Abstract: Square root calculation is one of the most useful and vital operations in digital signal processing, the operation which in recent generations of processors is performed by the hardware. The hardware implementation of the square root operation can be achieved by different means, but it is very dependent on programmer's sense and ability to write efficient hardware designs. This paper offers universal and shortest verilog coding of modified non-restoring square root calculator. The main principle of the method is similar with conventional non-restoring algorithm, but it only uses subtract operation and append 01, while add operation and append 11 is not used. The strategy has been conducted to implement it successfully in FPGA hardware, and offer an efficient in hardware resource, and it is superior.

Keywords: FPGA, Non-Restoring Algorithm, Pipelined Architecture, Square Root Calculation.

I. INTRODUCTION

In many VLSI applications, it is an urgent requirement to provide the computation of square root of a binary coded number with low power dissipation and fast computation (low delay propagation). Square root calculation is one of the most useful and vital operations in computer graphics and scientific calculation applications, such as digital signal processing (DSP) algorithms, math coprocessor, data processing and control, and even multimedia applications. It is a classical problem in computational number theory, which is oftenly encountered and which is a hard task to get an exact result. Any square root calculation techniques have been proposed, such as Rough estimation, Babylonian method, exponential identity, Taylor-series expansion algorithm, Newton-Raphson method, Sweeney Robertson Tocher redundant and non redundant method, restoring and non-restoring algorithm (digit-by-digit method). However, the early processors carry out the square root operation of the algorithms above by software means, which have long delays for its completion. With the rapid advancement of technology which allows the integration of large circuits on a single chip and the increase in demand for faster computational execution time, the hardware realization of square root became more attractive.

Unfortunately because of the complexity of the square root algorithms, the square root calculation is not easy to be implemented on Field Programmable Gate Array (FPGA) technology. There are some algorithms of the square root computation which are already implemented on FPGA. They are generally grouped into two distinct categories. The first category is called estimation methods, which includes algorithms such as Rough estimation and Newton-Raphson method (and also its derivations: CORDIC, DeLugish's and Chen's), whereby the second category is called digit-by-digit method. The restoring algorithm has a big limitation at restoring step in the regular flow. Primarily for this reason, although initially having led the way for all the other methods, it has been declined in importance and nowadays it is no longer used.

The non restoring algorithm does not restore the remainder, which can be implemented with least hardware resource usage. It is the most suitable for FPGA implementation and allows for IEEE standard rounding to be readily implemented. Any strategies or architectures have conducted to implement the non restoring digit-by-digit square root algorithm in FPGA hardware. Yamin and Wanming have introduced a non restoring algorithm with fully pipelined and iterative version that requires neither multipliers nor multiplexers. They introduced the carry save adder (CSA) and the carry propagate adder (CPA) as basic building blocks. Although the algorithms have a good processing speed, they consume too many hardware resources as a trade-off, while the algorithms has low computation speed, despite costing less resource usage. The similar architectures have been introduced by Xiaoliang, Thakkar and Xiumin et al. In the other study, Samawi et al have introduced controlled add-sub (CAS) as basic building blocks.
The effort is done to reduce hardware consumed, with moderate delay. The other architecture which has also been proposed is a fully combinational architecture. However, FPGA is very suitable to adopt fully pipelined architecture because of the characteristics of its structure. Hence, very little or even needless extra cost is required, if the pipeline technology is implemented in FPGA. In this paper, a strategy to implement non restoring square root algorithm based on FPGA which adopt fully pipelined architecture, will be presented. The main principle of the method is only uses subtract operation and append 01 which is implemented in register transfer level (RTL) abstraction, but add operation and append 11 are not used. In the proposed strategy will needs fewer pipeline stages compared with the proposed algorithm. Next, the performance of the developed design will be compared to the one developed by Samawi et al.

II. MODIFIED NON-RESTORING SQUARE ROOT ALGORITHM

Samavi, et al has improved classical non-restoring digit-by-digit square root circuit by eliminating redundant blocks which still based on constant binary digit of 01 or 11 and adder-subtractor as the main building block. This paper offers a simple strategy while only uses subtract operation and appends 01. This strategy is implemented by Verilog programming at RTL abstraction. A hardware implementation of the non-restoring digit-by-digit algorithm for 6-bit unsigned square root by an array structure is shown in Figure 1.

![Figure 1. A simple hardware implementation of the non-restoring digit-by-digit algorithm for unsigned 6-bit square root.](image)

In platform of register transfer level (RTL) abstraction, the above can be described as follow:

- **Step 0**: Start
- **Step 1**: Initialization of the radicand (the n-bit number will be squared root), the quotient (the result of squared root), and the remainder. To calculate the square root of a 2n-bit number, it needs n stage pipelines to implement the proposed algorithm.
- **Step 3**: At the binary point, divide the radicand into groups of two digits in both direction (integer and fractional).
- **Step 4**: Beginning on the left (the most significant bit), select the first group of one or two digit (If n is odd then the first groups is one digit, and vice versa) (the 2n-bit radicand is always even?)
- **Step 5**: Choose 1 squared, and then subtract. First developed root is “1” if the result of subtract is positive, else “0”
- **Step 6**: Shift two bits, subtract guess squared with append 01. Nth-bit squared is “1” if the result of subtract is positive, and because of subtract operation is done else Nth-bit squared is “0”, and not subtraction has been performed
- **Step 7**: Repeat Step 5 until end group of two digits

III. PROPOSED VERILOG CODING

This paper proposes a universal and shortest Verilog coding of modified non-restoring square root calculator as shown below. In fact, this RTL code is easy-to-use and parameterizable, since the input radicand size can be modified by only setting the appropriate n value.

```verilog
module sqrt #(parameter DATA_WIDTH= 8, ANSWER_WIDTH= 8, TRIAL_WIDTH = 4 )
(clk, data, start, answer, done);
input clk; input wire[DATA_WIDTH-1:0] data; output
```

Figure 2. Internal structure of a CSM Block.

![Figure 2. Internal structure of a CSM Block.](image)
Design and Implementation of Modified Non-Restoring Square Root Calculator Based on Verilog HDL

In the previous sections, the hardware implementation of the non-restoring digit-by-digit algorithm for square root is described. The first observation is conducted to validate the output of the square root calculation in simulation, which has been performed by using ModelSim-Altera. Then, the results achieved are reproduced in the hardware test, by observing the ADC system output. The simulation and experiment result are shown in Figure 5.a and 5.b, respectively.

Simulation and hardware experiments have been conducted to validate the Verilog code. The code is implemented and evaluated based on Altera DE2 FPGA, as shown in Figure 4. To observe the output calculation of the square root, they are connected to an 8-bit ADC system.

IV. RESULTS AND ANALYSIS

In many VLSI applications, it is an urgent requirement to provide the computation of square root. The operation is one of the most useful and vital operations in digital signal processing. This paper has presented novel strategy of the FPGA implementation of non restoring square root calculator. It has provided a universal and shortest VHDL coding of modified non-restoring square root calculator, and offers an efficient in hardware resource, and it is superior.

VI. REFERENCES

International Journal of Scientific Engineering and Technology Research
Volume.03, IssueNo.09, May-2014, Pages: 1694-1697


Author's Profile:

ANURUDDH SINGH, He has received the B.E. degree in Electronics and Communication Engineering from R.E.C. Rewa affiliated to R.G.P.V Bhopal, India in 2011. He is currently pursuing M.Tech degree in Microelectronics and VLSI Design from S.G.S.I.T.S. Indore.

MANJEET SINGH, He has 3 years of Teaching experience and presently pursuing M.Tech in Micro Electronics & VLSI Design in (2012-2014) SGSITS Indore, India He received his Bachelor of Engineering in Electronics and Communication from UIT-RGPV Bhopal, India in 2008. His main research interests include Low Power VLSI design, Digital design.

D.S.AJNAR, He has received the B.E. degree in Electronics and Communication Engineering from S.G.S.I.T.S. affiliated to D.A.V.V.University(Formerly known as University Of Indore), Indore, India in 1993 and M.E. Degree in Digital Techniques & Instrumentation Engineering from Rajiv Gandhi Technical University Bhopal, India in 2000. He has been in teaching and Research Profession since 1995. He is now working as Associate Professor in Department of Electronics & Instrumentation Engineering, S.G.S.I.T.S, Indore, India. His interest of research is in designing of analog filter and Current conveyer.

P.K.JAIN, He has received the B.E. degree in Electronics and Communication Engineering from D.A.V.V. University (Formerly known as University Of Indore), Indore, India in 1987 and M.E. Degree in Digital Techniques & Instrumentation Engineering from D.A.V.V. University, Indore, India in 1993. He has been in teaching and Research Profession since 1988. He is now working as Associate Professor in Department of Electronics & Instrumentation Engineering, S.G. S.I.T.S., Indore. He has also worked as a computer Engineer. His interest of research is in Analog and digital system design.