Experiment #4 - SOC Clock Generation

Mohammad.heydari
810197494

INTRODUCTION

System on Chip is an integrated circuit that integrates multiple components including digital, analog, hardware and software programs all in a single chip. The main core of an SoC is a processor that handles different computational tasks within the system. In addition to the processor, the system includes a memory, Input/Output ports and accelerators. In this experiment we follow some instructions to implement simple SOC clock generation as you see the whole details will be attached in further.

1 EXPONENTIAL ACCELERATOR

The accelerator that you are going to use is a exponential circuit .You are familiar with this accelerator design.As Figure 2 shows, this module receives a 16-bit input ”x” and generates an 16- bit output ”Fractionalpart” and 2-bit ”Integerpart”. The accelerator starts working with a complete pulse on signal ”start” and when the computation is completed signal ”done” will be sent to the processor to acknowledge it.

in this part using some knowledge we are going to write a Tb and only test the module of EXPONENTIAL ACCELERATOR in following figure you can see the result.

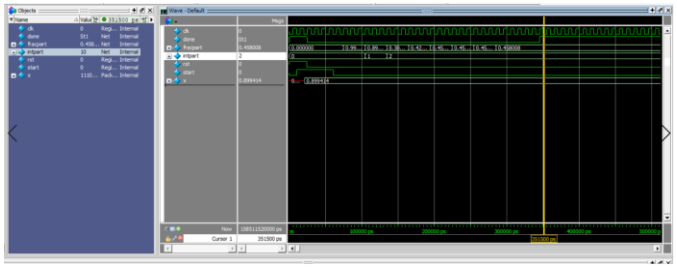


Fig. 1 result1 with x=0.9

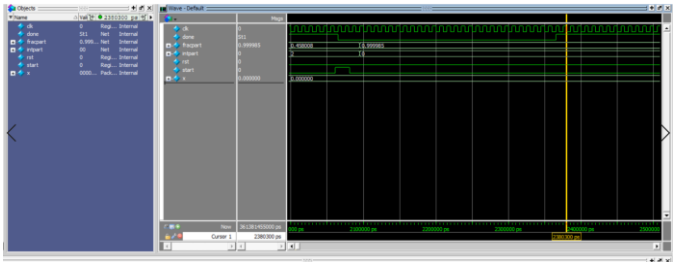


Fig. 2 result2 with x=0

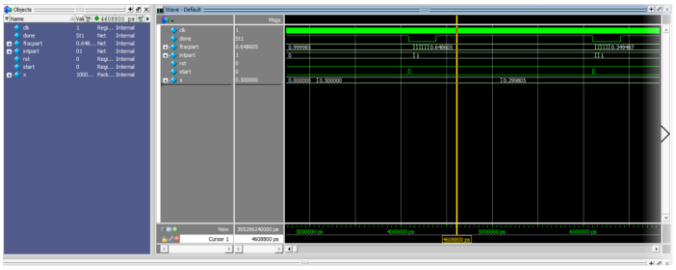


Fig. 3 result3 wtrh x=0.5

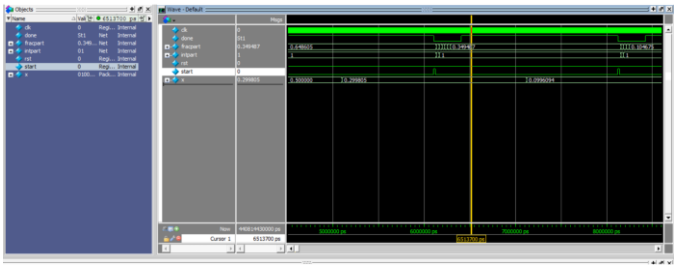


Fig. 4 result4 wtrh x=0.3

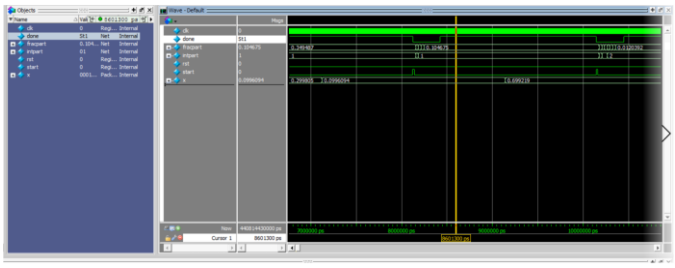


Fig. 5 result5 with x=0.1

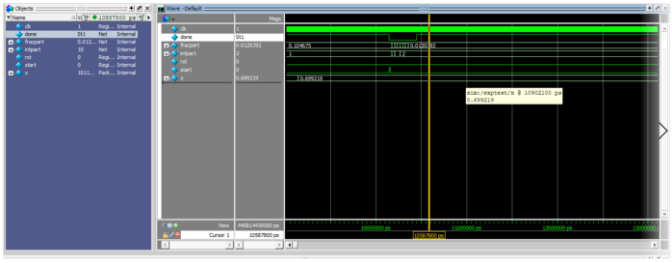


Fig. 6 result5 with x= 0.7

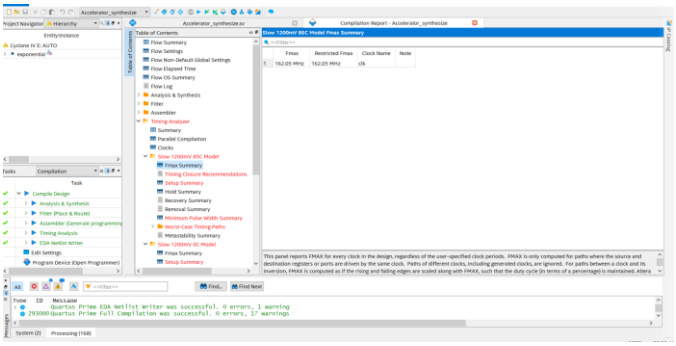
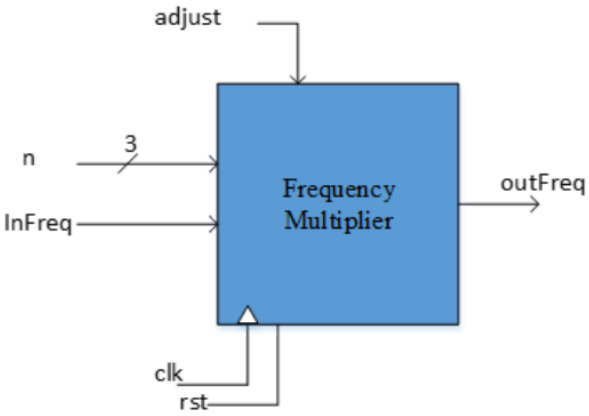


Fig. 7 result of synthesis with device 1200mV 85C

2 Frequency Multiplier

In this part, you are to design a frequency multiplier.Block diagram of this module is shown in Figure 4. A frequency multiplier takes an input signal frequency (InFreq with frequency f) and multiplies it by a value determined by multiplication factor (MultFactor). The output is a signal (OutFreq) with a frequency equal to InFreq*MultFactor. Multiplication factor is a power of two and could be shown as 2 n, where n is between 1 and 5.Therefore the circuit takes log2MultF actor as the input value. For example if MultFactor is 8, the corresponding input value would be 3. InFreq frequency range is between 1MHz and 50MHz. A reference clock input is used for reference working clock of this circuit and is provided via an input named clock RefClk, with a high frequency of 150 MHz. The output of the circuit that carries the faster signal is outFreq. The frequency of this signal is f × 2 n. The circuit has a valid output that is asserted when frequency multiplication is appearing on outFreq.

Figure 4: Block diagram of the frequency multiplier



All of code will be attached in modelsim folder

We show the result for frequency of 1Mhz for Finreq and 150 MHz for ClcRef TB module with various n

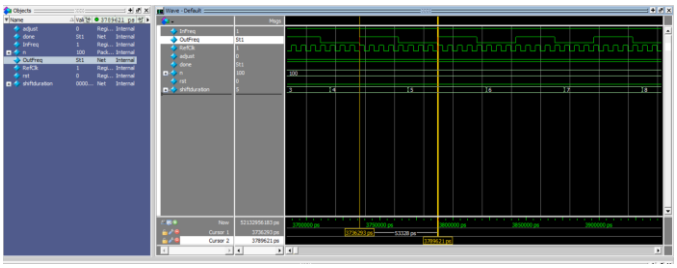


Fig 17 result1 with n=4

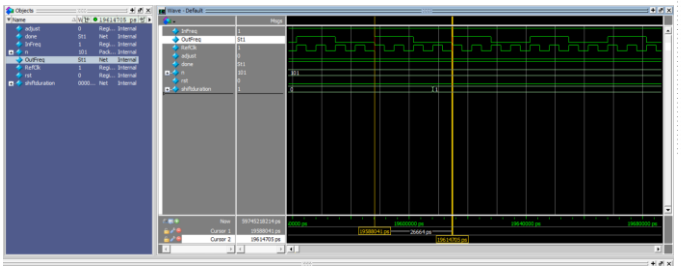


Fig 17 result1 with n=5

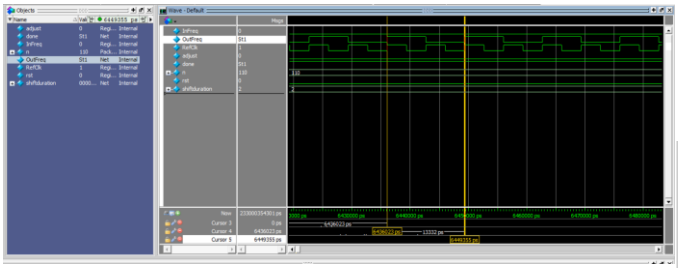


Fig 17 result1 with n=6

.

The results have been attached with best quallities

3 Integrated Circuit

In this part, you are to connected the accelerator and the frequency multiplier in an integrated circuit. For this purpose connect this two module based on the Figure 1. Although the accelerator is working with a higher frequency than the processor, for the handshaking signals of ”start” and ”done” the accelerator have to wait for the processor to send and recieve these signals with its low frequency. This imposes

some timing overhead to the accelerator performance. This overhead can be defined as below: Overhead = T Acc ÷ THsh In which, T-Acc is the clock cycles for executing the accelerator computation and T-Hsh is the clock cycles waiting for hand shaking signals.

The schematic of integrated circuits is like this that you see in below figuer

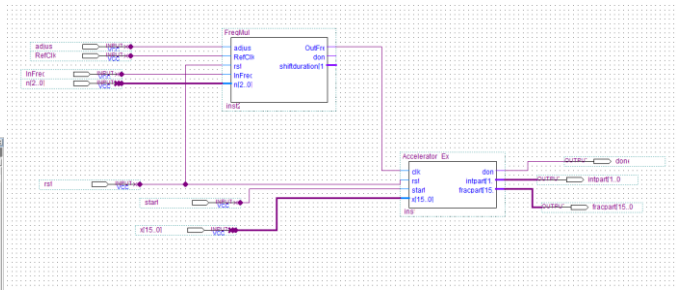


Fig 18 schem of integrated circiut in quartus

You can see the results of my works in below figures.

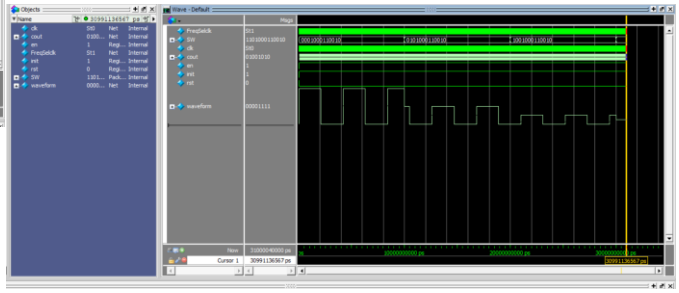


Fig 19 schem of amplitude selector in quartus

Answers to questions : in this part we implement the circuit using wiring between accelator and the frequency_multiplier after that we simulate the behaviour of CPU in my TB

Because the relativity of codes and Wave I decided to comment the result and the calculation to the files of modelsims.

CONCLUSIONS

in this experiment we are going to design an function generator that have the abilities such as changing frequency & amplitude changing and also 8 waveform that stored in out block memory via 1port_ROM. Also function generator is a specific form of signal generator that is able to generate waveforms with common shapes. Unlike RF generators and some others that only create sine waves, the function generator is able to create repetitive waveforms with a number

of common shapes. And as we know this is very useful in basic of logical circuits.

REFERENCES

[1] Some entries at github
[2] Some entries at stackoverflow.com
[3] Some entries to debug my codes
[4] A Review of below site

https://www.electronics-notes.com/articles/test-methods/signal-generators/function-generator.php