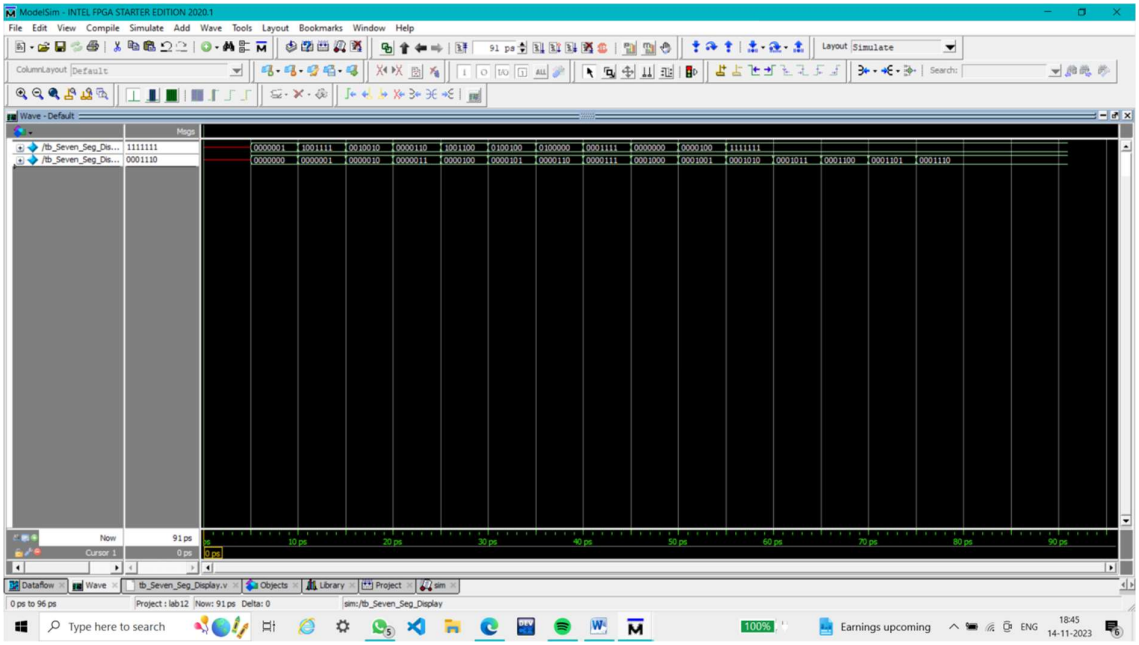
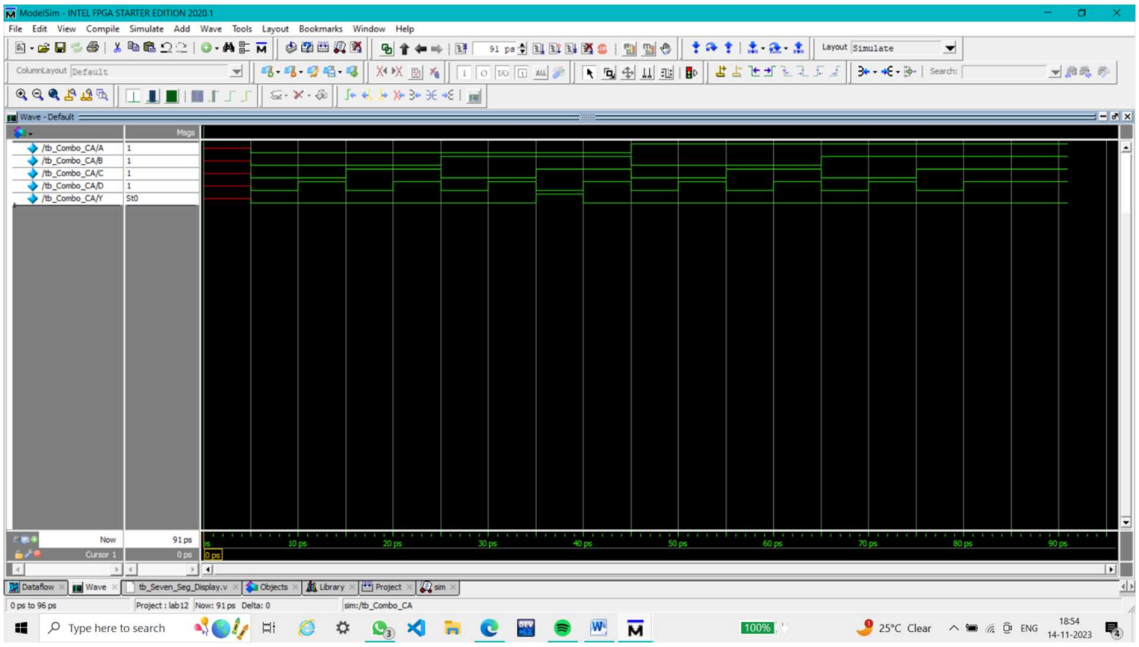
(LAB 16)

Task 1

1.

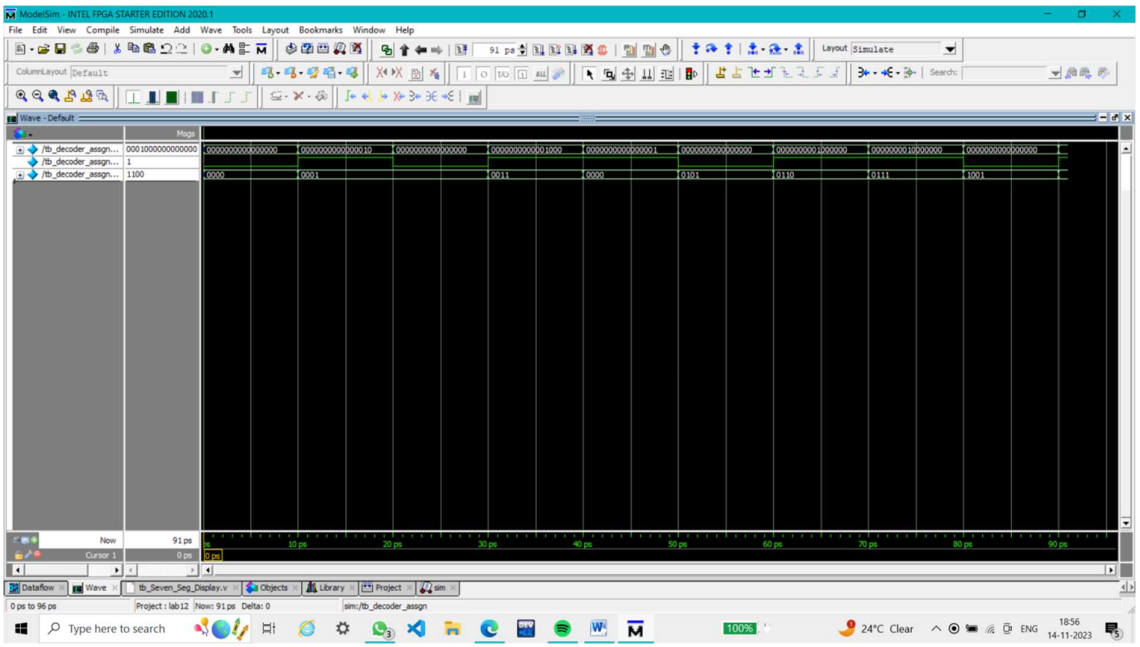


2.

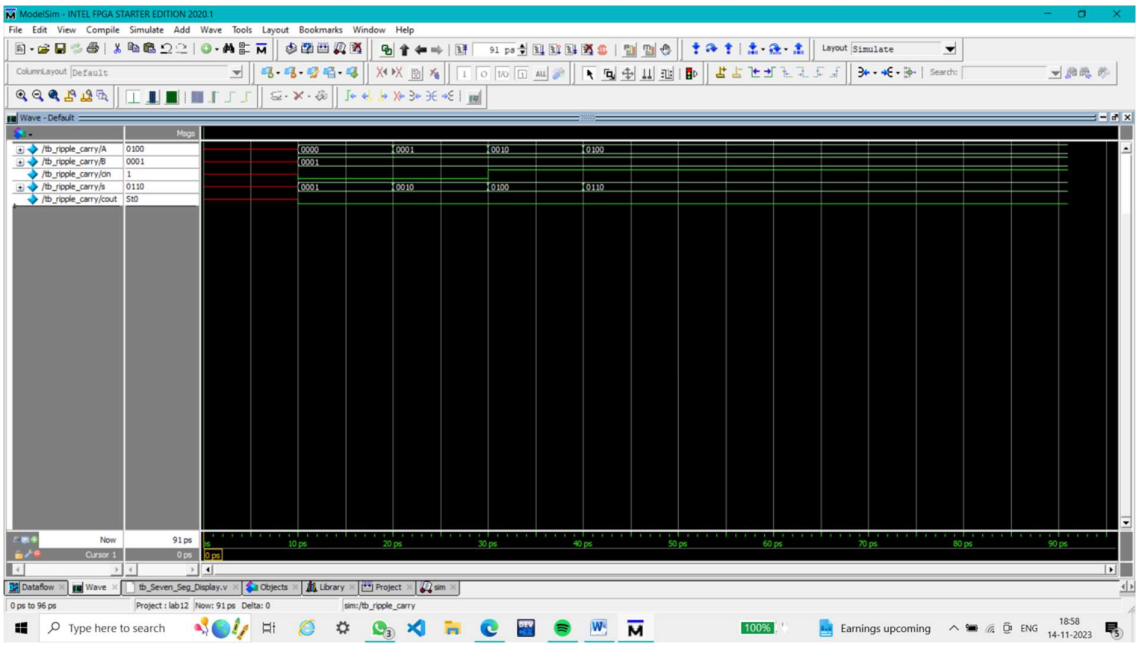


(LAB 16)

3.

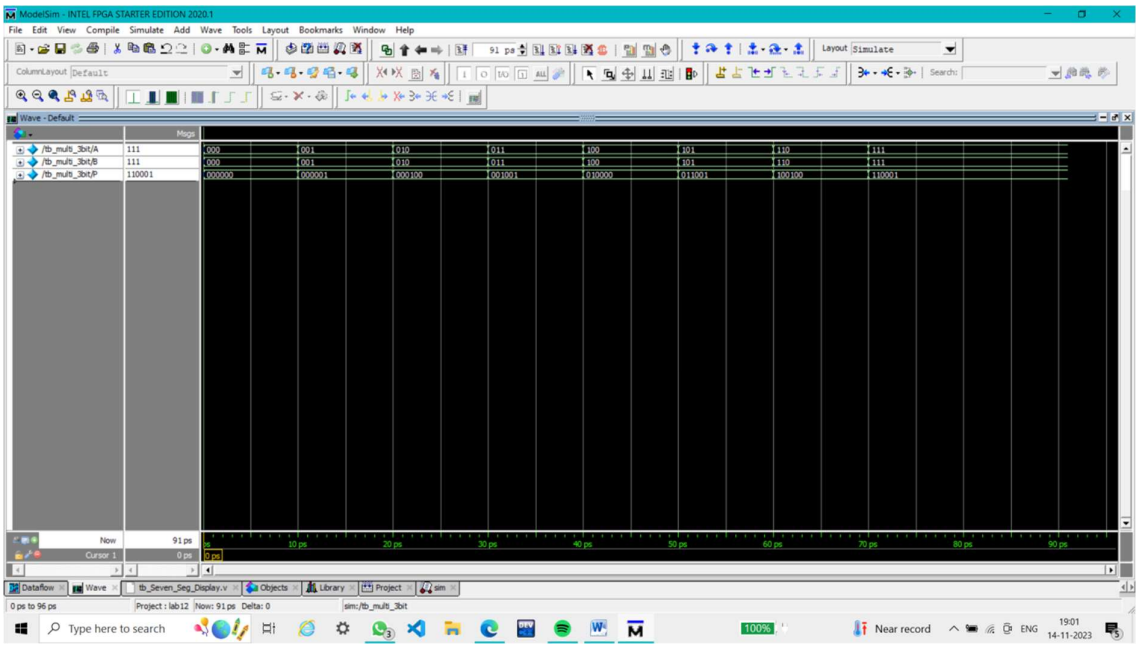

ModelSim - INTEL FPGA STARTER EDITION 2020.1
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Wave - Default
/tb_decoder_assgn...
/tb_decoder_assgn...
/tb_decoder_assgn...
0000
0001
0011
0000
0101
0110
0111
1001
Now
91 ps
Cursor 1
0 ps
Type here to search
24°C Clear
ENG
18:56
14-11-2023

4.

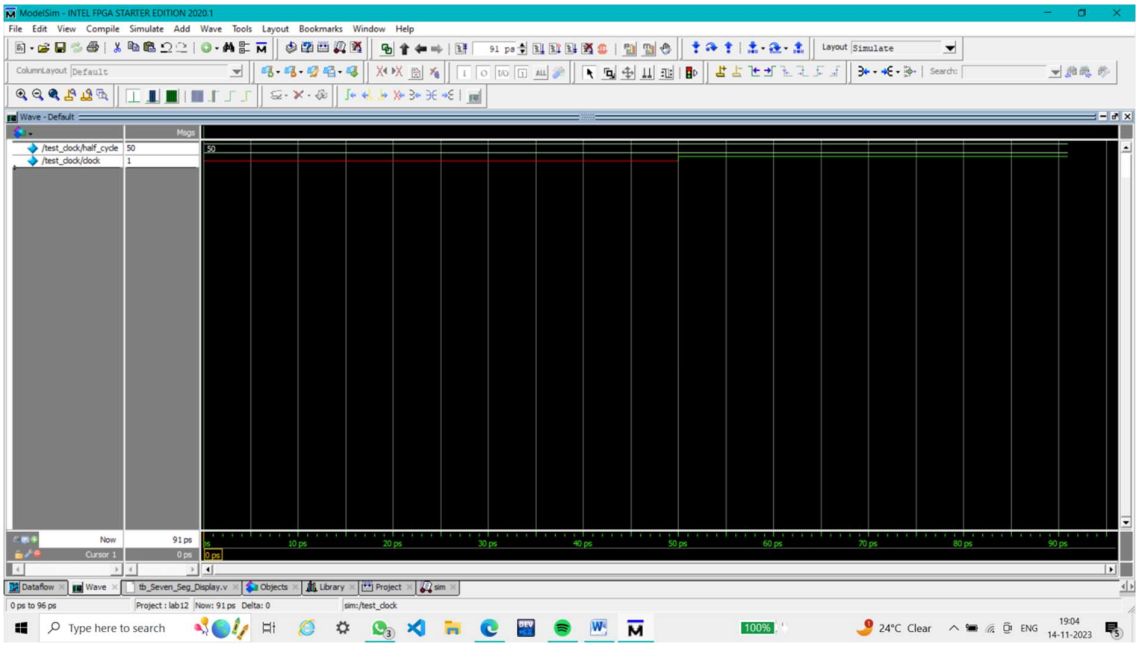

ModelSim - INTEL FPGA STARTER EDITION 2020.1
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Wave - Default
/tb_ripple_carry/A
/tb_ripple_carry/B
/tb_ripple_carry/cin
/tb_ripple_carry/s
/tb_ripple_carry/cout
0100
0001
1
0110
St0
0000
0001
0010
0100
0110
Now
91 ps
Cursor 1
0 ps
Type here to search
100%
Earnings upcoming
ENG
18:58
14-11-2023

(LAB 16)

5.

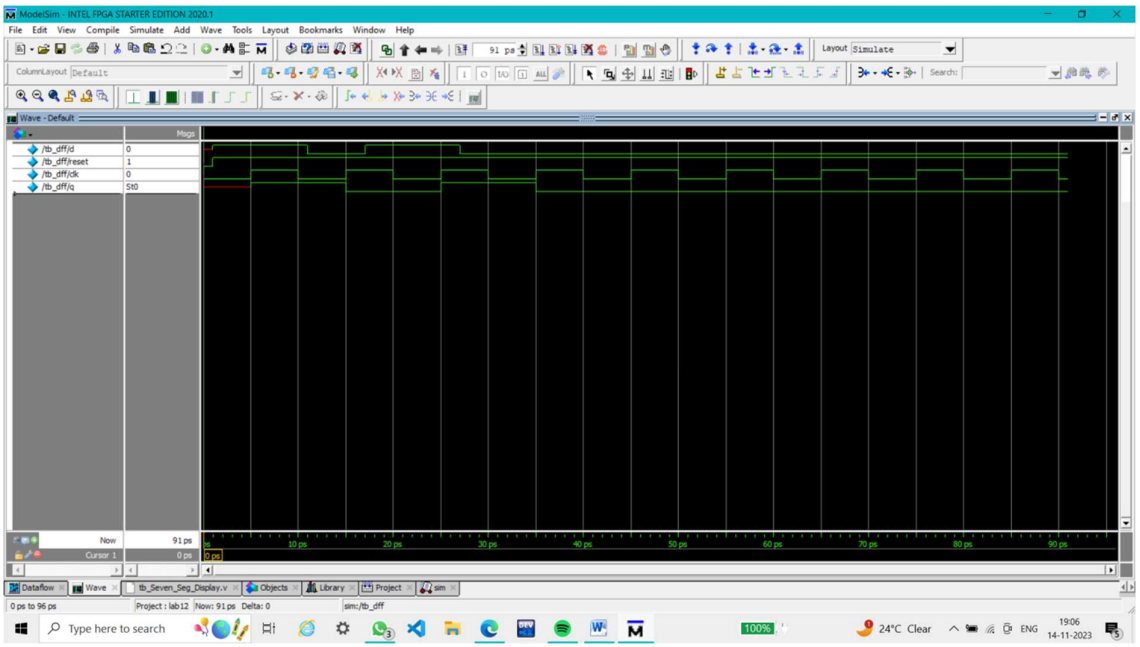

ModelSim - INTEL FPGA STARTER EDITION 2020.1
/tb_multi_3bit/A
/tb_multi_3bit/B
/tb_multi_3bit/P

6.

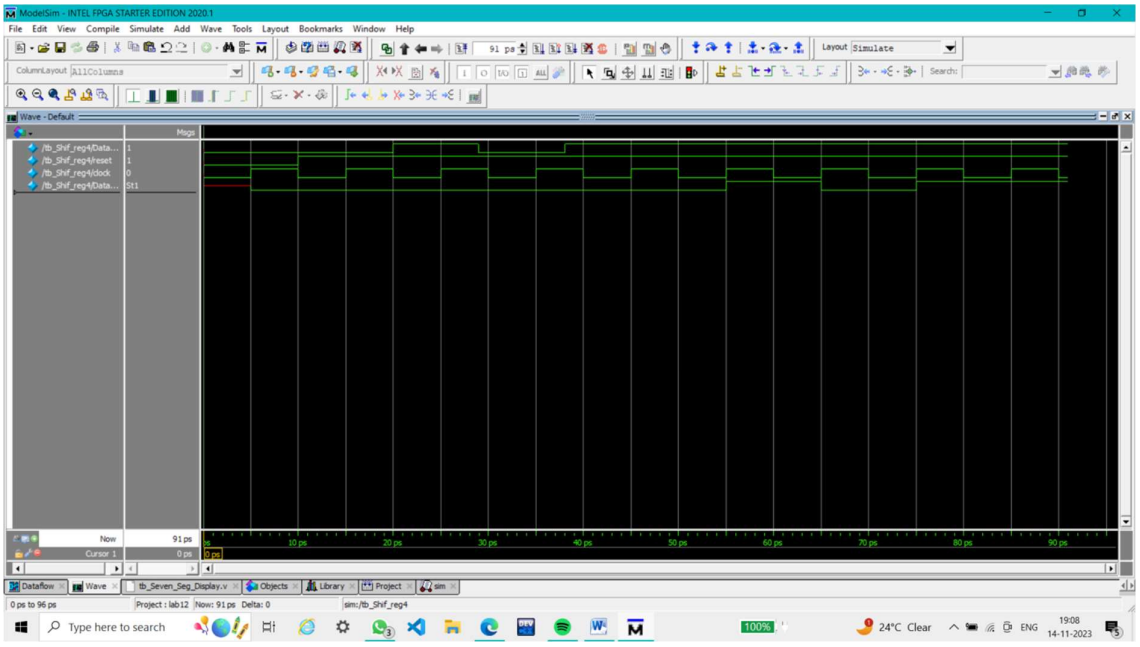

ModelSim - INTEL FPGA STARTER EDITION 2020.1
/test_clock/half_cycle
/test_clock/clock

(LAB 16)

7.

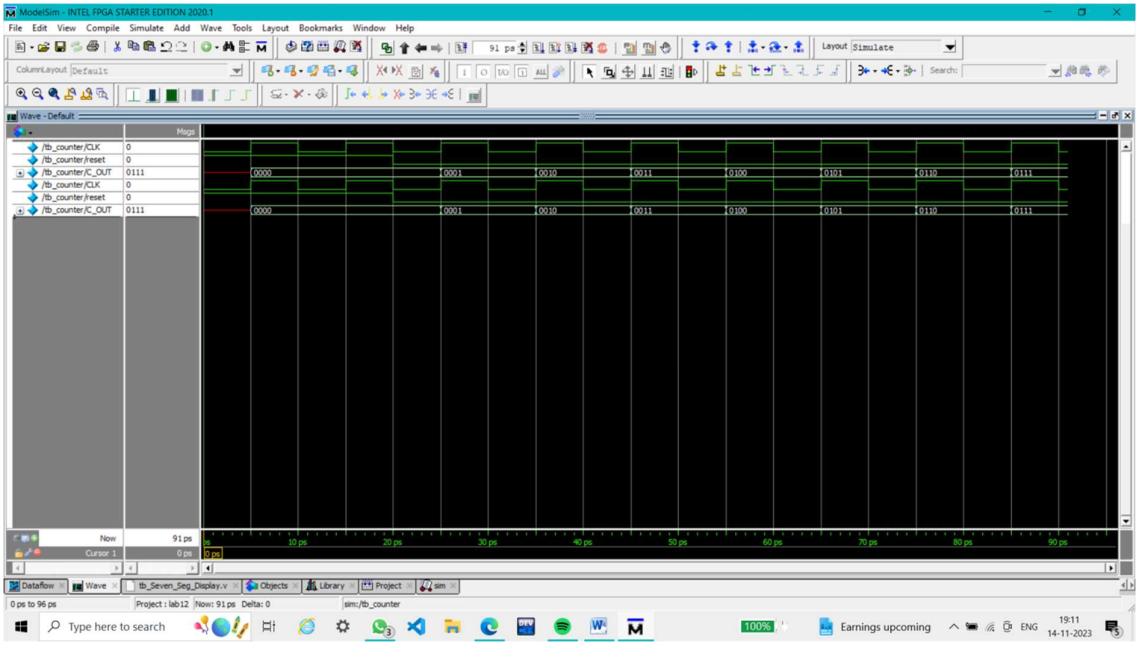

ModelSim - INTEL FPGA STARTER EDITION 2020.1
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Wave - Default
/tb_dff/d
/tb_dff/reset
/tb_dff/clk
/tb_dff/q
Type here to search
24°C Clear
ENG
19:06
14-11-2023

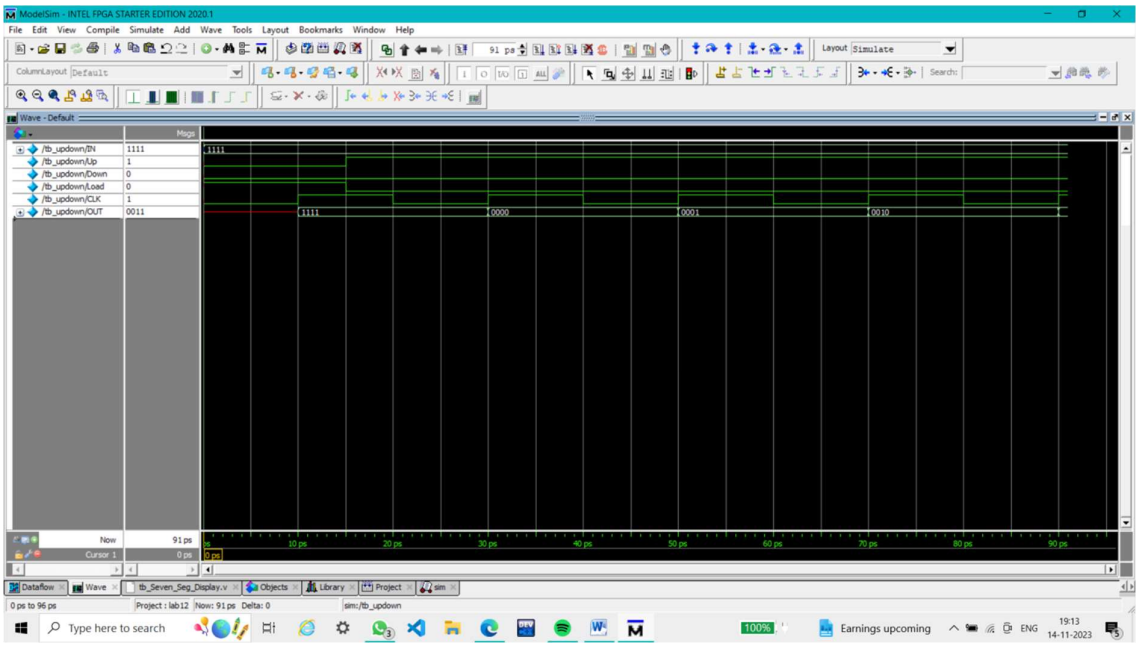
8.


ModelSim - INTEL FPGA STARTER EDITION 2020.1
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Wave - Default
/tb_Shif_reg4/Data...
/tb_Shif_reg4/reset
/tb_Shif_reg4/clock
/tb_Shif_reg4/Data...
Type here to search
24°C Clear
ENG
19:08
14-11-2023

(LAB 16)

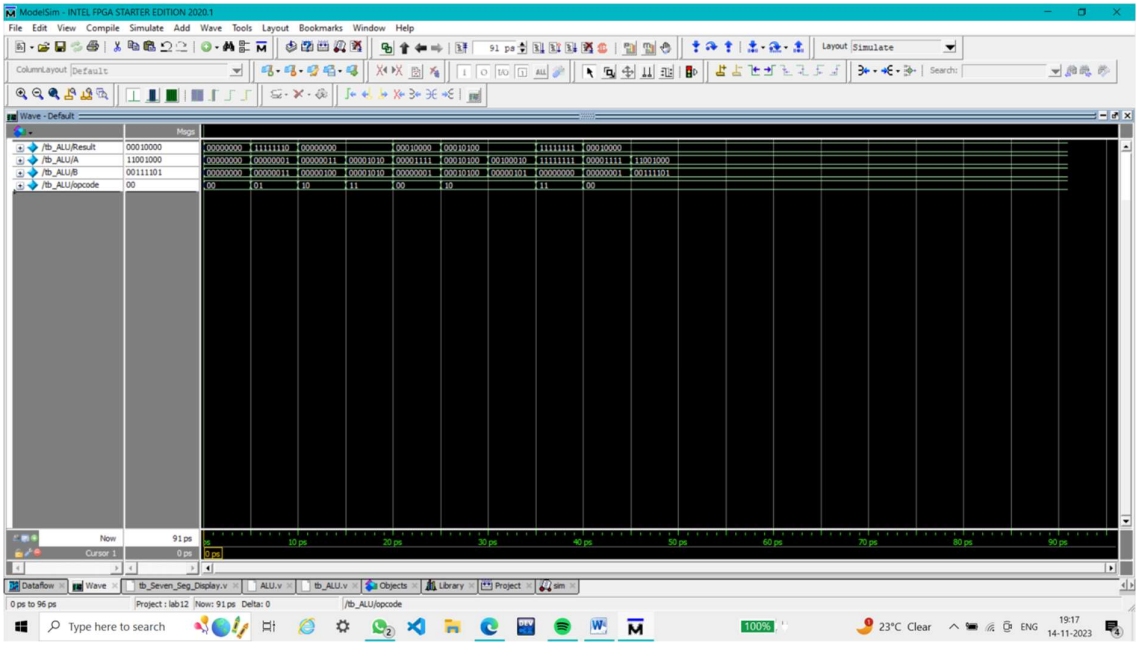
9.


ModelSim - INTEL FPGA STARTER EDITION 2020.1
/tb_counter/CLK
/tb_counter/reset
/tb_counter/C_OUT
0000
0001
0010
0011
0100
0101
0110
0111

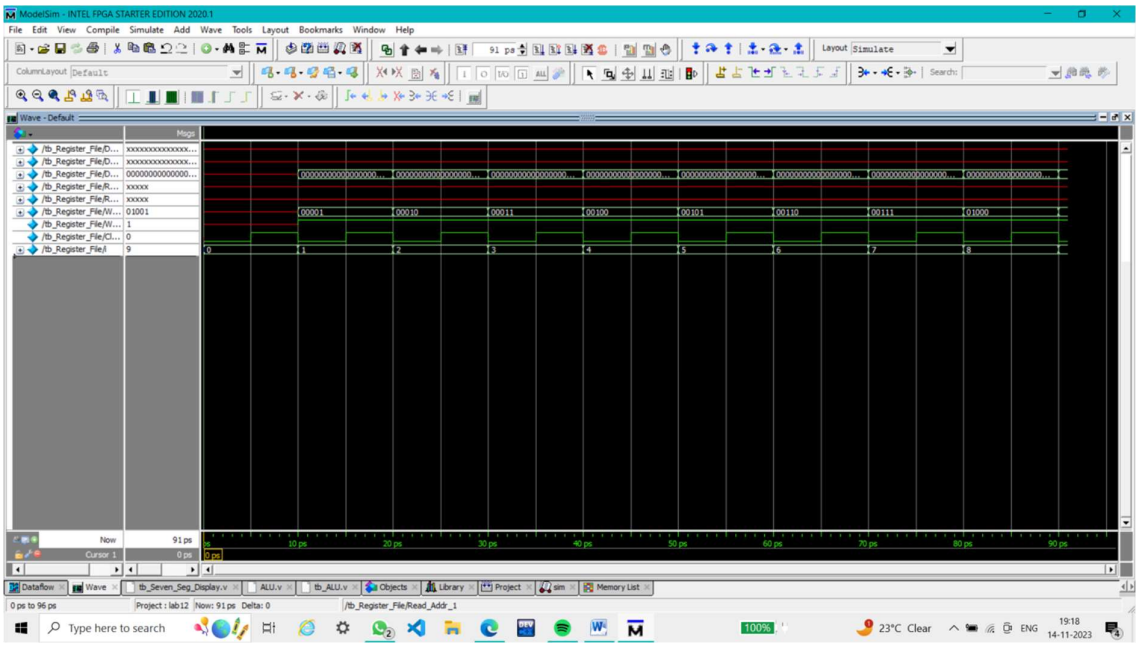

ModelSim - INTEL FPGA STARTER EDITION 2020.1
/tb_updown/IN
/tb_updown/Up
/tb_updown/Down
/tb_updown/Load
/tb_updown/CLK
/tb_updown/OUT
1111
0000
0001
0010

(LAB 16)

10. ALU

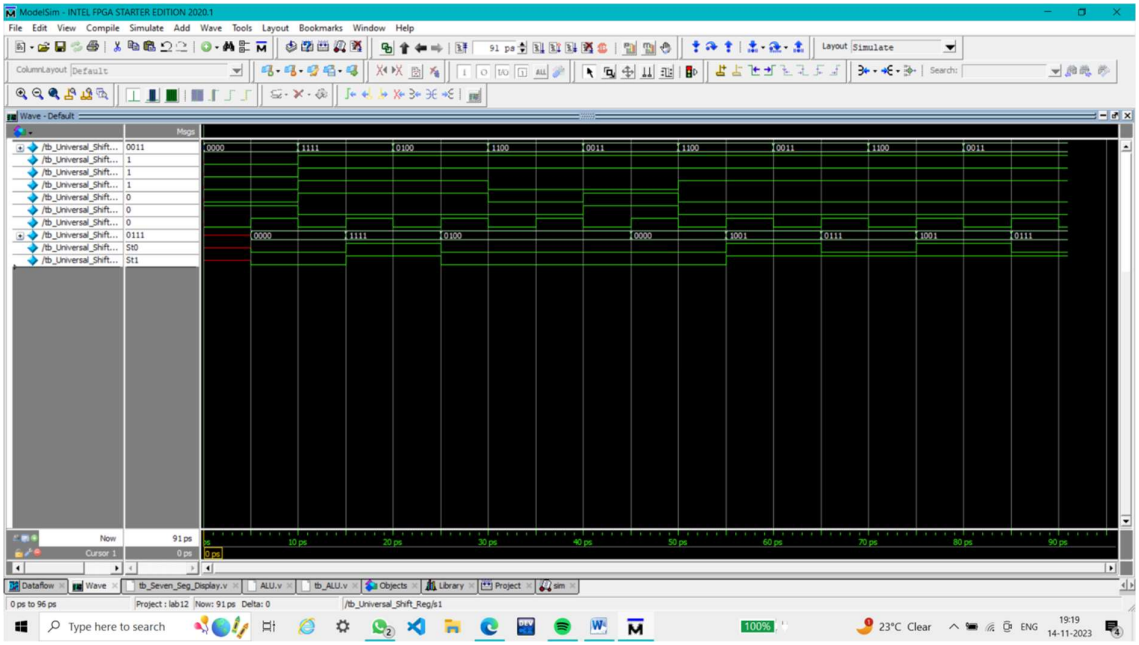


REGISTER FILE



UNIVERAL SHIFT REGISTER

(LAB 16)


ModelSim - INTEL FPGA STARTER EDITION 2020.1
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
Wave - Default
/tb_Universal_Shift...
0011
0000
1111
0100
1100
0011
1100
0011
1100
0011
0000
1111
0100
0000
1001
0111
1001
0111
St0
St1
Now
91 ps
Cursor 1
0 ps
10 ps
20 ps
30 ps
40 ps
50 ps
60 ps
70 ps
80 ps
90 ps
Dataflow
Wave
tb_Seven_Seg_Display.v
ALU.v
tb_ALU.v
Objects
Library
Project
sim
0 ps to 96 ps
Project : lab12
Now: 91 ps
Delta: 0
/tb_Universal_Shft_Reg/s1
Type here to search
100%
23°C Clear
ENG
19:19
14-11-2023

(LAB 16)

TASK 2

BIT WIDTH	***LOGIC ELEMENT***
3 BITS	14 BIT LOGIC
4 BITS	18 BIT LOGIC
5 BITS	22 BIT LOGIC
6 BITS	26 BIT LOGIC
8 BITS	34 BIT LOGIC
9 BITS	38 BIT LOGIC
10 BITS	42 BIT LOGIC
11 BITS	46 BIT LOGIC
12 BITS	50 BIT LOGIC
16 BITS	66 BIT LOGIC

(LAB 16)

CODES FOR 10 DIFFERENT BITS:

1)8 bits

```
module lab16(Result,opcode,A,B);

output[7:0] Result;

input [7:0] A,B;

input [1:0] opcode;

reg   [7:0] Result;
```

(LAB 16)

```
always @ (opcode)
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
     2'd3:    Result <= A^B;
     default:        Result <= 8'd0;
   endcase
endmodule
```

2) 4 bits

```
module lab16(Result,opcode,A,B);
output[3:0] Result;
input [3:0] A,B;
input [1:0] opcode;
reg   [3:0] Result;

always @ (opcode)
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
     2'd3:    Result <= A^B;
     default:        Result <= 3'd0;
   endcase
```

```
endmodule
```

3) 5 bits

```
module lab16(Result,opcode,A,B);

output[4:0] Result;

input [4:0] A,B;

input [1:0] opcode;

reg   [4:0] Result;


always @ (opcode)

      case (opcode)

     2'd0:  Result <= A+B;

     2'd1:  Result <= A-B;

     2'd2:  Result <= A &B;

     2'd3:    Result <= A^B;

     default:        Result <= 4'd0;

   endcase

endmodule
```

4) 6 bits

```
module lab16(Result,opcode,A,B);

output[5:0] Result;

input [5:0] A,B;

input [1:0] opcode;

reg   [5:0] Result;


always @ (opcode)
```

(LAB 16)

```
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
     2'd3:    Result <= A^B;
     default:        Result <= 5'd0;
   endcase
endmodule
```

5)9 bits

```
module lab16(Result,opcode,A,B);
output[8:0] Result;
input [8:0] A,B;
input [1:0] opcode;
reg   [8:0] Result;

always @ (opcode)
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
     2'd3:    Result <= A^B;
     default:        Result <= 9'd0;
   endcase
endmodule
```

6)10 bits

(LAB 16)

```
module lab16(Result,opcode,A,B);
output[9:0] Result;
input [9:0] A,B;
input [1:0] opcode;
reg   [9:0] Result;

always @ (opcode)
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
     2'd3:    Result <= A^B;
     default:        Result <= 10'd0;
   endcase
endmodule
```

7) 11 bits

```
module lab16(Result,opcode,A,B);
output[10:0] Result;
input [10:0] A,B;
input [1:0] opcode;
reg   [10:0] Result;

always @ (opcode)
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
```

(LAB 16)

```
    2'd3:    Result <= A^B;
    default:        Result <= 11'd0;
  endcase
endmodule
```

8) 12 bits

```
module lab16(Result,opcode,A,B);
output[11:0] Result;
input [11:0] A,B;
input [1:0] opcode;
reg   [11:0] Result;

always @ (opcode)
     case (opcode)
    2'd0:  Result <= A+B;
    2'd1:  Result <= A-B;
    2'd2:  Result <= A &B;
    2'd3:    Result <= A^B;
    default:        Result <= 12'd0;
  endcase
endmodule
```

9)16 bits

```
module lab16(Result,opcode,A,B);
output[15:0] Result;
input [15:0] A,B;
input [1:0] opcode;
reg   [15:0] Result;
```

(LAB 16)

```
always @ (opcode)
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
     2'd3:    Result <= A^B;
     default:        Result <= 16'd0;
   endcase
endmodule
```

10)3 bits

```
module lab16(Result,opcode,A,B);
output[2:0] Result;
input [2:0] A,B;
input [1:0] opcode;
reg   [2:0] Result;

always @ (opcode)
      case (opcode)
     2'd0:  Result <= A+B;
     2'd1:  Result <= A-B;
     2'd2:  Result <= A &B;
     2'd3:    Result <= A^B;
     default:        Result <= 3'd0;
   endcase
endmodule
```

(LAB 16)