

We are generating a 5-to-32 decoder hierarchically in Verilog. We have generated 2-to-4 decoder defined as follows:

```
module decoder2_4(  
  input [1:0] in,  
  output [3:0] out  
);  
  assign out[0] = ~in[1] & ~in[0];  
  assign out[1] = ~in[1] & in[0];  
  assign out[2] = in[1] & ~in[0];  
  assign out[3] = in[1] & in[0];  
endmodule
```

Please include the previous module(s) in your response and use them to hierarchically generate a 3-to-8 decoder defined as:

```
module decoder3_8(in[2:0], out[7:0]);  
  //Insert code here  
endmodule
```

```
module decoder2_4(  
  input [1:0] in,  
  output [3:0] out  
);  
  assign out[0] = ~in[1] & ~in[0];  
  assign out[1] = ~in[1] & in[0];  
  assign out[2] = in[1] & ~in[0];  
  assign out[3] = in[1] & in[0];  
endmodule  
  
module decoder3_8(  
  input [2:0] in,  
  output [7:0] out  
);  
  wire [3:0] lower, upper;  
  
  decoder2_4 lower_decoder(  
    .in(in[1:0]),  
    .out(lower)  
  );  
  
  decoder2_4 upper_decoder(  
    .in(in[1:0]),  
    .out(upper)
```

```
    );  
  
    assign out[3:0] = lower & {4{~in[2]}};  
    assign out[7:4] = upper & {4{in[2]}};  
endmodule
```