Experiment 3 – Function Generator

Shirin Jamshidi - 810199570, Mahya Shahshahani - 810199598

***Abstract*:** In this experiment we made a function generator which generates waves such as: rectangular, triangular, semi-triangular, reciprocal, sinusoidal, full-wave sine, and half-wave sine. We were able to change the amplitude and frequency of these signals.

***Keywords*:** *Function generator – waveform generator – frequency selector – amplitude selector – Direct Digital Synthesis (DDS) – PWM - ROM*

Introduction

This experiment focuses on designing an Arbitrary Function Generator (AFG) and understanding its essential components, such as the waveform generator, function selectors, frequency selector, amplitude selector and ROM memory. The experiment also covers the integration of ROM memories into the AFG design and provide an opportunity to develop proficiency in schematic design using Quartus II.

I. Experiments

A. *Waveform Generator*

This module produces functions shown below:

Func[2:0]	Waveform(Function)
3'b000	Reciprocal
3'b001	Square
3'b010	Triangle
3'b011	Semi-triangle
3'b100	Sine
3'b101	Full wave rectified
3'b110	Half-wave rectified

Table 1: Function selection

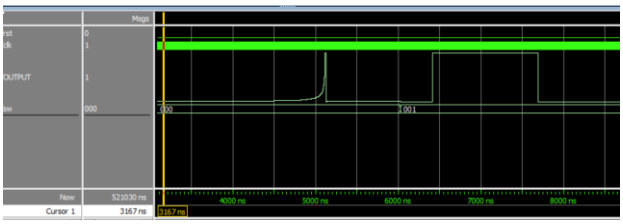

Figure 1: Reciprocal and Square wave simulation

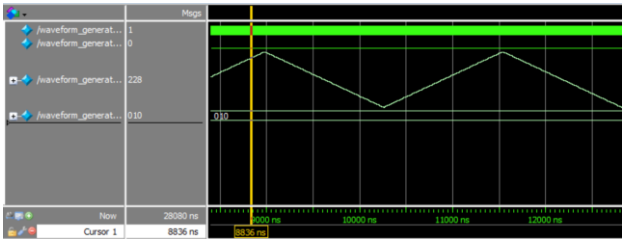

Figure 2: Triangle wave simulation

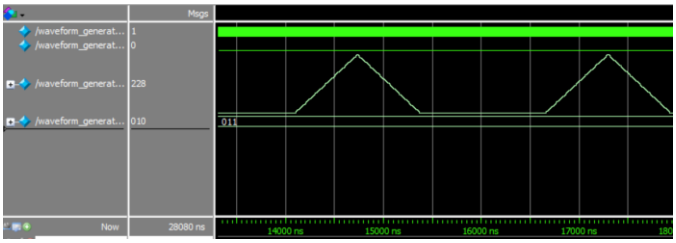

Figure 3: Semi-Triangle wave simulation

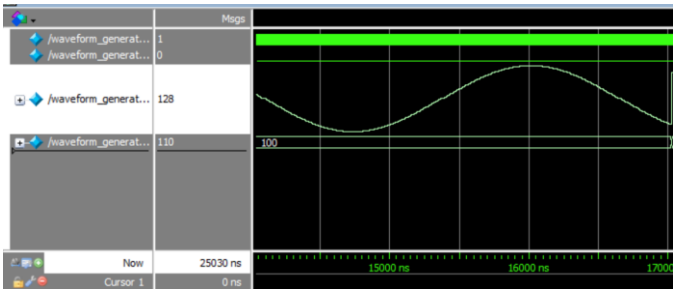

Figure 4: Sine wave simulation

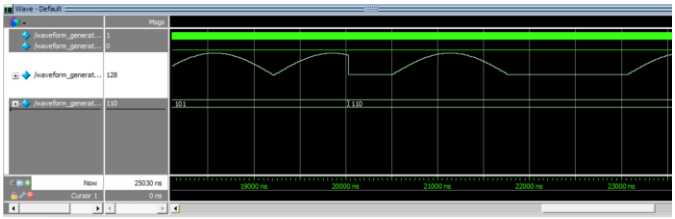

Figure 5: full and half wave simulation

Using the romstyle keyword can optimize FPGA resource utilization by explicitly instructing the synthesis tool on how to implement ROM. Without the romstyle keyword, the synthesis tool's default behaviour might not always yield the most efficient resource utilization, particularly for larger ROMs.

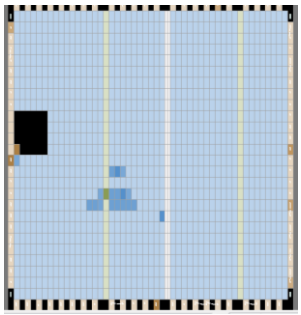

Figure 6: With ROM

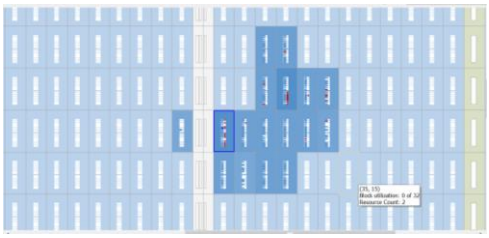

Figure 7: Without ROM

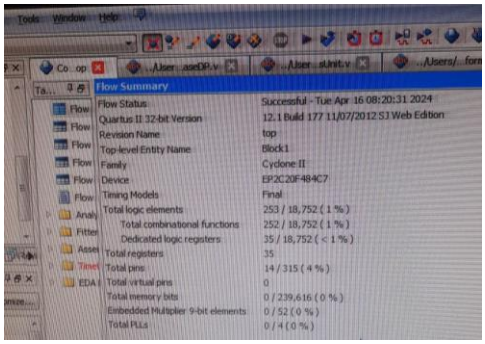

Figure 8: Quartus Synthesis report

B. PWM (Pulse Width Modulation)

We use PWM as a DAC. There are different methods for digital to analog conversion. We can either use an external chip like using an add-on-board card that consists of both ADC and DAC or we can implement that by Pulse Width Modulation (PWM). In this experiment, the period of PWM is fixed to 256 clocks and its pulse width is the value on 8-bit input of the module.

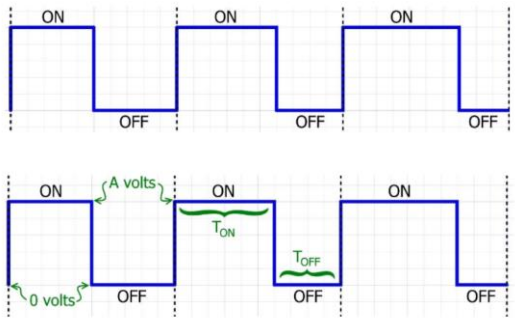

Figure 9: Pulse Width Modulation (PWM)

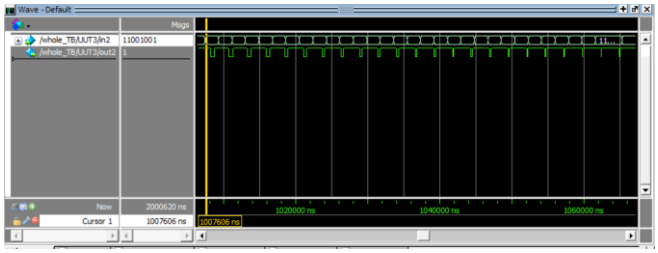

Figure 10: Pulse 3 PWM signals generated.

C. Frequency Selector

To set the output signal frequency, a frequency selector is necessary. This frequency selector uses a counter to divide a high-frequency input signal down to the desired frequency.

Simulations for three different frequencies are shown below:

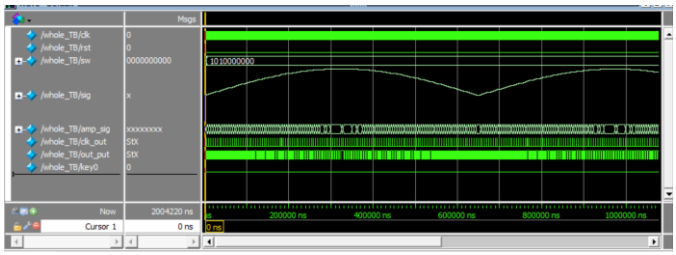

Figure 11: simulation for three different frequencies, frequency one.

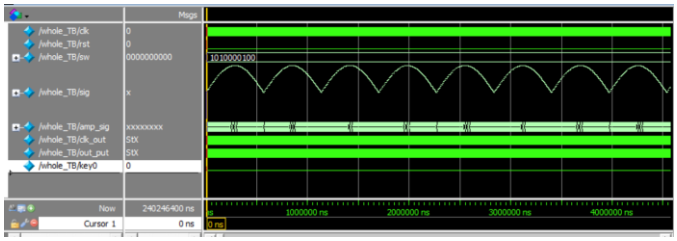

Figure 12: simulation for three different frequencies, frequency one

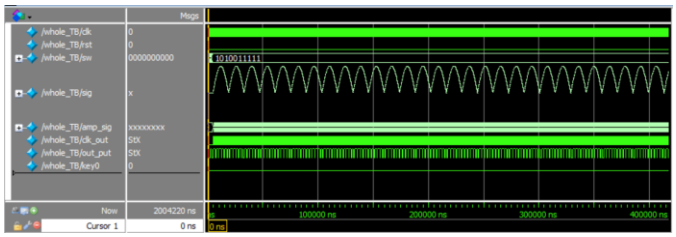

Figure 13: simulation for three different frequencies, frequency three.

D. Amplitude Selector

Simulations for three different amplitudes are shown below:

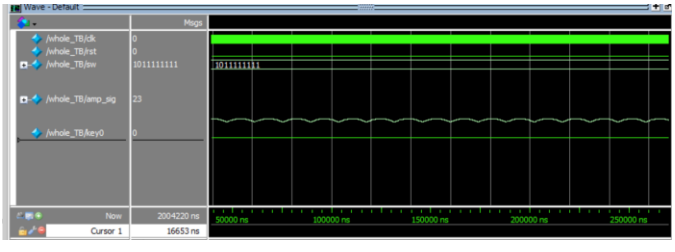

Figure 14: simulation for three different amplitudes, amplitude one.

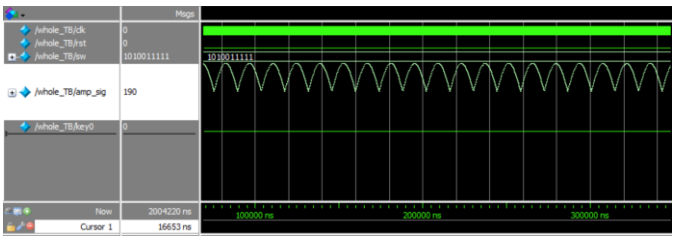

Figure 15: simulation for three different amplitudes, amplitude two.

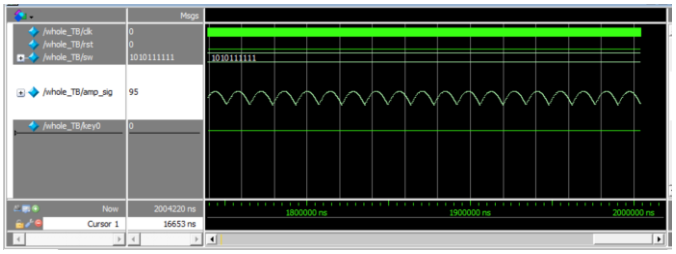

Figure 16: simulation for three different amplitudes, amplitude three.

E. Implementation

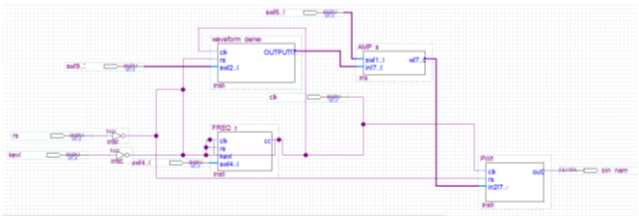

Figure 17: Schematic diagram of the Function Generator within Quartus.

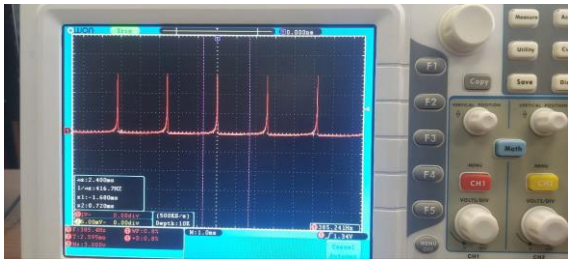

Figure 18: Reciprocal wave

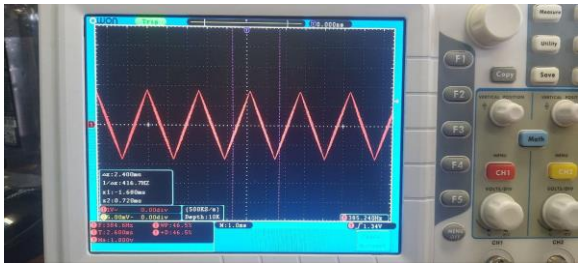

Figure 19: Triangle wave

Figure 20: Square wave

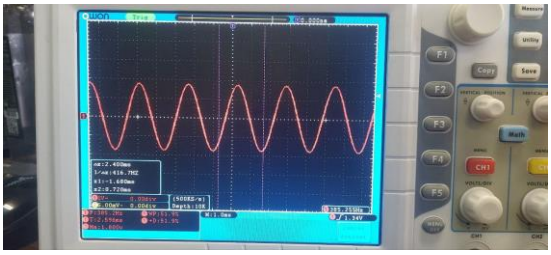

Figure 21: Sine wave

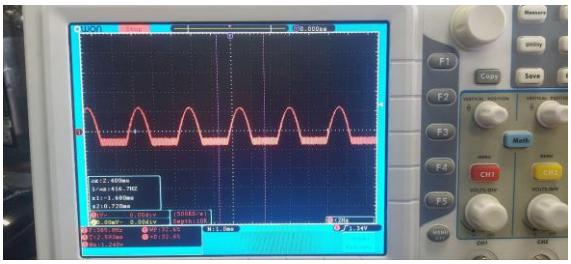

Figure 22: Sine half wave

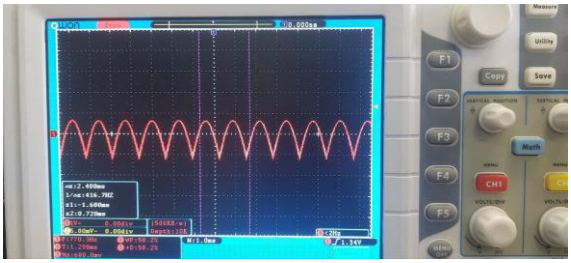

Figure 23: Sine full wave

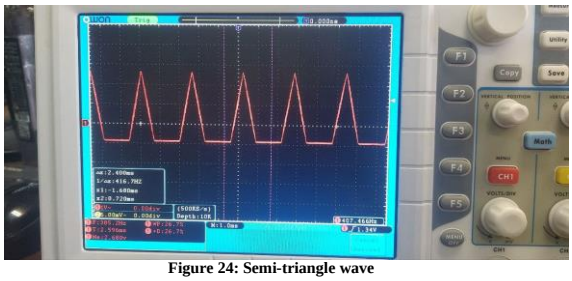

Figure 24: Semi-triangle wave

Figure 25: Sine wave with different frequencies.

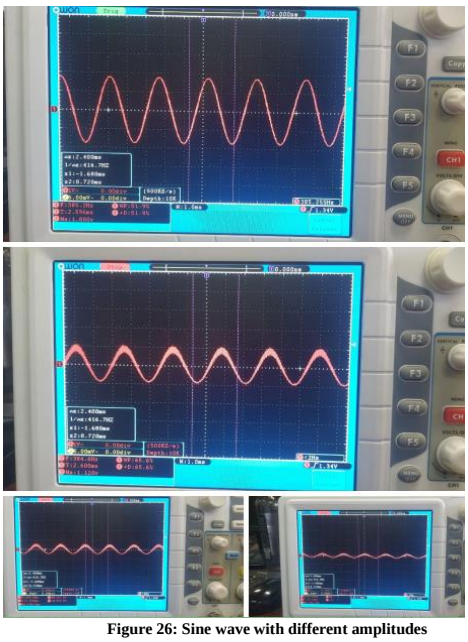

Figure 26: Sine wave with different amplitudes

F. References

[1] Katayoon Basharkhah and Zahra Jahanpeima and Prof. Zain Navabi, *Digital Logic Laboratory*, University of Tehran, Spring 1403.