FREQUENCY DIVIDER

A frequency divider, also called clock divider is a circuit that takes an input signal of a frequency fin. And generates an output signal of a frequency:



Integer Clock Dividers: where N is an integer.

N

**EVEN DIVIDER**

The division factor N is an even number (2, 4, 6, 8…).

Simple to implement using toggle flip-flops or counters.

Divide by 2

Divide by 4

Divide by 6

**ODD DIVIDER**

The division factor NNN is an odd number (e.g., 3, 5, 7).

Odd dividers inherently produce asymmetric output waveforms with unequal high and low phases (non-50% duty cycle).

Additional logic is needed for duty cycle correction if symmetry is required.

Divide by 3

Divide by 5

module clk\_div2(input clk, input reset, output reg clk\_out);

always @(posedge clk or posedge reset) begin

if (reset)

clk\_out <= 0;

else

clk\_out <= ~clk\_out; // Toggle every clock cycle

end

endmodule

module clk\_div3(input clk, input reset, output reg clk\_out);

reg [1:0] count;

always @(posedge clk or posedge reset) begin

if (reset) begin

count <= 0;

clk\_out <= 0;

end else if (count == 2) begin

count <= 0;

clk\_out <= ~clk\_out;

end else

count <= count + 1;

end

endmodule

module clk\_div4(input clk, input reset, output reg clk\_out);

reg [1:0] count;

always @(posedge clk or posedge reset) begin

if (reset) begin

count <= 0;

clk\_out <= 0;

end else if (count == 2'b11) begin

count <= 0;

clk\_out <= ~clk\_out;

end else

count <= count + 1;

end

endmodule

module clk\_div5(input clk, input reset, output reg clk\_out);

reg [2:0] count; // 3 bits to count up to 5

always @(posedge clk or posedge reset) begin

if (reset) begin

count <= 0;

clk\_out <= 0;

end else if (count == 4) begin

count <= 0;

clk\_out <= ~clk\_out; // Toggle every 5 cycles

end else

count <= count + 1;

end

endmodule

module clk\_div6(input clk, input reset, output reg clk\_out);

reg [2:0] count; // 3 bits to count up to 6

always @(posedge clk or posedge reset) begin

if (reset) begin

count <= 0;

clk\_out <= 0;

end else if (count == 5) begin

count <= 0;

clk\_out <= ~clk\_out; // Toggle every 6 cycles

end else

count <= count + 1;

end

endmodule