Low Power Multiplier Using Algorithmic Noise Tolerant Architecture

INDIGA JYOTHI PRIYANKA
Aditya Enginnering College, Surampalem

K CHANDRA SEKHAR
M. Tech, Aditya Enginnering College, Surampalem

Abstract: **A multiplier is one of the key hardware blocks in most digital signal processing (DSP) systems. Typical DSP applications where a multiplier plays an important role include digital filtering, digital communications and spectral analysis (Ayman.A et al (2001)). Many current DSP applications are targeted at portable, battery-operated systems, so that power dissipation becomes one of the primary design constraints. Since multipliers are rather complex circuits and must typically operate at a high system clock rate, reducing the delay of a multiplier is an essential part of satisfying the overall design.**

In this project a multiplier block has been designed through the algorithmic noise tolerance architectures (ANT) by using Wallace multiplier. A reliable low power multiplier design with the fixed width multiplier block through the reduced precision replica redundancy (RPR) and main block design with Wallace multiplier . The new architecture can meet the high accuracy, low power consumption and area efficiency when compared with previous multiplier circuit.

Keywords: **Truncated Multiplier; Array Multiplier; Modified Wallace Multiplier; Multiplexer;**

I. INTRODUCTION

The rapidgrowth of portable and wireless computing systems in recent years drives the need for ultralow powersystems. To lower the power dissipation, supply voltage scalingis widely used as an effective low-power technique sincethe power consumption in CMOS circuits is proportional to thesquare of supply voltage [1]. However, in deep-submicrometerprocess technologies, noise interference problems have raiseddifficulty to design the reliable and efficient microelectronicssystems; hence, the design techniques to enhance noise tolerance have been widely developed [2]–[12].

An aggressive low-power technique, referred to as voltageoverscaling (VOS), was proposed in [4] to lower supplyvoltage beyond critical supply voltage without sacrificingthe throughput. However, VOS leads to severe degradationin signal-to-noise ratio (SNR). A novel algorithmic noisetolerant (ANT) technique [2] combined VOS main block with reduced-precision replica (RPR), which combats softerrors effectively while achieving significant energy saving.Some ANT deformation designs are presented in [5]–[9] andthe ANT design concept is further extended to system level in [10]. However, the RPR designs in the ANT designs of [5]–[7] are designed in a customized manner, which are not easily adopted and repeated. The RPR designs in the ANTdesigns of [8] and [9] 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 [2] is still the mostpopular design because of its simplicity. However, adopting with RPR in [2] should still pay extra area overhead and powerconsumption. In this paper, we further proposed an easy wayusing the fixed-width RPR to replace the full-width RPR blockin [2]. Using the fixed-width RPR, the computation error canbe corrected with lower power consumption and lower areaoverhead. We take use of probability, statistics, and partialproduct weight analysis to find the approximate compensationvector for a more precise RPR design. In order not to increasethe critical path delay, we restrict the compensation circuit inRPR must not be located in the critical path. As a result, wecan realize the ANT design with smaller circuit area, lower power consumption, and lower critical supply voltage.

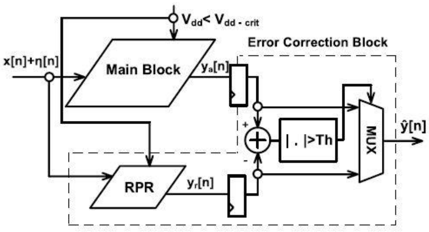


Fig. 1. ANT architecture [2].

II. ANT ARCHITECTURE DESIGNS

The ANT technique [2] includes both main digital signal processor (MDSP) and error correction (EC) block, as shown in Fig. 1. To meet ultralow power demand, VOS is used in MDSP. However, under the VOS, once the critical path delay *T*cp of the system becomes greater than thesampling period *T*samp, the soft errors will occur. It leads to severe degradation in signal precision. In the ANT technique[2], a replica of the MDSP but with reduced precisionoperands and shorter computation delay is used as ECblock. Under VOS, there are a number of input-dependentsoft errors in its output *ya*[*n*]; however, RPR output *yr* [*n*]is still correct since the critical path delay of the replicais smaller than *T*samp [4]. Therefore, *yr* [*n*] is applied to detect errors in the MDSP output *ya*[*n*]. Error detection isaccomplished by comparing the difference |*ya*[*n*] − *yr* [*n*]| against a threshold *Th*. Once the difference between *ya*[*n*] and *yr* [*n*] is larger than *Th*, the output ˆ*y*[*n*] is *yr* [*n*] instead of *ya*[*n*]. As a result, ˆ*y*[*n*] can be expressed as

$$\hat{y}[n] = \begin{cases} y_a[n], & \text{if } |y_a[n] - y_r[n]| \leq Th \\ y_r[n], & \text{if } |y_a[n] - y_r[n]| > Th. \end{cases}$$

where *yo*[*n*] is error free output signal. In this way, the power consumption can be greatly lowered while the SNR can still be maintained without severe degradation [2].

III. ANT MULTIPLIER DESIGN USING FIXED-WIDTH RPR

In this paper, we further proposed the fixed-width RPR toreplace the full-width RPR block in the ANT design [2], 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*-bitleast significant bit (LSB) output is a popular solution to constructa fixed-width DSP with *n*-bit input and *n*-bit output. Thehardware complexity and power consumption of a fixed-widthDSP 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 [13]–[22] have been presented to reduce the truncation error with constant correction value [13]–[15] or with variable correction value [16]–[22]. 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.

In [16]–[22], 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 [16]–[22], 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.

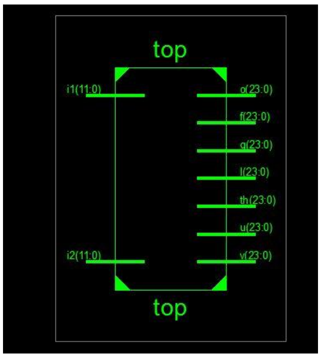
To achieve more precise error compensation, we compensate the truncation error with variable correction value. We construct the error compensation circuit mainly using the partial product terms with the largest weight in the least significant segment. The error compensation algorithm makes use of probability, statistics, and linear regression analysis to find the approximate compensation value [16]. 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 [17]. 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 in [15], the proposed fixed-width RPR multiplier not only performs with higher SNR but also with lower circuitry area and lower power consumption.

IV. ALGORITHMIC NOISE TOLERANCE (ANT)

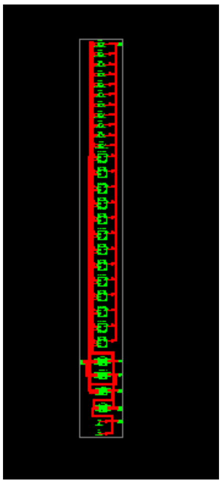
ALGORITHMIC NOISE TOLERANCE is to reduce power of the traditional methods for noise tolerance. Using ANT technique to improve the performance of DSP algorithms in presence of bit error rates. There are two blocks present in the ANT architecture. One is main digital signal processing block another one is error correction block. Error correction block contains a reduced precision.

SIMULATION RESULTS OF EXISITING:

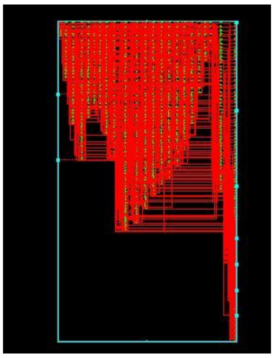
RTL SCHMATIC:



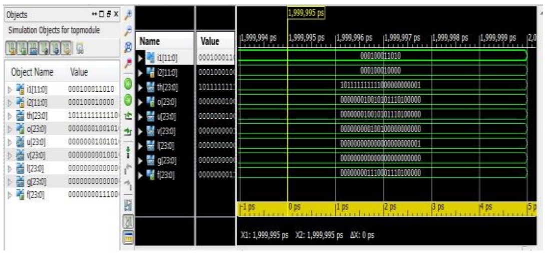
RTL INTERNAL DIAGRAM:



TECHNOLOGY SCHEMATIC:



SIMULATED WAVE FORMS:



Objects

Simulation Objects for topmodule

Object Name	Value
i1[11:0]	000100011010
i2[11:0]	000100010000
th[23:0]	1011111111110
o[23:0]	0000000100101
u[23:0]	0000000100101
v[23:0]	0000000001001
l[23:0]	0000000000000
g[23:0]	0000000000000
f[23:0]	0000000011100

Name	Value	1,999,995 ps
i1[11:0]	0001000110	000100011010
i2[11:0]	0001000100	000100010000
th[23:0]	1011111111	101111111111000000000001
o[23:0]	0000000100	000000010010101110100000
u[23:0]	0000000100	000000010010101110100000
v[23:0]	0000000001	000000000100100000000000
l[23:0]	0000000000	000000000000000000000001
g[23:0]	0000000000	000000000000000000000000
f[23:0]	0000000011	000000001110001110100000

X1: 1,999,995 ps X2: 1,999,995 ps ΔX: 0 ps

V. PROPOSED ANT ARCHITETURE WITH WALLACE MULTIPLIER

MULTIPLIER ARCHITECTURES

The composition of an array multiplier is shown in the Fig 2. There is a one to one topological correspondence between this hardware structure and the manual multiplication. The generation of n partial products requires N*M two bit AND gates. Most of the area of the multiplier is devoted to the adding of n partial products, which requires N-1, M-bit adders. The shifting of the partial products for their proper alignment's performed by simple routing and does not require any logic. The overall structure can be easily be compacted into rectangle, resulting in very efficient layout.

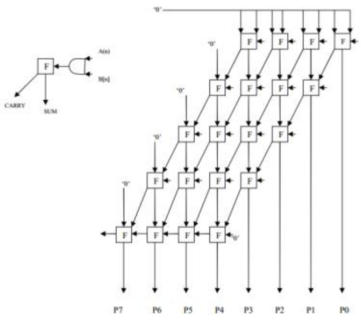


Fig: 2 Array Multiplier Architecture

Truncated multiplication is a technique which is shown in Fig 3 where only the most significant columns of the multiplication matrix are used and therefore area requirements can be reduced. In Truncation is a method where the least significant columns in the partial product matrix are not formed. The amount of columns not formed in this way, *T*, defines the degree of truncation and the *T* least significant bits of the product always result in *'0'*. The method of truncation will follows some steps in the process of multiplying of the partial product bits in the multiplier by the adders. The three steps involved in method are Deletion, Truncation, Rounding.

In truncated multiplier we start the multiplication process with deletion only.

In the partial product bits we remove the more than half of the bits,then remaining bits become the partial products in the process. This is the main criteria of deletion. Truncation is a method where the least significant columns in the partial product matrix are not formed. The amount of columns not formed in this way, T, defines the degree of truncation and the T Least Significant Bits (LSB) of the product always results in 0. The algorithm behind fixed width multiplication is the same as when dealing with non fixed width multiplication regardless of the truncation degree.

Conventionally an n-bit multiplicand and an n-bit multiplier would render a 2n-bit product. Sometimes an n-bit output is desired to reduce the number of stored bits. By the rounding process helps in the obtain of the faithfully rounded value. By these steps the truncated multiplier will gives the faithfully rounded values after truncate of the least significant part in the result. Truncated multiplication provides an efficient method for reducing the power dissipation and area of rounded parallel multiplier. The truncated multiplier is preferable as per the power related parameters, delay and area also it gives nominal results compare with the other multipliers. Truncated multiplier technique is an area reduced technique and it also it gives the low power values than the other one. With the truncated multipliers only the cost factor will be reduced in FIR filters.

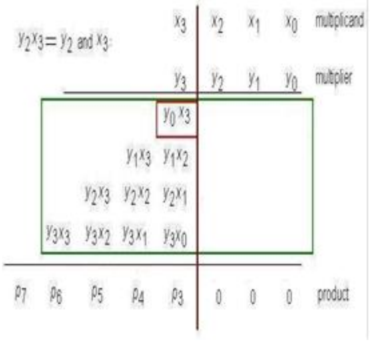


Fig:3 4x4 bit Binary Multiplication with truncation

WALLACE MULTIPLIER:

A modified Wallace multiplier is an efficient hardware implementation of digital circuit which multiplies two integers whose flow chart is shown in Fig 4. Generally in the reduction phase of conventional Wallace multipliers, many full adders and half adders are used when compared to modified Wallace multipliers. As we know that half adders do not reduce the number of partial product bits. Therefore, it is necessary to minimize the number of half adders used in a multiplier which reduces the hardware complexity. Hence, a modification to the Wallace reduction is done in which the delay is the same as for the conventional Wallace reduction. The modified reduction method greatly reduces the number of half adders with a very slight increase in the number of full adders. Reduced complexity Wallace multiplier reduction consists of three stages. First stage the N x N product matrix is formed and before passing on to the second phase the product matrix is rearranged to take the shape of inverted pyramid. During the second phase the rearranged product matrix is grouped into non-overlapping group of three as shown below, single bit and two bits in the group will be passed on to the next stage and three bits are given to a full adder. The number of rows in each stage of the reduction phase is calculated by the formula

$$r_{i+1} = 2[r_i/3]+r_i \text{mod} 3$$
$$\text{If } r_i \text{mod} 3 = 0, \text{ then } r_{i+1} = 2r_i/3$$

If the value calculated from the above equation for number of rows in each stage in the second phase and the number of rows that are formed in each stage of the second phase does not match, only then the half adder will be used. The final product of the second stage will be in the height of two bits and passed on to the third stage. During the third stage the output of the second stage is given to the carry propagation adder to generate the final output.

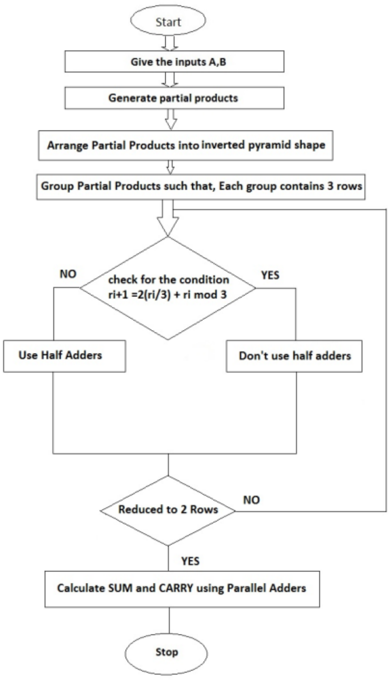
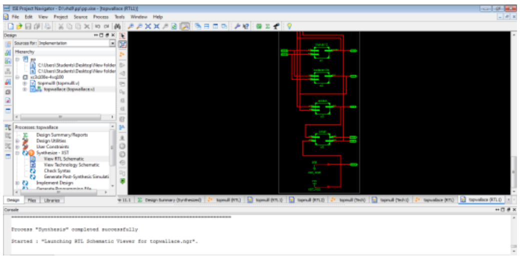

Start
Give the inputs A,B
Generate partial products
Arrange Partial Products into inverted pyramid shape
Group Partial Products such that, Each group contains 3 rows
NO
check for the condition ri+1 =2(ri/3) + ri mod 3
YES
Use Half Adders
Don't use half adders
Reduced to 2 Rows
NO
YES
Calculate SUM and CARRY using Parallel Adders
Stop

Fig 4: MODIFIED WALLACE FLOW CHART

VI. RESULTS

RTL schematic

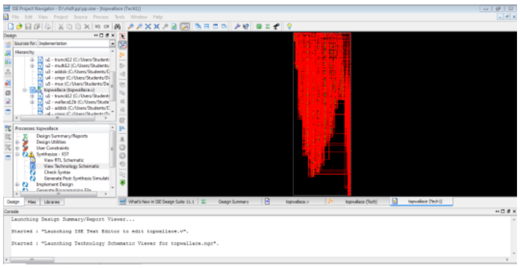


```
================================================================

Process "Synthesis" completed successfully

Started : "Launching RTL Schematic Viewer for topwallace.ngr".
```

Technological schematic

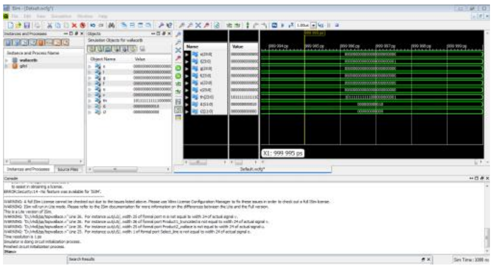


```
Launching Design Summary/Report Viewer...

Started : "Launching ISE Text Editor to edit topwallace.v".

Started : "Launching Technology Schematic Viewer for topwallace.ngc".
```

SIMULATED WAVEFORM



```
     to assist in obtaining a license.
ERROR:Security:14 - No 'feature' was available for 'ISIM'.
----------------------------------------------------------------
WARNING: A full ISim License cannot be checked out due to the issues listed above. Please use Xilinx License Configuration Manager to fix these issues in order to check out a full ISim license.
WARNING: ISim will run in Lite mode. Please refer to the ISim documentation for more information on the differences between the Lite and the Full version.
This is a Lite version of ISim.
WARNING: "D:/vhdl/pp/topwallace.v" Line 26. For instance uut/u1/, width 25 of formal port m is not equal to width 24 of actual signal r.
WARNING: "D:/vhdl/pp/topwallace.v" Line 26. For instance uut/u2/, width 26 of formal port Product1_truncated is not equal to width 24 of actual signal n.
WARNING: "D:/vhdl/pp/topwallace.v" Line 26. For instance uut/u3/, width 25 of formal port Product2_wallace is not equal to width 24 of actual signal u.
WARNING: "D:/vhdl/pp/topwallace.v" Line 23. For instance uut/u5/, width 1 of formal port Select_line is not equal to width 24 of actual signal o.
Time resolution is 1 ps
Simulator is doing circuit initialization process.
Finished circuit initialization process.
ISim>
```

Comparison Table

multiplier	No .of LUT's	Delay	Memory
Truncated with modified wallace	689	30.679ns	201564 kilobytes
Truncated with array	719	41.102ns	234524 kilobytes

VII. CONCLUSION

Here, in this project two different multipliers are designed which are array multiplier and modified Wallace multiplier along with the combination of truncated multiplier. In the proposed design which is nothing but truncated with modified Wallace the area (in terms of LUT's) is less which are 618 when compare to the existing truncated with array multiplier which are 648. So obviously the power is also reduced because it is calculated based on the number of LUT's. At the same time the delay and memory requirements for the proposed design is better when compare with the existed design . This multipliers output are derived depending on multiplexer selection line, which depends on the user. In future based on the requirements there may be a chance to change the multipliers.

VIII. REFERENCES

[1] Shen-Fu Hsiao, Jun-Hong Zhang Jian, and Ming-Chih Chen," Low-Cost FIR Filter Designs Based on Faithfully Rounded Truncated Multiple Constant Multiplication/Accumulation" ieee transactions on circuits and systems—ii: express briefs, vol. 60, no. 5, may 2013.

[2] M. M. Peiro, E. I. Boemo, and L. Wanhammar, "Design of high-speed multiplierless filters using a nonrecursive signed common subexpression algorithm," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.,vol. 49, no. 3, pp. 196–203, Mar. 2002.

[3] C.-H. Chang, J. Chen, and A. P. Vinod, "Information theoretic approach to complexity reduction of FIR filter design," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 8, pp. 2310–2321, Sep. 2008.

[4] F. Xu, C. H. Chang, and C. C. Jong, "Contention resolution—A new approach to versatile subexpressions sharing in multiple constant multiplications,"IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 2,pp. 559–571, Mar. 2008.

[5] F. Xu, C. H. Chang, and C. C. Jong, "Contention resolution algorithms for common subexpression elimination in digital filter design," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 52, no. 10, pp. 695–700, Oct. 2005.

[6] I.-C. Park and H.-J. Kang, "Digital filter synthesis based on an algorithm to generate all minimal signed digit representations," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 21, no. 12, pp. 1525–1529, Dec. 2002.

[7] C.-Y. Yao, H.-H. Chen, T.-F. Lin, C.-J. J. Chien, and X.-T. Hsu, "A novel common-subexpression-elimination method for synthesizing fixed-point FIR filters," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 51, no. 11,pp. 2215–2221, Sep. 2004.

[8] O. Gustafsson, "Lower bounds for constant multiplication problems," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 54, no. 11, pp. 974–978, Nov. 2007.

[9] Y. Voronenko and M. Puschel, "Multiplierless multiple constant multiplication," ACM Trans. Algorithms, vol. 3, no. 2, pp. 1–38, May 2007.

[10] D. Shi and Y. J. Yu, "Design of linear phase FIR filters with high probability of achieving minimum number of adders," IEEE Trans. Circuits Syst.I, Reg. Papers, vol. 58, no. 1, pp. 126–136, Jan. 2011.

[11] P. K. Meher, "New approach to look-up-table design and memory-based realization of FIR digital filter," IEEE Trans. Circuits Syst. I, Reg. Papers,vol. 57, no. 3, pp. 592–603, Mar. 2010.

[12] P. K. Meher, S. Candrasekaran, and A. Amira, "FPGA realization of FIR filters by efficient and flexible systolization using distributed arithmetic,"IEEE Trans. Signal Process., vol. 56, no. 7, pp. 3009–3017, Jul. 2008.

[13] S. Hwang, G. Han, S. Kang, and J.-S. Kim, "New distributed arithmetic algorithm for low-power FIR filter implementation," IEEE Signal Process.Lett., vol. 11, no. 5, pp. 463–466, May 2004.

[14] H.-J. Ko and S.-F. Hsiao, "Design and application of faithfully rounded and truncated multipliers with combined deletion, reduction, truncation,and rounding," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 58, no. 5, pp. 304–308, May 2011.