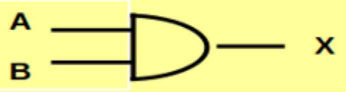
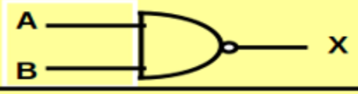
Basic Logic Gates

COMBINATIONAL GATES

Name	Symbol	Function	Truth Table
AND		X = A • B or X = AB	A B | X 0 0 | 0 0 1 | 0 1 0 | 0 1 1 | 1
OR		X = A + B	A B | X 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 1
NOT		X = A’	A | X 0 | 1 1 | 0
Buffer		X = A	A | X 0 | 0 1 | 1
NAND		X = (AB)’	A B | X 0 0 | 1 0 1 | 1 1 0 | 1 1 1 | 0
NOR		X = (A + B)’	A B | X 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 0
XOR Exclusive OR		X = A ⊕ B or X = A’B + AB’	A B | X 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 0
XNOR Exclusive NOR or Equivalence		X = (A ⊕ B)’ or X = A’B’+ AB	A B | X 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 1

Rules of Boolean Algebra

1. $A + 0 = A$
2. $A + 1 = 1$
3. $A \cdot 0 = 0$
4. $A \cdot 1 = A$
5. $A + A = A$
6. $A + \overline{A} = 1$
7. $A \cdot A = A$
8. $A \cdot \overline{A} = 0$
9. $\overline{\overline{A}} = A$
10. $A + AB = A$
11. $A + \overline{A}B = A + B$
12. $(A + B)(A + C) = A + BC$

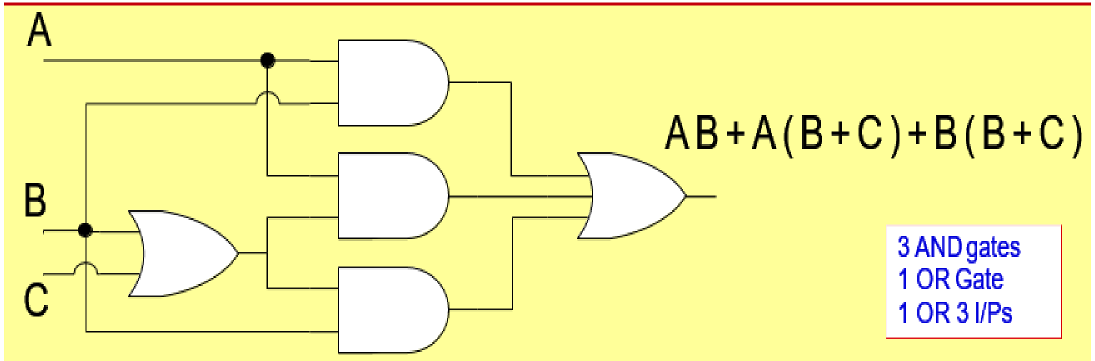
DeMorgan's Theorems

$$\overline{X + Y} = \overline{X}\,\overline{Y}$$

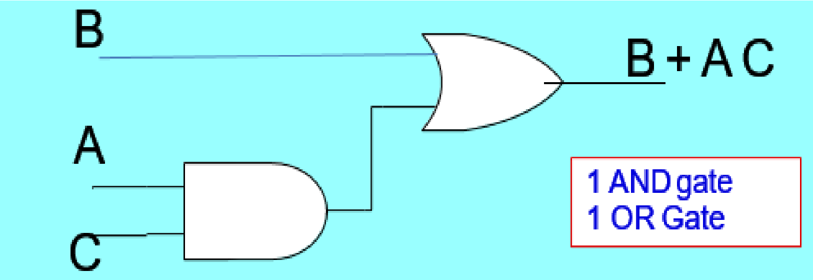
$$\overline{XY} = \overline{X} + \overline{Y}$$

Simplify using boolean Algebra

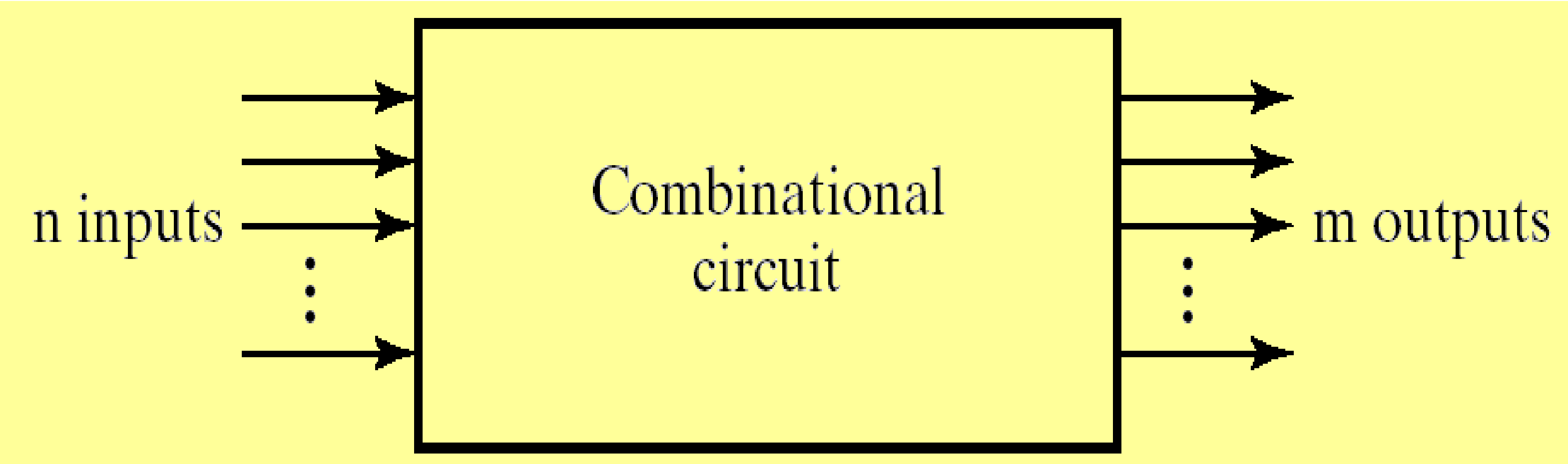
AB + A(B+C) + B(B+C)
= AB + AB + AC + BB + BC
= AB + AC + B + BC
= AB + AC + B
= B + AB + AC



= B + AC



Combinational Circuit



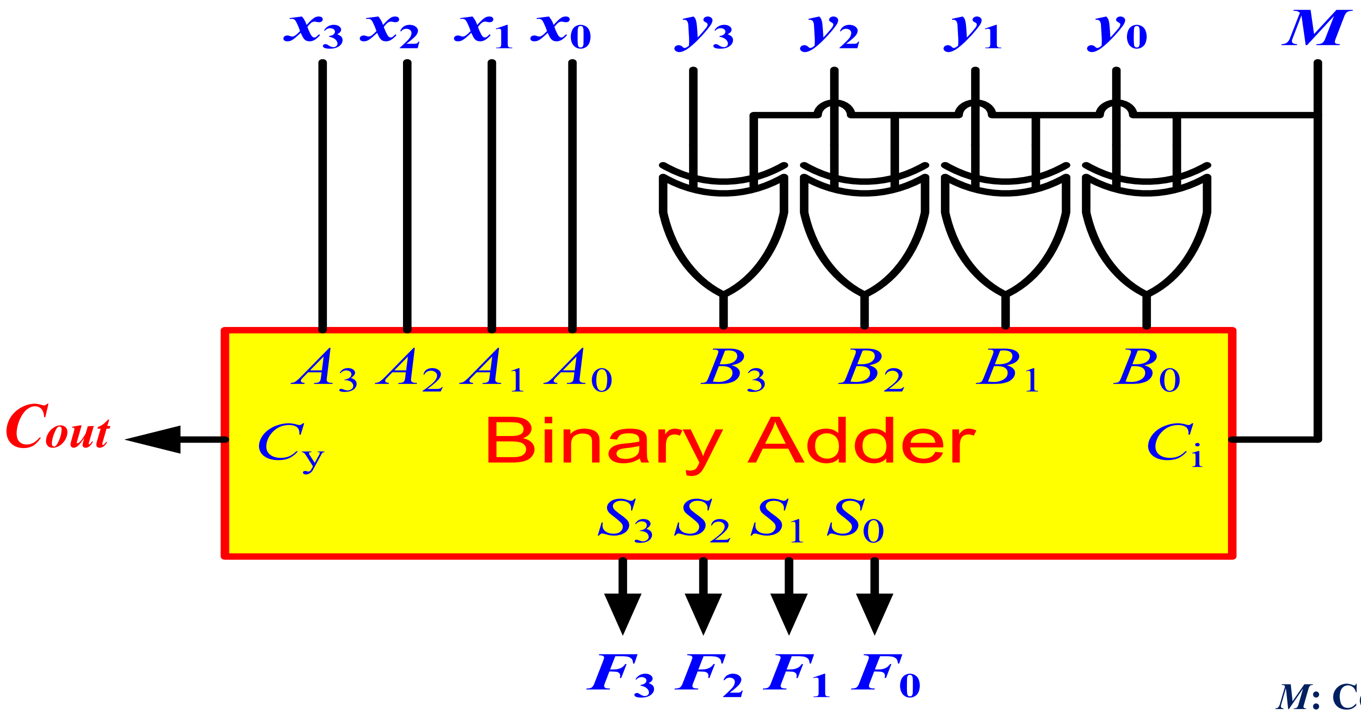
Combinational Circuits

- Full Adder
- Decoder
- Multiplexer

Binary Adder / Subtractor

$M=0 \rightarrow F = x + y$

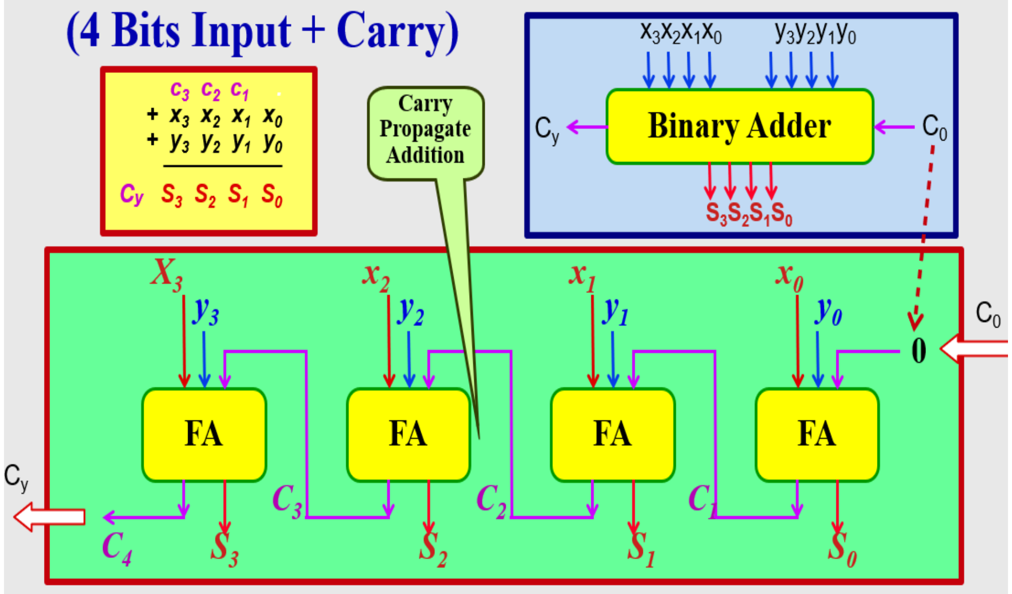
$M=1 \rightarrow F = x - y$



M: Control Signal Mode: Add / Sub

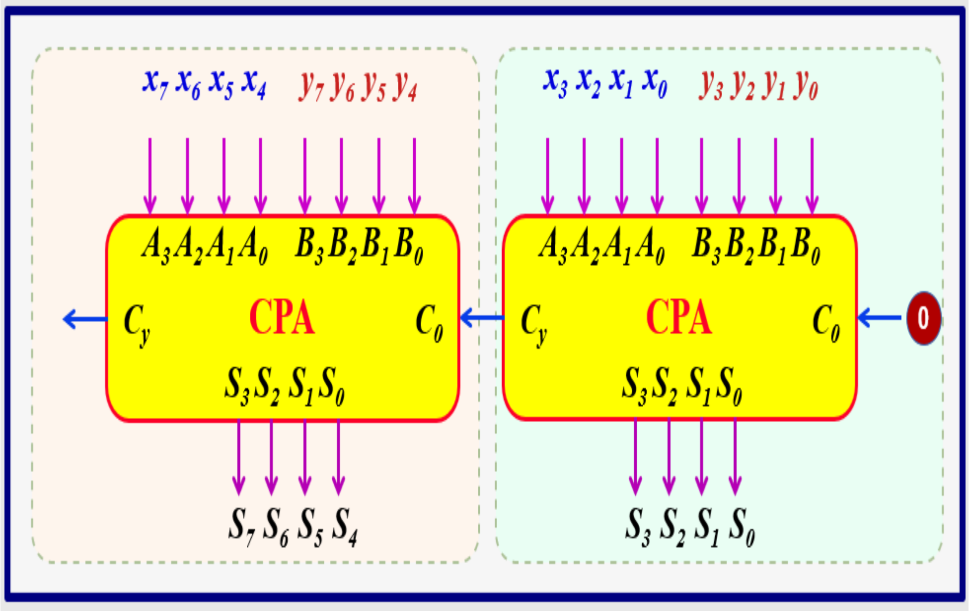
Carry Propagate Adder (CPA)

Adding Two 4-Bits Numbers



Carry Propagate Adder (CPA)

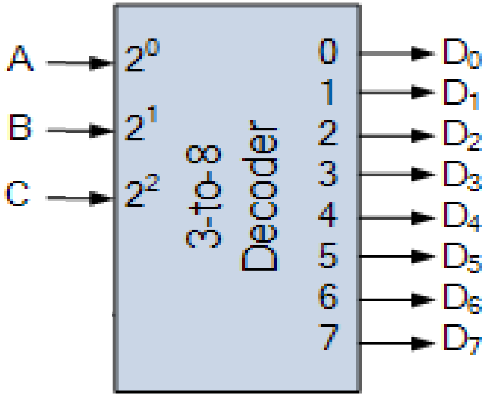
Adding Two 8-Bits Numbers



Decoders

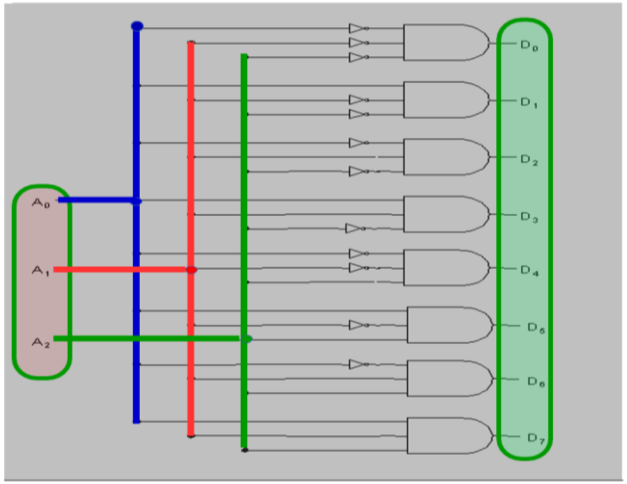
Decoders

DEC 3x8



3 Input Bits

Inputs: A_0, A_1, A_2



8 Output bits

Outputs $D_0, D_1, D_2, ..D_7$

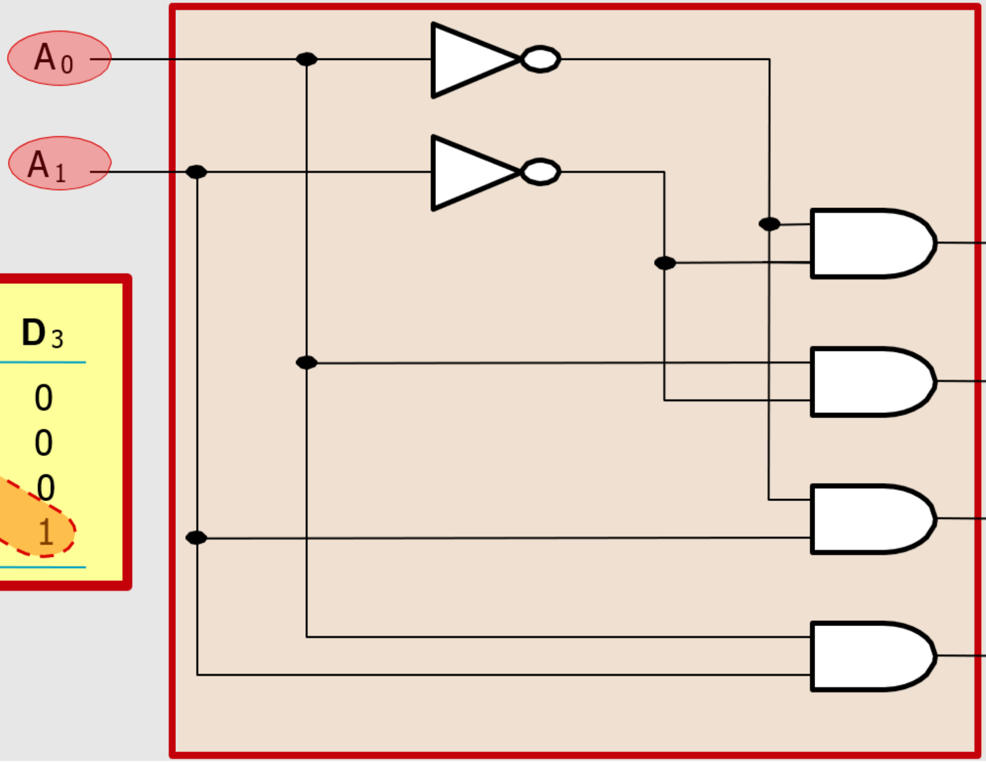
Inputs			Outputs							
A_2	A_1	A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

Decoders

DEC 2x4

2 Input Bits

A_1	A_0	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



4 Outputs

$D_0 = \overline{A_1}\ \overline{A_0}$

$D_1 = \overline{A_1}\ A_0$

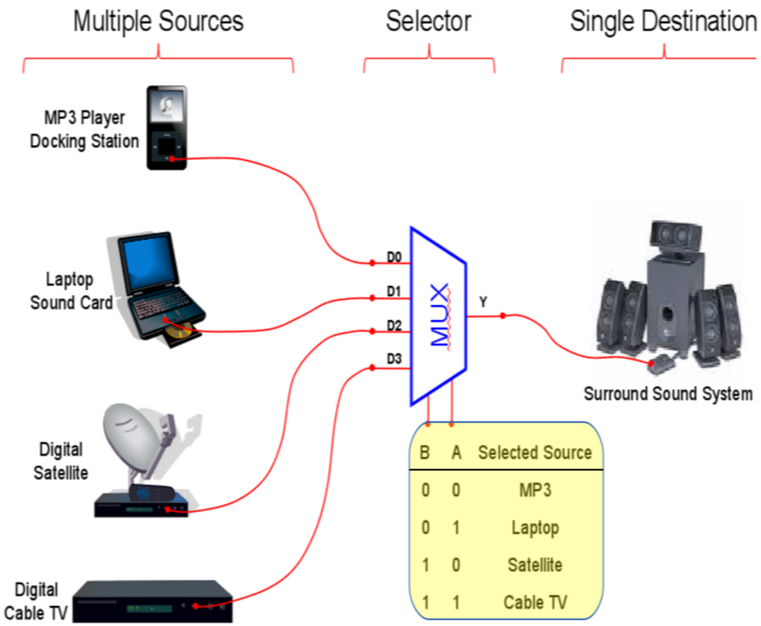
$D_2 = A_1\ \overline{A_0}$

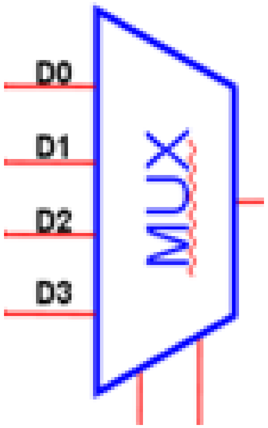
$D_3 = A_1\ A_0$

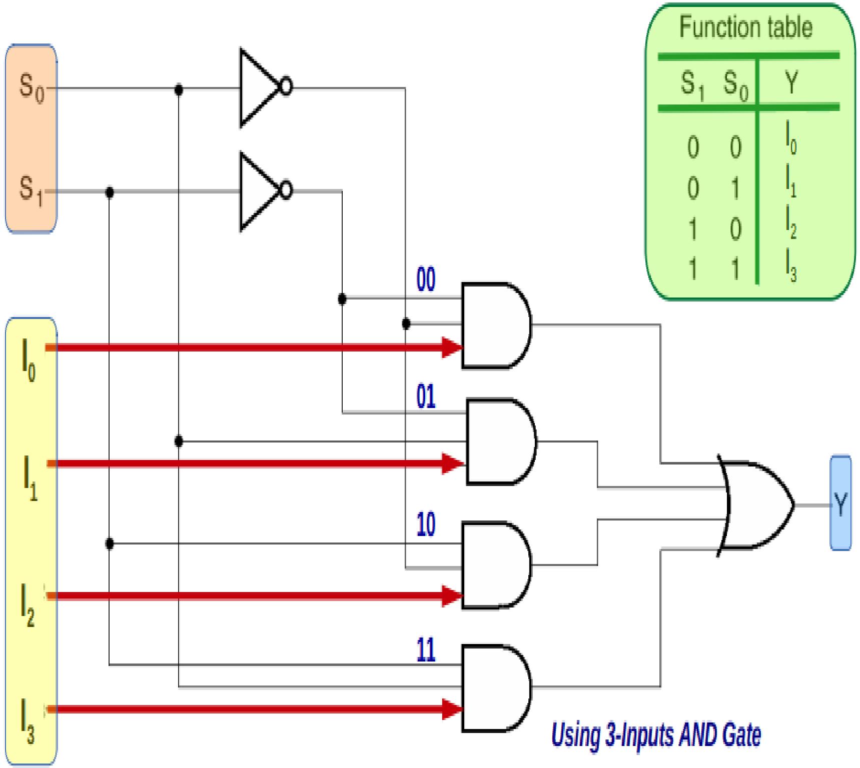
Combinational Logic Implementation using Decoder & OR Gates

Multiplexers

Multiplexers

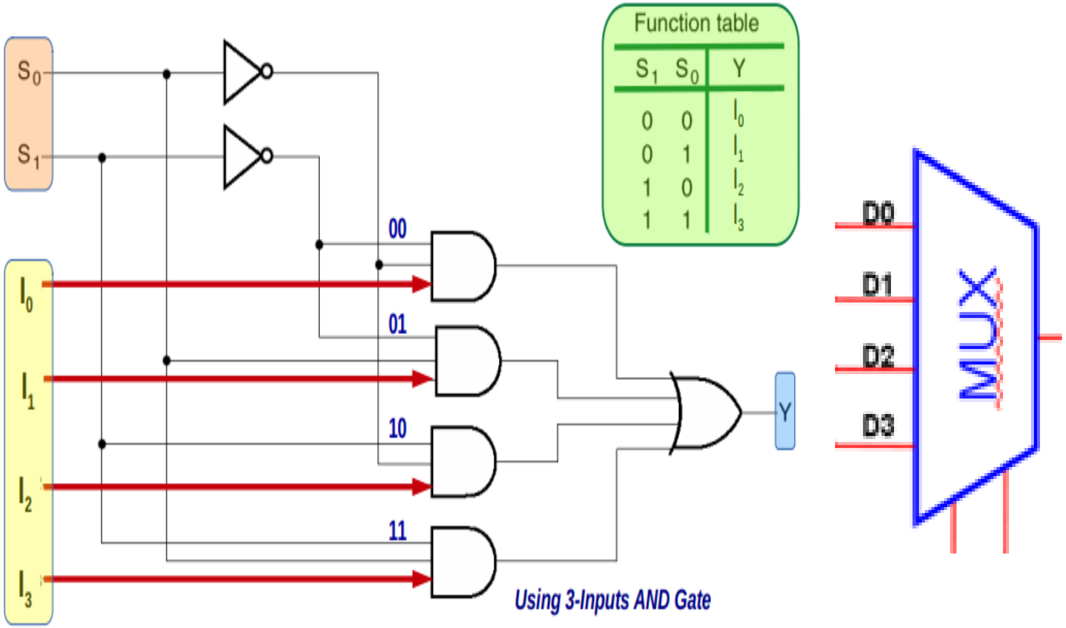






4-to-1-Line Multiplexer

A Single Bit 4-to-1 Line Multiplexer



4-to-1-Line Multiplexer

Sequential Circuits

[1] Flip Flops (FF)

[2] Design of Sequential Circuits

Review of Flip Flops

Characteristic and Excitation Table

D Flip flop

Excitation Table of "D" Flip Flop

Given: $Q(t)$, $Q(t+1)$ — Required: D

Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

Function Table

D	Q(t+1)	
0	0	Output Follows Input
1	1	

Equations

$$Q(t+1) = D$$

T Flip flop

Excitation Table of "T" Flip Flop

Given: $Q(t)$, $Q(t+1)$ — Required: T

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

Function Table

T	$Q(t+1)$	
0	$Q(t)$	No change
1	$\overline{Q}(t)$	Toggle

$$Q(t+1) = T \oplus Q(t)$$

JK Flip flop

Simplified Excitation Table

Given | Required

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Function Table

J	K	$Q(t+1)$	
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	$\overline{Q}(t)$	Toggle

$$Q(t+1) = J\overline{Q}(t)+\overline{K}Q(t)$$

Design of Sequential Circuit

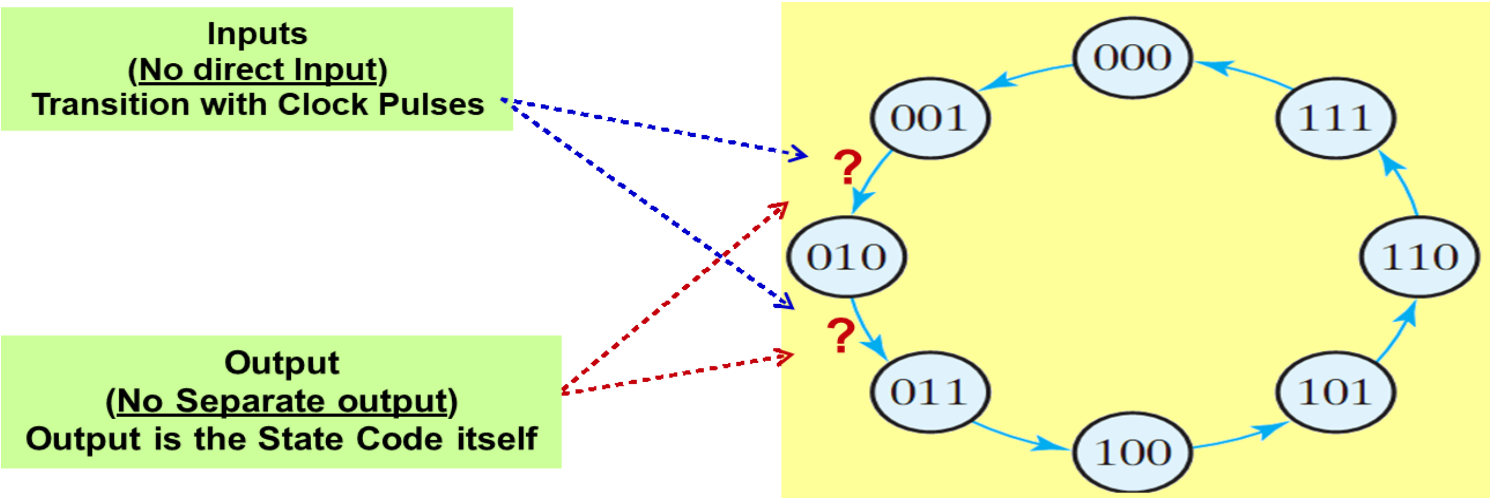
Example:

Design a 3 bit Counter (Using "T" FF) which counts in binary form as follows;

000, 001, 010, … 111, 000, 001, …

Solution

2- State diagram:



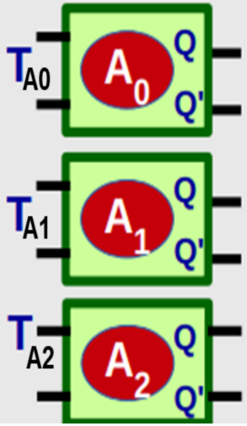
4- State Table:

Present State			Next State			Flip-Flop Inputs		
A_2	A_1	A_0	A_2^+	A_1^+	A_0^+	T_{A2}	T_{A1}	T_{A0}
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

INPUTS

Excitation Table of "T" flip flop

T	$Q(t+1)$	
0	$Q(t)$	No change
1	$Q'(t)$	Toggle



5- K-Map for FFs inputs and circuit Outputs

Present State			Next State			Flip-Flop Inputs		
A_2	A_1	A_0	A_2^+	A_1^+	A_0^+	T_{A2}	T_{A1}	T_{A0}
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	1	1
1	1	1	0	0	0	1	1	1

INPUTS

T_{A0}

A_2 \ A_1A_0	00	01	11	10
0	1	1	1	1
1	1	1	1	1

T_{A1}

A_2 \ A_1A_0	00	01	11	10
0	0	1	1	0
1	0	1	1	0

T_{A2}

A_2 \ A_1A_0	00	01	11	10
0	0	0	1	0
1	0	0	1	0

5- K-Map for FFs inputs and circuit Outputs

T_{A0}

$A_2 \backslash A_1A_0$	00	01	11	10
0	1	1	1	1
1	1	1	1	1

$T_{A0}=1$

T_{A1}

$A_2 \backslash A_1A_0$	00	01	11	10
0	0	1	1	0
1	0	1	1	0

$T_{A1}=A_0$

T_{A2}

$A_2 \backslash A_1A_0$	00	01	11	10
0	0	0	1	0
1	0	0	1	0

$T_{A2}=A_1A_0$

6- Circuit diagram: