Reliable Low Power Multiplier Design Using Fixed Width Replica Redundancy Block

P.Tharani[1], R.Padmaja[2]
[1]*M.Tech., PG Scholar, Gouthami Institute of Technology & Management For Women, Proddatur*
[2]*Assistant Professor, Gouthami Institute of Technology & Management For Women, Proddatur*

Abstract- **In this paper, we propose a reliable low-power multiplier design by adopting algorithmic noise tolerant (ANT) architecture with the fixed-width multiplier to build the reduced precision replica redundancy block (RPR). The proposed ANT architecture can meet the demand of high precision, low power consumption, and area efficiency. We design the fixed-width RPR with error compensation circuit via analyzing of probability and statistics. Using the partial product terms of input correction vector and minor input correction vector to lower the truncation errors, the hardware complexity of error compensation circuit can be simplified. In a 12 × 12 bit ANT multiplier, circuit area in our fixed-width RPR can be lowered by 44.55% and power consumption in our ANT design can be saved by 23% as compared with the state-of-art ANT design.**

I. INTRODUCTION

The fast growth of portable and wireless computing systems in recent years drives the need for ultralow power systems. Low power consumption and smaller area are some of the most important criteria in the DSP systems and high performance systems. The low power technique is the voltage over scaling (VOS), was proposed in lower supply voltage beyond critical supply voltage without sacrificing the throughput. A novel algorithmic noise tolerant (ANT) technique combined VOS main block with reduced precision replica (RPR), which combats soft errors effectively while achieving significant energy saving. VOS increase the delay in all paths of a system and may limit high performance required by today complex applications. ANT is the combined VOS block and RPR block, the error is occur. It is a very fast manner but hardware complexity is too difficult.

The Rapid growth of portable and wireless computing systems in recent years drives the need for ultralow power systems. To lower the power dissipation, supply voltage scaling is widely used as an effective low-power technique since the power consumption in CMOS circuits is proportional to the square of supply voltage. However, in deep-sub micrometer process technologies, noise interference problems have raised difficulty to design the reliable and efficient microelectronics systems. hence the design techniques to enhance noise tolerance have been widely developed. An aggressive low-power technique, referred to as voltage over scaling (VOS), was proposed to lower supply voltage beyond critical supply voltage without sacrificing the throughput. However, VOS leads to severe degradation in signal-to-noise ratio (SNR). A novel algorithmic noise tolerant (ANT) technique combined VOS main block with reduced-precision replica (RPR), which combats soft errors effectively while achieving significant energy saving. Some ANT deformation designs are presented and the ANT design concept is further extended to system level. However, the RPR designs in the ANT designs are designed in a customized manner, which are not easily adopted and repeated. The RPR designs in the ANT designs can operate in a very fast manner, but their hardware complexity is too complex. As a result, the RPR design in the ANT design of is still the most popular design because of its simplicity. However, adopting with RPR in should still pay extra area overhead and power consumption. In this paper, we further proposed an easy way using the fixed-width RPR to replace the full-width RPR block in . Using the fixed-width RPR, the computation error can be corrected with lower power consumption and lower area overhead. We take use of probability, statistics, and partial product weight analysis to find

the approximate compensation vector for a more precise RPR design. In order not to increase the critical path delay, we restrict the compensation circuit in RPR must not be located in the critical path. As a result, we can realize the ANT design with smaller circuit area, lower power consumption, and lower critical supply voltage.

Multipliers are key components of many high performance systems such as FI R filters, micro processors, digital signal processors, etc. A system's performance is generally determined by the performance of the multiplier because the multiplier is generally the slowest element in the system.

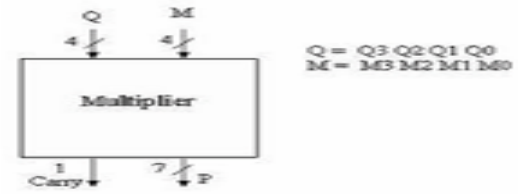
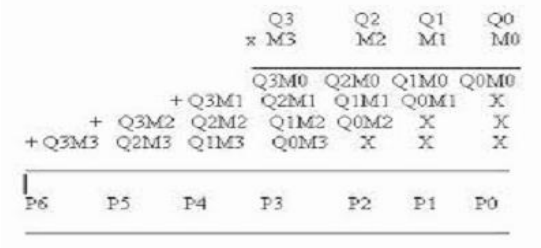


Fig: 1.*Multiplier*

The basic algorithm for multiplication of two binary numbers, M (multiplier) and N (multiplicand), makes use of the property of multiplication numbers.



II. PRELIMINARIES

The fixed-width designs adopted in DSP applications to avoid infinite growth of bit width. Cutting off n-bit LSB output is used to construct a fixed-width DSP with n-bit input and n-bit output. The circuit complexity and consumption of power of a fixed-width DSP is usually about half of the full length one. However, cutting of LSB part will result in rounding error, which should be dealt separately and compensated precisely. Many literatures have been presented to decrease the truncation error with constant correction value or with variable correction value.

The hardware complexity to compensate with constant correction value will be much more simpler than that of variable correction value; but the variable correction method are usually more accurate usually compensation method is to compensate the truncation error between the full-length multiplier and the fixed-width multiplier. In fixed-width RPR of an ANT multiplier, the compensation error we need to correct is the overall truncation error of MDSP block. Unlike , our compensation method is to compensate the truncation error between the full-length Main DSP multiplier and the fixed-width RPR multiplier. However nowadays, there are many fixed-width multiplier structures applied to the full-width multipliers. However, there is still no fixed-width RPR design applied to the ANT multiplier designs.

A.*FPGA Implementation of Low Power And Area efficient Multiplier Design Using Fixed-Width RPR:*

In this paper, we propose a new compensation method to reduce the error of fixed width multiplier for DSP applications. The input number based compensation method is carried out on array multiplier. The proposed architecture not only achieves low power consumption and area efficiency also involves in efficient precision reduction. By using the partial product terms of input correction vector and minor input correction vector, the truncation error is lowered and the hardware complexity of the fixed-width multiplier is simplified. In the 32 x 32 bit ANT multiplier, the circuit area in the proposed fixed-width RPR is reduced as compared with full width RPR design.

B. *Design for Low Power Multiplier Based On Fixed Width Replica Redundancy Block & Compressor Trees:*

This paper establishes designing multipliers that are of high-speed, low power, and regular in layout are of substantial research interest. Multiplier speed can be increased by reducing the generated partial products. Many attempts are done to reduce the number of partial products generated in a multiplication process. One of them is Wallace tree multiplier. Wallace Tree CSA structures are used to sum the partial products in reduced time. Speed can be increased by incorporating compressors with Wallace tree technique . Therefore, minimizing the number of half adders used in a multiplier which will reduce the circuit complexity.

C. Low Power Truncated Binary Multiplier Using Replica Redundancy Block:

A reliable low-power multiplier design by adopting algorithmic noise tolerant (ANT) architecture with truncated binary multiplier to build the fixed width reduced precision replica redundancy block (RPR). The ANT architecture can meet the high speed, low power, and area efficiency. To design the fixed-width RPR with error compensation circuit via analyzing of probability and statistics. Using this partial product terms of input correction vector and minor input correction vector to lower the truncation errors, the hardware complexity of error compensation circuit can be simplified.

D. *Reliable Low Power Multiplier Design Using Reduced Precision Redundancy Block:*

We develop a new method for designing a reliable low power multiplier by using algorithmic noise tolerant technique. To propose the fixed width RPR to replace the full width RPR block in the ANT design. RPR is a reduced precision replica whose output is taken as the corrected output in case the original system computes error. It reduces the truncation error and then construct a lower error fixed width Wallace tree multiplier. It is efficient for VLSI Implementation. The ANT technique having high accurate, low power consumption and area efficiency. To design the fixed width RPR by using the partial product terms of input correction vector and minor input correction vector to reduce the errors.

1) *Fixed Width Multiplier:*

The fixed-width multipliers have been widel y used in the design of di gital signal processor(DSP) due to their smaller area and lower power dissipation. I n order to reduce the chip area of channel detector for cognitive radio, many fixed width Booth multipliers have been used. However, they reduce the detection accuracy because of truncated partial products. This method can reduce the truncated error by using variable compensation value. The third categor y is hybrid error compensation, which uses both constant and adaptive QEC techniques together to reduce the truncated error. I n order to overcome the disadvantages of has presented a method of dividing the truncated partial products into the major truncated section and the minor truncated section.

2) *Algorithmic Noise Tolerance:*

The motivation is to reduce power of the traditional methods for noise tolerance. ANT can be mainly divided into Prediction based ANT and Reduced Precision redundancy based ANT. Using ANT technique to improve the performance of DSP algorithms in presence of bit error rates. Therefore ANT can produce more effective signals. ANT to compensate for degradation in the system output due to errors from soft computations.

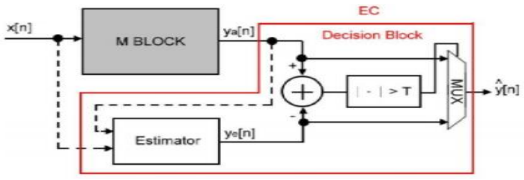


Figure2. *Algorithmic noise tolerance*

3) *Reduced Precision Redundancy:*

The MDSP block is subject to VOS, which results in soft errors in its output . When a soft error in MDSP is detected using an error control (EC) block, the RPR output is used as an output . Next, we describe the error characteristics of a system under VOS and then present the proposed error control algorithm.

4) *Soft Error Characteristics:*

Voltage overscaling introduces input dependent soft errors whenever a path with delay greater than the sample period is excited. Since the arithmetic units employed in DSP systems are based on least significant bit (LSB) first computation, soft errors appear first in the most significant bits (MSBs), resulting in errors of large magnitude. These errors severely degrade the performance but are desirable because they are easy to detect. This fraction depends upon the delay distribution of a system, which in turn depends on the architecture. The path delay distribution for all possible input combinations of an 8× 8 Baugh Wooley multiplier.

5) *ANT Multiplier design Using Fixed-Width RPR:*

We further proposed the fixed-width RPR to replace the full-width RPR block in the ANT design, as shown in Fig. 2, which can not only provide higher computation precision, lower power consumption, and lower area overhead in RPR, but also perform with higher SNR, more area efficient, lower operating supply voltage, and lower power consumption in realizing the ANT architecture. We demonstrate our fixed-width RPR-based ANT design

in an ANT multiplier. The fixed-width designs are usually applied in DSP applications to avoid infinite growth of bit width. Cutting off n-bit least significant bit (LSB) output is a popular solution to construct a fixed-width DSP with n-bit input and n-bit output. The hardware complexity and power consumption of a fixed-width DSP is usually about half of the full-length one. However, truncation of LSB part results in rounding error, which needs to be compensated precisely. Many literatures have been presented to reduce the truncation error with constant correction value or with variable correction value. The circuit complexity to compensate with constant corrected value can be simpler than that of variable correction value; however, the variable correction approaches are usually more precise.

Their compensation method is to compensate the truncation error between the full-length multiplier and the fixed-width multiplier. However, in the fixed-width RPR of an ANT multiplier, the compensation error we need to correct is the overall truncation error of MDSP block. Unlike, our compensation method is to compensate the truncation error between the full-length MDSP multiplier and the fixed-width RPR multiplier. In nowadays, there are many fixed-width multiplier designs applied to the full-width multipliers. However, there is still no fixed-width RPR design applied to the ANT multiplier designs.

To achieve more precise error compensation, we compensate the truncation error with variable correction value. We construct the error compensation circuit mainly using the partial product terms with the largest weight in the least significant segment. The error compensation algorithm makes use of probability, statistics, and linear regression analysis to find the approximate compensation value. To save hardware complexity, the compensation vector in the partial product terms with the largest weight in the least significant segment is directly inject into the fixed-width RPR, which does not need extra compensation logic gates. To further lower the compensation error, we also consider the impact of truncated products with the second most significant bits on the error compensation. We propose an error compensation circuit using a simple minor input correction vector to compensation the error remained. In order not to increase the critical path delay, we locate the compensation circuit in the noncritical path of the fixed-width RPR. As compared with the full-width RPR design the proposed fixed-width RPR multiplier not only performs with higher SNR but also with lower circuitry area and lower power consumption.

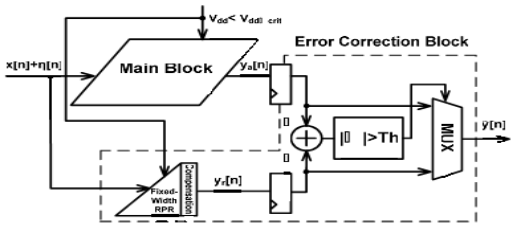


Fig. 3. *Proposed ANT architecture with fixed-width RPR*

E. *Proposed Precise Error Compensation Vector for Fixed-Width RPR Design*

In the ANT design, the function of RPR is to correct the errors occurring in the output of MDSP and maintain the SNR of whole system while lowering supply voltage. In the case of using fixed-width RPR to realize ANT architecture, we not only lower circuit area and power consumption, but also accelerate the computation speed as compared with the conventional full-length RPR. However, we need to compensate huge truncation error due to cutting off many hardware elements in the LSB part of MDSP. In the MDSP of n-bit ANT Baugh–Woolley array multiplier, its two unsigned n-bit inputs of X and Y can be expressed as

$$X = \sum_{i=0}^{n-1} x_i \cdot 2^i, \qquad Y = \sum_{j=0}^{n-1} y_j \cdot 2^j.$$

The multiplication result P is the summation of partial products of xiyj, which is expressed as

$$P = \sum_{k=0}^{2n-1} p_k \cdot 2^k = \sum_{j=0}^{n-1} \sum_{i=0}^{n-1} x_i y_j \cdot 2^{i+j}.$$

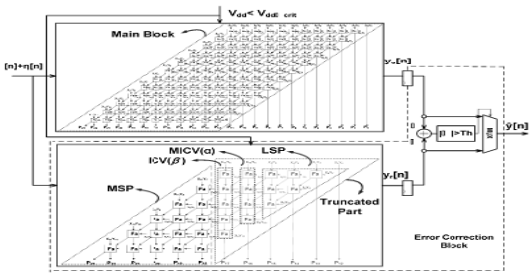


Fig. 4 *12×12 bit ANT multiplier is implemented with*

the six-bit fixed width replica redundancy block.

The (n/2)-bit unsigned full-width Baugh–Wooley partial product array can be divided into four subsets, which are most significant part (MSP), input correction vector [ICV(β)], minor ICV [MICV(α)], and LSP, as shown in Fig. 3. In the fixed width RPR, only MSP part is kept and the other parts are removed. Therefore, the other three parts of ICV(β), MICV(α), and LSP are called as truncated part. The truncated ICV(β)and MICV(α)are the most important parts because of their highest weighting. Therefore, they can be applied to construct the truncation error compensation algorithm. To evaluate the accuracy of a fixed-width RPR, we can exploit the difference between the (n/2)-bit fixed-width RPR output and the 2n-bit full-length MDSP output, which is expressed as

$$\varepsilon = P - P_t$$

Where P is the output of the complete multiplier in MDSP and Pt is the output of the fixed-width multiplier in RPR. Pt can be expressed as

$$\begin{aligned} P_t = & \sum_{j=\frac{n}{2}+1}^{n-1} y_j 2^j \sum_{i=\frac{3n}{2}-j}^{n-1} x_i 2^i \\ & + f\left(x_{n-1}y_{\frac{n}{2}}, x_{n-2}y_{\frac{n}{2}+1}, x_{n-3}y_{\frac{n}{2}+2}, \ldots, x_{\frac{n}{2}}y_{\frac{n}{2}+2}\right) \\ & + f\left(x_{n-2}y_{\frac{n}{2}}, x_{n-3}y_{\frac{n}{2}+1}, x_{n-4}y_{\frac{n}{2}+2}, \ldots, x_{\frac{n}{2}}y_{n-2}\right) \\ = & \sum_{j=\frac{n}{2}+1}^{n-1} y_j 2^j \sum_{i=\frac{3n}{2}-j}^{n-1} x_i 2^i + f(\mathrm{ICV}) + f(\mathrm{MICV}) \\ = & \sum_{j=\frac{n}{2}+1}^{n-1} y_j 2^j \sum_{i=\frac{3n}{2}-j}^{n-1} x_i 2^i + f(\mathrm{EC}) \end{aligned}$$

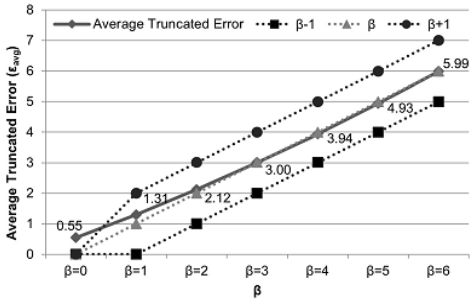


Fig.5. *Statistical curves of average truncation error in the LSP block and the curves of compensation function with β−1,β,andβ+1 in the 12-bit fixed-width RPR-based ANT multiplier.*

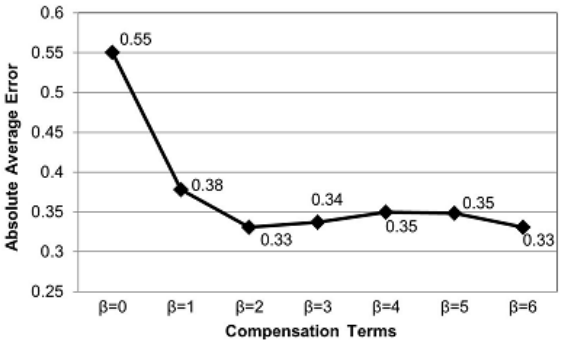


Fig. 6. *Analysis of absolute average compensation error under various βvalues in the 12-bit fixed-width RPR-based ANT multiplier.*

Where f(EC) is the error compensation function, f(ICV) is the error compensation function contributed by the input correction vector ICV(β),andf(MICV)is the error compensation function contributed by minor input correction vector MICV(α). The source of errors generated in the fixed-width RPR is dominated by the bit products of ICV since they have the largest weight. It is reported that a low-cost EC circuit can be designed easily if a simple relationship between f(EC) and β is found. It is noted that β is the summation of all partial products of ICV. By statistically analyzing the truncated difference between MDSP and fixed-width RPR with uniform input distribution, we can find the relationship between f(EC) andβ. As shown in Fig. 4, the statistical results show that the average truncation error in the fixed-width RPR multiplier is approximately distributed betweenβandβ+1. More precisely, as β=0, the average truncation error is close toβ+1. As β>0, the average truncation error is very close to β. If we can select β as the compensation vector, the compensation vector can directly inject into the fixed-width RPR as compensation, which does not need extra compensation logic gates.

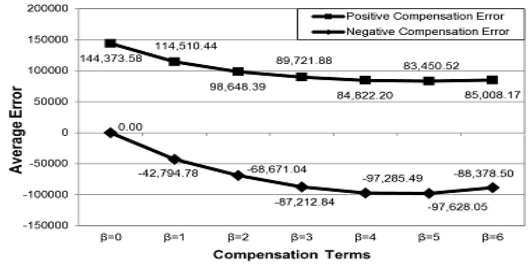


Fig. 7. *Analysis of average positive and negative compensation error under various β values in the 12-bit fixed-width RPR-based ANT multiplier.*

We go further to analyze the compensation precision by selecting β as the compensation vector. We can find that the absolute average error in β =0 is much larger than that in other β cases, as shown in Fig. 5. Moreover, the absolute average error in β =0 is larger than 0.5 ∗2(3n/2),while the absolute average error in other β situations is smaller than 0.5∗2(3n/2). Therefore, we can apply multiple input error compensation vectors to further enhance the error compensation precision. For theβ>0 case, we can still select βas the compensation vector. For the β=0 case, we select β+1 combining with MICV as the compensation vector. Before directly injecting the compensation vector β into the fixed-width RPR, we go further to double check the weight for the partial product terms in ICV with the same partial product summation value β but with different locations. As shown in Table I

TABLE I
RELATION BETWEEN THE PARTIAL PRODUCT TERM'S LOCATION IN ICV AND THE STATISTICAL COMPENSATION ERROR E_{AVG}

Row	*ICV*	ε_{avg}	*F(ICV)*
1	(1,0,0,0,0,0)	1.328	1
2	(0,1,0,0,0,0)	1.303	1
3	(0,0,1,0,0,0)	1.293	1
4	(0,0,0,1,0,0)	1.293	1
5	(0,0,0,0,1,0)	1.303	1
6	(0,0,0,0,0,1)	1.328	1

The average error value for each ICV vector with the same partial product term summation value is nearly the same even their partial product term's location is different. That is to say that no matter ICV=(1,0,0,0,0,0), ICV=(0,1,0,0,0,0),ICV =(0,0,1,0,0,0),ICV=(0,0,0,1,0,0),ICV= (0,0,0,0,1,0),orICV=(0,0,0,0,0,1), their weight in each partial product term for truncation error compensation is nearly the same. Therefore, we apply the same weight of unity to each input correction vector element. This conclusion is beneficial for us to inject the compensation vector β into the fixed-width RPR directly. In this way, no extra compensation logic gates are needed for this part compensation and only wire connections are needed.

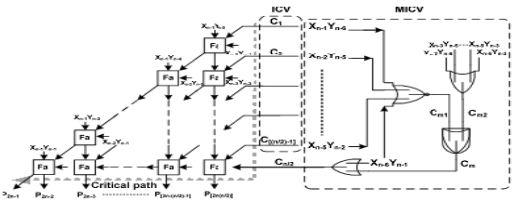


Fig. 8. *Proposed high-accuracy fixed-width RPR multiplier with compensation constructed by the multiple truncation EC vectors combined ICV together with MICV.*

F. *Proposed Precise Error Compensation Vector for Fixed-Width RPR Design*

To realize the fixed-width RPR, we construct one directly injecting ICV(β)to basically meet the statistic distribution and one minor compensation vector MICV(α) to amend the insufficient error compensation cases. The compensation vector ICV(β)is realized by directly injecting the partial terms ofXn−1Yn/2,Xn−2Y(n/2)+1, Xn−3Y(n/2)+2,...,X(n/2)+2Yn−2. These directly injecting compensation terms are labeled as C1,C2,C3,...,C(n/2)−1 in Fig. 9. The other compensation vector used to mend the insufficient error compensation case is constructed by one conditional controlledORgate. One input of ORgate is injected by X(n/2)Yn−1, which is designed to realize the function of compensation vector β. The other input is conditional controlled by the judgment formula used to judge whether β=0andβ1 =0 as well. As shown in Fig. 8, the term Cm1 is used to judge whetherβ=0 or not. The judgment function is realized by oneNORgate, while its inputs are Xn−1Yn/2, Xn−2Y(n/2)+1, Xn−3Y(n/2)+2,...,X(n/2)+2Yn−2. The termCm2is used to judge whether β1 =0. The judgment function is realized by one ORgate, while its inputs are Xn−2Yn/2,Xn−3Y(n/2)+1,Xn−4Y(n/2)+2,...,X(n/2)+ 1Yn−2.If both of these two judgments are true, a compensation term Cm is generated via a two-input ANDgate. Then, Cm is injected together with X(n/2)Yn−1 into a two-input ORgate to correct the insufficient error compensation. Accordingly, in the case of β =0andβ1 =0 as well, one additional carry-in signal C(n/2) is injected into the compensation vector to modify the compensation value asβ+1 instead of β. Moreover, the carry-in signal C(n/2) is injected in the bottom of error compensation vector, which is the farthest location away from the critical path. Therefore, not only the error compensation precision in the fixed-width RPR can be enhanced, the computation delay will also not be postponed. Since the critical supply voltage is dominated by the critical delay time of the RPR circuit, preserving the critical path of RPR not be postponed is very important. Finally, the proposed high-precision fixed-width RPR multiplier circuit is shown in Fig. 9. In our presented fixed-width RPR design, the adder cells can be saved by half as compared with the conventional full-width

RPR. Moreover, the proposed high-precision fixed-width RPR design can even provide higher precision as compared with the full-width RPR design

III. IMPORTANCE OF HDLS

HDLs have many advantages compared to traditional schematic-based design.

1) Design can be described at a very abstract level by us of HDLs.Designers can write their RTL description without choosing a specific fabrication technology. Logic synthesis tools can automatically convert the design to any fabrication technology. If a new technology emerges, designers do not need to redesign their circuit. They simply input the RTL description to the logic synthesis tool and create a new gate level netlist,using the new fabrication technology. The logic synthesis tool will optimize the circuit in area and timing for the new technology.
2) By describing designs in HDLs,functional verification of the design can be done early in the design cycle. Since designers work at the RTL level, they can optimize and modify the RTL description until it meets the desired functionality. Most design bugs are eliminated at this point. This cuts down design cycle time significantly because the probability of hitting a functional bug at a later time in the gate-level netlist or physical layout is minimized.
3) Designing with HDLs is analogous to computer programming. A textual description with comments is an easier way to develop and debug circuits. This also provides a concise representation of the design, compared to gate-level schematics. Gate-level schematics are almost incomprehensible for very complex designs.
4) HDL-based designs are here to stay. With rapidly increasing complexities of digital circuits and increasingly sophisticated EDA tools,HDLs are now the dominant method for large digital designs. No digital circuit designer can afford to ignore HDL based design.

   Verilog HDL has evolved as a standard hardware description language. Verilog HDL offers many useful features
5) Verilog HDL is a general-purpose hardware description language that is easy to learn and easy to use. It is similar in syntax to the C programming language. Designers with C programming experience will find it easy to learn Verilog HDL.
6) Verilog HDL allows different levels of abstraction to be mixed in the same model. Thus, a designer can define a hardware model in terms of switches, gates, RTL, or behavioral code. Also, a designer needs to learn only one language for stimulus and hierarchical design.
7) Most popular logic synthesis tools support Verilog HDL. This makes it the language of choice for designers.
8) All fabrication vendors provide Verilog HDL libraries for post logic synthesis simulation. Thus, designing a chip in Verilog HDL allows the widest choice of vendors.
9) The Programming Language Interface (PLI) is a powerful feature that allows the user to write custom C code to interact with the internal data structures of Verilog. Designers can customize a Verilog HDL simulator to their needs with the PLI.
10) The speed and complexity of digital circuits have increased rapidly. Designers have responded by designing at higher levels of abstraction. Designers have to think only in terms of functionality. EDA tools take care of the implementation details. With designer assistance, EDA tools have become sophisticated enough to achieve a close-to-optimum implementation.
11) The most popular trend currently is to design in HDL at an RTL level, because logic synthesis tools can create gate-level net lists from RTL level design. Behavioral synthesis allowed engineers to design directly in terms of algorithms and the behavior of the circuit, and then use EDA tools to do the translation and optimization in each phase of the design.
12) However, behavioral synthesis did not gain widespread acceptance. Today, RTL design continues to be very popular. Verilog HDL is also being constantly enhanced to meet the needs of new verification methodologies.
13) Formal verification and assertion checking techniques have emerged. Formal verification applies formal mathematical techniques to verify the correctness of Verilog HDL descriptions and to establish equivalency between RTL and gate-level net lists. However, the need to describe a

design in Verilog HDL will not go away. Assertion checkers allow checking to be embedded in the RTL code. This is a convenient way to do checking in the most important parts of a design.

14) New verification languages have also gained rapid acceptance. These languages combine the parallelism and hardware constructs from HDLs with the object oriented nature of C++. These languages also provide support for automatic stimulus creation, checking, and coverage. However, these languages do not replace Verilog HDL. They simply boost the productivity of the verification process. Verilog HDL is still needed to describe the design.
15) For very high-speed and timing-critical circuits like microprocessors, the gate-level netlist provided by logic synthesis tools is not optimal. In such cases, designers often mix gate-level description directly into the RTL description to achieve optimum results. This practice is opposite to the high-level design paradigm, yet it is frequently used for high-speed designs because designers need to squeeze the last bit of timing out of circuits, and EDA tools sometimes prove to be insufficient to achieve the desired results.
16) Another technique that is used for system-level design is a mixed bottom-up methodology where the designers use either existing Verilog HDL modules, basic building blocks, or vendor-supplied core blocks to quickly bring up their system simulation. This is done to reduce development costs and compress design schedules. For example, consider a system that has a CPU, graphics chip, I/O chip, and a system bus.
17) The CPU designers would build the next-generation CPU themselves at an RTL level, but they would use behavioral models for the graphics chip and the I/O chip and would buy a vendor-supplied model for the system bus. Thus, the system-level simulation for the CPU could be up and running very quickly and long before the RTL descriptions for the graphics chip and the I/O chip are completed.

A. *Typical Design Flow:*

A typical design flow for designing VLSI-IC circuits show the level of design representation shaded blocks show processes in the design flow. The design flow used by designers who use HDLs. In any design, specifications are written first. Specifications describe abstractly the functionality, interface, and overall architecture of the digital circuit to be designed. At this point, the architects do not need to think about how they will implement this circuit. A behavioral description is then created to analyze the design in terms of functionality, performance, and compliance to standards, and other high-level issues. Behavioral descriptions are often written with HDLs. The behavioral description is manually converted to an RTL description in an HDL.

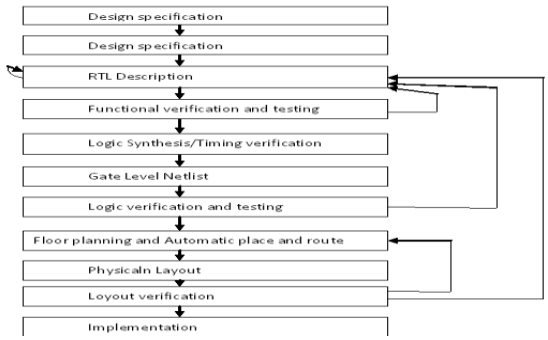


Fig *Typical Design Flow*

Logic synthesis tools convert the RTL description to a gate-level net list. A gate-level net list is a description of the circuit in terms of gates and connections between them. Logic synthesis tools ensure that the gate-level net list meets timing, area, and power specifications. The gate-level net list is input to an Automatic Place and Route tool, which creates a layout. The layout is verified and then fabricated on a chip.

Thus, most digital design activity is concentrated on manually optimizing the RTL description of the circuit. After the RTL description is frozen, EDA tools are available to assist the designer in further processes. Designing at the RTL level has shrunk the design cycle times from years to a few months. It is also possible to do many design iterations in a short period of time. Behavioral synthesis tools have begun to emerge recently. These tools can create RTL descriptions from a behavioral or algorithmic description of the circuit. As these tools mature, digital circuit design will become similar to high-level computer programming. Designers will simply implement the algorithm in an HDL at a very abstract level. EDA tools will help the designer convert the behavioral description to a final IC chip.

It is important to note that, although EDA tools are available to automate the processes and cut design cycle times, the designer is still the person who controls how the tool will perform. EDA tools are also susceptible to the "GIGO : Garbage In Garbage Out" phenomenon. If used improperly, EDA tools will lead to inefficient designs. Thus, the designer still needs to understand the nuances of design methodologies, using EDA tools to obtain an optimized design.

IV. SIMULATION TOOLS

SCHEMATIC VIEW:

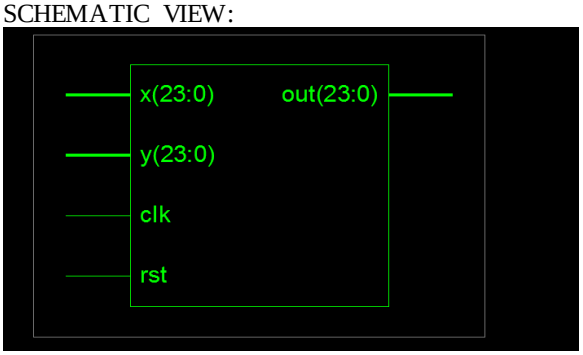


Fig 9 Schematic View

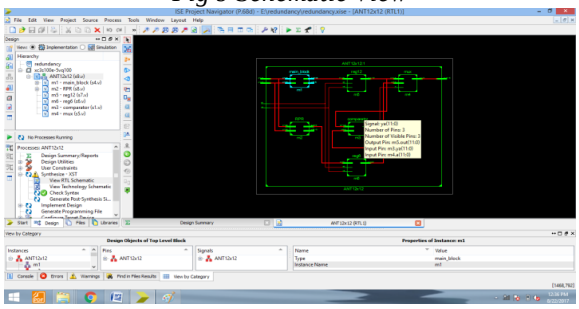


Fig 10.RTL SCHEMATIC

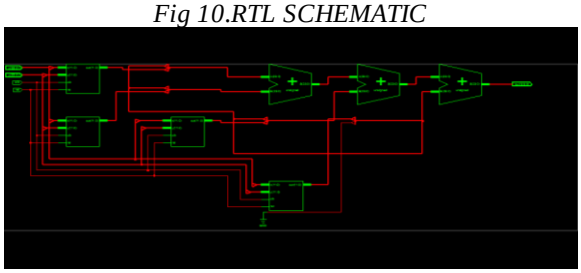


Fig 11. TECHNOLOGY SCHEMATIC

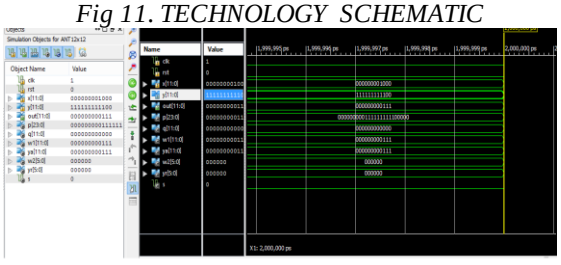
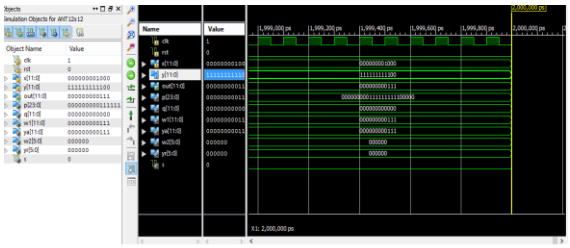


Fig 12. SIMULATION RESULTS



VI. CONCLUSION

In this paper, a low-error and area-efficient fixed-width RPR-based ANT multiplier design is presented. The proposed 12-bit ANT multiplier circuit is implemented in TSMC 90-nm process and its silicon area is 4616.5μm2. Under 0.6 V supply voltage and 200-MHz operating frequency, the power consumption is 0.393 mW. In the presented 12-bit by 12-bit ANT multiplier, the circuitry area in our fixed-width RPR can be saved by 45%, the lowest reliable operating supply voltage in our ANT design can be lowered to 0.623 VDD, and power consumption in our ANT design can be saved by 23% as compared with the state-of-art ANT design.

REFERENCE

[1] (2009). The International Technology Roadmap for Semiconductors [Online]. Available: http://public.itrs.net/

[2] B. Shim, S. Sridhara, and N. R. Shanbhag, "Reliable low-power digital signal processing via reduced precision redundancy,"IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 12, no. 5, pp. 497–510, May 2004.

[3] B. Shim and N. R. Shanbhag, "Energy-efficient soft-error tolerant digital signal processing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 14, no. 4, pp. 336–348, Apr. 2006.

[4] R. Hedge and N. R. Shanbhag, "Energy-efficient signal processing via algorithmic noise-tolerance," inProc. IEEE Int. Symp. Low Power Electron. Des., Aug. 1999, pp. 30–35.

[5] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," IEEE Trans. Comput. Added Des. Integr. Circuits Syst., vol. 32, no. 1, pp. 124–137, Jan. 2013.

[6] Y. Liu, T. Zhang, and K. K. Parhi, "Computation error analysis in digital signal processing systems with overscaled supply voltage,"IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 4, pp. 517–526, Apr. 2010.

[7] J. N. Chen, J. H. Hu, and S. Y. Li, "Low power digital signal processing scheme via stochastic logic protection," inProc. [8] J. N. Chen and J. H. Hu, "Energy-efficient digital signal processing via voltage-overscaling-based residue

number system," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 7, pp. 1322–1332, Jul. 2013.

[8] P. N. Whatmough, S. Das, D. M. Bull, and I. Darwazeh, "Circuit-level timing error tolerance for low-power DSP filters and transforms," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 6, pp. 12–18, Feb. 2012.

[9] G. Karakonstantis, D. Mohapatra, and K. Roy, "Logic and memory design based on unequal error protection for voltage-scalable, robust and adaptive DSP systems," J. Signal Process. Syst., vol. 68, no. 3, pp. 415–431, 2012.

[10] Y. Pu, J. P. de Gyvez, H. Corporaal, and Y. Ha, "An ultra low energy/frame multi-standard JPEG co-processor in 65-nm CMOS with sub/near threshold power supply,"IEEE J. Solid State Circuits, vol. 45, no. 3, pp. 668–680, Mar. 2010.

[11] H. Fuketa, K. Hirairi, T. Yasufuku, M. Takamiya, M. Nomura, H. Shinohara, et al., "12.7-times energy efficiency increase of 16-bit integer unit by power supply voltage (VDD) scaling from 1.2V to 310mV enabled by contention-less flip-flops (CLFF) and separated VDD between flip-flops and combinational logics," inProc. ISLPED, Fukuoka, Japan, Aug. 2011, pp. 163–168.

[12] Y. C. Lim, "Single-precision multiplier with reduced circuit complexity for signal processing applications,"IEEE Trans. Comput., vol. 41, no. 10, pp. 1333–1336, Oct. 1992.

[13] M. J. Schulte and E. E. Swartzlander, "Truncated multiplication with correction constant," inProc. Workshop VLSI Signal Process.,vol.6. 1993, pp. 388–396.

[14] S. S. Kidambi, F. El-Guibaly, and A. Antoniou, "Area-efficient multipliers for digital signal processing applications,"IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 43, no. 2, pp. 90–95, Feb. 1996.

[15] J. M. Jou, S. R. Kuang, and R. D. Chen, "Design of low-error fixed-width multipliers for DSP applications,"IEEE Trans. Circuits Syst., vol. 46, no. 6, pp. 836–842, Jun. 1999.

[16] S. J. Jou and H. H. Wang, "Fixed-width multiplier for DSP application," in Proc. IEEE Int. Symp. Comput. Des., Sep. 2000, pp. 318–322.

[17] F. Curticapean and J. Niittylahti, "A hardware efficient direct digital frequency synthesizer," in Proc. 8th IEEE Int. Conf. Electron., Circuits,Syst., vol. 1. Sep. 2001, pp. 51–54.

[18] A. G. M. Strollo, N. Petra, and D. D. Caro, "Dual-tree error compensation for high performance fixed-width multipliers," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 52, no. 8, pp. 501–507, Aug. 2005.

[19] S. R. Kuang and J. P. Wang, "Low-error configurable truncated multipliers for multiply-accumulate applications," Electron. Lett., vol. 42, no. 16, pp. 904–905, Aug. 2006.

[20] N. Petra, D. D. Caro, V. Garofalo, N. Napoli, and A. G. M. Strollo, "Truncated binary multipliers with variable correction and minimum mean square error,"IEEE Trans. Circuits Syst., vol. 57, no. 6,pp. 1312–1325, Jun. 2010.

[21] I. C. Wey and C. C. Wang, "Low-error and area-efficient fixedwidth multiplier by using minor input correction vector," inProc. IEEE Int. Conf. Electron. Inf. Eng., vol. 1. Kyoto, Japan, Aug. 2010, pp. 118–122.

[22] IEEE Int. Symp. Circuits Syst., May 2012, pp. 3077–3080.