NOVATEUR PUBLICATIONS
INTERNATIONAL JOURNAL OF INNOVATIONS IN ENGINEERING RESEARCH AND TECHNOLOGY [IJIERT]
ISSN: 2394-3696
VOLUME 4, ISSUE 3, Mar.-2017

MODIFIED FREQUENCY SCALING METHOD USED TO 32 BIT X 32 BIT MULTIPRECISION MULTIPLIER

MISS.BHOSALE SHUBHANGI ASHOK
Department of Electronics & Telecommunication, VVPIET, Solapur, MS, India
Shubhangibhosale284@gmail.com

PROF.MANTRI D.B.
Department of Electronics & Telecommunication, VVPIET, Solapur, MS, India
dbmantri@yahoo.co.in

ABSTRACT

In this paper, we present a multiprecision (MP) reconfigurable multiplier that incorporates variable precision, parallel processing (PP), razor-based dynamic voltage scaling (DVS), and MP operands scheduling are used to provide a variety of operating conditions. In Previous paper the PLL used for the frequency division. If use the PLL for frequency division its hardware complexity increases. To decrease the hardware complexity and also speed is increases are done by software using some frequency division methods. The reconfigurable multiplier can either work as independent smaller-precision multipliers or work in parallel to perform higher-precision multiplications. To operate at the proper precision and frequency the user's require to configure a dynamic voltage/frequency scaling management unit.

KEYWORDS: Computer arithmetic, dynamic voltage scaling, low power design, multi-precision multiplier.

INTRODUCTION

Now days, the demand for low power, high performance portable devices has been greatly increased. The demands for microelectronic circuits design with low power dissipation and it is mainly due to internal components [1]. In Digital Signal Processing most frequently used arithmetic operation is multiplication. So today's in digital signal processing a Multipliers play an important role and also in various others application. With advanced technology, the following design targets – high speed, low power consumption and hence less area or even combination of them in one multiplier thus making them suitable for various high speed, low power VLSI implementation.

Multipliers are used for applications have large area and consume considerable power. Therefore design of low-power multiplier has been an important part in low- power VLSI system design. Multipliers have large area, long latency and consume considerable power. A multiplier play an important part in digital signal processing systems, like frequency domain filtering (FIR and IIR), frequency-time transformations (FFT), Correlation, Digital Image processing etc.

While focused on implementing high-speed multipliers, recently there have been many attempts to reduce power consumption. This is due to the increased demand for portable multimedia applications, which require low power consumption with high speed. There is wide range of multipliers like serial, parallel and serial-parallel multipliers and it is based on the way the data is processed, Dynamic Voltage Scaling (DVS) can be used to match the circuit's real working load and further reduce the power consumption.

METHODOLOGY

In block diagram consist of input operands scheduler, frequency voltage management unit[2] voltage scaling unit, frequency scaling unit, multiprecision multiplier. In this paper only we focus on frequency division module.

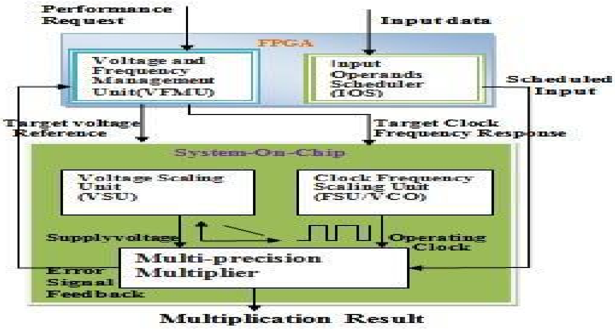

Performance Request
Input data
FPGA
Voltage and Frequency Management Unit(VFMU)
Input Operands Scheduler (IOS)
Scheduled Input
Target voltage Reference
Target Clock Frequency Response
System-On-Chip
Voltage Scaling Unit (VSU)
Clock Frequency Scaling Unit (FSU/VCO)
Supplyvoltage
Operating Clock
Error Signal Feedback
Multi-precision Multiplier
Multiplication Result

Fig 1.Overall multiplier system architecture

FREQUENCY DIVISION

In this paper use the three multiprecision multiplier like 8x8bits,16x16bits and 32x32bits.The 8x8bits multiplier required less frequency as compare to the 16x16 bits multiplier and 16x16bits multiplier required less frequency as compare to 32x32bits multiplier. If 32x32bits multiplier require f frequency then 16x16bits require f/2 and 8x8bits require f/4 frequency.

In Previous paper used the PLL circuit for the frequency division it's very complex hardware, in this paper we replace the PLL by softwaerlly frequency division module. In this module we use the f frequency as the circuit frequency and is divide into the f/2,f/4.

FREQUENCY SCALING & RESULTS

1) Using FSM;
2) Using JK FF;
3) Using T FF;
4) Using D FF;

1) USING FSM:

We can done the frequency scaling using FSM. It's the simple process. Frequency is divided by 2 require only two state and frequency divided 4 require the 4 state. In our project require frequency division by 2 and 4.

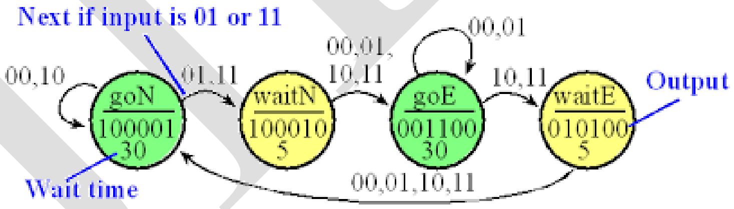
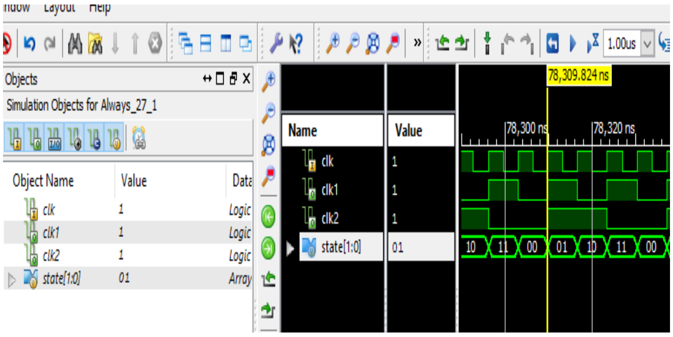


Fig 2. State diagram of frequency division by 2 and 4

SIMULATION RESULT

The clock is main frequency and the frequency divided by 2 is the clock1 and frequency divided by 4 is clock2. An every positive clock change the state to next state .The FSM is in cyclic from its generating continuously clock.



Simulation result of FSM.

Maximum output required time after clock: 4.202ns and require Number of Slice Registers=3, Number of Slice LUTs=3.

2) USING JK FF:

We can done the frequency scaling using JK FF. It's the simple process. Frequency is divided by 2 require only one JK FF and frequency divided 4 require the Two JK FF. In our project require frequency division by 2 and 4.

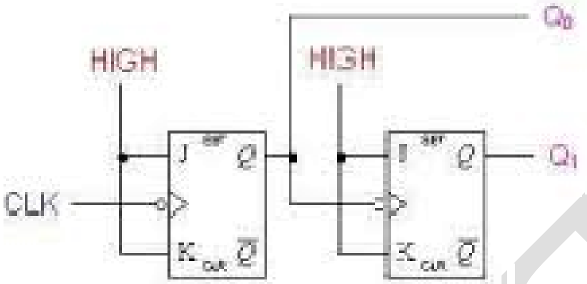
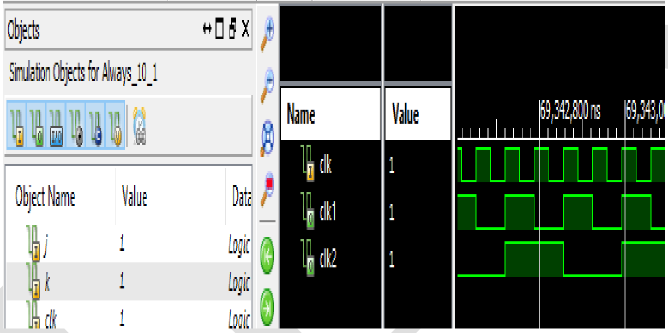


Fig 3. Block diagram of frequency division by JK FF.

SIMULATION RESULT:

The clock is main frequency and the frequency divided by 2 is the clock1 and frequency divided by 4 is clock2. An every positive clock change the first FF output is the clock1 mean frequency divided by 2 and output of the second is clk2 means frequency divided by 4.The JK FF input is given by 1 for J and k.


Objects

Simulation Objects for Always_10_1

Object Name	Value	Data
j	1	Logic
k	1	Logic
clk	1	Logic

Name	Value
clk	1
clk1	1
clk2	1

Simulation result of JK FF.

Maximum output required time after clock: 4.202ns and require Number of Slice Registers=2, Number of Slice LUTs=3.

3) USING T FF:

We can done the frequency scaling using T FF. It's the simple process. Frequency is divided by 2 require only one T FF and frequency divided 4 require the Two T FF. In our project require frequency division by 2 and 4.

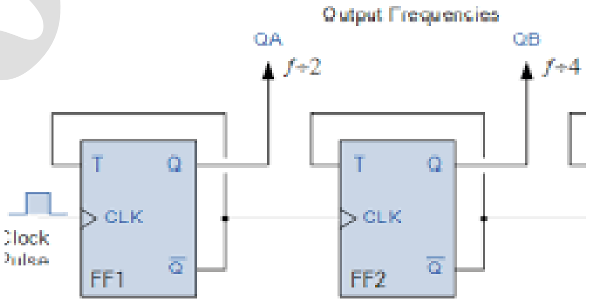
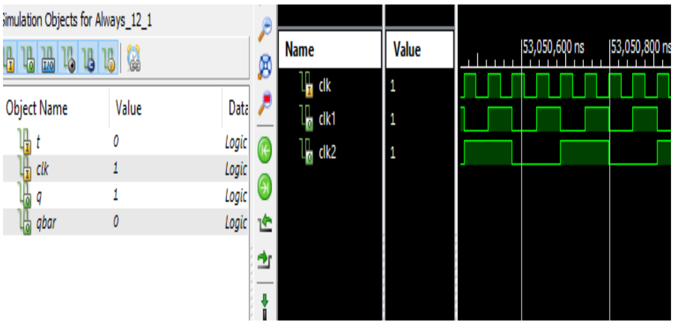


Fig 4. Block diagram of frequency division by T FF.

SIMULATION RESULT:

The clock is main frequency and the frequency divided by 2 is the clock1 and frequency divided by 4 is clock2. An every positive clock change the first FF output is the clock1 mean frequency divided by 2 and output of the second is clock2 means frequency divided by 4.



Simulation Objects for Always_12_1

Object Name	Value	Data
t	0	Logic
clk	1	Logic
q	1	Logic
qbar	0	Logic

Name	Value
clk	1
clk1	1
clk2	1

Simulation result of T FF.

Maximum output required time after clock: 4.162ns and require Number of Slice Registers=4, Number of Slice LUTs =4.

3) USING D FF:

We can done the frequency scaling using D FF . It's the simple process. Frequency is divided by 2 require only one D FF and frequency divided 4 require the Two D FF. In our project require frequency division by 2 and 4.

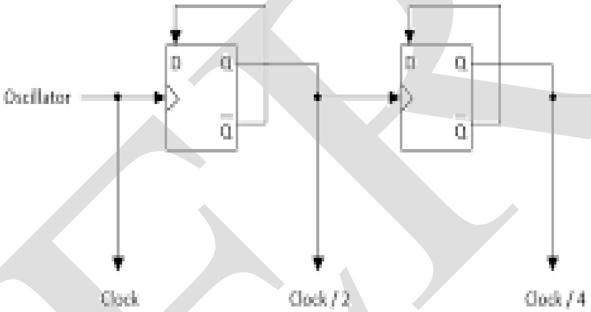
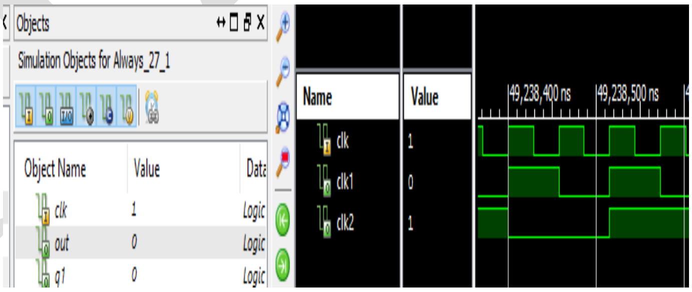


Fig5.Block diagram of frequency division by D FF.

SIMULATION RESULT:

The clock is main frequency and the frequency divided by 2 is the clock1 and frequency divided by 4 is clock2. An every positive clock change the first FF output is the clock1 mean frequency divided by 2 and output of the second is clock2 means frequency divided by 4.



Objects

Simulation Objects for Always_27_1

Object Name	Value	Data
clk	1	Logic
out	0	Logic
q1	0	Logic

Name	Value
clk	1
clk1	0
clk2	1

Simulation result of D FF.

Maximum output required time after clock: 4.202ns and require Number of Slice Registers=4, Number of Slice LUTs =2.

CONCLUSION

The frequency scaling using JKFF is better than the other methods. Frequency scaling using JK FF require only 2 registers and 3 Slice LUT and delay is 4.202 ns. But T FF require less delay as compare to JK but require 4 registers and 4 Slice LUT.Then JK FF frequency method is used in 32x32 multiplier.

REFERENCES

1) Xiaoxiao Zhang, and Amine Bermak,”*32 Bit×32 Bit Multiprecision Razor-Based DynamicVoltage Scaling Multiplier With Operands Scheduler”, IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 22, no. 4,Apr 2014.*

2) Amine Bermak, Farid Boussaid and Xiaoxiao Zhang , IEEE,”*32 Bit×32 Bit Multiprecision Razor-Based Dynamic Voltage Scaling Multiplier With Operands Scheduler” IEEE Transactions On Very Large Scale Integration (Vlsi) Systems, Vol. 22, No. 4, April 2014 759. [3] Shyamal pampattiwar “literature survey on NFC, applications” IJSER feb-2012.*

3) M. Anis, M. Elmasry, and A. Youssef, “*A comparative study between static and dynamic sleep signal generation techniques for leakage tolerant designs,” IEEE Trans. Very Large Scale Integration multiplier,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57,no. 3, pp. 568–580, Mar. 2010.*