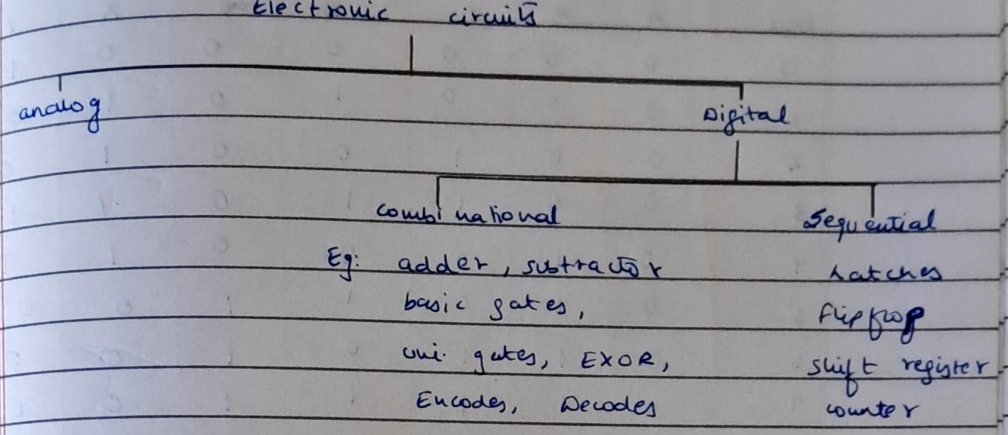
Page No.
Date / /20

Module 2: Combinational logic & Sequential logic



combinational	Sequential
• present i/p	• present & past i/p
• does not store data	• stores data
• no memory, no feedback	• have memory & feedback

→ draw logic dig

→ Design a BCD to Excess 3 code converter.

	input BCD				o/p Excess - 3			
Deci	A	B	C	D	W	X	Y	Z
0	0	0	0	0+3	0	0	1	1
1	0	0	0	1+3	0	1	0	0
2	0	0	1	0+3	0	1	0	1
3	0	0	1	1+3	0	1	1	0
4	0	1	0	0+3	0	1	1	1
5	0	1	0	1+3	1	0	0	0
6	0	1	1	0+3	1	0	0	1
7	0	1	1	1+3	1	0	1	0
8	1	0	0	0+3	1	0	1	1
9	1	0	0	1+3	1	1	0	0
10	1	0	1	0	x	x	x	x
11	1	0	1	1	x	x	x	x
12	1	1	0	0	x	x	x	x
13	1	1	0	1	x	x	x	x
14	1	1	1	0	x	x	x	x
15	1	1	1	1	x	x	x	x

W

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
	0	1	3	2
$\bar{A}\bar{B}$	0	0	0	0
	4	5	7	6
$\bar{A}B$	0	1	1	1
	12	13	15	14
AB	x	x	x	x
	8	9	11	10
$A\bar{B}$	1	1	x	x

$W = A + B\bar{C}D + BC$

X

AB \ $C\bar{D}$	$C\bar{D}$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0 (0)	1 (1)	1 (3)	1 (2)
$\bar{A}B$	1 (4)	0 (5)	0 (7)	0 (6)
AB	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$	0 (8)	1 (9)	X (11)	X (10)

$X = B\bar{C}\bar{D} + D\bar{B} + \bar{B}C$

Y

AB \ $\bar{C}\bar{D}$	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1 (0)	0 (1)	1 (3)	0 (2)
$\bar{A}B$	1 (4)	0 (5)	1 (7)	0 (6)
AB	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$	1 (8)	0 (9)	X (11)	X (10)

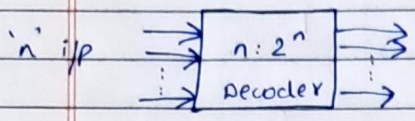
$Y = \bar{C}\bar{D} + CD$

Z

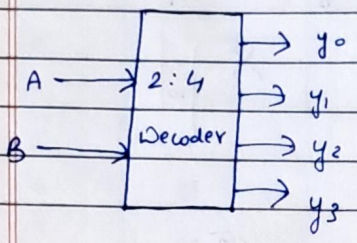
$Z = \bar{D}$

AB \ $\bar{C}\bar{D}$	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1 (0)	0 (1)	0 (3)	1 (2)
$\bar{A}B$	1 (4)	0 (5)	0 (7)	1 (6)
AB	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$	1 (8)	0 (9)	X (11)	X (10)

Decoder - also called binary to decimal decoder



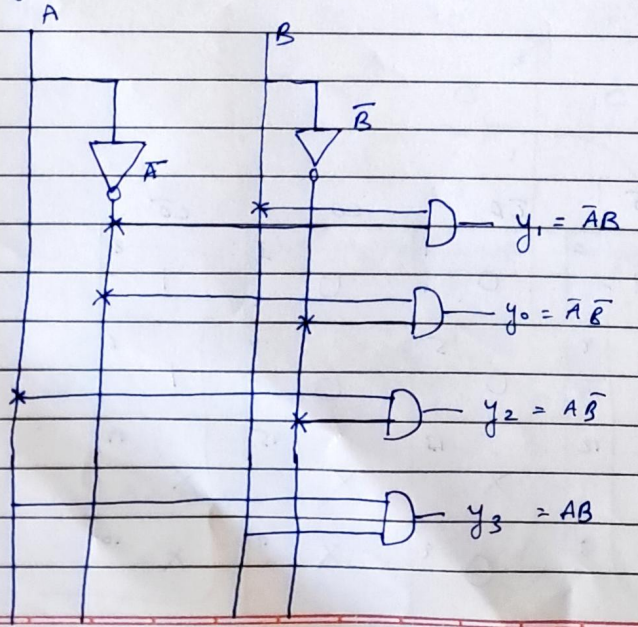
→ Design 2 to 4 Binary Decoder (1 of 4 Decoder)



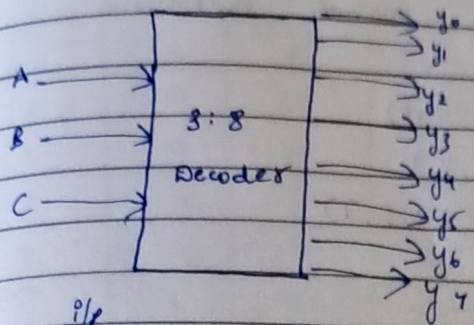
deci	i/p		o/p			
	A	B	y_0	y_1	y_2	y_3
0	0	0	1	0	0	0
1	0	1	0	1	0	0
2	1	0	0	0	1	0
3	1	1	0	0	0	1

$y_0 = \bar{A}\bar{B}$ $\quad y_3 = A\bar{B}$

$y_1 = \bar{A}B$ $\quad y_3 = AB$

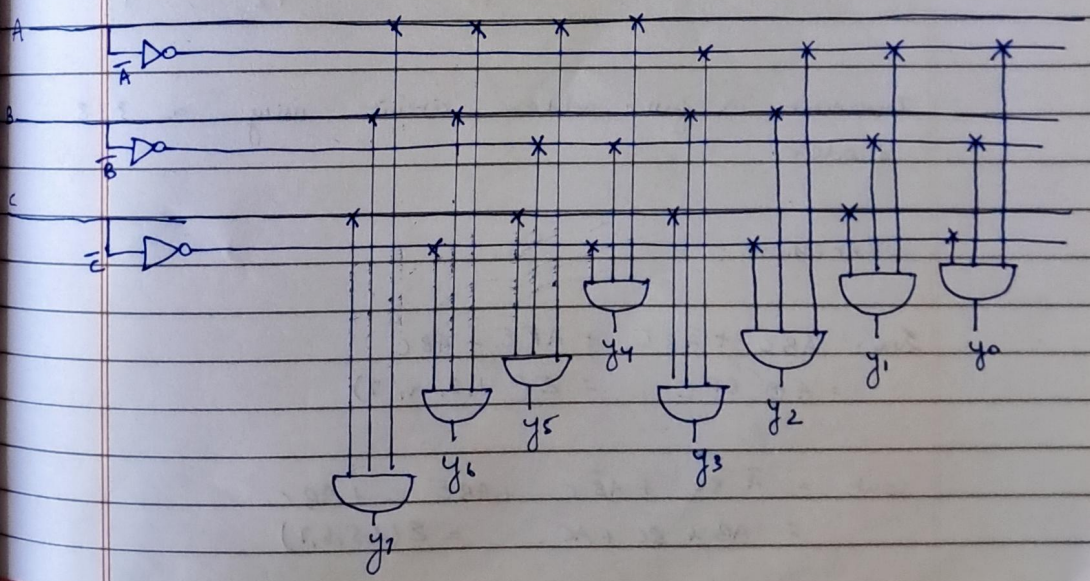


→ Doing 3 to 8 decoder. (If 1 of 8 decoder)



	i/p			o/p							
decimal	A	B	C	y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7
0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	0	0	0	0	0
3	0	1	1	0	0	0	1	0	0	0	0
4	1	0	0	0	0	0	0	1	0	0	0
5	1	0	1	0	0	0	0	0	1	0	0
6	1	1	0	0	0	0	0	0	0	1	0
7	1	1	1	0	0	0	0	0	0	0	1

$y_0 = \bar{a}\bar{b}\bar{c}$, $y_1 = \bar{a}\bar{b}c$, $y_2 = \bar{a}b\bar{c}$, $y_3 = \bar{a}bc$, $y_4 = a\bar{b}\bar{c}$,
$y_5 = a\bar{b}c$, $y_6 = ab\bar{c}$, $y_7 = abc$

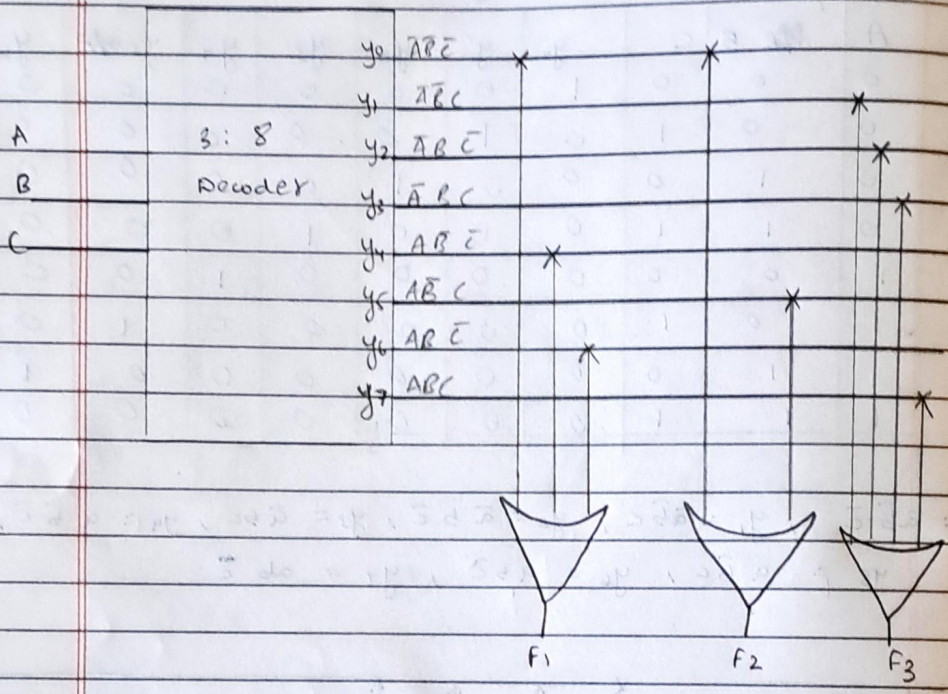


Realise using a 3 to 8 decoder & multi-input OR gates following Boolean expressions simultaneously.

$F_1(A,B,C) = \Sigma m(0,4,6)$

$F_2(A,B,C) = \Sigma m(0,5)$

$F_3(A,B,C) = \Sigma m(1,2,3,7)$



Implement a full-adder circuit using a 3-8 decoder.

Expressions:

Sum: $\bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$

$= A \oplus B \oplus C \quad = \Sigma(1,2,4,7)$

Cout $= \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

$= AB + BC + AC. \quad = \Sigma(4,5,6,7)$

Page no.
Date / /20

Full adder expressions

dec	A	B	C	sum	cout
0	0	0	0	0	0
1	0	0	1	1	0
2	0	1	0	1	0
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	1
6	1	1	0	0	1
7	1	1	1	1	1