The material covered in this class will be as follows:

⇒Multiplexers.

⇒Boolean function implementation.

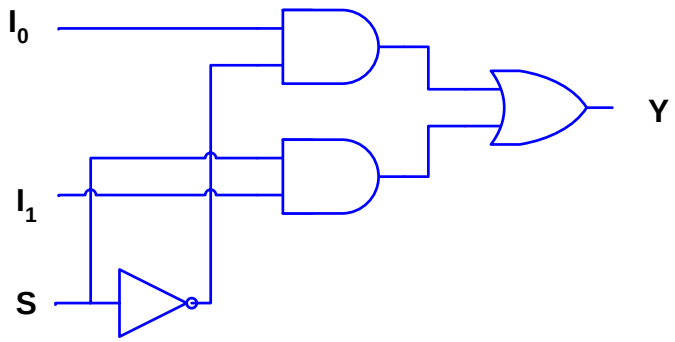
⇒Three-state gates.

Multiplexers

A multiplexer is a combinational circuit that selects one of many input lines (normally 2^n lines) and directs it to a single output line. The selection of a particular input line is controlled by a set of selection lines (normally n selection lines).

2-to-1 Line Multiplexer

A 2-to-1 line multiplexer has two inputs, one selection line and one output. This is shown in the following logic circuit.

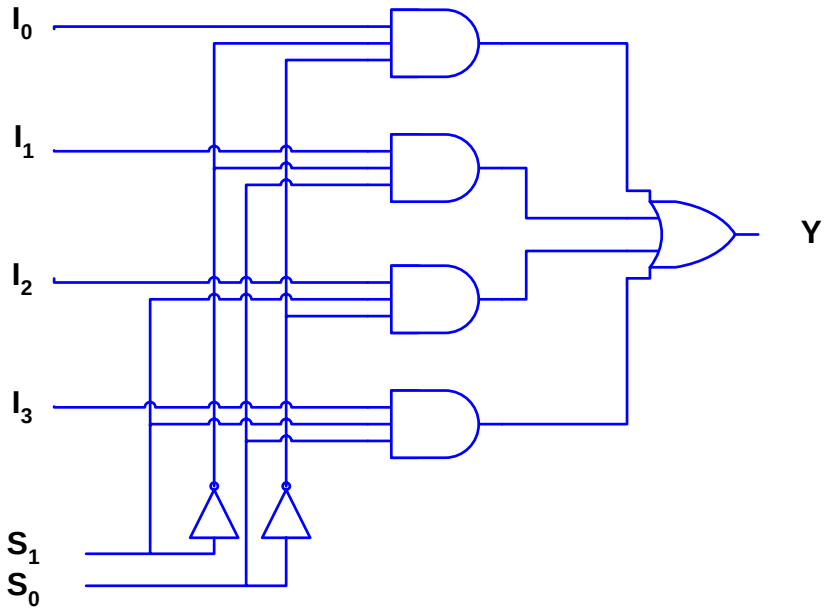


4-to-1 Line Multiplexer

A 4-to-1 line multiplexer consists of four AND gates. Each input is connected to one AND gate. Selection lines S1 and S0 are decoded to select a particular AND gate. The outputs of the AND gates are applied to a single OR gate that provides the output of the multiplexer Y.

The output of the multiplexer is then given by:

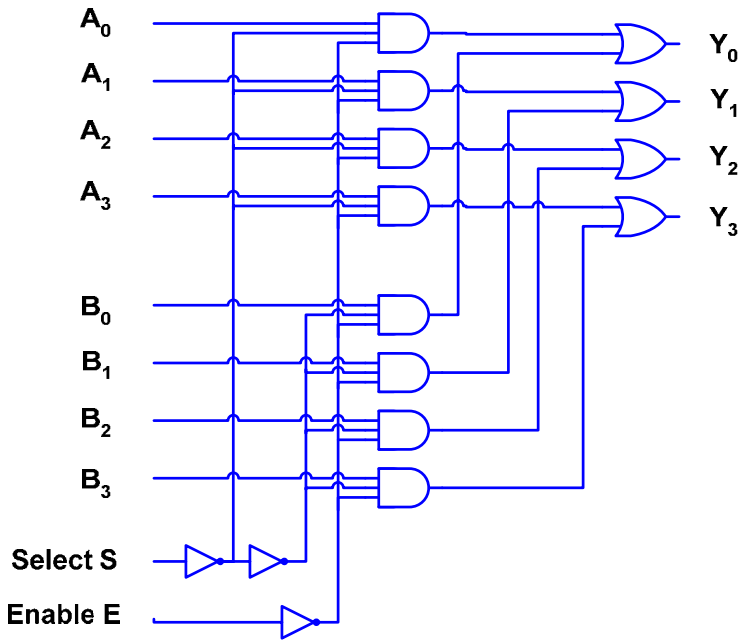
$$Y = S_1' S_0' I_0 + S_1' S_0 I_1 + S_1 S_0' I_2 + S_1 S_0 I_3$$



The function table of the multiplexer is shown next.

S_1	S_0	Y
0	O	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Multiplexers may have an enable input, similar to decoders, to control the operation of the unit. A quadruple 2-to-1 multiplexer with enable input is shown next.



The function table of the quadruple 2-to-1 multiplexer with the enable input will be as follows:

E	S	Y
1	X	All 0's
0	0	Select A
0	1	Select B

Boolean Function Implementation

A multiplexer is a decoder and an OR gate that provides the output. The multiplexer can be used to implement Boolean functions of n variables. This can be achieved using either 2^n-to-1 multiplexer or $2^{(n-1)}$-to-1 multiplexer.

1. Using 2^n-to-1 multiplexer

The n variables are connected to the n selection lines. Each input of the multiplexer is set to 0 or 1, depending on which minterm of the function is present.

Example: Implement $F(x,y,z) = \Sigma(1,2,6,7)$ using 8-to-1 multiplexer.

Solution: Connect the variables x, y, z to the selection inputs S_2, S_1, and S_0. Then set $I_0 = I_3 = I_4 = I_5 = 0$ and $I_1 = I_2 = I_6 = I_7 = 1$.

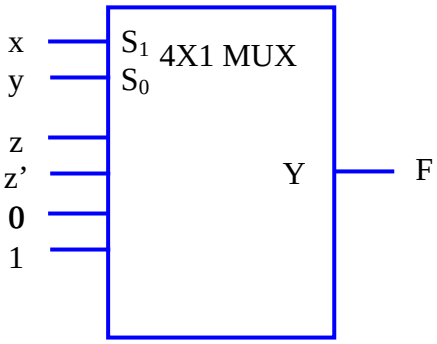
2. Using $2^{(n-1)}$-to-1 multiplexer

We connect (n-1) variables to the selection lines. The multiplexer inputs are going to be either 0 or 1 or the remaining variable or the complement of the remaining variable.

Example: Implement $F(x,y,z) = \Sigma(1,2,6,7)$ using 4-to-1 multiplexer.

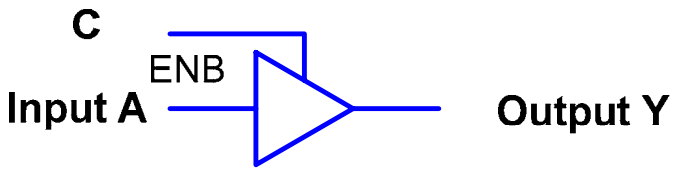
Solution: Connect the variables x and y to the selection inputs S_1, and S_0. The inputs of the multiplexers can be obtained from the truth table as shown below.

x	y	z	F	
0	0	0	0	F=z
0	0	1	1	
0	1	0	1	F=z'
0	1	1	0	
1	0	0	0	F=0
1	0	1	0	
1	1	0	1	F=1
1	1	1	1	



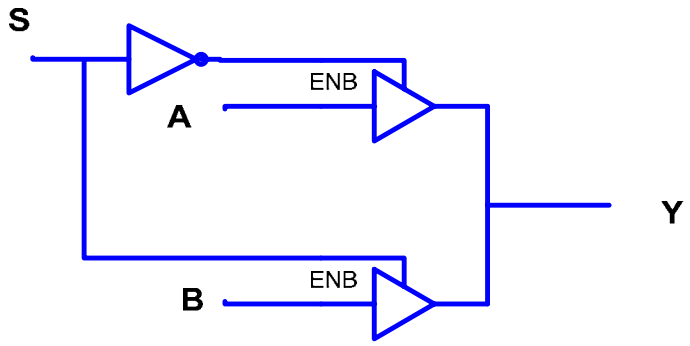
Three State Gates

A three state gate is a digital circuit that exhibits three states. Two states are equivalent to logic 0 and 1. The third state is a high impedance state which is controlled by a control input C. The most commonly used 3-state gate is the buffer.



The output Y = A when C =1 and is high impedance state when C = 0.

It is possible to implement multiplexers using 3-state buffers as shown.



A 4-to-1 multiplexer may be constructed using four 3-state buffers and a 2-to-4 decoder.