Experiment #3 – Function Generator

Mohammad
Saadati,
810198410

***Abstract*— This document is a student report to experiment #3 of Digital Logic Laboratory course at ECE Department, University of Tehran. The goal of this experiment is to design an arbitrary function generator which is capable of generating many waveforms such as Rhomboid, Square, Reciprocal, and etc with wide range for frequency selection.**

***Keywords*— Function Generator, Waveforms, Frequency Selector, Amplitude Selector, Cyclone IV E**

I. Introduction

An overall view of the function generator system is showed in figure 1. It consists of 3 main units:

1. Waveform Generator.
2. Frequency Selector.
3. Amplitude Selector.

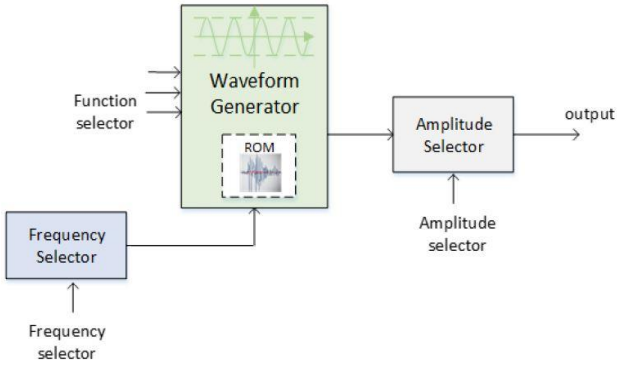

Fig. 1 Overall view of function generator

II. Waveform Generator

This module is the heart of the project. It inputs 3 bits indicating which function (waveform) needs to be generated by this unit. There are 8 choices:

Func[2:0]	Waveform
3'b000	Rhomboid
3'b001	Square
3'b010	Reciprocal
3'b011	Triangle
3'b100	Full-wave rectified
3'b101	Half-wave rectified
3'b110	Modulated square wave
3'b111	DDS

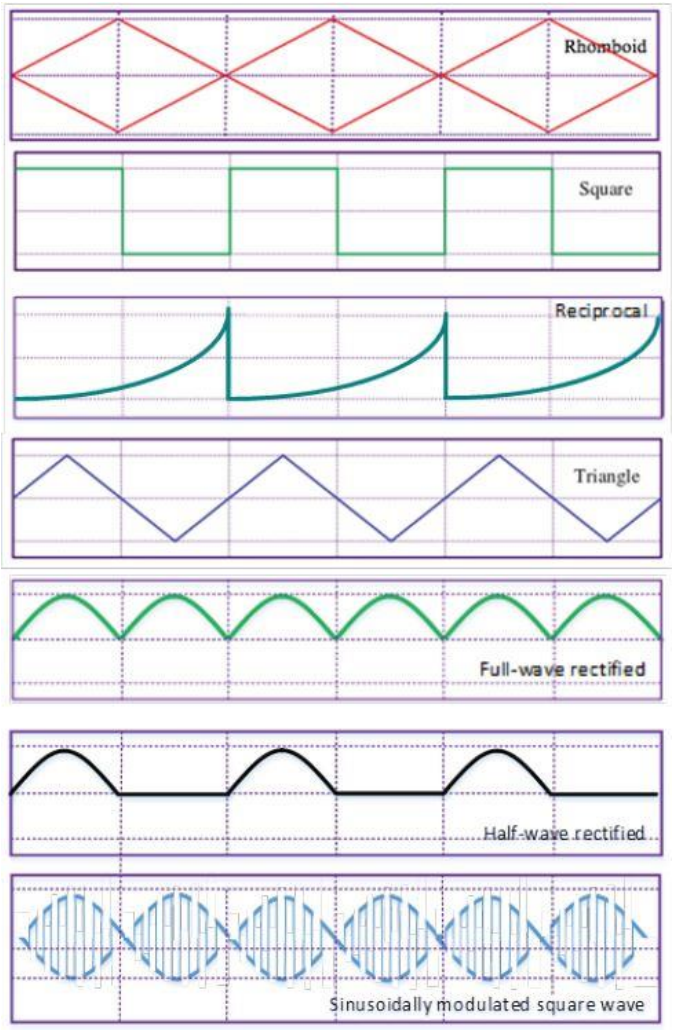

Fig. 2 Supported waveforms

The last case (3'b111) will generate a waveform stored in a ROM. The ROM is initialized using a sine.mif file with Quartus MegaWizard.

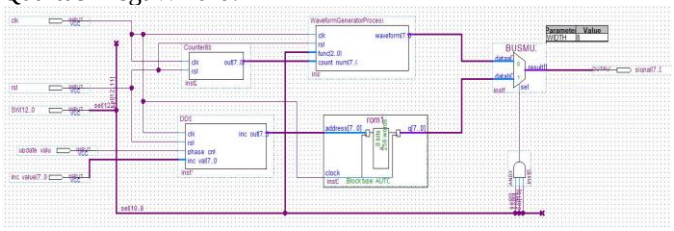

Fig 3. Block Diagram of WaveForm Generator unit

Processor unit in Fig. 3 outputs the desired function (waveforms) based on input SW[10..8]. 256 points are generated for each function. Each wave is represented by a function *f(x)* which *x* is the output of counters. An 8 bit

counter is used for Processor unit and a DDS is used for the ROM since it stores 256 words (8 bit values). The multiplexer puts the desired wave (generated by processor or ROM) on the output. This output will later be used as an input for Amplitude Selector which we discuss later in this report. Details on generating the functions can be found in WaveformGeneratorProcessor.v file.

Analysis & Synthesis Status	Successful - Fri May 21 20:24:28 2021
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	WaveformGenerator
Top-level Entity Name	WaveformGenerator
Family	Cyclone IV E
Total logic elements	261
Total combinational functions	261
Dedicated logic registers	48
Total registers	48
Total pins	32
Total virtual pins	0
Total memory bits	2,048
Embedded Multiplier 9-bit elements	0
Total PLLs	0

Fig 4. synthesis results of Waveform Generator

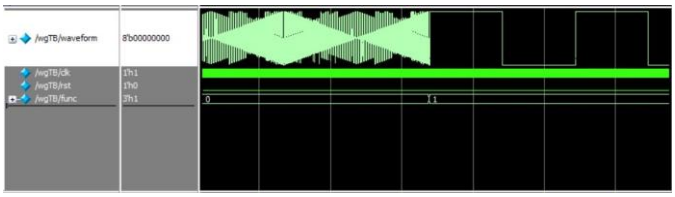

Fig. 5: Synthesize result of WaveForm Generator: Rhomboid and Square are illustrated

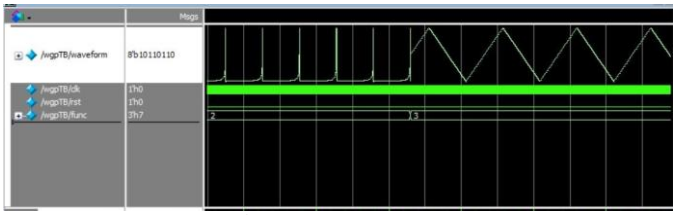

Fig. 6: Synthesize result of WaveForm Generator: Reciprocal and Triangle are illustrated

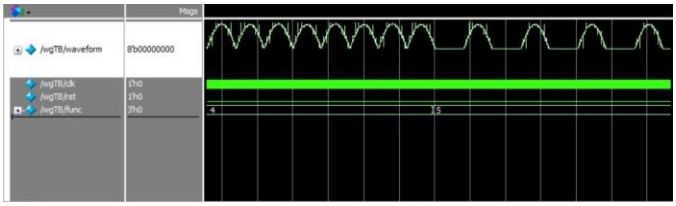

Fig. 7: Synthesize result of WaveForm Generator: Full-wave rectified and Half-wave rectified are illustrated

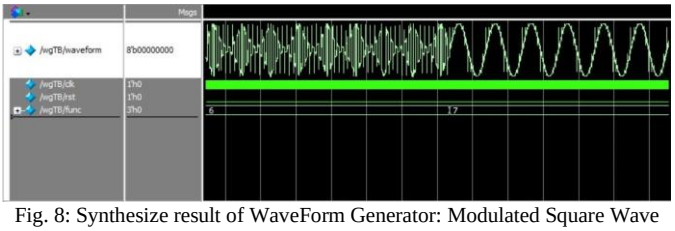

Fig. 8: Synthesize result of WaveForm Generator: Modulated Square Wave and DDS waves are illustrated

III. FREQUENCY SELECTOR

This section divides the FPGA clock (50MHz) and the divided lock is used for the WaveGenerator unit clock. The division is done by building a 256-counter with parallel load inputs. Counting starts from PL[7..0] and ends at 255 = $(11111111)_2$. When the counter reaches 255, the carry out bar bit of a 74193 IC becomes high, therefore there is a toggle on the T Flip Flop (Built using a 7476 JK Flip Flop). The output of the T Flip Flop is the divided clock. The divided clock has a duty cycle of 50%. It toggles whenever the counter reaches 255, therefore $T_{divided\_clock} = T_{FPGA} \times (255 - PL[7..0])$ where T indicates the period of a clock signal. The desired frequency can be set using the suitable parallel load values using the formula above.

```
module clkdiv8(input clk,input rst,input [7:0] pin, output reg cout, output reg [7:0] pout);
    always @(posedge clk) begin
        if (rst) begin pout <= pin; cout <= 1'b0; end
        else {cout, pout} <= pout + 8'b00000001;
        if (pout == 8'b00000000) pout <= pin;
    end
endmodule
```

Fig. 9: Verilog description of Clock Divider

A block diagram of the clock divider (Fig. 9) is built and added to the WaveForm Generator explained in section II and synthesis is done.

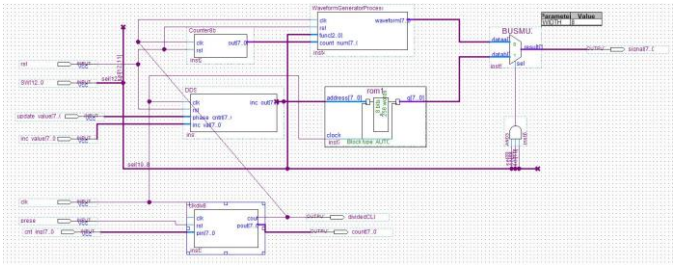

Fig. 10: Block Diagram of Wave Generator with a Frequency Selector (Clock divider)

Flow Status	Successful - Sat May 22 11:06:24 2021
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	WaveformGenFreqSelec
Top-level Entity Name	WaveformGenFreqSelec
Family	Cyclone IV E
Total logic elements	282 / 6,272 (4 %)
Total combinational functions	282 / 6,272 (4 %)
Dedicated logic registers	57 / 6,272 (< 1 %)
Total registers	57
Total pins	57 / 92 (62 %)
Total virtual pins	0
Total memory bits	2,048 / 276,480 (< 1 %)
Embedded Multiplier 9-bit elements	0 / 30 (0 %)
Total PLLs	0 / 2 (0 %)
Device	EP4CE6E22C6
Timing Models	Final

Fig 11. synthesis results of Wave Generator with a Frequency Selector

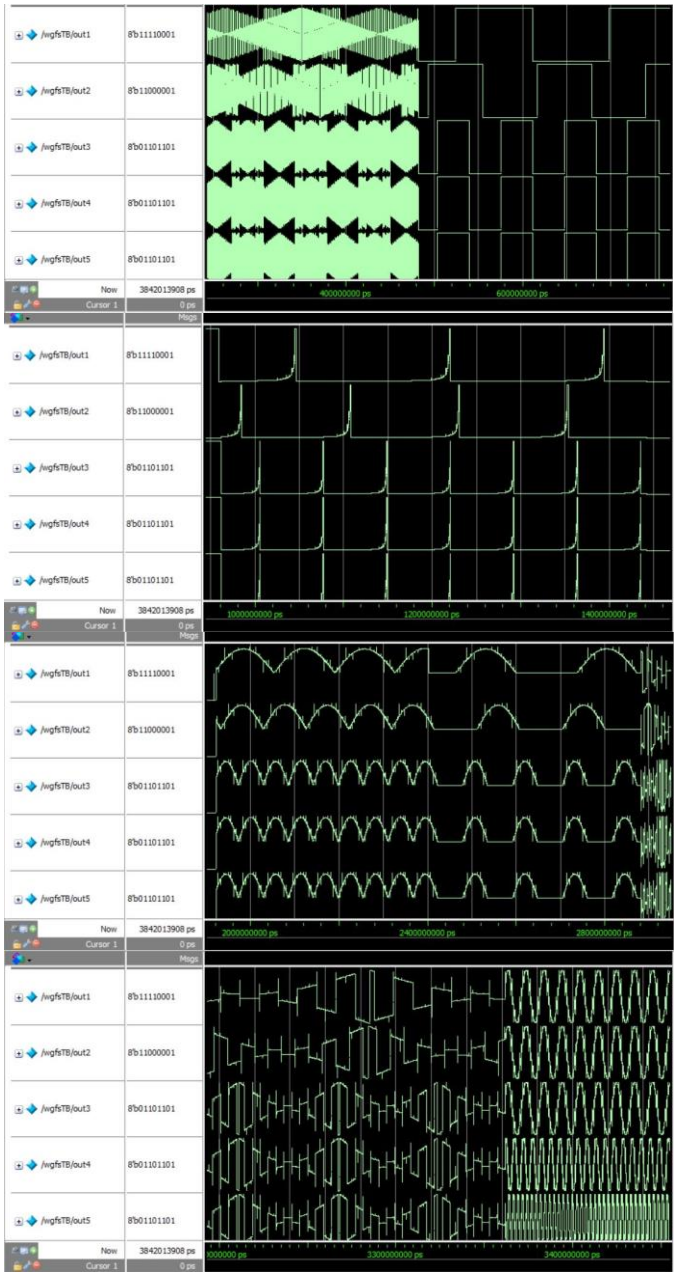

Fig. 12: Simulation result of 5 different units with different parallel loads causing 3 different frequencies and different phase_cntrl causing 3 different frequencies for DDS wave.

The unit with least parallel load (240) will have the highest clock period (It takes longer for the counters to reach 255 and for the T Flip Flop to toggle), therefore less number of positive edges in a fixed length of time leading to a lower frequency wave. The unit with highest parallel load (250) will have the lowest clock period, therefore higher number of positive edges in a fixed length of time leading to a higher frequency wave. This is can be validated in Figure 12.

$T_{divided\_clock} = T_{FPGA} \times (255 - PL[7..0])$

$T_{FPGA} = 1 / f_{FPGA} = 1 / 50MHz = 20ns$

$T_1 = 20 \times (255 - 240) = 300ns \quad -> \quad f_1 = 3.33MHz$

$T_2 = 20 \times (255 - 245) = 200ns \quad -> \quad f_2 = 5MHz$

$T_3 = 20 \times (255 - 250) = 100ns \quad -> \quad f_3 = 10MHz$

IV. AMPLITUDE SELECTOR

This is an option in the Function Generator to scale down the generated waveform. The 8 bit output of the WaveForm Generator unit is passed to this unit and scaling is executed based on a 2 bit input indicating the scaling intensity. Based on the 2 bit input, the output is divided by a number shown in the following table:

SW[12..11]	Amplitude
2'b00	1
2'b01	2
2'b10	4
2'b11	8

Dividing each waveform by 2^i is equivalent to shifting *i* units to the right, therefore sign extension needs to be applied to each signal.

```
module AmplitudeSelector(input [7:0]INsignal, input [1:0] division, output reg [7:0] signal);
    always @(INsignal or division) begin
        signal = 8'b00000000;
        case (division)
            2'b00: signal = INsignal;
            2'b01: signal = {{1{INsignal[7]}}, INsignal[7:1]};
            2'b10: signal = {{2{INsignal[7]}}, INsignal[7:2]};
            2'b11: signal = {{3{INsignal[7]}}, INsignal[7:3]};
        endcase
    end
endmodule
```

Fig. 13: Verilog description of Amplitude Selector

Analysis & Synthesis Status	Successful - Sat May 22 12:07:00 2021
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	WaveformGenFreqSelecAmpSelec
Top-level Entity Name	WaveformGenFreqSelecAmpSelec
Family	Cyclone IV E
Total logic elements	303
Total combinational functions	303
Dedicated logic registers	57
Total registers	57
Total pins	57
Total virtual pins	0
Total memory bits	2,048
Embedded Multiplier 9-bit elements	0
Total PLLs	0

Fig 14. synthesis results of Wave Generator with a Frequency Selector and a Amplitude Selector.

A block symbol of the Amplitude Selector section is added to the previous block diagrams. The final (complete) block diagram is as followed:

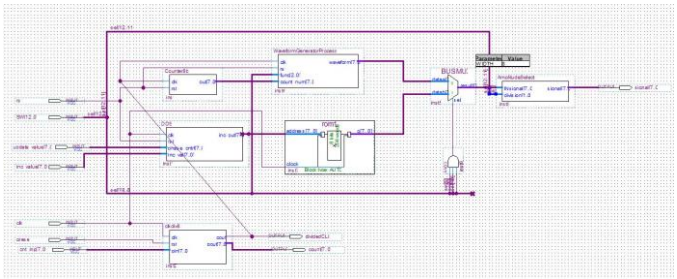

Fig. 15: Complete Block Diagram of Function Generator. For further details check WaveformGenFreqSelecAmpSelec.bdf

The final module has 13 bit inputs (SW[12..0]). 3 bits are used for function selection. 2 bits are used to amplitude selection. The remaining 8 bits are used for parallel load of counters.

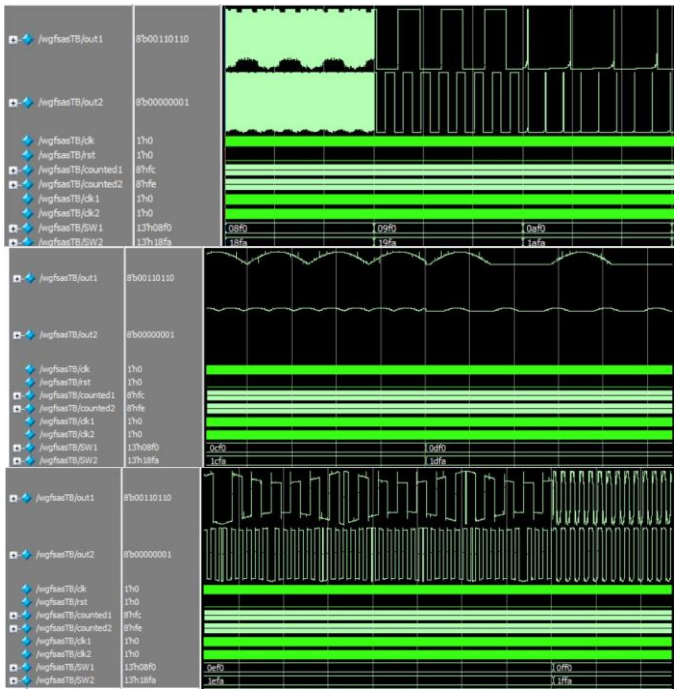

Fig. 16: Simulation results of two units with different Amplitudes. Unit 2, has Amplitude select 11 which divides the wave by 8. Meanwhile Unit 1, has Amplitude select 01 which divides the wave be 2. It can be seen that out2 has a smaller output range in compare to out1.