CS224_2021_quiz1_A

Points: 14/20

1. Write Verilog code to design a 8-bit Barrel shifter that rotates the input number 'a' by an arbitrary number of positions to the right. The number of positions to rotate is given as a control input 'ctrl' which is a 3-bit value. Use 'case' statement to design the module. The output will go in variable 'out'.

   Write the Verilog code for this with help of Xilinx ISE and paste it in the space below.
   (2/8 Points)

```
module barrel(
    input [7:0] A,
    input [2:0] ctrl,
    output reg [7:0] out
    );

always@(*)
        begin
                case(ctrl)
                        3'b000 : begin
                                        out[0]= A[0];
                                        out[1]= A[1];
                                        out[2]= A[2];
                                        out[3]= A[3];
                                        out[4]= A[4];
                                        out[5]= A[5];
                                        out[6]= A[6];
                                        out[7]= A[7];
                                        end

                        3'b001 : begin
                                        out[0]= A[7];
                                        out[1]= A[0];
                                        out[2]= A[1];
                                        out[3]= A[2];
                                        out[4]= A[3];
                                        out[5]= A[4];
                                        out[6]= A[5];
                                        out[7]= A[6];
                                        end

                        3'b010 : begin
                                        out[0]= A[6];
                                        out[1]= A[7];
                                        out[2]= A[0];
                                        out[3]= A[1];
                                        out[4]= A[2];
                                        out[5]= A[3];
                                        out[6]= A[4];
                                        out[7]= A[5];
                                        end

                        3'b011 : begin
                                        out[0]= A[5];
                                        out[1]= A[6];
                                        out[2]= A[7];
                                        out[3]= A[0];
                                        out[4]= A[1];
                                        out[5]= A[2];
                                        out[6]= A[3];
                                        out[7]= A[4];
```

```
                        end
                3'b100 : begin
                        out[0]= A[4];
                        out[1]= A[5];
                        out[2]= A[6];
                        out[3]= A[7];
                        out[4]= A[0];
                        out[5]= A[1];
                        out[6]= A[2];
                        out[7]= A[3];
                        end

                3'b101 : begin

                        out[0]= A[3];
                        out[1]= A[4];
                        out[2]= A[5];
                        out[3]= A[6];
                        out[4]= A[7];
                        out[5]= A[0];
                        out[6]= A[1];
                        out[7]= A[2];
                        end

                3'b110 : begin
                        out[0]= A[2];
                        out[1]= A[3];
                        out[2]= A[4];
                        out[3]= A[5];
                        out[4]= A[6];
                        out[5]= A[7];
                        out[6]= A[0];
                        out[7]= A[1];
                        end

                3'b111 : begin
                        out[0]= A[1];
                        out[1]= A[2];
                        out[2]= A[3];
                        out[3]= A[4];
                        out[4]= A[5];
                        out[5]= A[6];
                        out[6]= A[7];
                        out[7]= A[0];
                        end


            endcase
        end

endmodule
```

“you have done rotate left”

2. For the barrel shifter designed in previous question, write the components synthesised in the RTL. Use the synthesis report to write this answer.
(2/2 Points)

one 8-bit 8-to-1 multiplexer

3. Modify the barrel shifter code from previous question and design a 4-bit barrel shifter. This 4-bit shifter has one more input 'dir' which indicates the direction of rotation. if dir=1 rotate right else rotate left. Write the Verilog code for this with help of Xilinx ISE and paste it in the space below.
(8/8 Points)

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date:    14:30:48 02/02/2021
// Design Name:
// Module Name:    barrel4
// Project Name:
// Target Devices:
// Tool versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////
module barrel4(
    input [3:0] A,
    input dir,
    input [1:0] ctrl,
    output reg [3:0] out
    );
always@(*)
        begin
                case(ctrl)
                        3'b000 : begin
                                                case(dir)
                                                        1'b0:begin

                                                out[0]= A[0];
                                                out[1]= A[1];
                                                out[2]= A[2];
                                                out[3]= A[3];
                                                        end
                                                        1'b1:begin

                                                out[0]= A[0];
                                                out[1]= A[1];
                                                out[2]= A[2];
                                                out[3]= A[3];
                                                        end
                                                endcase
                                                end

                        3'b001 : begin
                                                case(dir)
                                                        1'b0:begin
```

```
                    out[0]= A[1];
                    out[1]= A[2];
                    out[2]= A[3];
                    out[3]= A[0];
                        end
                        1'b1:begin

                    out[0]= A[3];
                    out[1]= A[0];
                    out[2]= A[1];
                    out[3]= A[2];
                        end
                    endcase
                    end

            3'b010 : begin
            case(dir)
                        1'b0:begin

                    out[0]= A[2];
                    out[1]= A[3];
                    out[2]= A[0];
                    out[3]= A[1];
                        end
                        1'b1:begin

                    out[0]= A[2];
                    out[1]= A[3];
                    out[2]= A[0];
                    out[3]= A[1];
                        end
                    endcase
                    end

            3'b011 : begin
            case(dir)
                        1'b0:begin

                    out[0]= A[3];
                    out[1]= A[0];
                    out[2]= A[1];
                    out[3]= A[2];
                        end
                        1'b1:begin

                    out[0]= A[1];
                    out[1]= A[2];
                    out[2]= A[3];
                    out[3]= A[0];
                        end
                    endcase
                    end
```

```
            endcase
        end


endmodule
```

4. For the new barrel shifter in Q-3 write the components synthesised in the RTL. Use the synthesis report to write this answer
(2/2 Points)

```
4-bit 2-to-1 multiplexer                : 2
 4-bit 4-to-1 multiplexer                : 1
```