Experiment 1 - Clock and Periodic Signal Generation

Mohammad
Saadati,
810198410

***Abstract*— This document is Mohammad Saadati report on experiment 1 for digital logic design laboratory. In this experiment we will discuss about different method of clock generation, Frequency Divider and Baud Generator for UART Serial Communication.**

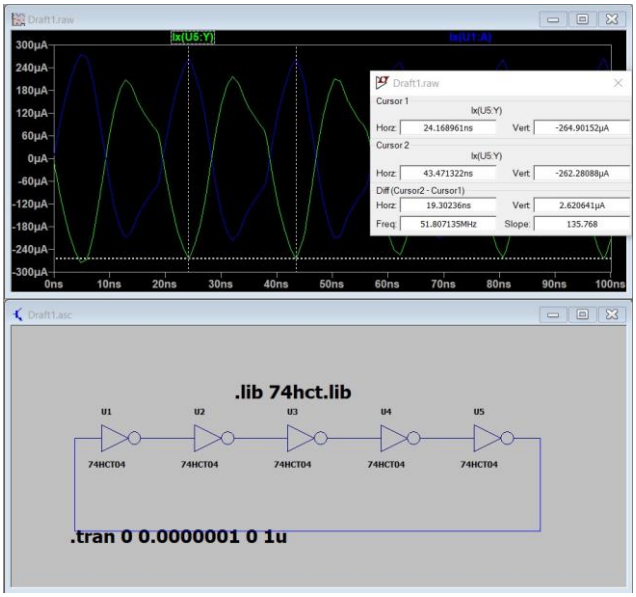
***Keywords*— Ring Oscillator, LM555, Schmitt Trigger, Frequency Divider, Baud Rate Generation for UART Serial Communication.**

INTRODUCTION

Most of digital circuits use a system called clock to work property and there are many methods to generate clock but we discuss about three of them in this experiment then we will design a Baud Rate Generation for UART Serial Communication.

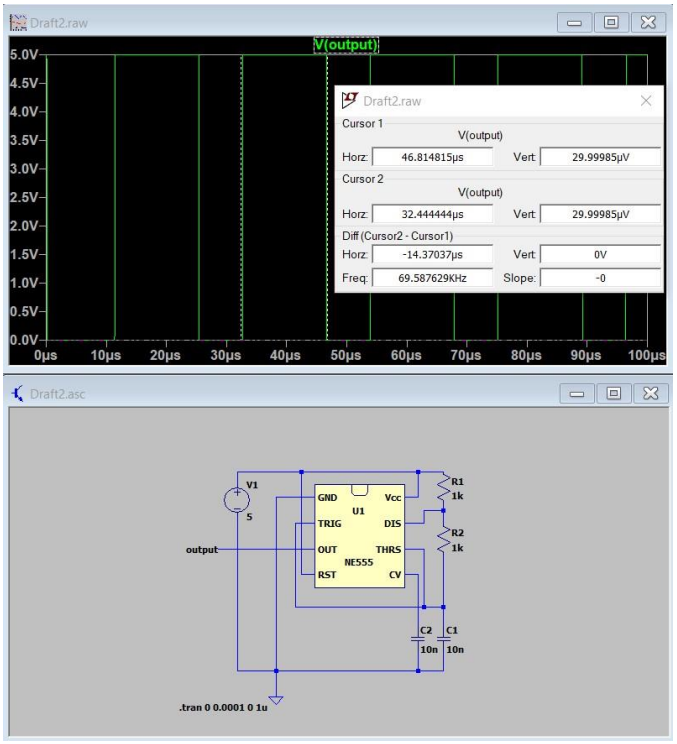
1. CLOCK GENERATION USING ICS AND ANALOG COMPONENTS

A. Ring Oscillator

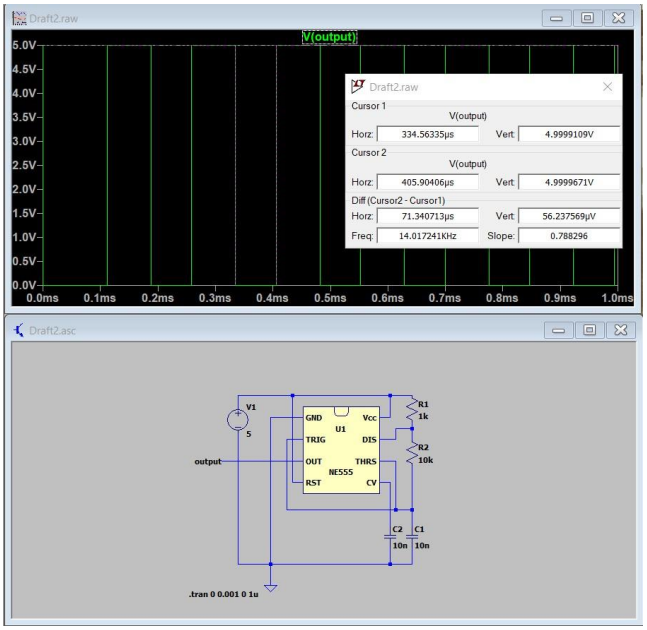


$T_{clock} = 2N * Delay_{inv}$
$N = 5$, $T_{clock} = 19.3$ ns
$Delay_{inv} = 1.93$ ns

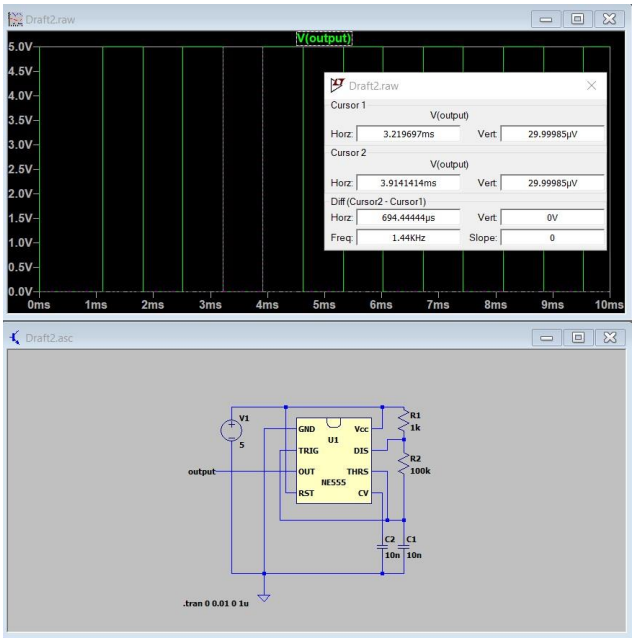
B. LM555 Timer



$R_2 = 1k$, $T_{on} = 14_{\mu s}$, $T_{off} = 7 + 14$ µs
Duty Cycle (Theory) = 2/ 3 = 66.67%
Duty Cycle = 14 / 21 = 66.67%



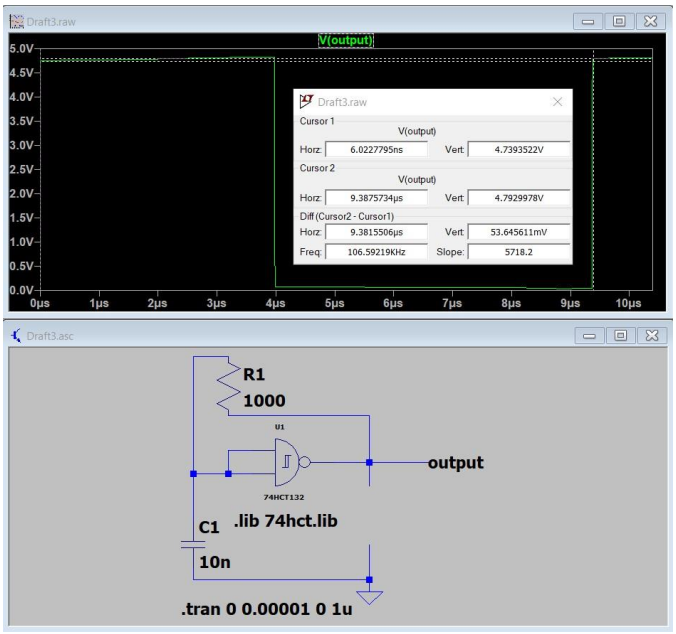
R_2 = 10k, T_{on} = 76µs, T_{off} = 71 + 76 µs
Duty Cycle (Theory) = 11/ 21 = 52.38 %,
Duty Cycle = 76 / 147 = 51.70 %



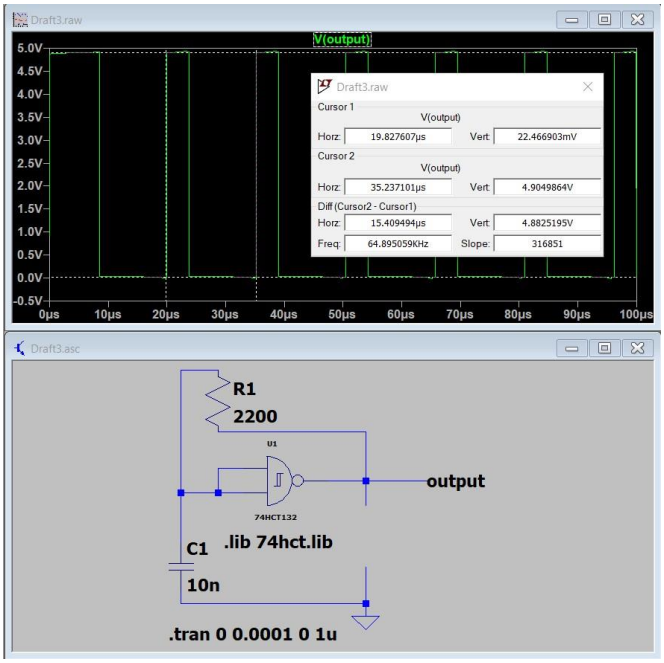
R_2 = 100k, T_{on} = 707µs, T_{off} = 707 + 694 µs
Duty Cycle (Theory) = 101/ 201 = 50.24 %,
Duty Cycle = 707 / 1401 = 50.46 %

C. Schmitt Trigger Oscillator

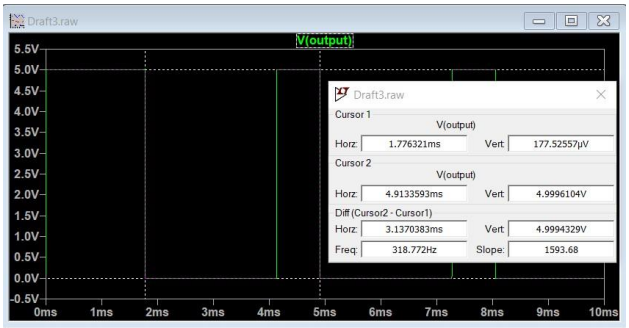
f = α / RC , f = 1/ T
α = RC / T

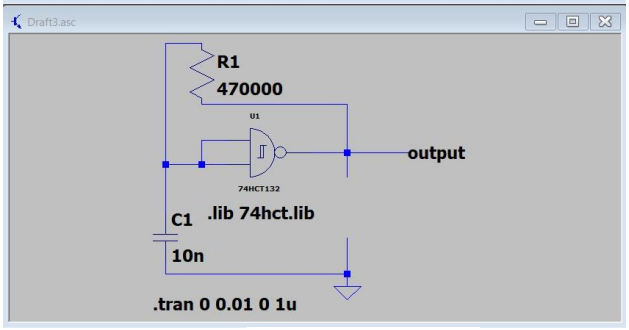


T = 9.38 µs , R=1k , C = 10n
α = 1.06



T = 15.4 µs , R=2.2k , C = 10n
α = 1.42



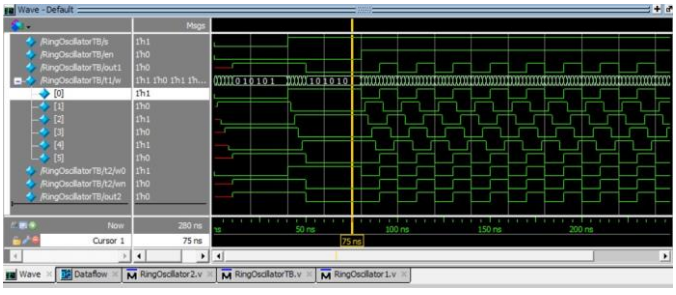


T = 3137 $_{\mu s}$, R=470k , C = 10n
α = 1.49c.

2. FPGA DESIGN

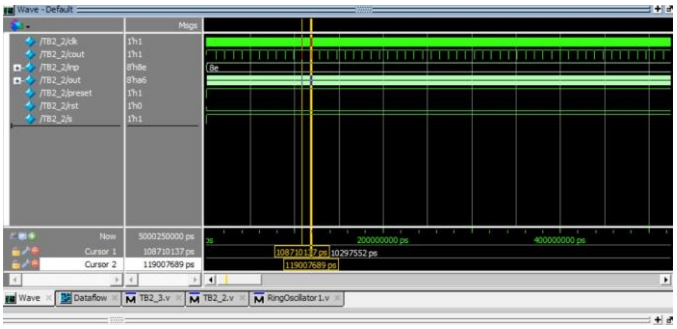
A. *Ring Oscillator*

```
module RingOscillator1 #(parameter n = 5 , delay = 10)(input start, enable, output clk);
        wire w[0:n];
        assign w[0] = enable ? w[n] : start;
        assign clk = w[n];
        genvar i;
        generate
                for (i = 0; i < n; i = i+ 1) begin : inv
                        not #delay invg(w[i+1], w[i]);
                end
        endgenerate
endmodule

module RingOscillator2 #(parameter n = 5 , delay = 10)(input start, enable, output clk);
        wire w0, wn;
        assign w0 = enable ? wn : start;
        assign clk = wn;
        not #(delay*n) invg(wn, w0);
endmodule

module RingOscillatorTB();
        wire out1, out2;
        reg en = 1'b0;
        reg s = 1'b0;
        RingOscillator1 #(5, 2) t1(s,en,out1);
        RingOscillator2 #(5, 2) t2(s,en,out2);
        initial begin
        #40 s = 1'b1;
        #40 en = 1'b1;
        #200 $stop;
        end
endmodule
```



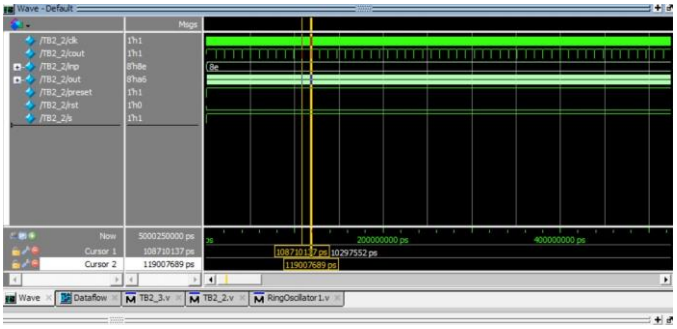
T = 20ns, f = 50KHz

B. *Synchronos Counter as a Frequency Divider*



T_{clk} = 10000ps, T_{co} = 11297552
T_{co} / T_{clk} = 113

C. *T Flip-Flop*



T_{clk} = 10000ps, T_{co} = 11297552
T_{co} / T_{clk} = 113