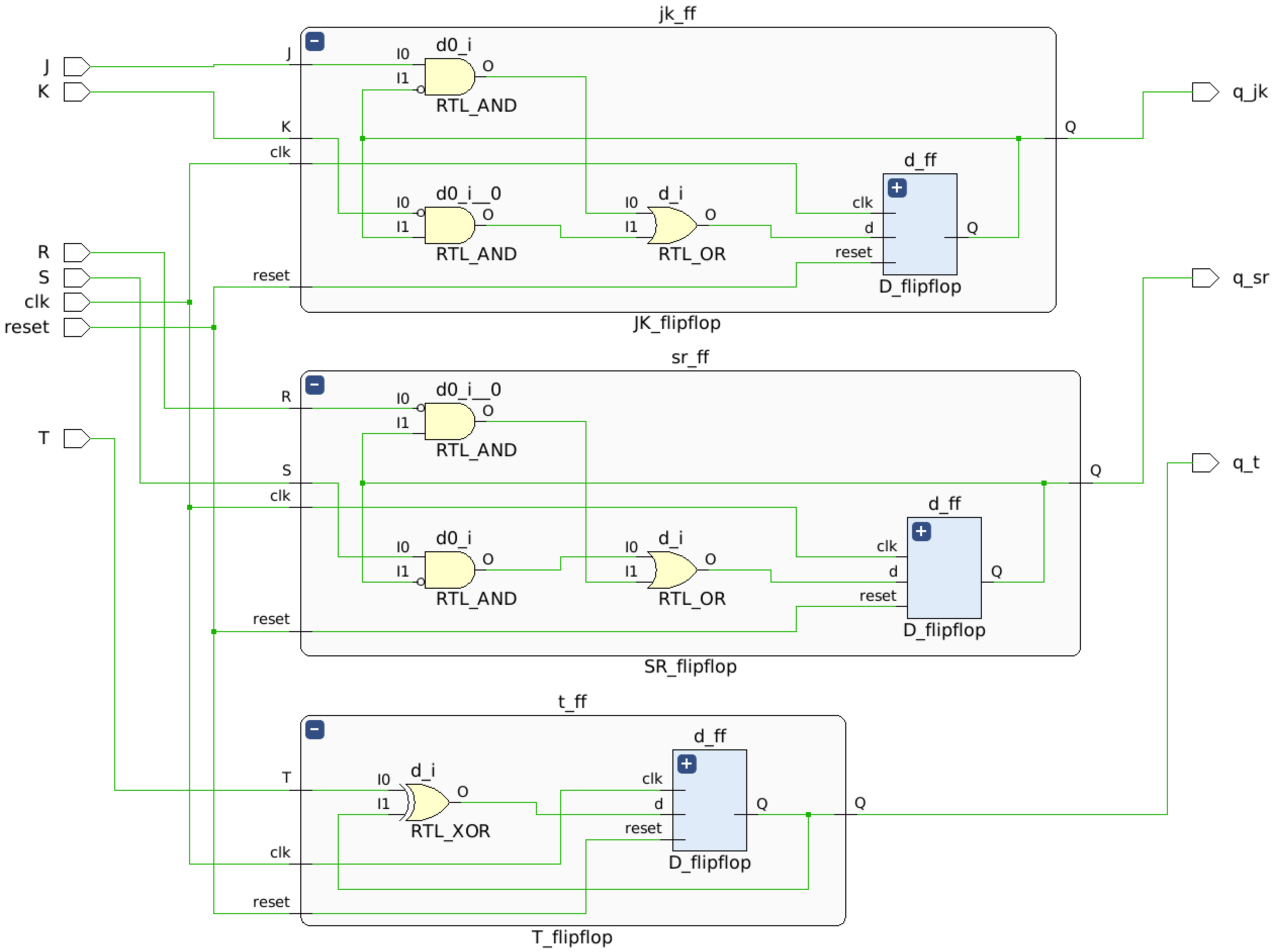
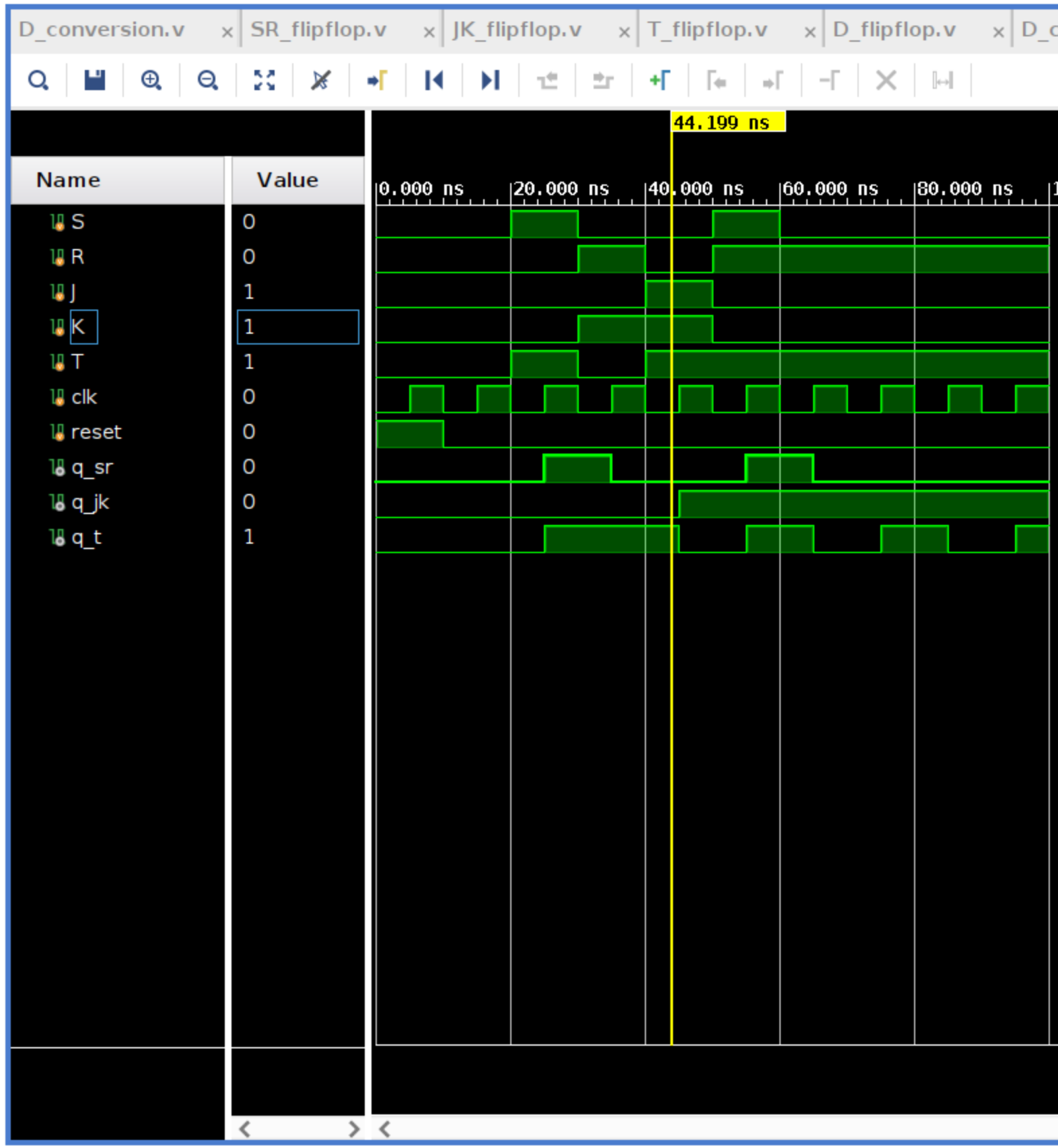
14 Cells 10 I/O Ports 35 Nets



D_conversion.v | SR_flipflop.v | JK_flipflop.v | T_flipflop.v | D_flipflop.v | D_c



Tcl Console | Messages | Log

D_conversion.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_053/project_2/project_2.srcs/sources_1/new/D_conversion.v

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Engineer: Anjan Prasad
4  // Create Date: 11/13/2024 05:44:45 AM
5  // Module Name: D_conversion
6  //////////////////////////////////////////////////////////////////////////////////
7
8  module D_conversion (
9      input S, R, J, K, T, clk, reset,
10     output q_sr, q_jk, q_t
11 );
12     // Instantiate SR, JK, and T Flip-Flops
13     SR_flipflop sr_ff (.S(S), .R(R), .clk(clk), .reset(reset), .Q(q_sr));
14     JK_flipflop jk_ff (.J(J), .K(K), .clk(clk), .reset(reset), .Q(q_jk));
15     T_flipflop t_ff (.T(T), .clk(clk), .reset(reset), .Q(q_t));
16 endmodule
17
```

D_conversion_tb.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_053/project_2/project_2.srcs/sim_1/new/D_conversion_tb.v

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Engineer: Anjan Prasad
4  // Create Date: 11/13/2024 05:52:13 AM
5  // Module Name: D_conversion_tb
6  //////////////////////////////////////////////////////////////////////////////////
7
8  module D_conversion_tb;
9      reg S, R, J, K, T, clk, reset;
10     wire q_sr, q_jk, q_t;
11
12     D_conversion DUT (.S(S),.R(R),.J(J),.K(K),.T(T),.clk(clk),.reset(reset),.q_sr(q_sr),.q_jk(q_jk),.q_t(q_t));
13
14     initial begin
15         clk = 0;
16         forever #5 clk = ~clk;
17     end
18     initial begin
19
20         S = 0; R = 0; J = 0; K = 0; T = 0; reset = 1;
21         #10 reset = 0;
22         // Test SR Flip-Flop
23         #10 S = 1; R = 0; // Set
24         #10 S = 0; R = 1; // Reset
25         #10 S = 0; R = 0; // Hold
26         #10 S = 1; R = 1; // Invalid (both set and reset)
27         #10 S = 0; R = 1;
28     end
29     initial begin
30         // Test JK Flip-Flop
31         #20 S = 0; R = 0; // Disable SR inputs
32         J = 1; K = 0;     // Set
33         #10 J = 0; K = 1; // Reset
34         #10 J = 1; K = 1; // Toggle
35         #10 J = 0; K = 0; // Hold
36         #50 $stop;
37     end
38     initial begin
39         // Test T Flip-Flop
40         #20 J = 0; K = 0; // Disable JK inputs
41         T = 1;            // Toggle
42         #10 T = 0;        // Hold
43         #10 T = 1;        // Toggle again
44     end
45 endmodule
46
47
```

D_flipflop.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_053/project_2/project_2.srcs/sources_1/new/D_flipflop.v

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Engineer: Anjan Prasad
4  // Create Date: 11/13/2024 05:49:10 AM
5  // Module Name: D_flipflop
6  //////////////////////////////////////////////////////////////////////////////////
7
8  module D_flipflop (
9      input clk, reset, d,
10     output reg Q
11 );
12     always @(posedge clk or posedge reset) begin
13         if (reset)
14             Q <= 1'b0;
15         else
16             Q <= d;
17     end
18 endmodule
19
```

SR_flipflop.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_053/project_2/project_2.srcs/sources_1/new/SR_flipflop.v

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Engineer: Anjan Prasad
4  // Create Date: 11/13/2024 05:46:27 AM
5  // Module Name: SR_flipflop
6  //////////////////////////////////////////////////////////////////////////////////
7
8  module SR_flipflop (
9      input S, R, clk, reset,
10     output Q
11 );
12     wire d;
13     assign d = S & ~Q | ~R & Q; // SR logic converted to D input
14     D_flipflop d_ff (.clk(clk), .reset(reset), .d(d), .Q(Q));
15 endmodule
16
```

JK_flipflop.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_053/project_2/project_2.srcs/sources_1/new/JK_flipflop.v

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Engineer: Anjan Prasad
4  // Create Date: 11/13/2024 05:46:27 AM
5  // Module Name: JK_flipflop
6  //////////////////////////////////////////////////////////////////////////////////
7
8  module JK_flipflop (
9      input J, K, clk, reset,
10     output Q
11 );
12     wire d;
13     assign d = J & ~Q | ~K & Q; // JK logic converted to D input
14     D_flipflop d_ff (.clk(clk), .reset(reset), .d(d), .Q(Q));
15 endmodule
```

T_flipflop.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_053/project_2/project_2.srcs/sources_1/new/T_flipflop.v

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Engineer: Anjan Prasad
4  // Create Date: 11/13/2024 05:46:27 AM
5  // Module Name: T_flipflop
6  //////////////////////////////////////////////////////////////////////////////////
7
8  module T_flipflop (
9      input T, clk, reset,
10     output Q
11 );
12     wire d;
13     assign d = T ^ Q; // T logic converted to D input
14     D_flipflop d_ff (.clk(clk), .reset(reset), .d(d), .Q(Q));
15 endmodule
16
```