dual_port_ram.v × tb_dual_port_ram.v

day_087 > project_1 > project_1.srcs > sources_1 > new > dual_port_ram.v

```
1   `timescale 1ns / 1ps
2   //////////////////////////////////////////////////////////////////////////////////
3   // Engineer: Anjan Prasad
4   // Create Date: 12/17/2024 10:58:43 AM
5   // Module Name: dual_port_ram
6   //////////////////////////////////////////////////////////////////////////////////
7
8   module dual_port_ram #(
9       parameter DATA_WIDTH = 8,
10      parameter ADDR_WIDTH = 4
11  ) (
12      input clk,
13      input we_a,
14      input we_b,
15      input [ADDR_WIDTH-1:0] addr_a,
16      input [ADDR_WIDTH-1:0] addr_b,
17      input [DATA_WIDTH-1:0] data_a,
18      input [DATA_WIDTH-1:0] data_b,
19      output reg [DATA_WIDTH-1:0] q_a,
20      output reg [DATA_WIDTH-1:0] q_b
21  );
22
23      // Memory declaration
24      reg [DATA_WIDTH-1:0] ram [2**ADDR_WIDTH-1:0];
25
26      // Port A operations
27      always @(posedge clk) begin
28          if (we_a)
29              ram[addr_a] <= data_a;
30          else
31              q_a <= ram[addr_a];
32      end
33
34      // Port B operations
35      always @(posedge clk) begin
36          if (we_b)
37              ram[addr_b] <= data_b;
38          else
39              q_b <= ram[addr_b];
40      end
41
42  endmodule
43
```

```
1   `timescale 1ns / 1ps
2   //////////////////////////////////////////////////////////////////////////////////
3   // Engineer: Anjan Prasad
4   // Create Date: 12/13/2024 11:02:21 PM
5   // Module Name: tb_dual_port_ram
6   //////////////////////////////////////////////////////////////////////////////////
7
8   module tb_dual_port_ram;
9       parameter DATA_WIDTH = 8;
10      parameter ADDR_WIDTH = 4;
11
12      reg clk;
13      reg we_a, we_b;
14      reg [ADDR_WIDTH-1:0] addr_a, addr_b;
15      reg [DATA_WIDTH-1:0] data_a, data_b;
16      wire [DATA_WIDTH-1:0] q_a, q_b;
17
18      dual_port_ram #(DATA_WIDTH, ADDR_WIDTH) DUT (
19          .clk(clk),
20          .we_a(we_a),      .we_b(we_b),
21          .addr_a(addr_a), .addr_b(addr_b),
22          .data_a(data_a), .data_b(data_b),
23          .q_a(q_a),        .q_b(q_b)
24      );
25
26      initial clk = 0;
27      always #5 clk = ~clk;
28
29      initial begin
30          we_a = 0; we_b = 0;
31          addr_a = 0; addr_b = 0;
32          data_a = 0; data_b = 0;
33
34          // Write to Port A
35          #10 we_a = 1; addr_a = 4'b0010; data_a = 8'hAA;
36          #10 we_a = 0;
37
38          // Write to Port B
39          #10 we_b = 1; addr_b = 4'b0011; data_b = 8'hBB;
40          #10 we_b = 0;
41
42          // Read from Port A
43          #10 addr_a = 4'b0010;
44
45          // Read from Port B
46          #10 addr_b = 4'b0011;
47
48          // Write to both ports simultaneously
49          #10 we_a = 1; we_b = 1;
50              addr_a = 4'b0001; data_a = 8'hCC;
51              addr_b = 4'b0100; data_b = 8'hDD;
52          #10 we_a = 0; we_b = 0;
53
54          // Read from both ports
55          #10 addr_a = 4'b0001;
56              addr_b = 4'b0100;
57
58          #20 $stop;
59      end
60  endmodule
```

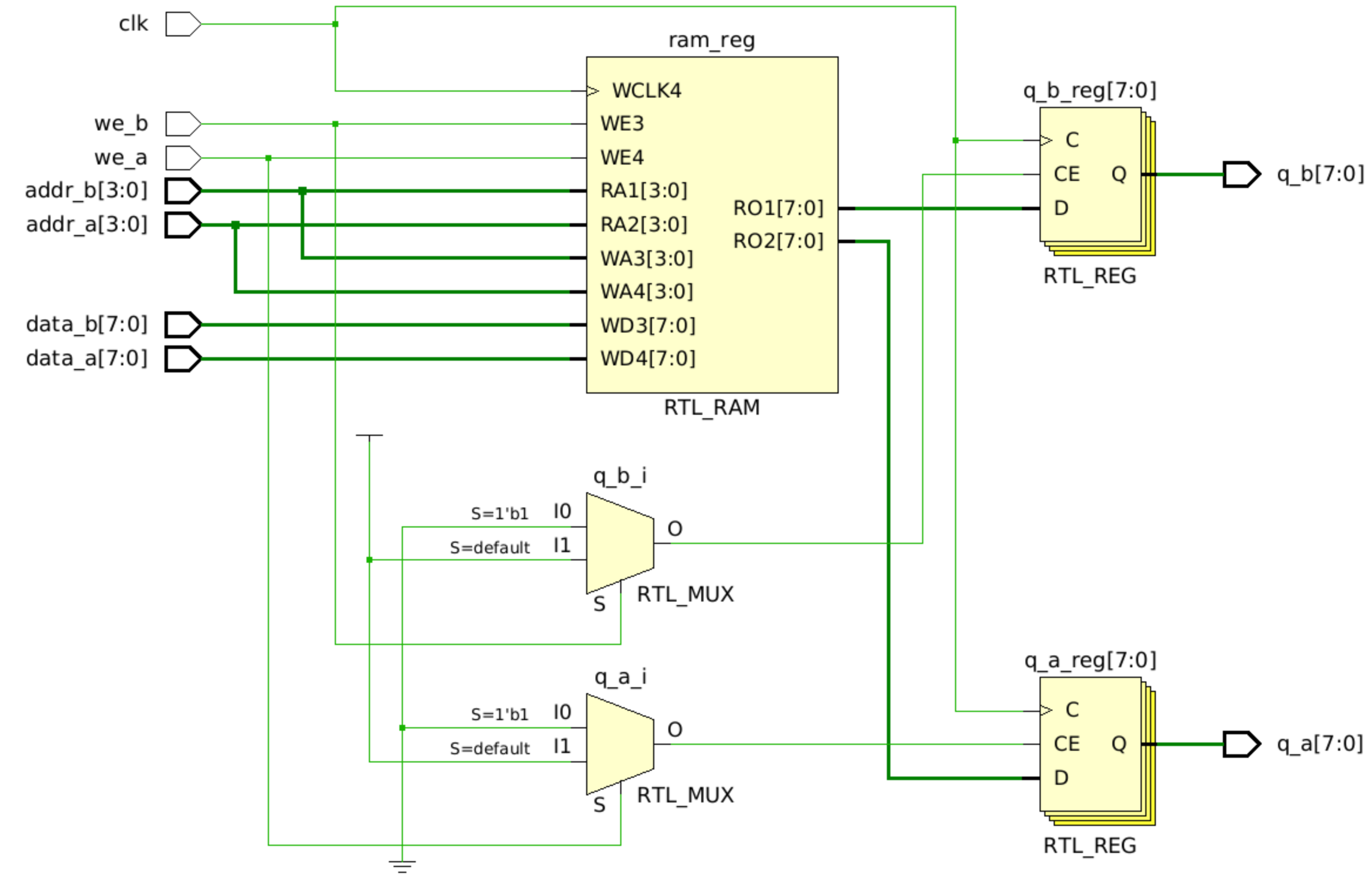
oject_1 - [/home/itzzinfinity/Cozy Drive/100daysofRTL/day_087/project_1/project_1.xpr] - Vivado 2024.1

Help | Quick Access

7a12ticsg325-1L

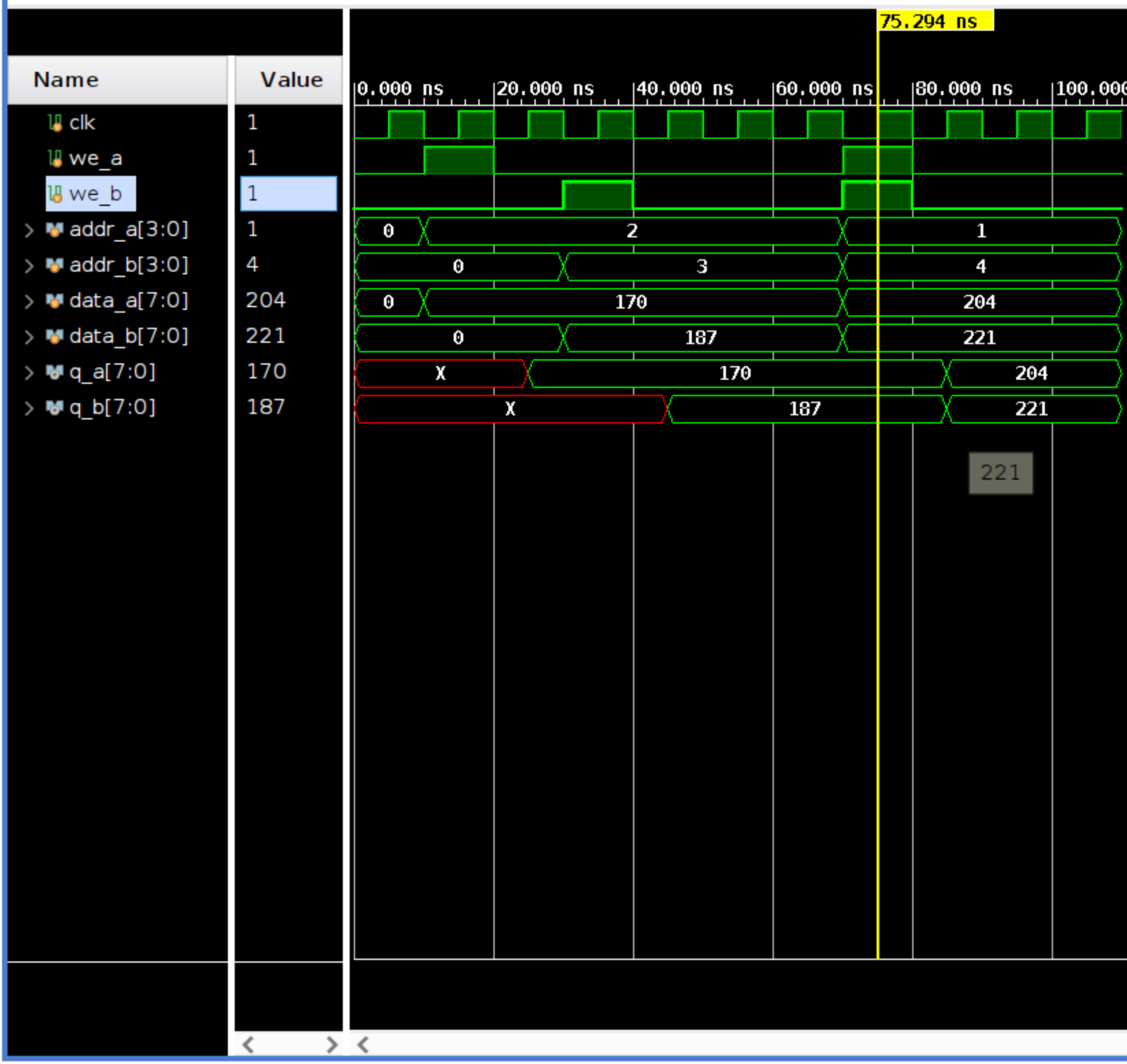
dual_port_ram.v | tb_dual_port_ram.v | Schematic

19 Cells | 43 I/O Ports | 63 Nets



Log | Reports | Design Runs

dual_port_ram.v | tb_dual_port_ram.v | Untitled 4*



Tcl Console | Messages | Log