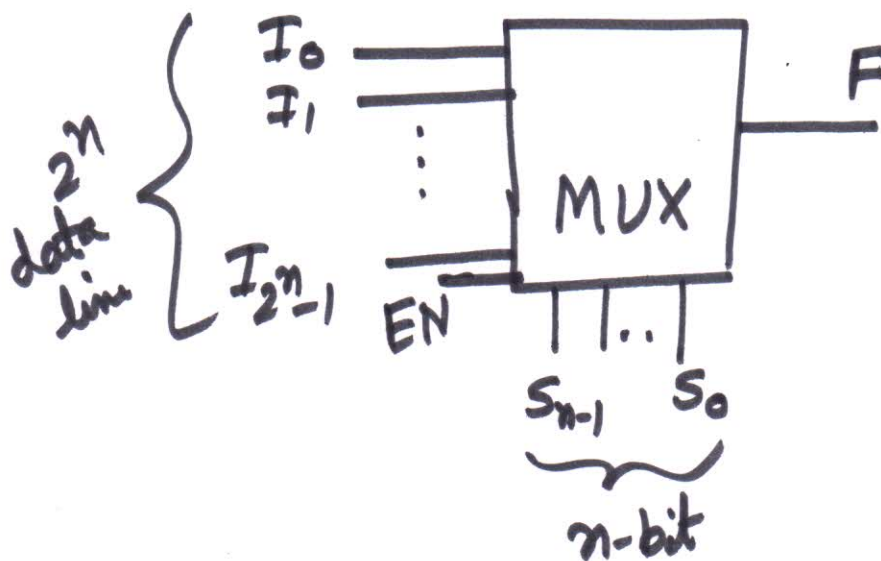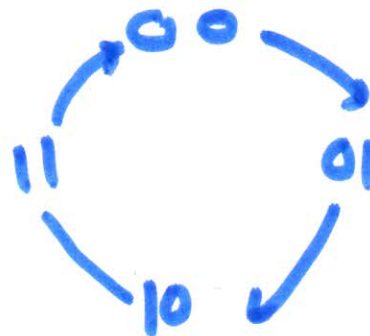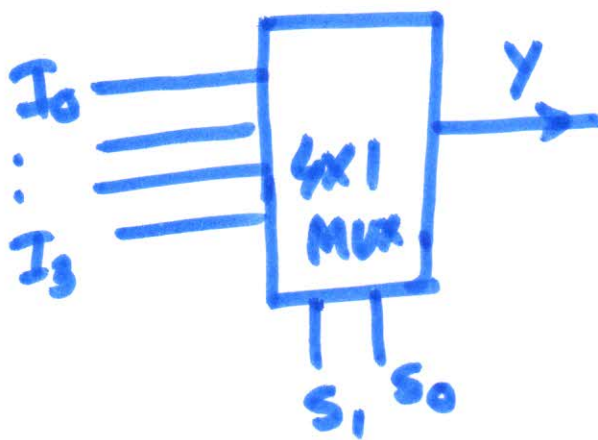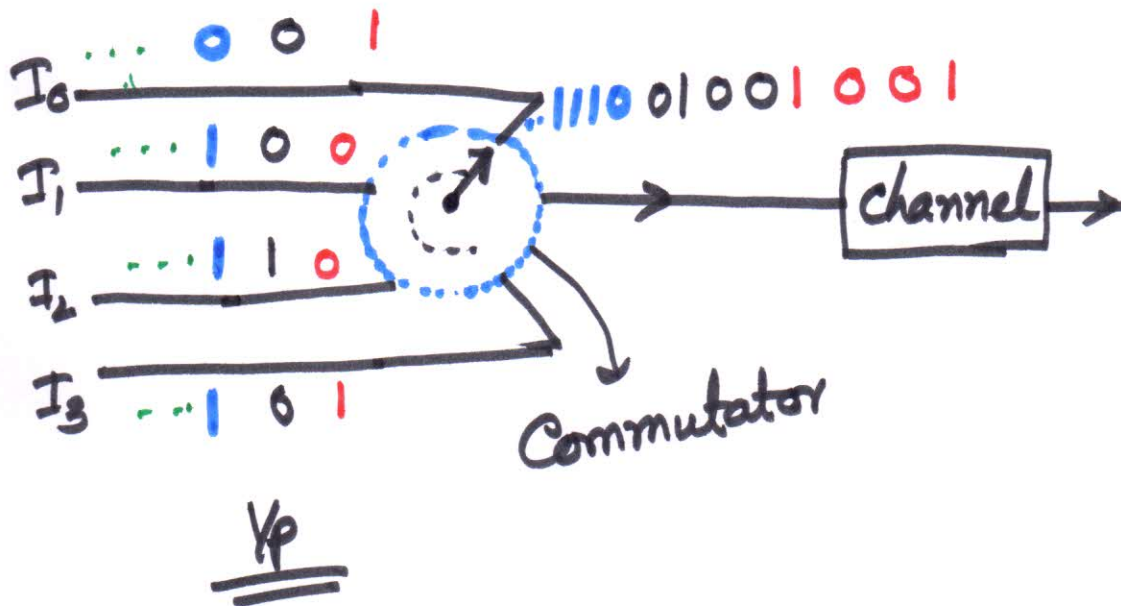# Multiplexers

- **A multiplexer or MUX is a combinational circuit with more than one input line, one output line and more than one selection line.**

- **In general, for n selection lines, there are 2^n data lines, and the multiplexer is referred to as a 2^n-to-1 MUX or $2^n \times 1$ MUX.**

- **The multiplexer sends the binary information present on any of the input line to the data line depending on the binary status of the select lines.**
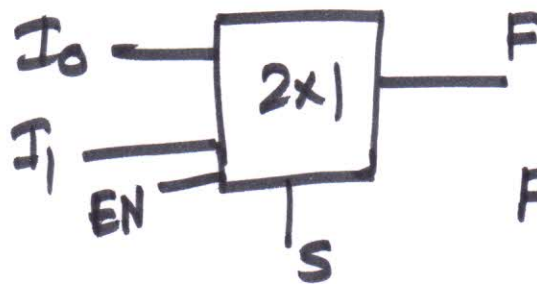

Application
(1) TDM/FDM
(2) parallel to serial Converter

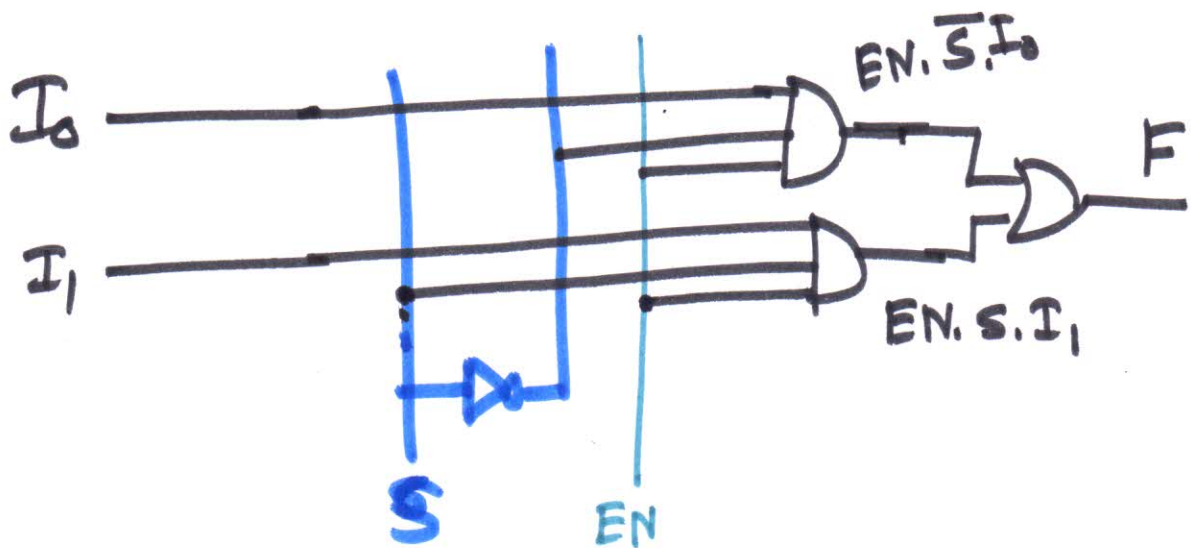# Time Division Multiplexing (TDM)

I_0 — $\cdots$ 0 0 1
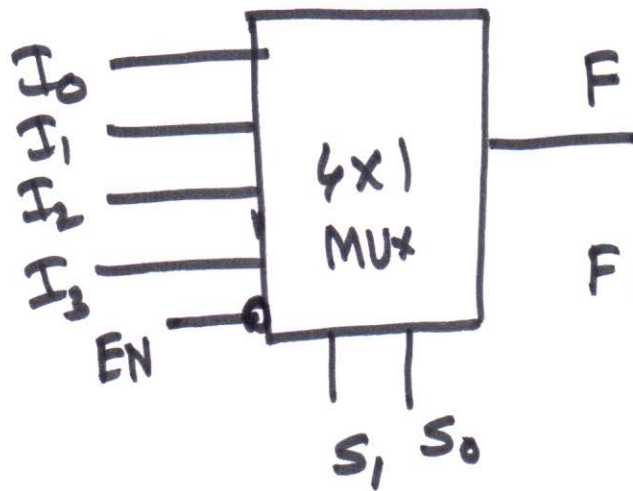
I_1 — $\cdots$ 1 0 0

I_2 — $\cdots$ 1 1 0

I_3 — $\cdots$ 1 0 1

1110 0100 1 0 0 1 → channel →

Commutator

V_p

4×1 MUX

I_0 — $\vdots$ — I_3 → Y

S_1 S_0

00
11   01
10

A) 2 × 1 MUX (with ENABLE input)

$$F = EN.\bar{S}.I_0 + EN.S.I_1$$

S	EN	F
X	0	0
0	1	I_0
1	1	I_1

$EN.\bar{S}.I_0$

$EN.S.I_1$

F

I_0

I_1

S   EN

B) 4 × 1 MUX (with ENABLE input)

$$F = \overline{EN}.\overline{S_0}.\overline{S_1}.I_0 +$$
$$\overline{EN}.S_0.\overline{S_1}.I_1 +$$
$$\overline{EN}.\overline{S_0}.S_1.I_2 +$$
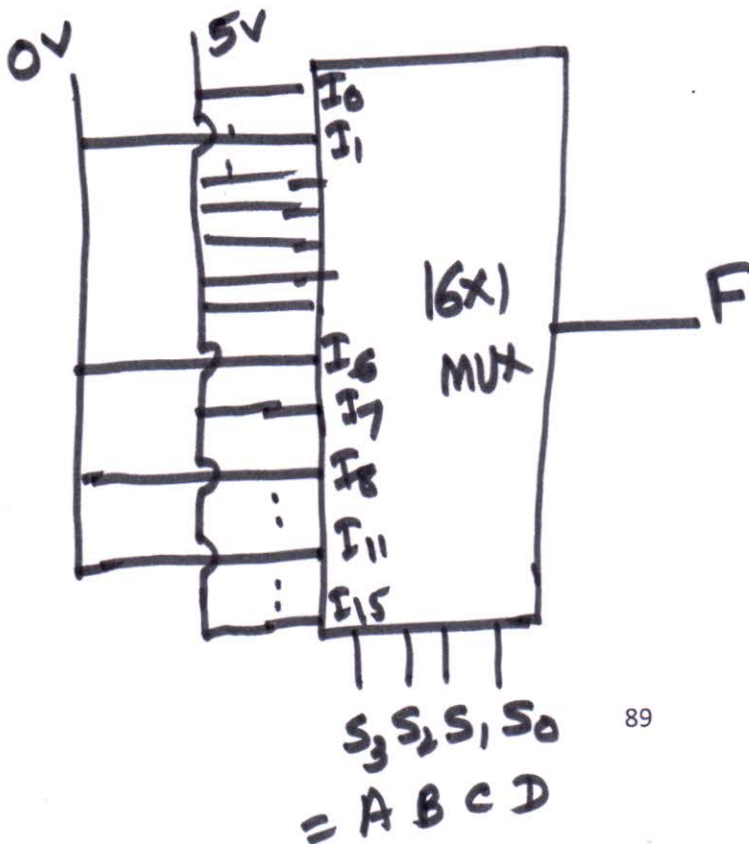$$\overline{EN}.S_0.S_1.I_3$$

II

EN	S_1	S_0	F
1	×	×	0
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3

88

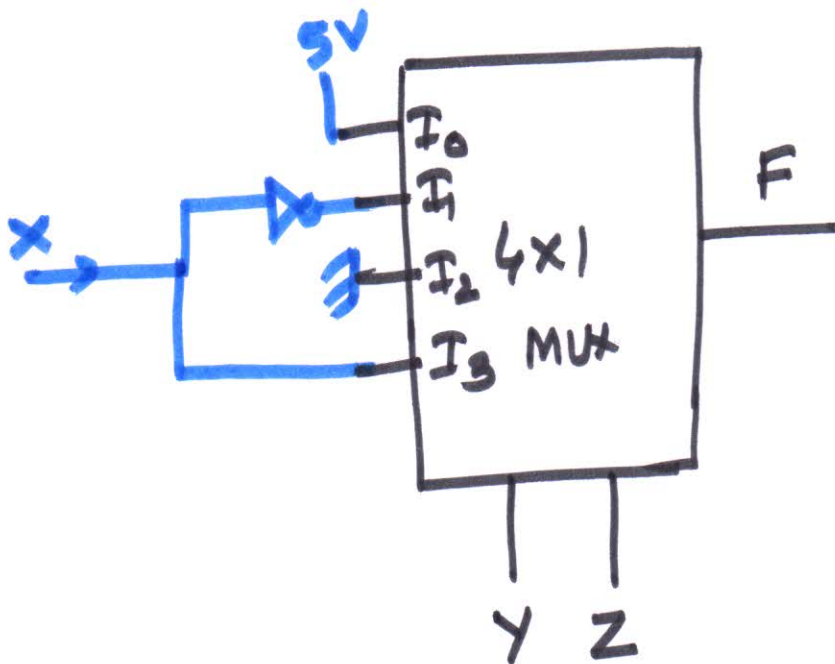## Implementing Boolean Functions with Multiplexers:

Ex. Implement $F(X, Y, Z) = \sum m(2, 4, 5)$ using 8-to-1 MUX.
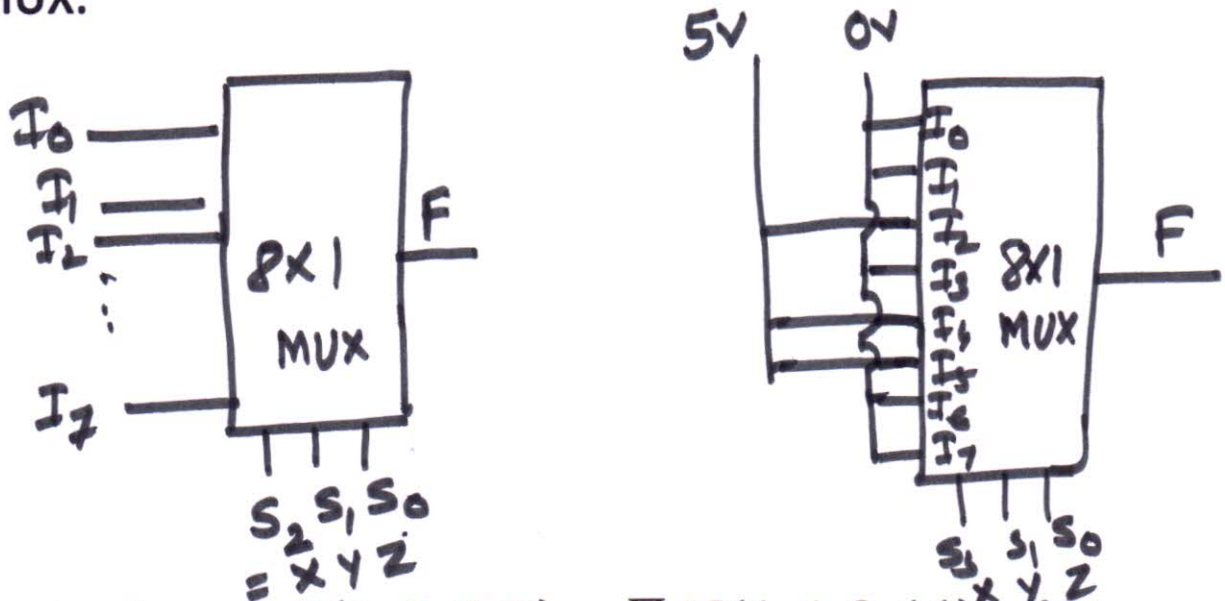
Ex. Implement $F(A, B, C, D) = \prod M(1, 6, 8, 11)$ using 16-to-1 MUX.

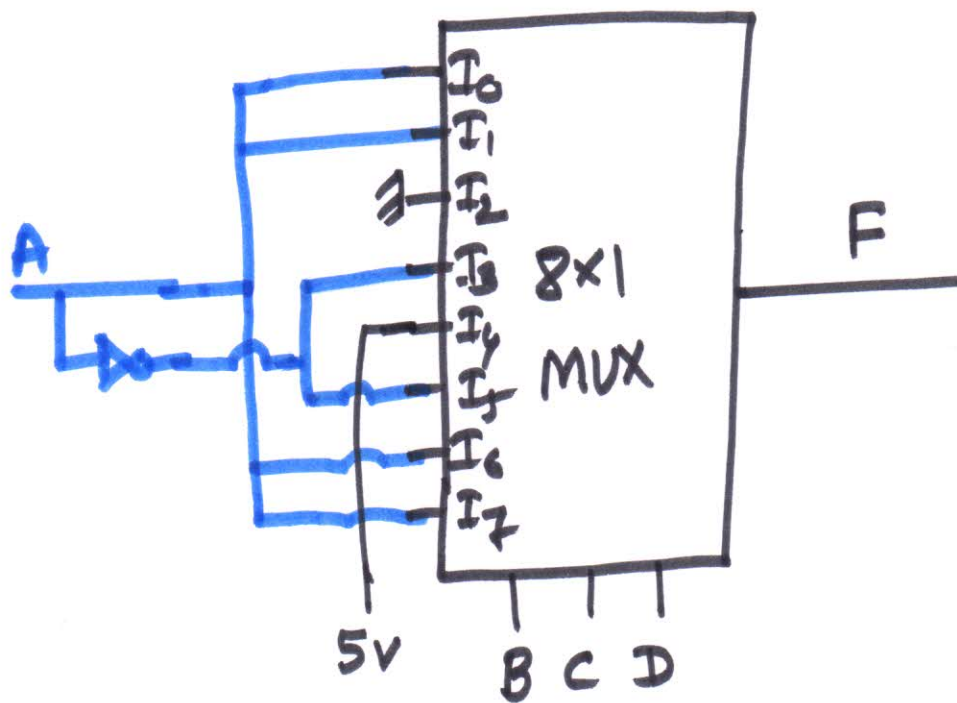**Ex. Implement** $F(X, Y, Z) = \sum m(0, 1, 4, 7)$ **using 4-to-1 MUX.**

X \ YZ	00	01	10	11
0	1	1	0	0
1	1	0	0	1

'1' $\bar{X}$ '0'   X

Ex.

Implement $F(A, B, C, D) = \sum m(3, 4, 5, 8, 9, 12, 14, 15)$ **using 8-to-1 MUX.**

A \ BCD	000	001	010	011	100	101	110	111
0	0	0	0	1	1	1	0	0
1	1	1	0	0	1	0	1	1
	A	A	0	$\bar{A}$	1	$\bar{A}$	A	A

Ex. Implement $F(X, Y, Z) = \sum m(3, 5, 6)$ using 2-to-1 MUX.

$$XY$$

$XY \backslash Z$	0	1
00	0	0
01	0	1
10	0	1
11	1	0

XY $X \oplus Y$

X

Y

I_0

I_1

2×1 MUX

F

Z