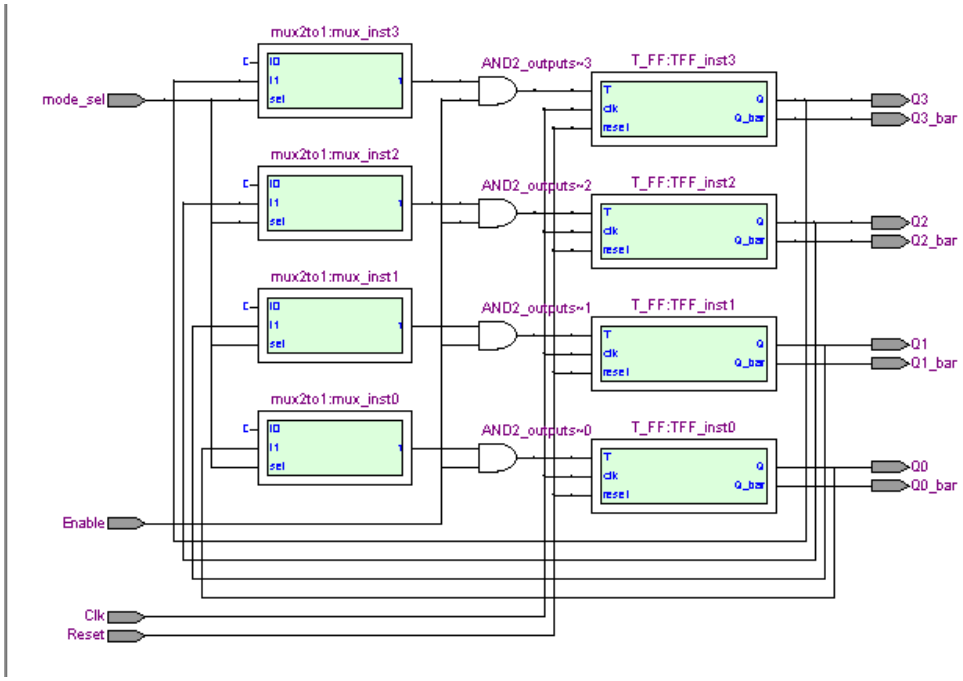
```
// D Flip-Flop module
module D_FF (
    input  wire  clk,
    input  wire  reset,
    input  wire  D,
    output reg  Q,
    output reg  Q_bar
);

  always @(posedge clk or posedge reset) begin
    if (reset) begin
      Q <= 1'b0;
      Q_bar <= 1'b1;
    end else begin
      Q <= D;
      Q_bar <= ~D;
    end
  end

endmodule

// T Flip-Flop module
module T_FF (
    input  wire  clk,
    input  wire  reset,
    input  wire  T,
    output reg  Q,
    output reg  Q_bar
);

  always @(posedge clk or posedge reset) begin
    if (reset) begin
      Q <= 1'b0;
      Q_bar <= 1'b1;
    end else begin
      case (T)
        1'b0: begin
          Q <= Q;
          Q_bar <= Q_bar;
        end
        1'b1: begin
          Q <= ~Q;
          Q_bar <= ~Q_bar;
        end
```

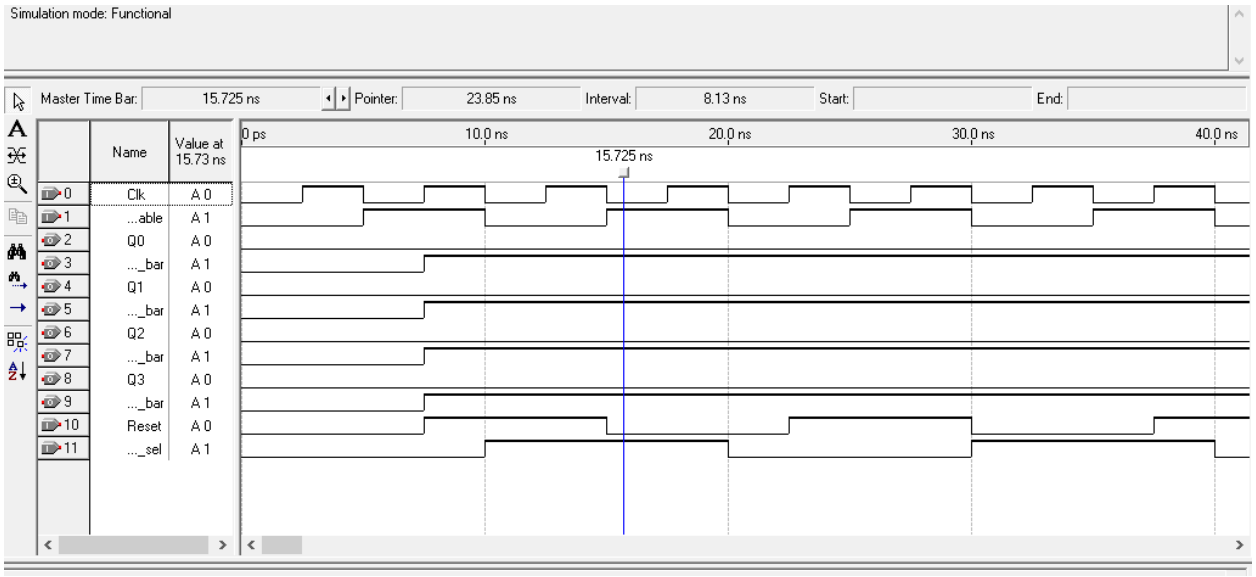
```
      endcase
    end
  end

endmodule

// 2-to-1 MUX module
module mux2to1 (
    input  wire  sel,
    input  wire  i0,
    input  wire  i1,
    output reg  f
);

  always @(*) begin
    case (sel)
      1'b0: f = i0;
      1'b1: f = i1;
    endcase
  end

endmodule
module Lab20301269 (
// Top module for the circuit in Figure 1
    input  wire  Enable,
    input  wire  Clk,
    input  wire  Reset,
    input  wire  mode_sel,
    output wire  Q0,
    output wire  Q0_bar,
    output wire  Q1,
    output wire  Q1_bar,
    output wire  Q2,
    output wire  Q2_bar,
    output wire  Q3,
    output wire  Q3_bar
);

  wire [3:0] sel_signals;
  wire [3:0] i0_signals;
  wire [3:0] i1_signals;
  wire [3:0] f_signals;
  wire [3:0] AND2_outputs;
```

```
  // Generate select signals for the multiplexers
  assign sel_signals[0] = mode_sel;
  assign sel_signals[1] = mode_sel;
  assign sel_signals[2] = mode_sel;
  assign sel_signals[3] = mode_sel;

  // Generate input signals for the multiplexers
  assign i0_signals[0] = 1'b0;
  assign i0_signals[1] = 1'b0;
  assign i0_signals[2] = 1'b0;
  assign i0_signals[3] = 1'b0;

  assign i1_signals[0] = Q0;
  assign i1_signals[1] = Q1;
  assign i1_signals[2] = Q2;
  assign i1_signals[3] = Q3;

  // Instantiate multiplexers
  mux2to1 mux_inst0 (.sel(sel_signals[0]),.i0(i0_signals[0]),.i1(i1_signals[0]),.f(f_signals[0]));
  mux2to1 mux_inst1 (.sel(sel_signals[1]),.i0(i0_signals[1]),.i1(i1_signals[1]),.f(f_signals[1]));
  mux2to1 mux_inst2 (.sel(sel_signals[2]),.i0(i0_signals[2]),.i1(i1_signals[2]),.f(f_signals[2]));
  mux2to1 mux_inst3 (.sel(sel_signals[3]),.i0(i0_signals[3]),.i1(i1_signals[3]),.f(f_signals[3]));

  // Instantiate AND2 gates
  assign AND2_outputs[0] = Enable & f_signals[0];
  assign AND2_outputs[1] = Enable & f_signals[1];
  assign AND2_outputs[2] = Enable & f_signals[2];
  assign AND2_outputs[3] = Enable & f_signals[3];
// Instantiate T Flip-Flops
  T_FF TFF_inst0 (.clk(Clk),.reset(Reset),.T(AND2_outputs[0]),.Q(Q0),.Q_bar(Q0_bar));
  T_FF TFF_inst1 (.clk(Clk),.reset(Reset),.T(AND2_outputs[1]),.Q(Q1),.Q_bar(Q1_bar));
  T_FF TFF_inst2 (.clk(Clk),.reset(Reset),.T(AND2_outputs[2]),.Q(Q2),.Q_bar(Q2_bar));
  T_FF TFF_inst3 (.clk(Clk),.reset(Reset),.T(AND2_outputs[3]),.Q(Q3),.Q_bar(Q3_bar));

endmodule
```

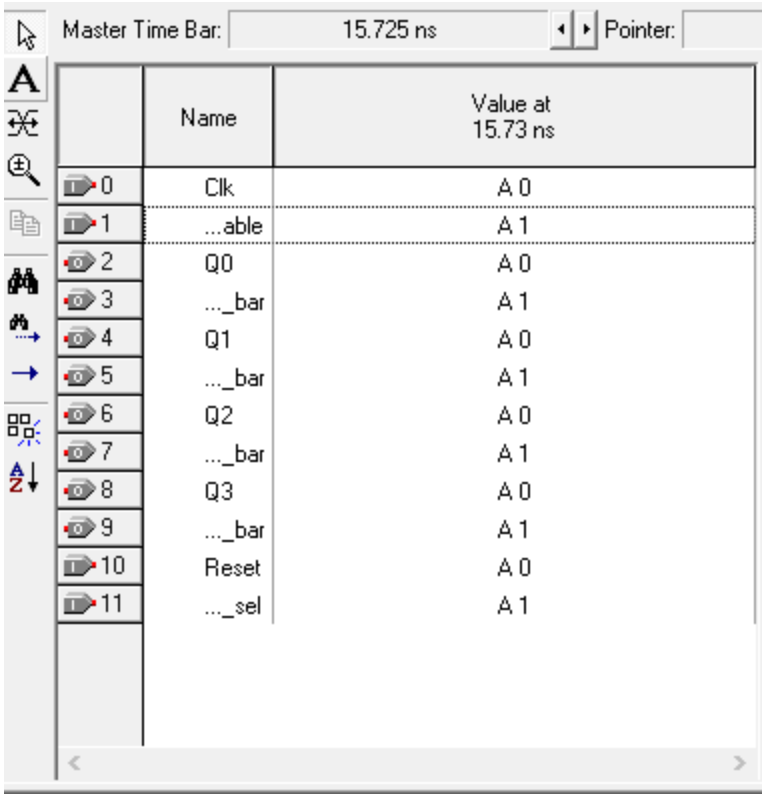
##########
RTL:


mux2to1:mux_inst3
mux2to1:mux_inst2
mux2to1:mux_inst1
mux2to1:mux_inst0
I0
I1
sel
AND2_outputs~3
AND2_outputs~2
AND2_outputs~1
AND2_outputs~0
T_FF:TFF_inst3
T_FF:TFF_inst2
T_FF:TFF_inst1
T_FF:TFF_inst0
T
clk
reset
Q
Q_bar
mode_sel
Enable
Clk
Reset
Q3
Q3_bar
Q2
Q2_bar
Q1
Q1_bar
Q0
Q0_bar

##########
SIMULATION:



TRUTH TABLE:



Clock	Reset	Select	Q0	Q1	Q2	Q3
0	0	0	0	0	0	0
1	0	0	1	0	0	0

0	0	0	0	1	0	0
1	0	0	1	1	0	0
0	0	0	0	0	1	0
1	0	0	1	0	1	0
0	0	0	0	1	1	0
1	0	0	1	1	1	0
0	0	0	0	0	0	1
1	0	0	1	0	0	1

0	1	0	0	0	0	0
1	1	0	0	0	0	0

##################

The Lab20301269 module has the following input-output pins:

- Enable: Enables or disables the T Flip-Flops
- Clk: Clock signal for the T Flip-Flops
- Reset: Reset signal for the T Flip-Flops
- mode_sel: Selects the input to the multiplexers
- Q0, Q0_bar, Q1, Q1_bar, Q2, Q2_bar, Q3, Q3_bar: Outputs of the T Flip-Flops

The overall functionality of the Lab20301269 module is as follows:

- The multiplexers select the input to the T Flip-Flops based on the mode_sel signal.
- The T Flip-Flops toggle their outputs based on the input from the multiplexers and the clock signal.
- The Enable signal enables or disables the T Flip-Flops.
- The Reset signal resets the T Flip-Flops to their initial state.
-