Experiment #1

Group 3

Nazanin Sabri

810194346

*nazanin.sabrii@gmail.com*

Nima Jarrahiyan

810194292

*jarrahian.nima76@gmail.com*

**Abstract**—working with FPGAs, ring oscillators, creating a steady rectangular (square) pulse.

Keywords— clock divider, oscillator, frequency divider, FPGA, voltage converter

1. Introduction

The goal of this experiment is to introduce the concepts of static characteristics of digital logic gates, delay times, clock frequency generation and digital system using schematic diagram and Verilog HDL.

1. Part 1

This part was omitted from the experiment due to the absence of signal generators.

1. Part 2

In this part, we understood different methods of clock generation in digital systems.

1. Ring Oscillator

Our duty cycle is about 50%, the number of inverters used in our circuit; need to be an odd number. Different ICs produce different frequencies.

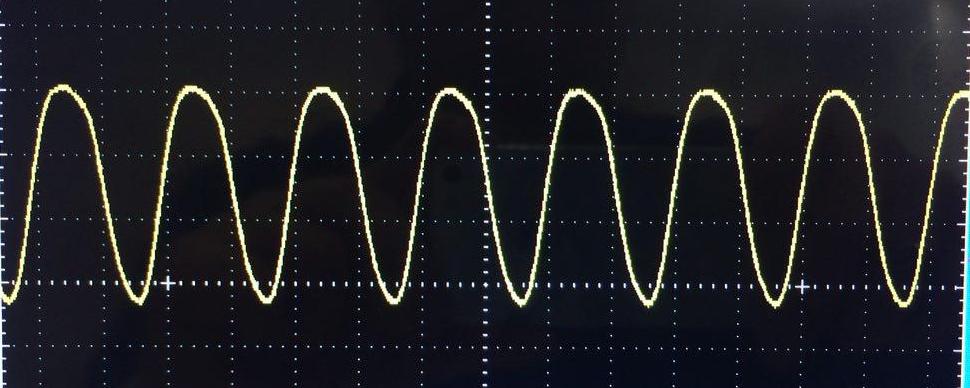


Fig. Example of a ring oscillator output signal

In order to measure the delay of each inverter in the ring oscillator, we divide the frequency of the output signal by 3.

1. LM555 Timer

This IC generates a square (rectangular) signal with different duty cycles. The duty cycle is computed using the following formula:

As the resistance of R2 increases, the duty cycle of the output pulse gets closer to 50%.

The required frequencies are:

For R1 = 1KOhm and R2 =

1KOhm, f =

10 KOhm, f =

100 KOhm, f =

1. Schmitt Inverter Oscillator

In this inverter, the formula for the duty cycle and the frequency are different and are as follows:

1. Part 3

The goal of this part is to measure the relation between the clock frequency and the frequency of FPGA.

In order to work with the FPGA, we need to make sure that the input signal is a steady square signal of 3.3 volts.

To build such a signal we went through the following steps:

1. Oscillator: we used the oscillator made in part 2 (A), to generate a clock frequency.
2. Counter: Using the counter of part 2 (D), we divided the oscillator frequency. More on how that was done is available in the mentioned section of the report. (Part 2 – D)
3. 74ls74: to build a steady square signal
4. Voltage Converter: to convert the 5v signal to a 3.3v one.

We then passed on the output of the counter to the 74ls74 as a clock generator. The output of 74ls74 was then converted to a 3.3v signal using the voltage converter and then passed to the FPGA as an entry signal.

We faced some difficulties in converting the output voltage. To solve the problem we used FPGA as a source for the 3.3v.

Following the steps of the appendix available in the end of the lab 1 we created a project in Quartus, using the VHDL code and the data sheet of out FPGA we assigned the pins and then compiled and ran the program.

The 7 segments displayed how many local (FPGA) clock cycles our signal was. The calculation that proves the display was correct is shown bellow:

Delay time of the oscillator = 1.7 \* 20ns

Frequency of the oscillator = 29 MHz

This frequency was supposed to be divided of 113 making the divided result 0.26 MHz. The delay time of the output of the counter was 2 \* 2 microseconds. Making the frequency of the counter 0.25 MHz.

The inside clock frequency of the FPGA is 50 Hz. Making our counter output frequency 200 of the FPGA clock frequency.

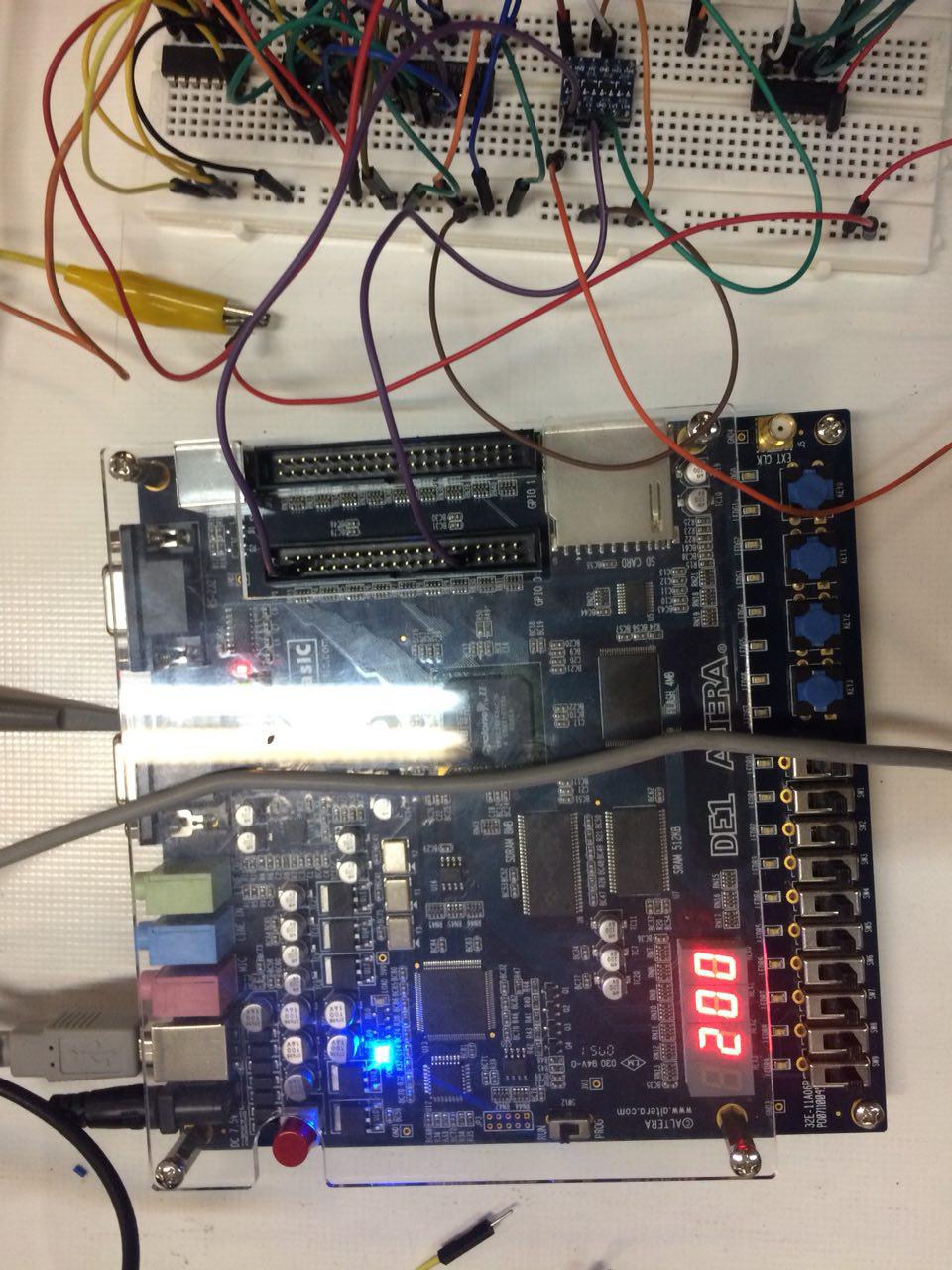


Fig. Display of FPGA seven-segment