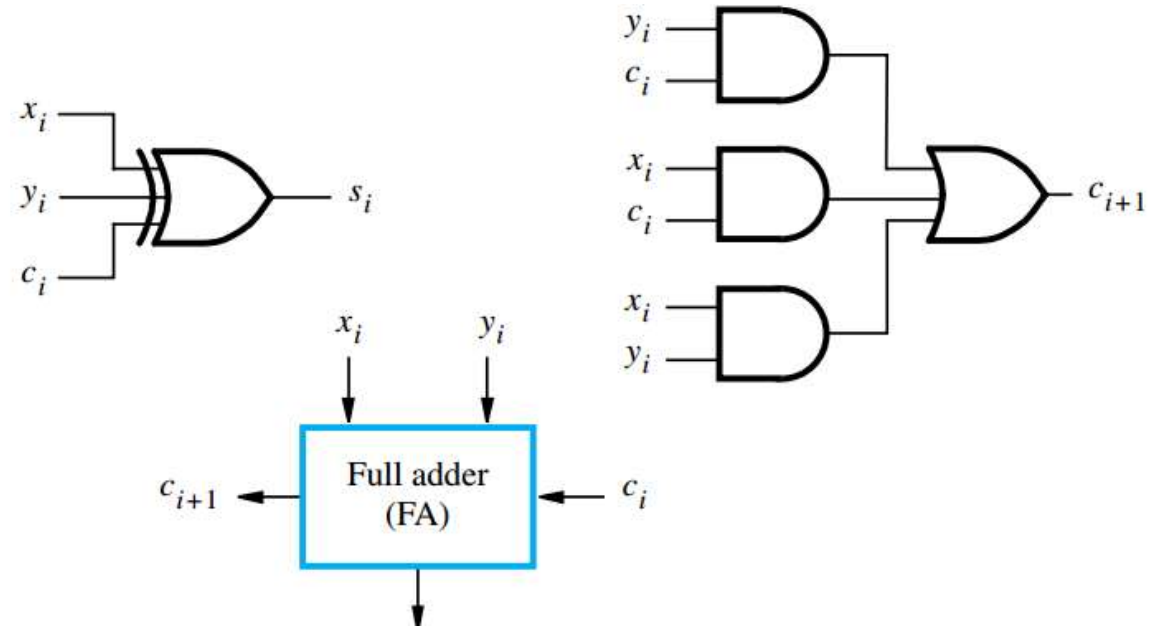


# Example for Binary Addition

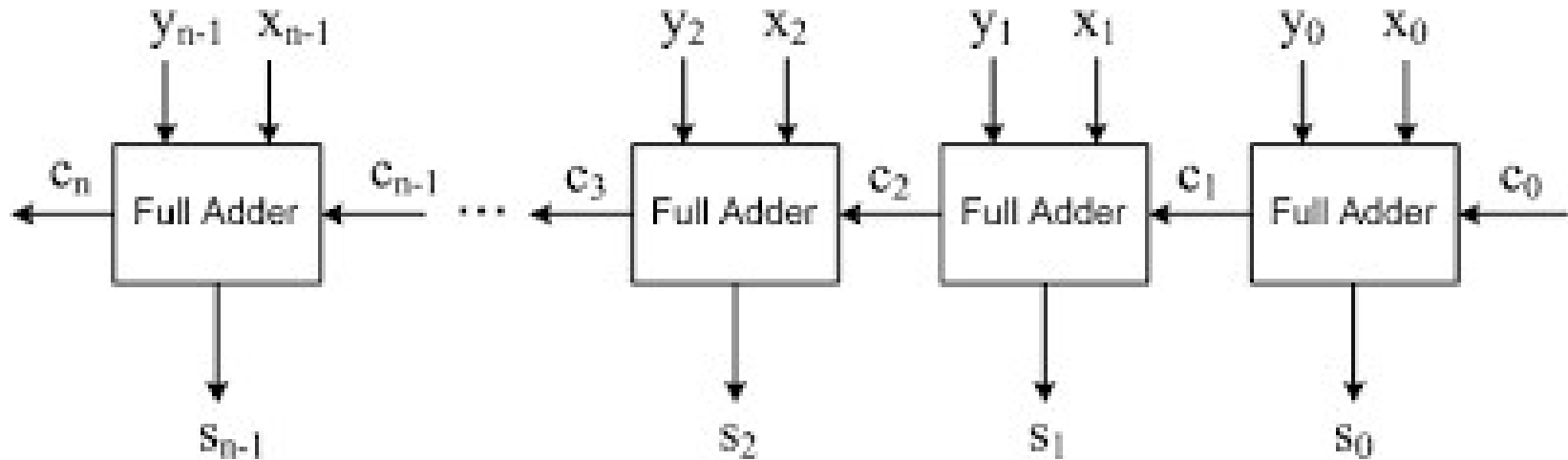


# Addition and Subtraction Logic Unit

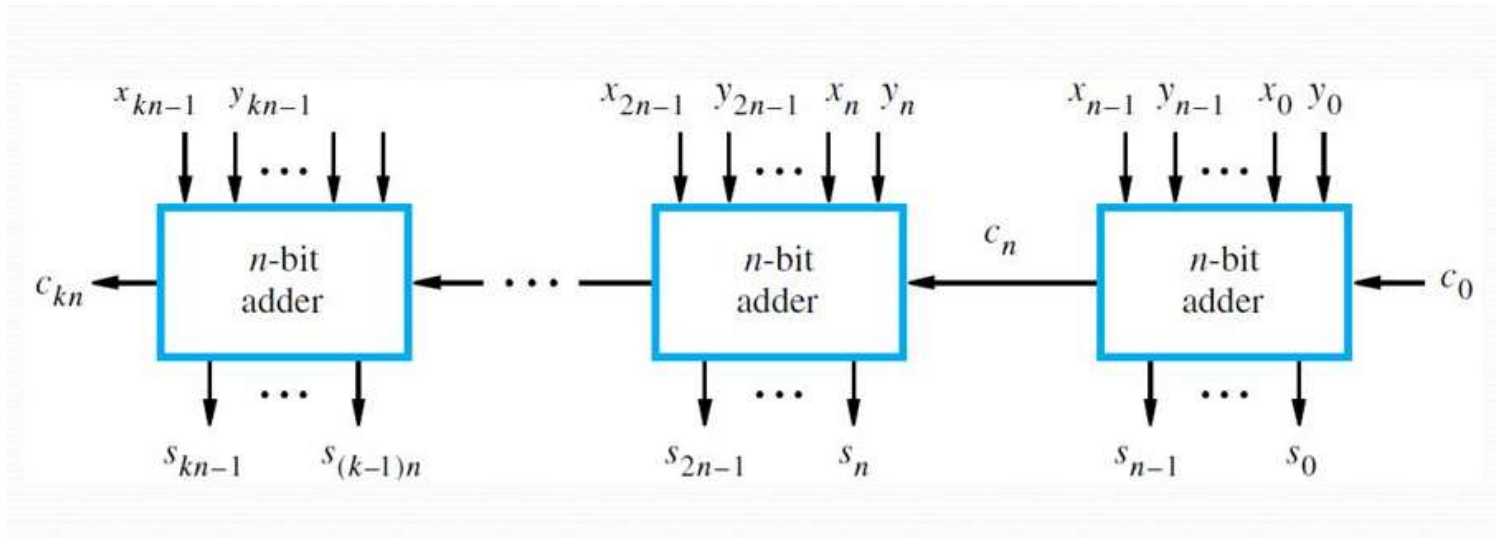
Logic for Single Stage



# n bit ripple carry adder

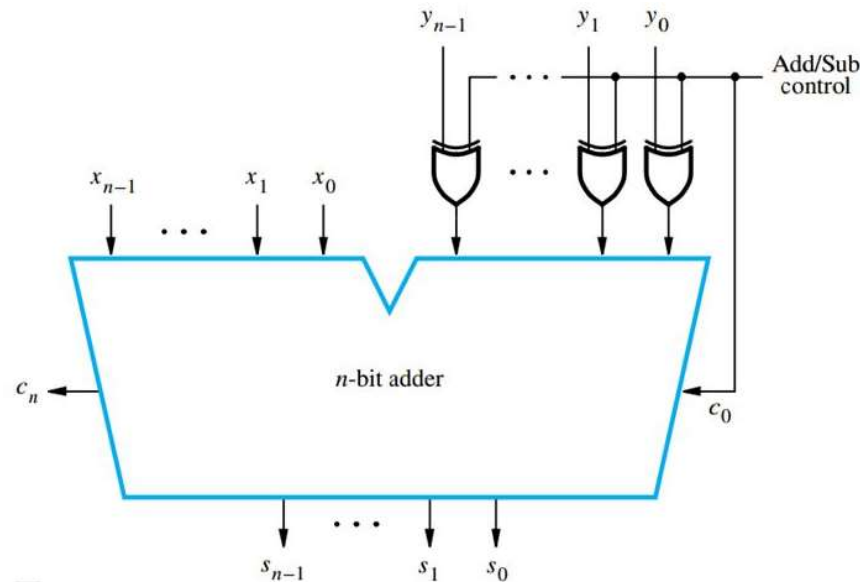


# cascaded k n bit adders

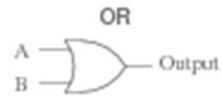




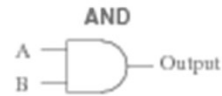
# Binary addition and subtraction logic network



- Addition  $\rightarrow$  Add/sub control = 0.
- Subtraction  $\rightarrow$  Add/sub control = 1



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1



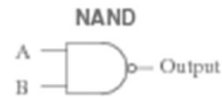
A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1



A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0



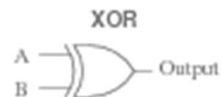
A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0



A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0



A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

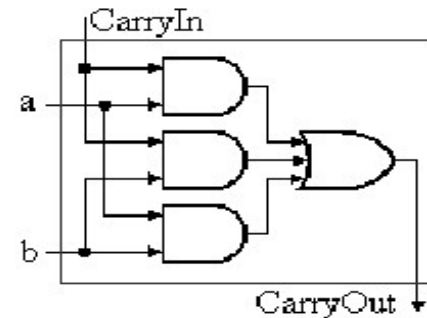
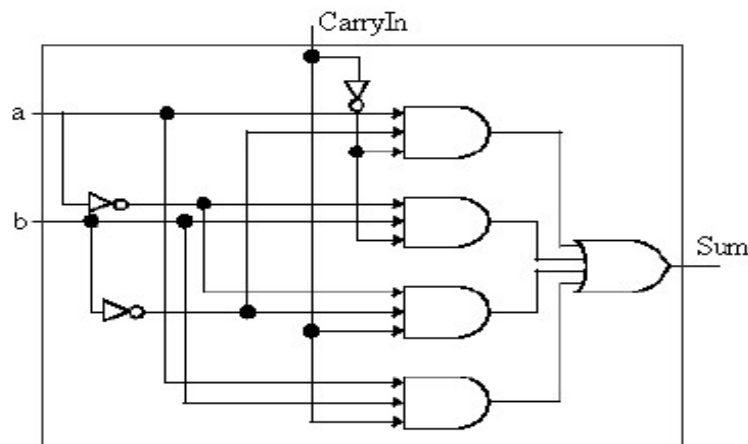


A	B	Output
0	0	1
0	1	0
1	0	0
1	1	1



A	Output
0	1
1	0





$$\text{Carryout} = (b \cdot \text{CarryIn}) + (a \cdot \text{CarryIn}) + (a \cdot b)$$

$$\text{Sum} = (a \cdot b' \cdot \text{CarryIn}') + (a' \cdot b \cdot \text{CarryIn}') + (a' \cdot b' \cdot \text{CarryIn}) + (a \cdot b \cdot \text{CarryIn})$$



**sns**  
INSTITUTIONS



*Thank You*