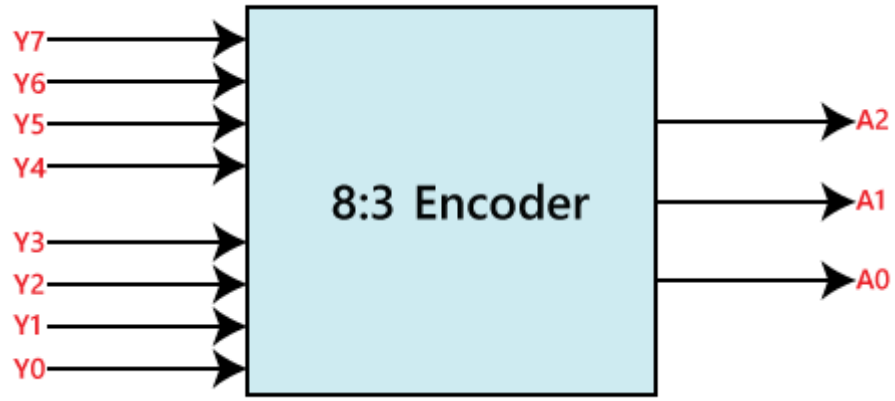
## ENCODERS:

An **Encoder** is a combinational circuit that performs the reverse operation of Decoder. It has maximum of $2^n$ input lines and 'n' output lines. It will produce a binary code equivalent to the input, which is active High. Therefore, the encoder encodes $2^n$ input lines with 'n' bits. It is optional to represent the enable signal in encoders…

The 8 to 3 line Encoder is also known as **Octal to Binary Encoder**. In 8 to 3 line encoder, there is a total of eight inputs, i.e., $Y_0$, $Y_1$, $Y_2$, $Y_3$, $Y_4$, $Y_5$, $Y_6$, and $Y_7$ and three outputs, i.e., $A_0$, A1, and $A_2$. In 8-input lines, one input-line is set to true at a time to get the respective binary code in the output side. Below are the block diagram and the truth table of the 8 to 3 line encoder.

### Block Diagram:

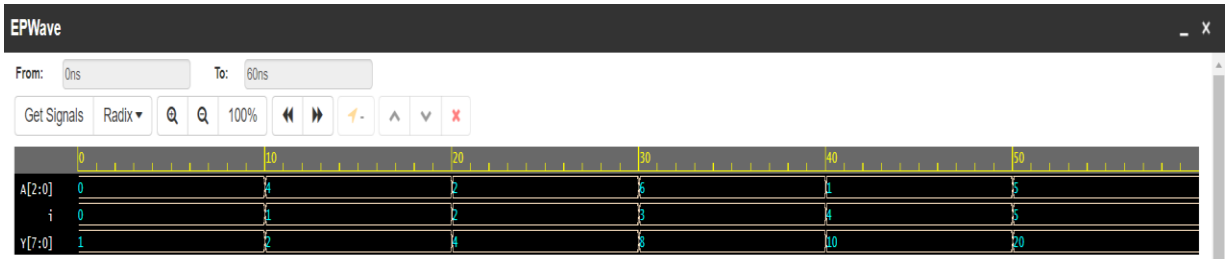



## RTL CODE:

```
module DeCoders(input [7:0]Y, output [2:0]A);
  assign A[0]= Y[4] | Y[5] | Y[6] | Y[7];
  assign A[1]= Y[2] | Y[3] | Y[6] | Y[7];
  assign A[2]= Y[1] | Y[3] | Y[5] | Y[7];
endmodule
```

## TESTBENCH:

```
module testbench;
  reg [7:0]Y;
  wire [2:0]A;
  int i;
  DeCoders a1 (Y,A);
  initial
    begin
      $dumpfile(".vcd");
      $dumpvars(1);
    end
  initial
    begin
      Y=8'b1;
      for(i=0;i<8;i++)
        begin
          #10 Y=Y<<1;
        end
    end
  initial
    begin
      #60 $finish();
    end
endmodule
```


EPWave
From:
0ns
To:
60ns
Get Signals
Radix
100%
0
10
20
30
40
50
A[2:0]
0
4
2
6
1
5
i
0
1
2
3
4
5
Y[7:0]
1
2
4
8
10
20