International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 08, August 2022)

VLSI Architecture of 32 Bit Approximate Square Root for DSP-FPGA Applications

Poonam Rajpoot[1], Dr. Tarun Verma[2]

[1]*Research Scholar,* [2]*Associate Professor, Department of Electronics and Communication Engineering, Lakshmi Narain College of Technology, Bhopal, India*

Abstract**-- Variable precision fixed and floating point operations have various fields of applications including scientific computing and signal processing. Field Programmable Gate Arrays (FPGAs) are a good platform to accelerate such applications because of their flexibility. Among those operations, the square root can differ based on the algorithm implemented. More complex square root problems require some work to find and optimize fast result and improved performance. This paper proposed VLSI architecture of 32 bit approximate square root for DSP-FPGA applications. Implementation and Simulation is performed using Xilinx ISE 14.7 software with verilog language.**

Keywords**-- FPGA, Simulation, Synthesis, VLSI, DSP, Square Root, Verilog, Xilinx.**

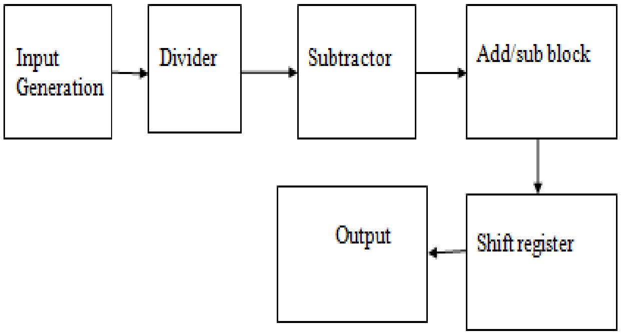
I. Introduction

Variable precision floating point operations are widely used in many fields of computer and IOT engineering. Floating point arithmetic operations are included in most processing units. There are many floating point operations including addition, subtraction, multiplication, division, reciprocal and square root. The Northeastern Reconfigurable Computing Lab has developed its own variable precision floating point library called VFLOAT, which is vender agnostic, easy to use and has a good tradeoff between hardware resources, maximum clock frequency and latency. Field Programmable Gate Arrays (FPGAs), due to their flexibility, low power consumption and short development time compared to Application Specific ICs (ASICs), are chosen as the platform for the VFloat library to run on. Very-high-speed integrated circuits Hardware Description Language (VHDL) is used to describe these components. Xilinx and Altera are the two main suppliers of programmable logic devices. Each company has its own Integrated Development Environment (IDE). Both IDE from Altera and Xilinx have been used to implement this cross platform project.

The corresponding bit widths of sign, exponent and mantissa for each format. Notice that there are many combinations not included in IEEE 754. An example is sign bit is 1, exponent is 9 bits and mantissa is 30 bits.

However using this non standard format could save some resources in a flexible technology like FPGAs and still accomplish the computing task. That is why variable precision floating point is also considered while building our floating point library.

II. Methodology



\ **Figure 1: Flow Chart**

Step-

- Firstly assign the input numbers in form of binary, decimal or hexa form.
- Now perform the implementation process, it generates the register transfer level (RTL) and technological view.
- Now simulate the results in the Xilinx test bench and run the simulation. The square root output will be generated.
- If its fixed number then square root will be also fixed and accurate and of the number is not fixed or floating then square root output will be nearest fixed number.

The methodology is based on the followings sub steps-

- Register
- Data memory block
- Square root
- Accumulator
- Control unit

New non-restoring square root algorithm that requires neither square roots nor multiplexors is presented. It generates the correct resulting value at each iteration and does not require extra circuitry for adjusting the result bit. The operation at each iteration is simple: addition or subtraction based on the result bit generated in previous iteration. The remainder of the addition or subtraction is fed via registers to the next iteration directly even it is negative. At the last iteration, if the remainder is non-negative, it is a precise remainder. Otherwise, we can obtain a precise remainder by an addition operation.

III. Simulation Results

The implementation of the proposed algorithm is done over Xilinx ISE 14.7. The ISE package processing toolbox helps us to use the functions available in Xilinx Library.

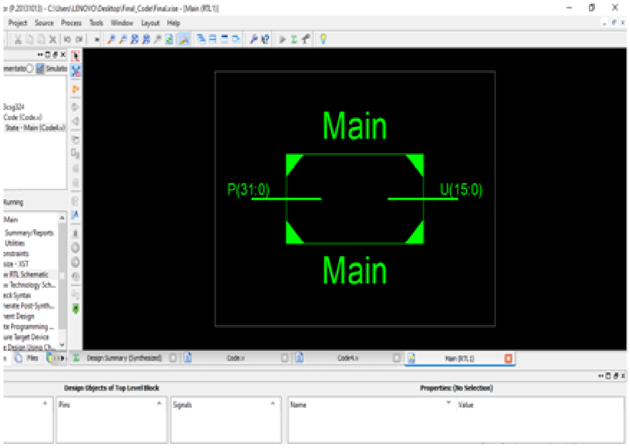


Figure 2: RTL View of Top module

Figure 2 presents the top level view of the proposed square root VLSI implementation. P stands for the input which is 32 bit and U stands for the output which is 16 bit.

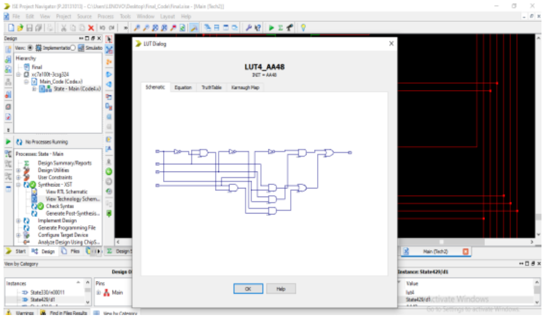
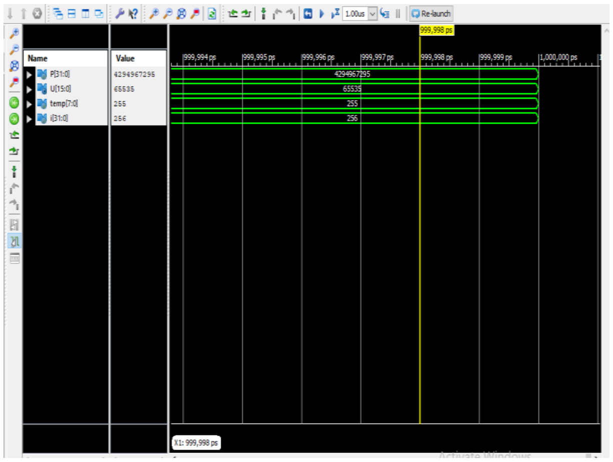


Figure 3: Look up table 4

Figure 3 is showing the look up table LUT4_AA48. The logical function in various combinations is carried out by the chip using the Lookup Table. Any combinatorial logic function can be implemented in a lookup table.

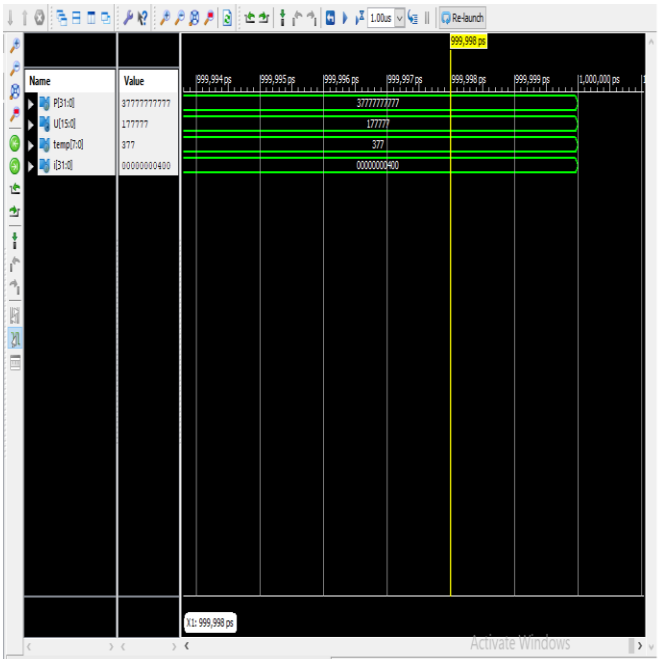


Name	Value
P[31:0]	4294967295
U[15:0]	65535
temp[7:0]	255
i[31:0]	256

Figure 4: Test bench results-3

Figure 4 shows numbers in the decimal form for the square root calculation.

Input (P) = 4294967295 (not fixed number)
Output (U) = 65535 (nearest square root)



Name	Value
P[31:0]	37777777777
U[15:0]	177777
temp[7:0]	377
i[31:0]	00000000400

Figure 5: Test bench results-4

Figure 5 shows input number in octal form.

Input = 37777777777
Output= 177777

Table 1:
Comparison of Simulation Results

Sr No.	Parameters	Previous Result [1]	Proposed Result
1	Order	16 bit square root	32 bit square root
2	Area	536	245 or 2.66 %
3	Delay	3.69 ns	3.01 ns
4	Power	0.87 mw	0.45 mw
5	Frequency	107.50 MHz	293MHz

IV. Conclusion

This paper proposes a new non-restoring square root algorithm that requires neither square roots nor multiplexors. Compared with previous square root algorithms, our algorithm is very efficient for VLSI implementation. The proposed square root is implemented for the 32 bit square root while previous it is designed for the 16 bit. The total area is optimized 245 number of component or 2.66% while previous it is 536. The delay is 3.01ns while previous it is 3.69ns.

REFERENCES

[1] N. Arya, T. Soni, M. Pattanaik and G. K. Sharma, "Area and Energy Efficient Approximate Square Rooters for Error Resilient Applications," 2020 33rd International Conference on VLSI Design and 2020 19th International Conference on Embedded Systems (VLSID), 2020, pp. 90-95, doi: 10.1109/VLSID49098.2020.00033.

[2] R. Nayar, P. Balasubramanian and D. L. Maskell, "Hardware Optimized Approximate Adder with Normal Error Distribution," 2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2020, pp. 84-89, doi: 10.1109/ISVLSI49217.2020.00025.

[3] Y. Fu, L. Li, Y. Liao, X. Wang, Y. Shi and D. Wang, "A 32-GHz Nested-PLL-Based FMCW Modulator With 2.16-GHz Bandwidth in a 65-nm CMOS Process," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 28, no. 7, pp. 1600-1609, July 2020, doi: 10.1109/TVLSI.2020.2992123.

[4] T. Fujibayashi and Y. Takeda, "A 76- to 81-GHz, 0.6° rms Phase Error Multi-channel Transmitter with a Novel Phase Detector and Compensation Technique," 2019 Symposium on VLSI Circuits, 2019, pp. C16-C17, doi: 10.23919/VLSIC.2019.8778158.

[5] S. U. Rehman, M. M. Khafaji, C. Carta and F. Ellinger, "A 10-Gb/s 20-ps Delay-Range Digitally Controlled Differential Delay Element in 45-nm SOI CMOS," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 27, no. 5, pp. 1233-1237, May 2019, doi: 10.1109/TVLSI.2019.2894736.

[6] S. Yang, J. Yin, P. Mak and R. P. Martins, "A 0.0056-mm2 −249-dB-FoM All-Digital MDLL Using a Block-Sharing Offset-Free Frequency-Tracking Loop and Dual Multiplexed-Ring VCOs," in IEEE Journal of Solid-State Circuits, vol. 54, no. 1, pp. 88-98, Jan. 2019, doi: 10.1109/JSSC.2018.2870551.

[7] J. -H. Hsieh, K. -C. Hung, Y. -L. Lin and M. -J. Shih, "A Speed- and Power-Efficient SPIHT Design for Wearable Quality-On-Demand ECG Applications," in IEEE Journal of Biomedical and Health Informatics, vol. 22, no. 5, pp. 1456-1465, Sept. 2018, doi: 10.1109/JBHI.2017.2773097.

[8] H. Fuketa, S. -i. O'uchi and T. Matsukawa, "A Closed-Form Expression for Minimum Operating Voltage of CMOS D Flip-Flop," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 25, no. 7, pp. 2007-2016, July 2017, doi: 10.1109/TVLSI.2017.2677978.

[9] J. J. Pimentel, B. Bohnenstiehl and B. M. Baas, "Hybrid Hardware/Software Floating-Point Implementations for Optimized Area and Throughput Tradeoffs," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 25, no. 1, pp. 100-113, Jan. 2017, doi: 10.1109/TVLSI.2016.2580142.

[10] T. Lee and P. A. Abshire, "Frequency-Boost Jitter Reduction for Voltage-Controlled Ring Oscillators," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 24, no. 10, pp. 3156-3168, Oct. 2016, doi: 10.1109/TVLSI.2016.2541718.

[11] Z. Yan, G. He, Y. Ren, W. He, J. Jiang and Z. Mao, "Design and Implementation of Flexible Dual-Mode Soft-Output MIMO Detector With Channel Preprocessing," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 62, no. 11, pp. 2706-2717, Nov. 2015, doi: 10.1109/TCSI.2015.2479055.

[12] K. Chen and Y. H. Kim, "Current source model of combinational logic gates for accurate gate-level circuit analysis and timing analysis," VLSI Design, Automation and Test (VLSI-DAT), 2015, pp. 1-4, doi: 10.1109/VLSI-DAT.2015.7114529.