DIGITAL

DESIGN

PROJECT

Question 1.A

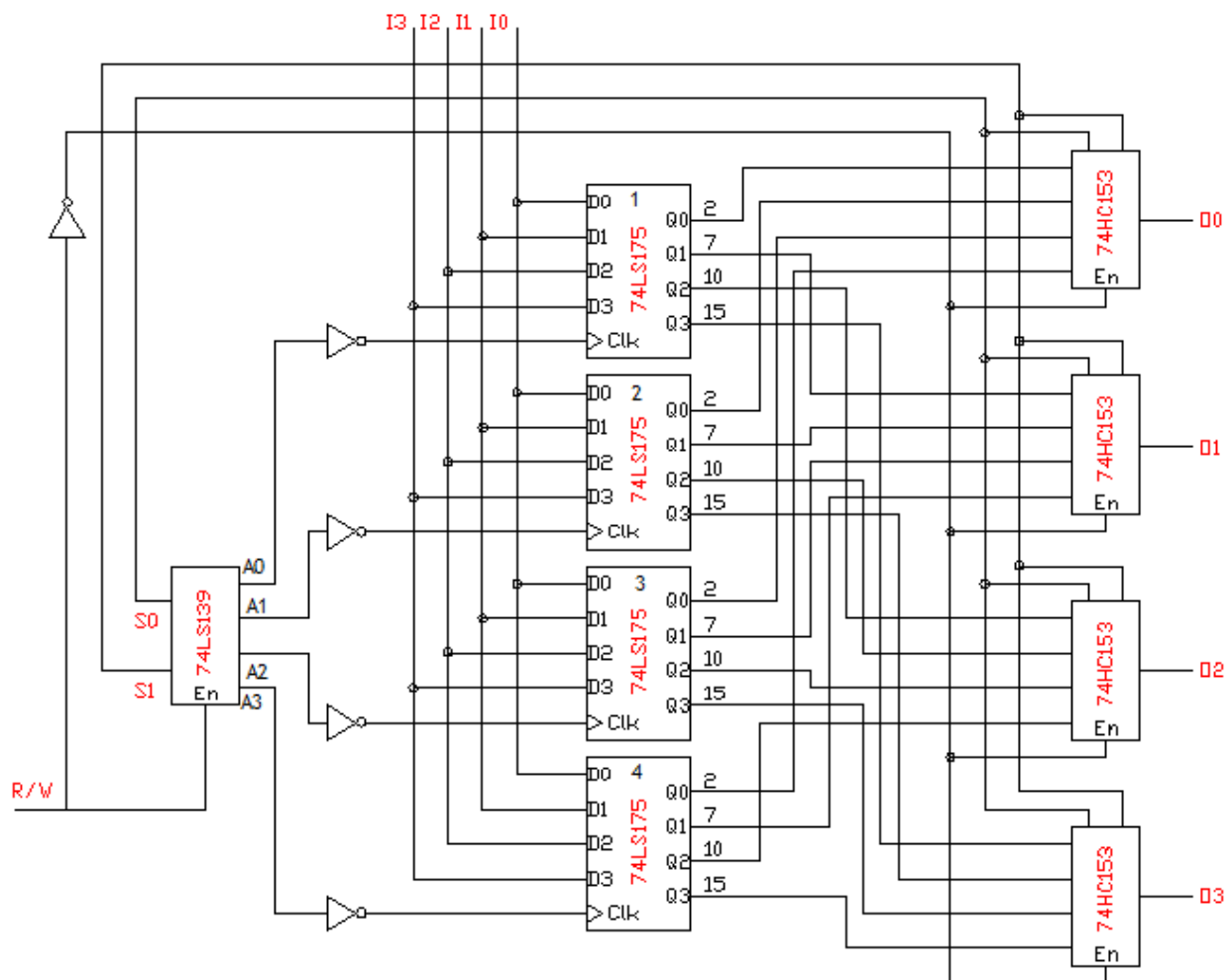Shatrujit Aditya Kumar - 2013A7PS044G (Batch 1)

Meghana Gupta - 2013A7PS042G (Batch 1)

Sanket Gupte - 2013A7PS078G (Batch 2)

Pratik Singhal - 2013A7PS019G (Batch 2)

Sreehari S. – 2013A7PS126G (Batch 3)

Karthika Nair - 2013A7PS202G (Batch 3)

I3 I2 I1 I0

D0   1
D1   74LS175
D2
D3
> Clk
Q0 2
Q1 7
Q2 10
Q3 15

D0   2
D1   74LS175
D2
D3
> Clk
Q0 2
Q1 7
Q2 10
Q3 15

D0   3
D1   74LS175
D2
D3
> Clk
Q0 2
Q1 7
Q2 10
Q3 15

D0   4
D1   74LS175
D2
D3
> Clk
Q0 2
Q1 7
Q2 10
Q3 15

74HC153
En
O0

74HC153
En
O1

74HC153
En
O2

74HC153
En
O3

A0
A1
A2
A3
74LS139
En

S0
S1

R/W

BLOCK 1

List of ICs required (Total circuit):

8 X 74LS175 - Quad-D flip-flops (Pos. edge triggered)

4 X 74HC153 - Dual 4to 1 multiplexer

1 X 74LS139 - 2 X4 Decoder

1 X 7404 - Hex inverter

(Note: Decoder, multiplexer and inverter ICs are common in both the blocks. Separate D flip-flop ICs are used for different blocks.)

D Flip-Flop IC:

The D flip-flops (DFF) are positive edge triggered. Each DFF can hold 1 bit, and since each IC contains 4 DFFs, each block can hold 16 bits of data. The input loaded into the DFF is stored and can be retrieved from the output pin on corresponding to the Q pin in a DFF.

Parallel loading is accomplished by applying the four bits of data. The output from the decoder is given using the selection bits (S0 and S1), and accordingly, the DFF IC in which the data is to be loaded is selected. The data is loaded into the associated DFF and appears at the output of the DFF after the low-to-high transition (positive edge) of the clock signal (the inverter makes, which is basically the decoder output. During loading, the entry of serial data is inhibited. At all other times, the data in the registers is retained.

2x4 Decoder:

A 2x4 decoder is used to select a DFF to carry out write operation. Our enable signal is W. To write into the DFFs, the W signal should be low (0). When both select signals S0 and S1 are low (0), the output A0 is low (0) and all others are high. Thus DFF IC 1 is selected. Similarly, different sets of S0 and S1 select different DFF ICs. When W is high, no DFF IC is selected and write operation cannot be performed.

Now the outputs of the decoder go as the clock signals to the 4 ICs respectively. During write operation, depending upon the select signals, one of the outputs of the decoder goes from high (1) to low (0). However, the inverter IC makes sure the clock is inverted, i.e., the transition is low (0) to high (1). During this positive edge trigger, information is loaded in the DFF in a parallel manner.

4x1 Multiplexer:

In a quadruple 2x1 line multiplexer, 4-bit signal is selected in accordance with the select signal S when En signal is low (0). When En is high (1) all outputs are low (0).

The multiplexers have the same select signal as the decoder, i.e., S0 and S1. The enable signal for the multiplexers is the opposite of that for the decoder. This means that when the enable signal for the decoder, which is the R/W signal, is high (1), the circuit is in read mode, and the multiplexers' enable signal is inverted to give a low signal which activates them, and vice versa. When S1 and S0 are both low (0), the first DFF IC in each block is selected and thus, the data from the first DFF IC is read. Similarly, different DFF ICs are selected this way, and 8 bits of data can be read or written at a time.

RTL VIEW :
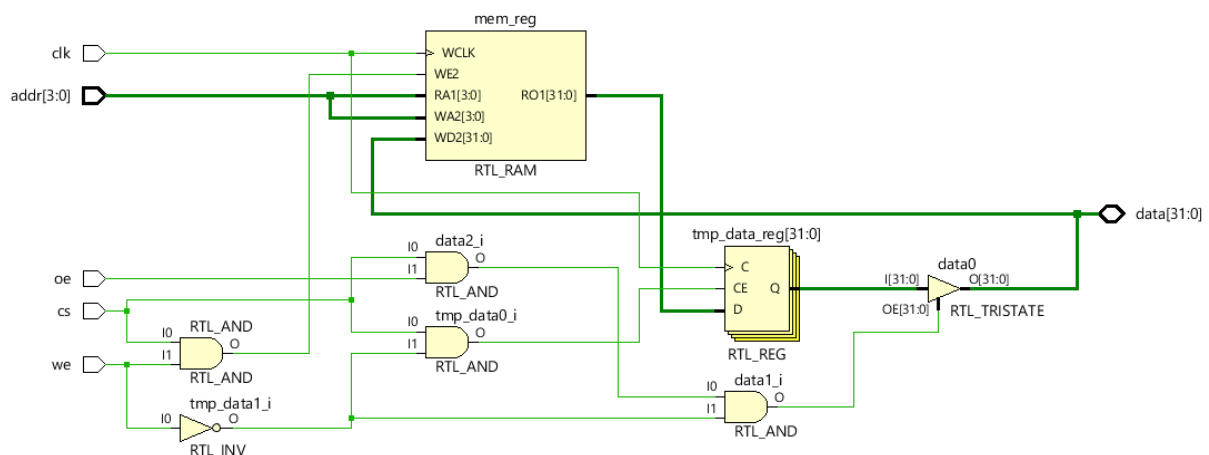
Appendix :

- **Codes -**

```verilog
module test_bench2;


        reg [7:0] i;      //Input variable

        wire [7:0] o;   //Output variable

        reg [1:0] select;

        reg read, clear;


        Circuit C1(i[0], i[1], i[2], i[3], i[4], i[5], i[6], i[7], o[0], o[1], o[2], o[3], o[4], o[5],
o[6], o[7], select[0], select[1], read, clear);


        initial

                begin



                        clear = 1'b0;   //active LOW reset for all DFFs to clear memory

                        #10 clear = 1'b1;   //disabling reset so that DFFs can store data



                        #10 read = 1'b0;   //Setting circuit to WRITE mode by giving
active LOW enable to decoder

                                select = 2'b00;  //Telling decoder to provide HIGH signals
to DFF ICs 0 and 4 for storage

                                i = 8'b00010001;   //Writing this value into DFF IC's


                        #200 read = 1'b1;
```

```verilog
        select = 2'b00;   //Telling multiplexers to read from DFF IC's 0 and 4




    #200 read = 1'b0;
                        #10 select = 2'b01;   //Telling decoder to provide HIGH signals to
DFF ICs 1 and 5 for storage

                            i = 8'b00100010;     //Writing this value into DFF IC's




                        #200 read = 1'b1;

    select = 2'b01;   //Telling multiplexers to read from DFF IC's 1 and 5






                        #200 read = 1'b0;

                        #10 select = 2'b10;   //Telling decoder to provide HIGH signals to
DFF ICs 2 and 6 for storage

                            i = 8'b01000100;     //Writing this value into DFF IC's

                        #200 read = 1'b1;     //Setting circuit to READ mode by giving
active LOW enable to multiplexers

                         select = 2'b10;   //Telling multiplexers to read from DFF IC's 2
and 6




                        #200 read = 1'b0;
```

```verilog
            #10 select = 2'b11;   //Telling decoder to provide HIGH signals to
DFF ICs 3 and 7 for storage

                i = 8'b10001000;      //Writing this value into DFF IC's


            #200 read = 1'b1;    //Setting circuit to READ mode by giving
active LOW enable to multiplexers

             select = 2'b11;   //Telling multiplexers to read from DFF IC's 3
and 7




        end
endmodule
```