



## DAY-14

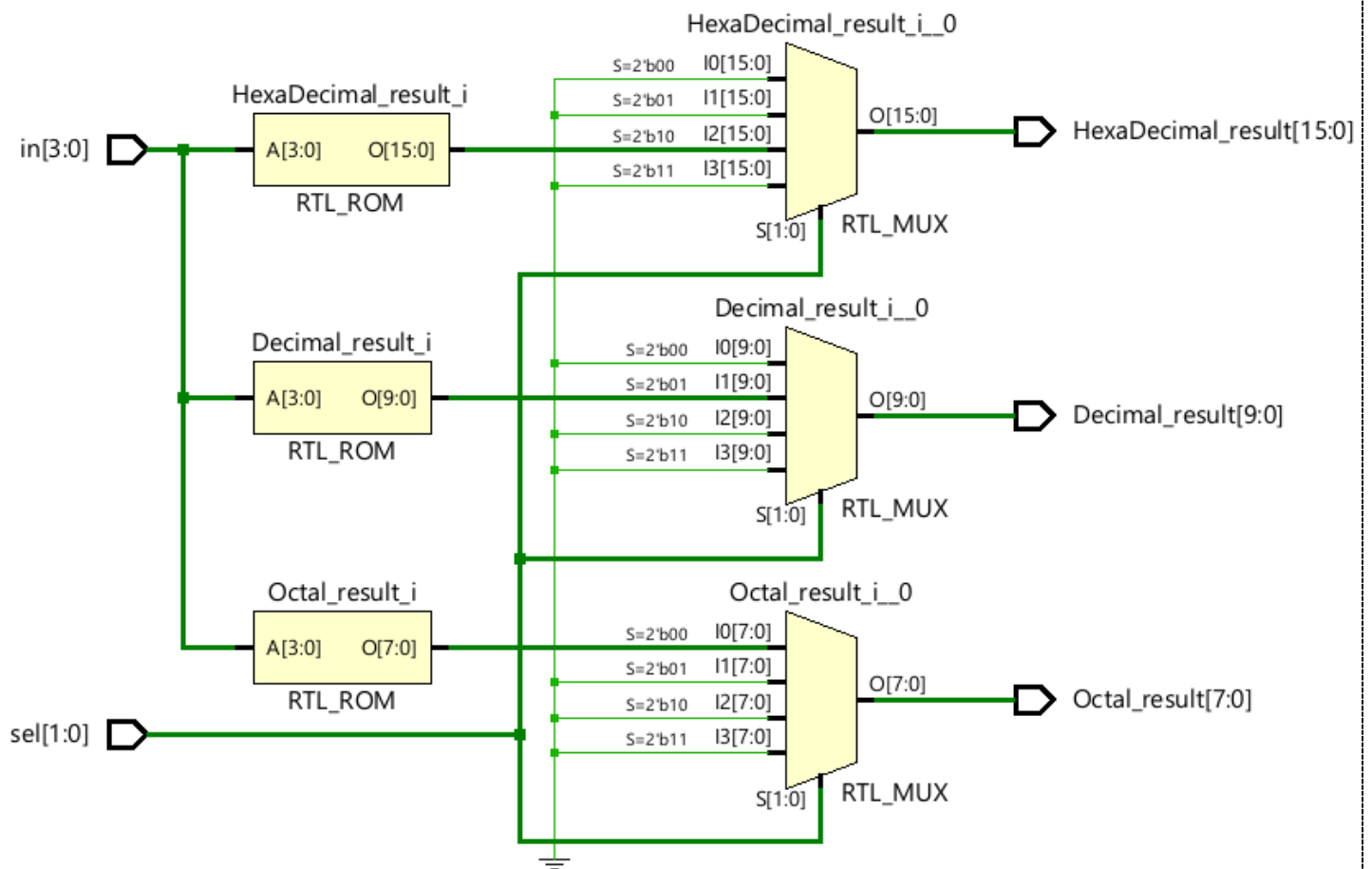
### #100DAYSRTL

**“Aim”:-** To Design the Decoder which converts the binary value to octal, decimal and Hexadecimal

**“Verilog Code”:-**

```
module Decoder(  
    input [3:0] in,  
    input [1:0] sel,  
    output reg [7:0] Octal_result,  
    output reg [9:0] Decimal_result,  
    output reg [15:0] Hexadecimal_result);  
always @(*) begin  
    case(sel)  
        2'b00: begin // Binary to Octal conversion  
            Decimal_result=0;  
            Hexadecimal_result=0;  
            case(in)  
                4'd0: Octal_result = {7'b0, 1};  
                4'd1: Octal_result = {6'b0, 1, 0};  
                4'd2: Octal_result = {5'b0, 1, 2'b0};  
                4'd3: Octal_result = {4'b0, 1, 3'b0};  
                4'd4: Octal_result = {3'b0, 1, 4'b0};  
                4'd5: Octal_result = {2'b0, 1, 5'b0};  
                4'd6: Octal_result = {0,1,6'b0};  
                4'd7: Octal_result = {1, 7'b0};  
                default: Octal_result = 8'b0;  
            endcase  
        end  
        2'b01: begin //Binary to decimal  
            Hexadecimal_result=0;  
            Octal_result=0;  
            case(in)  
                4'd0: Decimal_result = {9'b0, 1};  
                4'd1: Decimal_result = {8'b0, 1, 0};  
                4'd2: Decimal_result = {7'b0, 1, 2'b0};  
                4'd3: Decimal_result = {6'b0, 1, 3'b0};  
                4'd4: Decimal_result = {5'b0, 1, 4'b0};  
                4'd5: Decimal_result = {4'b0, 1, 5'b0};  
                4'd6: Decimal_result = {3'b0, 1, 6'b0};  
                4'd7: Decimal_result = {2'b0, 1, 7'b0};  
                4'd8: Decimal_result = {1'b0, 1, 8'b0};  
                4'd9: Decimal_result = {1, 9'b0};  
                default: Decimal_result = 10'b0;  
            endcase  
        end  
        2'b10: begin //binary to hexadecimal  
            Decimal_result=0;  
            Octal_result=0;  
            case(in)  
                4'd0: Hexadecimal_result = {15'b0, 1};  
                4'd1: Hexadecimal_result = {14'b0, 1, 0};  
                4'd2: Hexadecimal_result = {13'b0, 1, 2'b0};  
                4'd3: Hexadecimal_result = {12'b0, 1, 3'b0};  
                4'd4: Hexadecimal_result = {11'b0, 1, 4'b0};  
                4'd5: Hexadecimal_result = {10'b0, 1, 5'b0};  
                4'd6: Hexadecimal_result = {9'b0, 1, 6'b0};  
                4'd7: Hexadecimal_result = {8'b0, 1, 7'b0};  
                4'd8: Hexadecimal_result = {7'b0, 1, 8'b0};  
                4'd9: Hexadecimal_result = {6'b0, 1, 9'b0};  
                4'd10: Hexadecimal_result = {5'b0, 1, 10'b0};  
                4'd11: Hexadecimal_result = {4'b0, 1, 11'b0};  
                4'd12: Hexadecimal_result = {3'b0, 1, 12'b0};  
                4'd13: Hexadecimal_result = {2'b0, 1, 13'b0};  
                4'd14: Hexadecimal_result = {1'b0,1,14'b0};  
                4'd15: Hexadecimal_result = {1, 15'b0};  
                default: Hexadecimal_result = 16'b0;  
            endcase  
        end  
        2'b11: begin  
            // No operation  
        end  
    endcase  
end  
endmodule
```

## “Schematics”:-



## “Waveforms”:-

