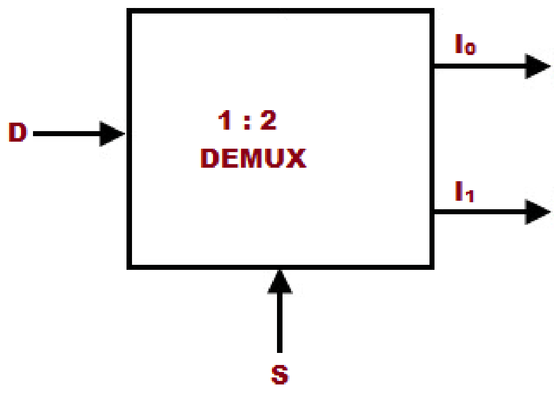
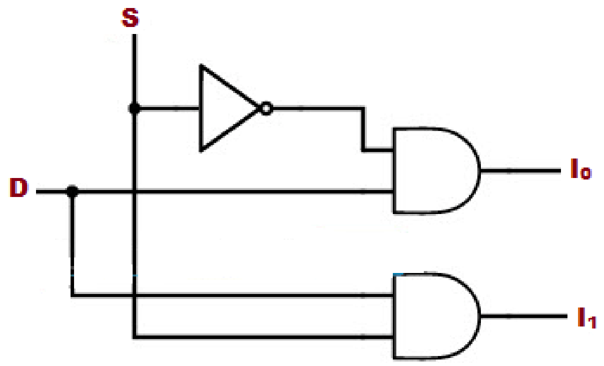
1 X 2 DEMUX

A 1-to-2 demultiplexer consists of one input line, two output lines and one select line. The signal on the select line helps to switch the input to one of the two outputs. The figure below shows the block diagram of a 1-to-2 demultiplexer with additional enable input.

In the figure, there are only two possible ways to connect the input to output lines, thus only one select signal is enough to do the demultiplexing operation. When the select input is LOW, then the input will be passed to Y0 and if the select input is HIGH, then the input will be passed to Y1.



(a)



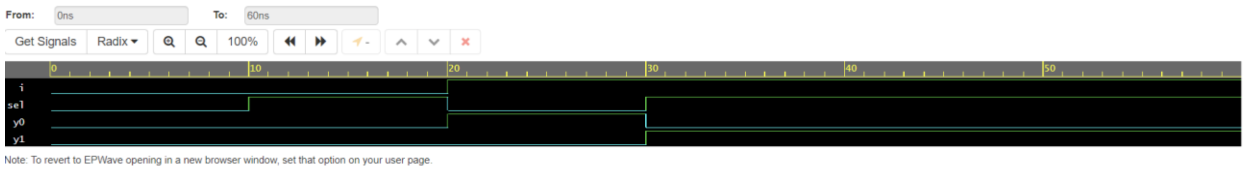
(b)

RTL CODE:

```
module demux_2_1(
  input sel,
  input i,
  output y0, y1);

  assign {y0,y1} = sel?{1'b0,i}: {i,1'b0};
endmodule
```

TESTBENCH:

```
module demux_tb;
  reg sel, i;
  wire y0, y1;

  demux_2_1 demux(sel, i, y0, y1);
  initial begin
    $dumpfile("dump.vcd");
    $dumpvars(1);
  end
  initial begin
    sel=0;i=0;
    #10 sel=1;i=0;
    #10 sel=0; i=1;
    #10 sel=1; i=1;
  end
  initial begin
    #10 $finish();
  end
endmodule
```



Note: To revert to EPWave opening in a new browser window, set that option on your user page.

1X4 DEMUX

A 1-to-4 demultiplexer has a single input (D), two selection lines (S1 and S0) and four outputs (Y0 to Y3). The input data goes to any one of the four outputs at a given time for a particular combination of select lines.

This demultiplexer is also called as a 2-to-4 Demultiplexer, which means that it has two select lines and 4 output lines. The block diagram of a 1:4 DEMUX is shown below.

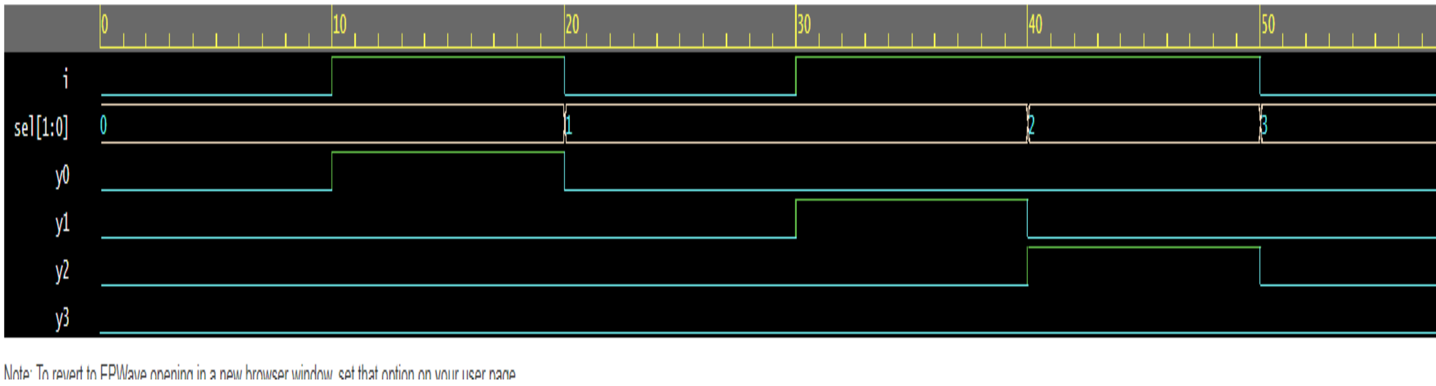
The picture can't be displayed.

RTL CODE:

```
module demux_1_4(
  input [1:0] sel,
  input  i,
  output reg y0,y1,y2,y3);

  always @(*) begin
    case(sel)
      2'h0: {y0,y1,y2,y3} = {i,3'b0};
      2'h1: {y0,y1,y2,y3} = {1'b0,i,2'b0};
      2'h2: {y0,y1,y2,y3} = {2'b0,i,1'b0};
      2'h3: {y0,y1,y2,y3} = {3'b0,i};
      default: $display("Invalid sel input");
    endcase
  end
endmodule
```

TESTBENCH:

```
module tb;
  reg [1:0] sel;
  reg i;
  wire y0,y1,y2,y3;

  demux_1_4 demux(sel, i, y0, y1, y2, y3);
  initial begin
    $dumpfile("dump.vcd");
    $dumpvars(1);
  end

  initial begin
    sel=2'b00; i=0;
    #10 sel=2'b00; i=1;
    #10 sel=2'b01; i=0;
    #10 sel=2'b01; i=1;
    #10 sel=2'b10; i=1;
    #10 sel=2'b11; i=0;
    #10 sel=2'b11; i=1;
  end
  initial begin
    #60 $finish();
  end
endmodule
```



Note: To revert to EPWave opening in a new browser window, set that option on your user page