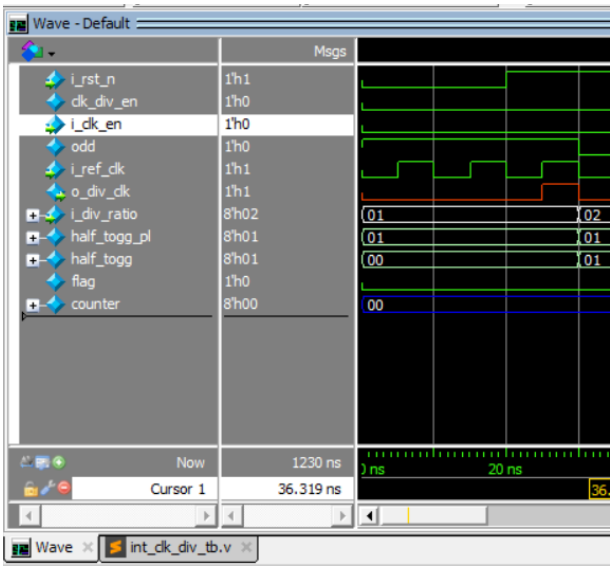
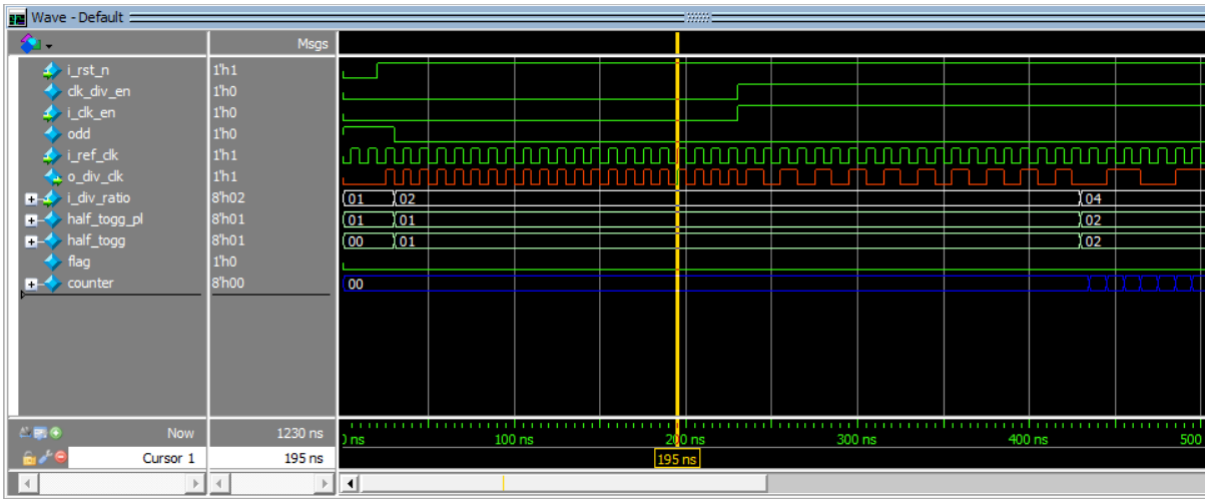
# Assignment Integer Clock Divider

| Name |
| --- |
| Shams Khaled Ezzat |

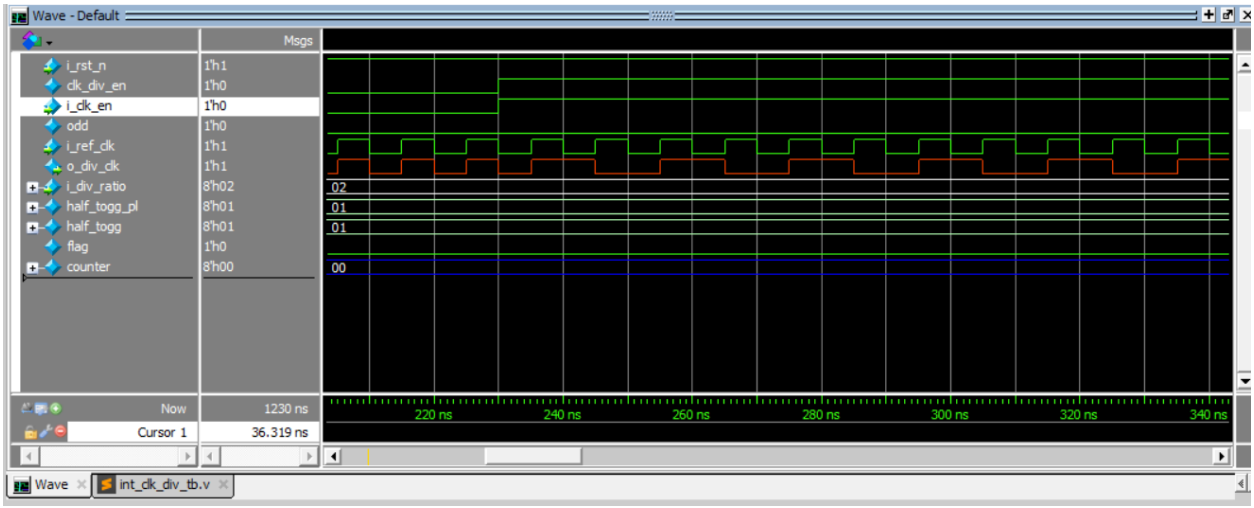
1. Initializing ports



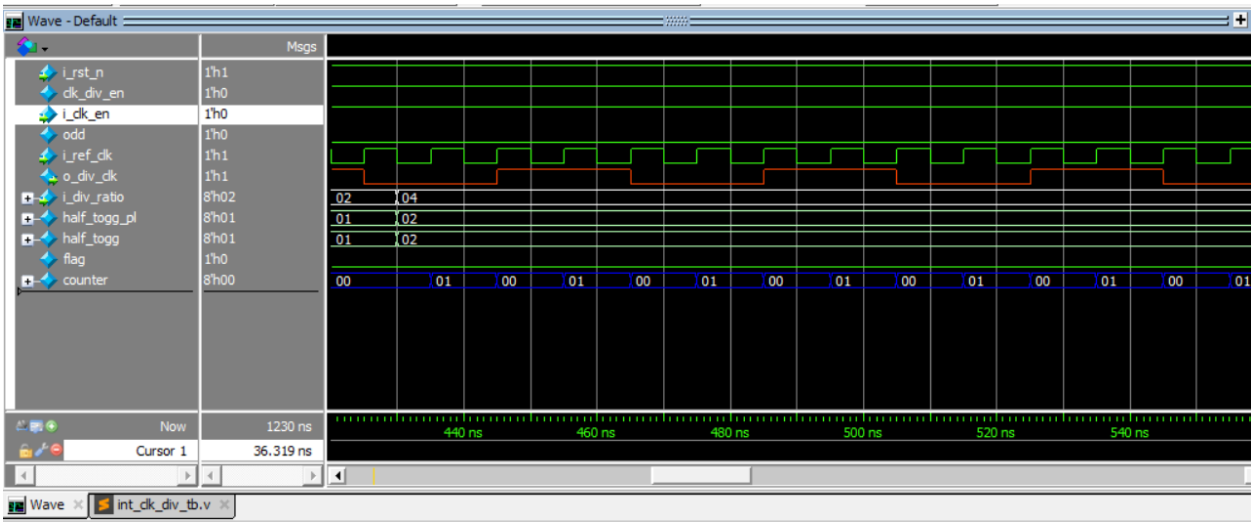

2. Dividing clk frequency by 2 with enable =0



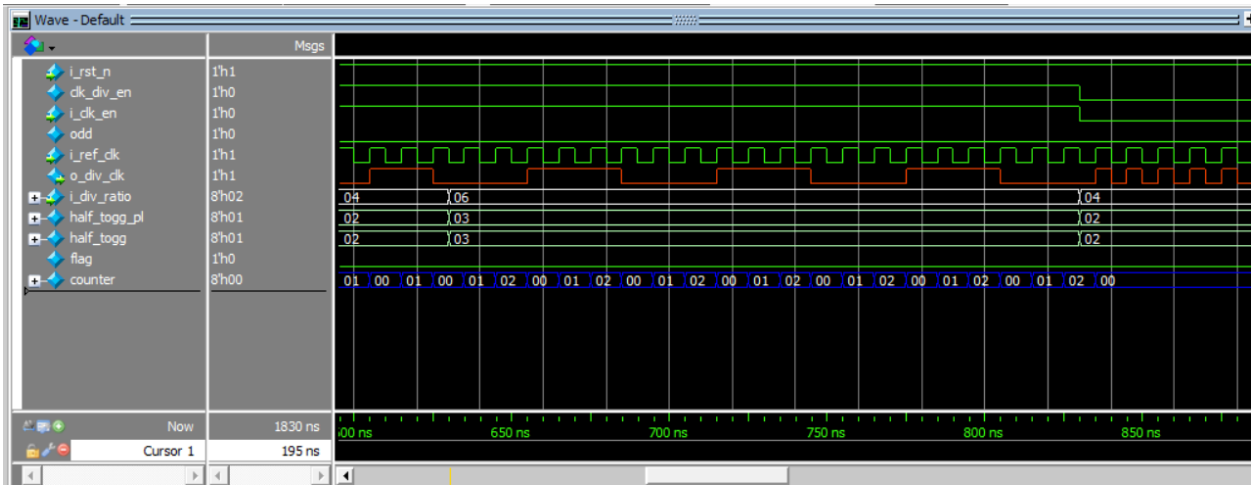
3. Dividing clk frequency by 2 with active enable



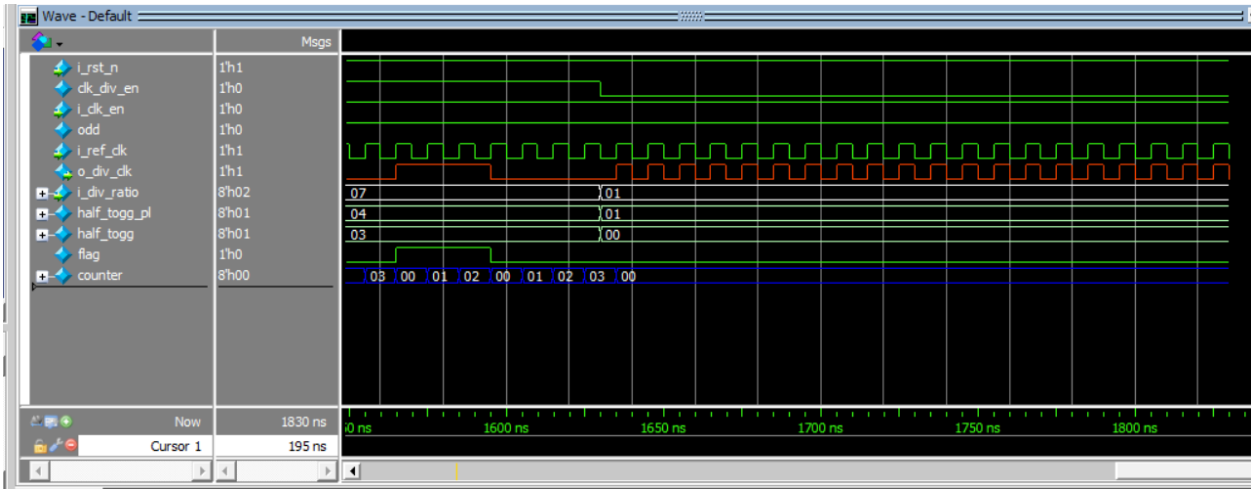

4. Dividing clk frequency by 4 with active enable



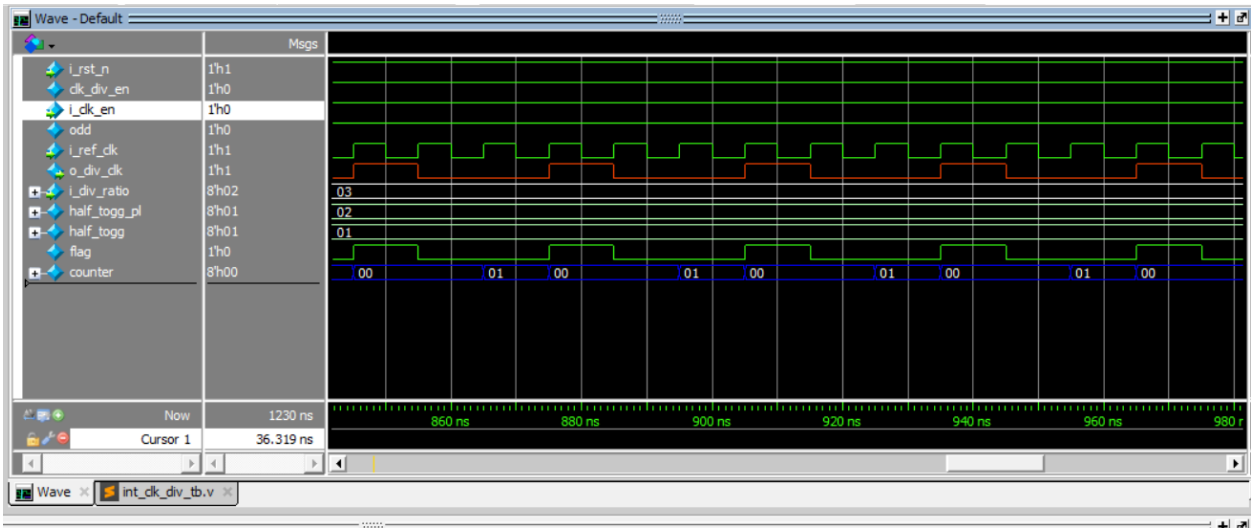
5. Dividing clk frequency by 6 with active enable



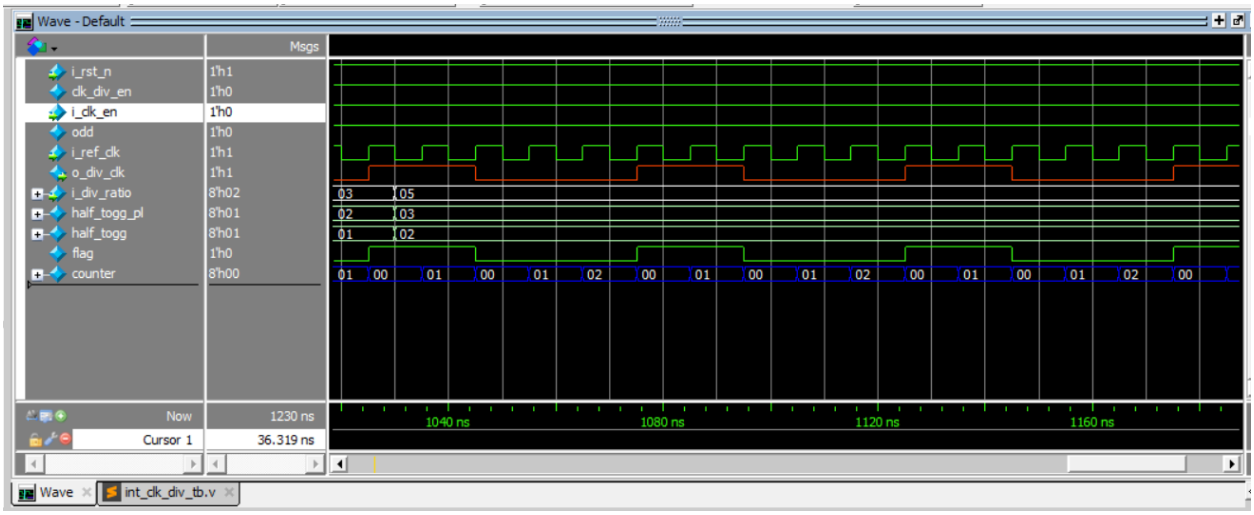

6. Dividing clk frequency by 1 with active enable



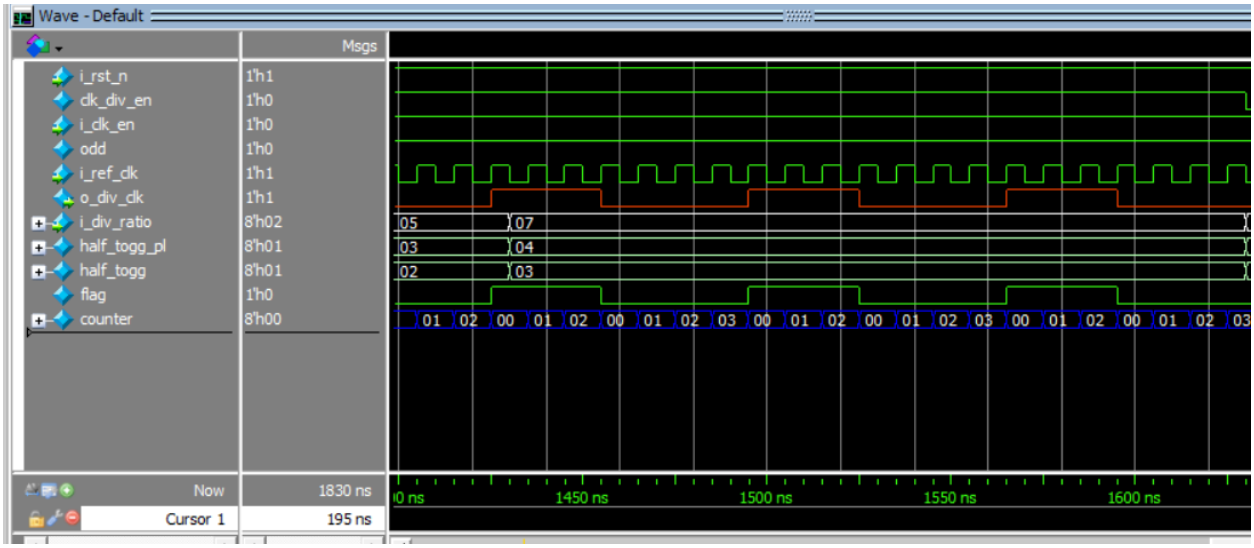

7. Dividing clk frequency by 3 with active enable



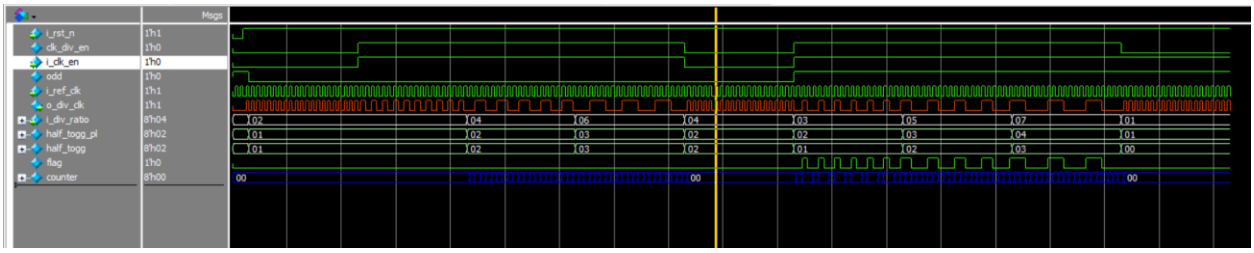
8. Dividing clk frequency by 5 with active enable




9. Dividing clk frequency by 7 with active enable




10. Full wave form



- Design

int_clk_div.v > int_clk_div

```
module int_clk_div(
    input i_ref_clk,
    input i_rst_n,
    input i_clk_en,
    input [7:0] i_div_ratio,
    output o_div_clk
);
wire clk_div_en;
wire odd;
wire [7:0] half_togg;
wire [7:0] half_togg_pl;
reg [7:0] counter;
reg flag;
reg div_clk_reg;
assign o_div_clk = (!i_rst_n)? 1'b0 :(i_clk_en && (i_div_ratio > 1)) ? div_clk_reg : i_ref_clk;
assign clk_div_en = i_clk_en && ( i_div_ratio != 0) && ( i_div_ratio != 1);
assign odd = i_div_ratio[0];
assign half_togg = i_div_ratio >> 1;
assign half_togg_pl = i_div_ratio - half_togg;

always @(posedge i_ref_clk or negedge i_rst_n) begin
    if (!i_rst_n)begin
        div_clk_reg <= 1'h0;
        counter <= 8'h0 ;
        flag <= 1'h0 ;
    end
    else if(clk_div_en)begin
        if (!odd && counter == half_togg-1)begin
            div_clk_reg <=! o_div_clk ;
            counter <= 'h0 ;
        end
        else if (odd && ((counter == half_togg-1 && flag) || (counter == half_togg_pl-1 && !flag)))begin
            div_clk_reg <= ! o_div_clk ;
            counter <= 'h0 ;
            flag <= !flag ;
        end
        else
            counter <= counter + 'h1 ;
    end
    else begin
        counter   <= 8'd0;
        flag      <= 1'b0;
    end
end
endmodule
```

- Testbench

```
int_clk_div_tb.v > int_clk_div_tb
1   `timescale 1ns/1ps
2   module int_clk_div_tb();
3       reg i_ref_clk;
4       reg i_rst_n;
5       reg i_clk_en;
6       reg [7:0] i_div_ratio;
7       wire o_div_clk;
8
9   int_clk_div DUT(
10      .i_ref_clk (i_ref_clk),
11      .i_rst_n   (i_rst_n),
12      .i_clk_en  (i_clk_en),
13      .i_div_ratio(i_div_ratio),
14      .o_div_clk (o_div_clk)
15  );
16
17
18  always #5 i_ref_clk =~i_ref_clk;
19
20  initial begin
21      $dumpfile("clk_div_DUMP.vcd") ;
22      $dumpvars;
23      initialize_ports();
24      reset();
25      randomize_inputs(0,2);
26
27      randomize_inputs(1,2);
28      randomize_inputs(1,4);
29      randomize_inputs(1,6);
30
31      randomize_inputs(0,4);
32      randomize_inputs(1,3);
33      randomize_inputs(1,5);
34      randomize_inputs(1,7);
35      randomize_inputs(1,1);
36
37  $stop;
38  end
39
```

- Tasks

```
39  task initialize_ports();
40  begin
41      i_ref_clk =0;
42      i_rst_n =0;
43      i_clk_en =0;
44      i_div_ratio =1;
45      @(negedge i_ref_clk);
46  end
47  endtask
48
49  task reset();
50  begin
51      i_rst_n =0;
52      @(negedge i_ref_clk);
53      i_rst_n =1;
54      @(negedge i_ref_clk);
55  end
56  endtask
57
58  task randomize_inputs(input clk_en, input [7:0] div_ratio);
59  begin
60      i_clk_en    = clk_en;
61      i_div_ratio = div_ratio;
62      repeat(20)@(negedge i_ref_clk);
63  end
64  endtask
65
66  endmodule
```

- Run file

```
run.do
1   vlib work
2   vlog -f src_files.list
3   vsim -voptargs=+acc work.int_clk_div_tb
4   do wave.do
5   run -all
6   #quit -sim
```

- Source list

```
src_files.list
1   int_clk_div.v
2   int_clk_div_tb.v
```

- Wave.do

```
wave.do
1   onerror {resume}
2   quietly WaveActivateNextPane {} 0
3   add wave -noupdate /int_clk_div_tb/DUT/i_rst_n
4   add wave -noupdate /int_clk_div_tb/DUT/clk_div_en
5   add wave -noupdate /int_clk_div_tb/DUT/i_clk_en
6   add wave -noupdate /int_clk_div_tb/DUT/odd
7   add wave -noupdate /int_clk_div_tb/DUT/i_ref_clk
8   add wave -noupdate -color {Orange Red} /int_clk_div_tb/DUT/o_div_clk
9   add wave -noupdate -color White /int_clk_div_tb/DUT/i_div_ratio
10  add wave -noupdate /int_clk_div_tb/DUT/half_togg_pl
11  add wave -noupdate /int_clk_div_tb/DUT/half_togg
12  add wave -noupdate /int_clk_div_tb/DUT/flag
13  add wave -noupdate -color Blue /int_clk_div_tb/DUT/counter
14  TreeUpdate [SetDefaultTree]
15  WaveRestoreCursors {{Cursor 1} {195 ns} 0}
16  quietly wave cursor active 1
17  configure wave -namecolwidth 150
18  configure wave -valuecolwidth 100
19  configure wave -justifyvalue left
20  configure wave -signalnamewidth 1
21  configure wave -snapdistance 10
22  configure wave -datasetprefix 0
23  configure wave -rowmargin 4
24  configure wave -childrowmargin 2
25  configure wave -gridoffset 0
26  configure wave -gridperiod 1
27  configure wave -griddelta 40
28  configure wave -timeline 0
29  configure wave -timelineunits ns
30  update
31  WaveRestoreZoom {0 ns} {557 ns}
32
```