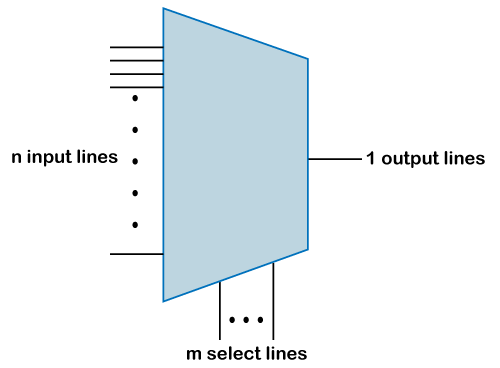
A multiplexer is a device that selects one output from multiple inputs. It is also known as a data selector. We refer to a multiplexer with the terms **MUX** and **MPX**.

Multiplexers are used in communication systems to increase the amount of data sent over a network within a certain amount of time and bandwidth. It allows us to squeeze multiple data lines into one data line.

It switches between one of the many input lines and combines them one by one to the output. It decides which input line to switch using a control signal.

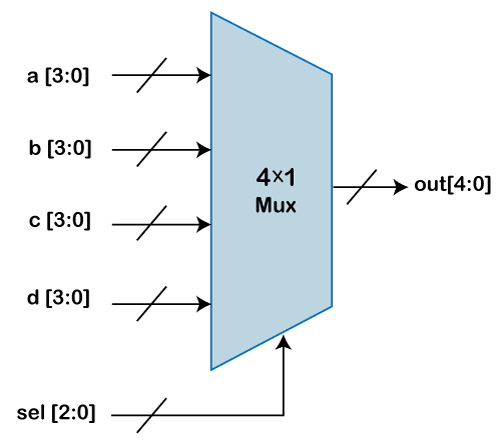
Physically, a multiplexer has n input pins, one output pin, and m control pins. n = 2^m. Since a multiplexer's job is to select one of the data input lines and send it to the output, it is also known as a **data selector**.



There are three main ways of constructing a multiplexer.

* Digital multiplexers are made up of logic gates.
* Analog multiplexers are made using transistors.
* Mechanical switches or rotary switches are made using rotating shafts.

The ***MUX*** itself acts like a digitally controlled multi-position switch where the binary code applied to the select inputs controls the data input, which will be switched to the output.

For example, it transfers data from one of the N inputs to the output based on the select signal. A 4-bit multiplexer would have N inputs each of 4 bits where each input can be transferred to the output by using a select signal. 

**Sel** is a 2-bit input and can have four values. Each value on the select line will allow one of the inputs to be sent to the output.

A 4x1 multiplexer can be implemented in multiple ways and here we show two of the most common ways:

**1. Using assign statement**

1. module mux\_4to1\_assign ( input [3:0] a,                 // 4-bit input called a
2. input [3:0] b,                 // 4-bit input called b
3. input [3:0] c,                 // 4-bit input called c
4. input [3:0] d,                 // 4-bit input called d
5. input [1:0] sel,               // input sel used to select between a,b,c,d
6. output [3:0] out);             // 4-bit output based on input sel
8. // When sel[1] is 0, (sel[0]? b:a) is selected and sel[1] is 1, (sel[0] ? d:c) is taken
9. // If sel[0] is 0, a is sent to output, else b and if sel[0] is 0, c is sent to output, else d
10. assign out = sel[1] ? (sel[0] ? d : c) : (sel[0] ? b : a);
12. endmodule

The module called **mux\_4x1\_assign** has four 4-bit data inputs, one 2-bit select input and one 4-bit data output. The multiplexer will select either a, b, c, or d based on the select signal sel using the assign statement.

**2. Using case statement**

When we use case statement, then the signal **out** is declared as a **reg** type because it is used in a **procedural** block.

1. module mux\_4to1\_case ( input [3:0] a,                 // 4-bit input called a
2. input [3:0] b,                 // 4-bit input called b
3. input [3:0] c,                 // 4-bit input called c
4. input [3:0] d,                 // 4-bit input called d
5. input [1:0] sel,               // input sel used to select between a,b,c,d
6. output reg [3:0] out);         // 4-bit output based on input sel
8. // This always block gets executed whenever a/b/c/d/sel changes value
9. // When it happens, output is assigned to either a/b/c/d
10. always @ (a or b or c or d or sel) begin
11. **case** (sel)
12. 2'b00 : out <= a;
13. 2'b01 : out <= b;
14. 2'b10 : out <= c;
15. 2'b11 : out <= d;
16. endcase
17. end
18. endmodule

The module called ***mux\_4x1\_case*** has four 4-bit data inputs, one 2-bit select input and one 4-bit data output. The multiplexer will select either a, b, c, or d based on the select signal sel using the case statement.

Uses of Multiplexer

The multiplexer includes the following useful points, such as:

1. In a communication system where we have a communication network, a multiplexer increases the system's efficiency by allowing audio and video data transmission on a single channel.
2. In optical fiber communication, a multiplexer does the same job to combine multiple fiber cables onto one fiber cable using a technique called ***Dense wavelength division multiplexing***.
3. In satellite communication, multiplexers transfer data from the satellite's computer system to the ground segment using ***GSM*** communication.
4. It also works as a parallel to serial data converter.
5. A computer decreases the number of copper lines necessary to connect the memory to other computer parts.

How to join multiplexers?

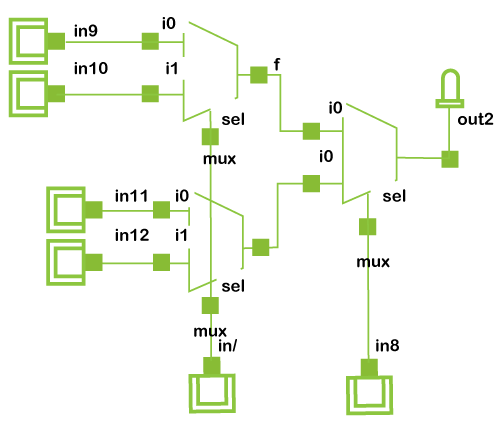
If we have small multiplexers, but we want to increase their functionality, we can join them to obtain a MUX with more inputs. The cascading of multiplexers is easy. Ensure that we connect to give the same number of inputs and control lines as the target MUX.

Let's make 4:1 MUX using 2:1 multiplexers. We know that a 2:1 MUX has two inputs and one select line. So joining two 2:1 multiplexers will give us four inputs and two select lines.

We can reduce the outputs to one, so we use another 2:1 MUX to combine two lines to get a single line.

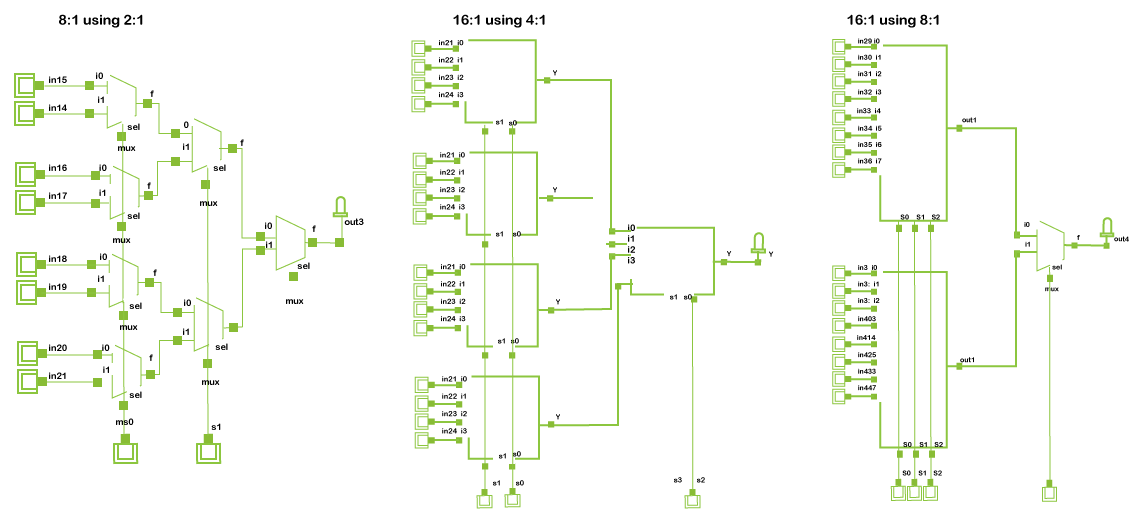
However, though that gives us the one output that we require, it gives us an additional select line. So now we have three select lines.

Now we reduce three select lines to two select lines. We can do that by joining two select lines. That would essentially reduce the two lines to one single line. The following image shows the result we get by applying our logic.

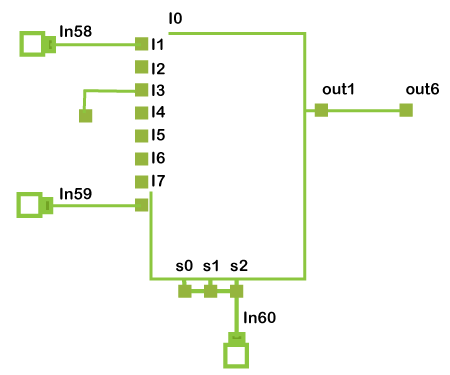


### 8:1 and 16:1 Multiplexers

Similar to the process we saw above, we can design an 8 to 1 multiplexer using 2:1 multiplexers, 16:1 MUX using 4:1 MUX, or 16:1 MUX using 8:1 multiplexer.



We can also go the opposite way and use a multiplexer with more inputs than required as a smaller MUX. Here's an 8:1 multiplexer being used as a 2:1 multiplexer.



\*\*\*\*\*\*\*\*