Journal of Nonlinear Analysis and Optimization
Vol. 13(12) (2022), December 2022
https://ph03.tci-thaijjo.org/

ISSN : 1906-9685

J.Nonlinear Anal. Optim

# Reconfigurable Architecture Pipelined Power Efficient Fixed Width Baugh-Wooley Multipliers

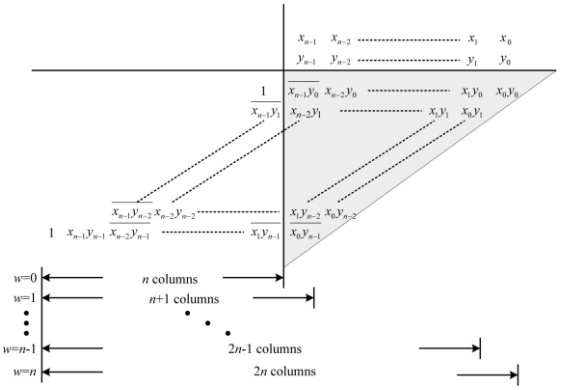
Dr Challa Venkateswara
Associate Professor, Department of ECE
Sri Sai Institute of Technology and Science, Rayachoti
Email: venkateswararaophd@gmail.com

*Abstract*—**This paper presents a pipelined reconfigurable fixed-width Baugh-Wooley multiplier design framework that provides four configuration modes (CMs): n * n fixed-width multiplier, two (n/2 * n/2) fixed-width multipliers, (n/2* n/2) fullprecision multiplier and two (n/4 * n/4) full-precision multipliers will be carried out. In this work pipeline is reconfigarableto save the power. In this work low-power schemes including gated clock and zero input techniques are employed to achieve the power-efficient pipelined reconfigurable design. The presented power-efficient pipelined reconfigurable fixed width multiplier design not only generates a family of widely used multipliers.**

***Keywords: Baugh-Wooley algorithm, full-precision multiplier, fixed-width multiplier, pipeline, power efficient, and reconfigurable,***

## I. INTRODUCTION

A core operation in actual circuits ,especially in Digital Signal Processing such as filtering, Modulation, Video processing, Neural networks , Satellite Communication , Graphics or Control Systems etc, is multiplication. Often the operational performance of a DSP system is limited by its computational performance of a DSP system is limited by its multiplication performance. Power dissipation is recognized as a critical parameter in modern VLSI design field. To satisfy MOORE'S law and to produce consumer electronics goods with more backup and less weight, low power VLSI design is necessary. Fast multipliers are essential parts of digital signal processing systems. The speed of multiply operation is of great importance in digital signal processing as well as in the general purpose processors today, especially since the media processing took off.




## II.FIXED WIDTH BAUGHWOOLEY MULTIPLIER STRUCTER AND SUBWORD MULTIPLICATION

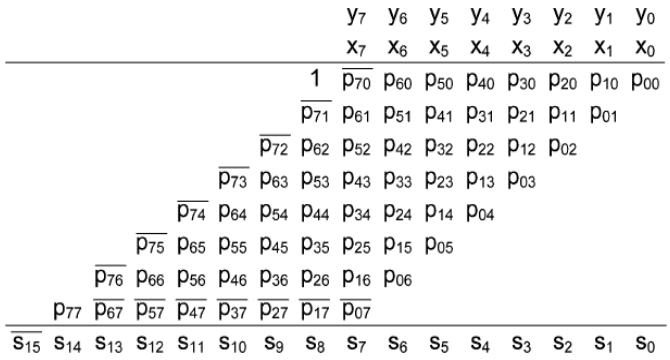
In this paper, we would like to reconfigure the fixed-width multiplication engine to generate four useful multipliers under the limited hardware resource. The partial product array for n* n 2s-complement multiplication are depicted in Fig. 1, where notation w means to keep n + w most significant columns of the partial products for fixed-width multiplications. If w = n, the fixed-width multiplier becomes a full-precision multiplier.

In this paper, we would like to reconfigure the fixed-width multiplication engine to generate four useful multipliers under the limited hardware resource. Moreover, many DSP and computer applications demand to operate at lower resolution, where the data can be expressed in a half word length. Generally, applying the sub word multiplication scheme, we can partition an n-bit operand into two independent n/2-bit operands or four independent n/4-bit operands; hence, the sub word multiplier can perform not only n*n full-precision multiplication but also two n/2* n/2 or four n/4 * n/4 full-precision.

Fig1: the low-power fixed-width multiplier structure

TABLE 1
Proposed Four Configuration Modes of the Reconfigurable
Fixed-Width Baugh-Wooley Multiplier

| Configuration Mode (CM) | Function Descriptions |
|---|---|
| CM1 | *nxn* fixed-width multiplier |
| CM2 | two *n*/2x*n*/2 fixed-width multipliers |
| CM3 | *n*/2x*n*/2 full-precision multiplier |
| CM4 | two *n*/4x*n*/4 full-precision multipliers |




*A. Baugh–Wooley Algorithm*

Fig:2 illustrates the algorithm for an 8-bit case, where the partial-product array

**i)** the most significant partial product of the first ***N-1*** rows and the last row of partial products except the most significant have to be negated,

**ii)** A constant one is added to the ***N***th column,

**iii)** the most significant bit (MSB) of the final result is negated.

Fig2: Illusration of multiplication of 8*8bit

he fixed-width multipliers derived from Baugh–Wooley [11]–[13] multiplier produce -bit output product with n-bit multiplier and n–bit multiplicand. Area saving of a fixed-width multiplier can be achieved either by directly truncating least significant columns and preserving most significant columns or by other efficient methods.

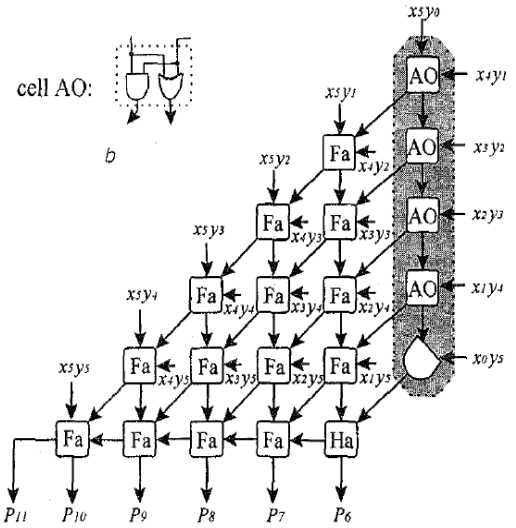
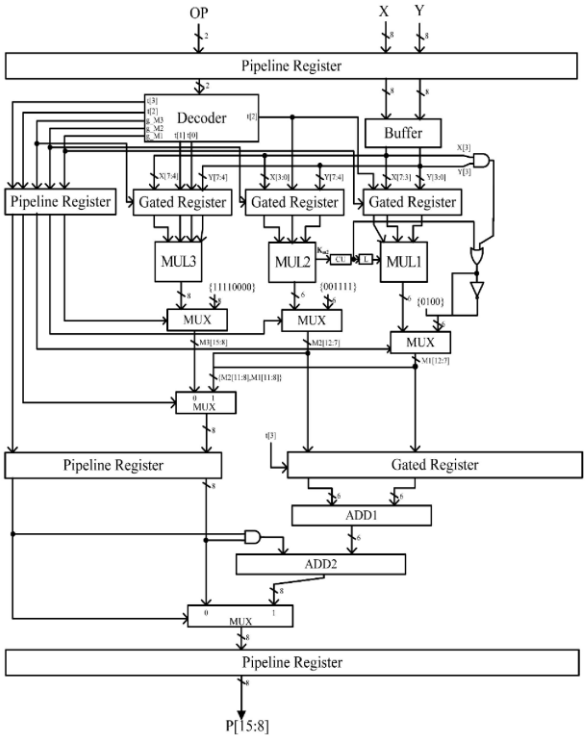



Fig3:Architectureof Fixed width multiplier

Truncating or roundup the lower part (LP) of the general 6-bitmultplier gives the 6-bit fixed width multiplier which gives product also 6-bit width.Truncatig the lower part (LP) of general multiplier by using series of **AND-OR (AO) logic** cells. The disadvantages of previous multipliers were low computational speed and low throughput and high power dissipation. Our multiplier enhances in these areas i.e. it has high computational speed, double throughput and also power reduction. Because of these advantages the proposed multiplier can be used in applications such as discrete cosine transform, discrete wavelet transform, image and signal processing applications. In image compression chip and data compression chip.

*III.Block Diagram*

The block diagram of the previous architecture is shown below. This architecture has some disadvantages in terms of optimization of power so again this was modified and a new architecture was proposed which optimizes in terms of power the block diagram for which is shown below. This can implement only one module in the fig4 all the multipliers will be used to produce the output. So that it can consume more power and it can't produce more no of modules so to reduce power consumption the pipelining stage is reconfigurable and low-power schemes including gated clock and zero input technique is




Fib4:proposed architecture

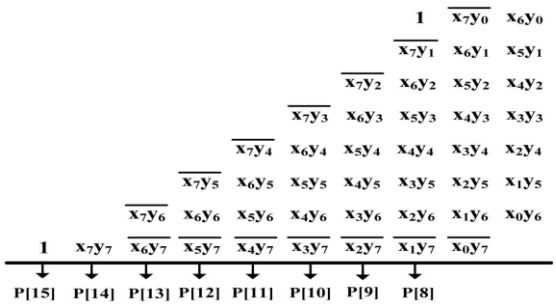
*A. Clock gating for the second and third stages*

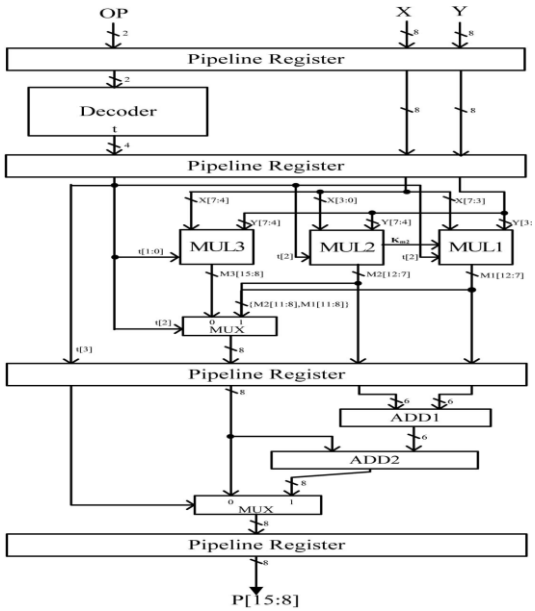
1.If CM1 is performed the MUL1,MUL2.MUL3 is conditionally disabled2. If CM2 is performed, input registers of MUL3 andADD1 can be disabled.3. If CM3 is performed, input registers MUL1,MUL2, and ADD1 can be disabled.4. If CM4 is performed, input registers of MUL1,MUL2, and ADD1 can be disabled

*B.Zero input for the third stage*

Zero input scheme working for CM2, CM3, and CM4 is mainly aimed at providing zero input sequences for adder to keep value unchanged at the third stage of Fig. 5.If CM2, CM3, or CM4 is performed, we use AND gates to generate zero sequence and feed into the ADD2. In this case, for ADD1, we can use t[3] as the control signal of the clock gating register to latch its input value. At the same time, for ADD2, one of the inputs comes from ADD1 that has been latched and we only need to set the other input to zero via AND operation with t[3]. Thus, we can further reduce the

implemented. our improved architecture is shown in thefig.5the pipelined reconfigurable architecture have MUL1,MUL2,MUL3works depending on configuration mode




of

Fig:5 Modified Architecture of pipelined reconfigurable power efficient fixed width multiplier

*IV.Design of Reconfigurable fixed width multiplier*

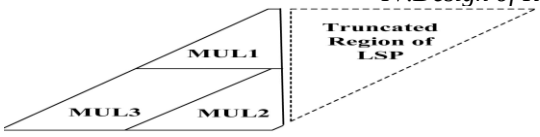



Fig 6:Prototype structure of the proposed reconfigurable fixed-width multiplier involving MUL1, MUL2, MUL3, and discarding truncated

The fig 6 how the LSB part is truncated and remaining part is distributed among three multipliers the rest partial products are decomposed into three multiplication modules MUL1, MUL2, and MUL3 as depicted in Fig6. The partial products of the three blocks are summed up independently and then the three summations are added together to produce final product.

*A. Proposed architecture in CM1 mode*

Configuration mode 1 (CM1) is in charge of operating n*n fixed-width multiplication that receives two n-bit numbers and produces an n-bit product. As mentioned above in this section, the rest partial products are decomposed into three multiplication modules MUL1, MUL2, and MUL3 as depicted in Fig.6 The partial products of the three blocks are summed up independently and then the three summations are added together to produce final product. , we provide five configuration parameters CP0, CP1, CP2, CP3, and CP4 combining with the proper partial product setting to generate other multipliers. In CM1, CP0, CP1, CP2, CP3, and CP4 are set to 0 as shown below in the fig 7.

*B. Proposed architecture in CM2 mode*

It is manifest that MUL1 and MUL2 are suitable for two n/2 * n/2 fixed-width multiplications. Due to the use of MUL1

Fig:7 Partial product array diagram for n*n fixed-width

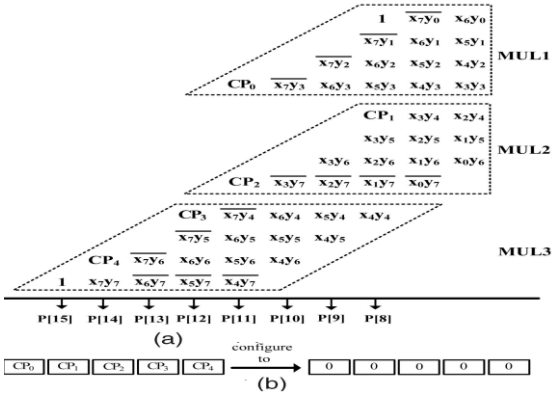
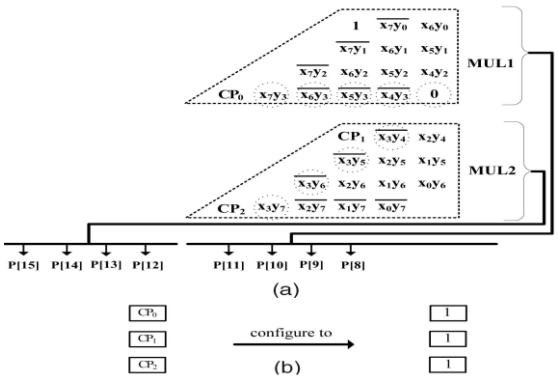



Fig.8. (a) proposed partial product array diagram using MUL1, MUL2, and MUL3 for CM1; and (b) configuration parameter settings

and MUL2, the corresponding fixed-width sub word operation of CM2 is illustrated. . we can depict the partial product array diagram using MUL1 and MUL2 in Fig. 9a, where the partial products circled by dot-line needed to be reconfigured in comparison with CM1. In Fig. 9a, compared with partial products MUL1 and MUL2 of CM1, x4y3, x5y3, x6y3, x7y3, x3y4, x3y5, x3y6, and x3y7 are complemented, x3y3 is configured to zero. The configuration parameters of CM2 can be set as addressed in Fig. where CP0, CP1, and CP2 are set to 1. The rest partial products are unchanged.




of 9b,

Fig.9 (a) proposed partial product array diagram using MUL1, MUL2 for CM2(b) configuration parameter settings

*C.Proposed architecture in CM3 mode*

Configuration mode 3 (CM3) serves as performing an n/2*n/2 full-precision multiplication. In behavior similar to that in CM2, the design procedures can be stated as follows: First, we have to determine which modules are suitable for n/2*n/2 full-precision multiplications with the minimum number of modules and partial product configuration settings. Under these constraints, since the proposed reconfigurable structure to implement full-precision multiplication is based on the fixed-width multiplier fabric.

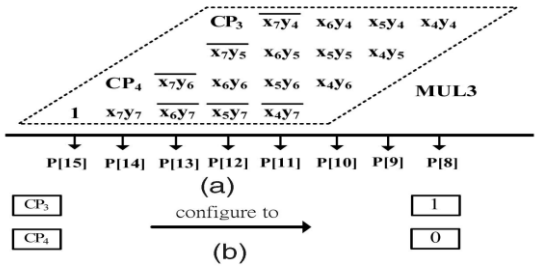



Fig10: (a) Proposed partial product array diagram for CM3, and(b) configuration parameter settings

*D.Proposed architecture in CM4 mode*

Configuration mode 4 (CM4) widely used in lower resolution operation serves as performing two n/4 * n/4 full-precision multiplications. Under the minimum number of modules and partial product configuration setting constraints, we make use of the MUL3 to fulfill the CM4operation. Due to the use of MUL3, the corresponding sub word operation of CM4 is illustrated in fig 11.Then, the partial product array diagram of two n/4 * n/4 full-precision multipliers can be obtained in Fig. 11a. In Fig. 11a, compared with partial products of the MUL3 of CM1, x5y4 and x4y5 are complemented, x6y4 and x6y5 are configured to one, and x7y4, x7y5, x4y6, x5y6, x4y7, and x5y7 are configured to zero. The configuration parameters of CM4 can be set as addressed in Fig. 9b, where CP3 and CP4 are set to 0 and 1, respectively. The rest partial products are unchanged.

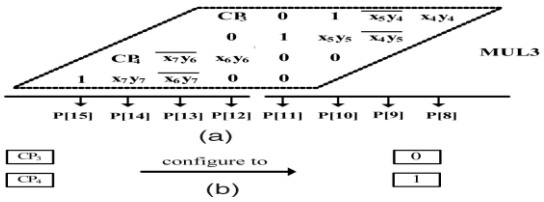


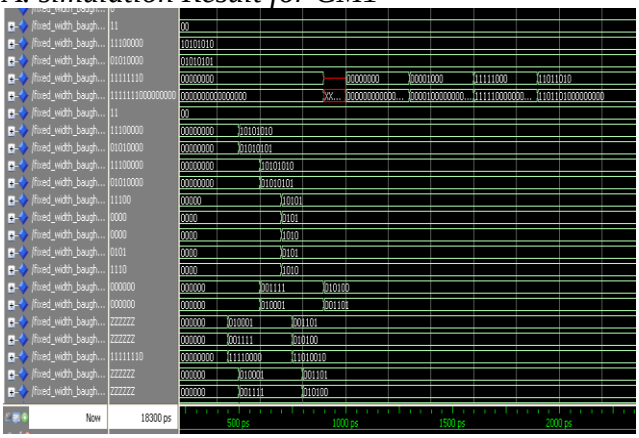

*Fig11. (a) Proposed partial product array diagram for CM4, and(b) Configuration parameter settings.*

*V.Design of reconfigurable architecture pipelined fixed width Bagaugh_wooley multiplier*

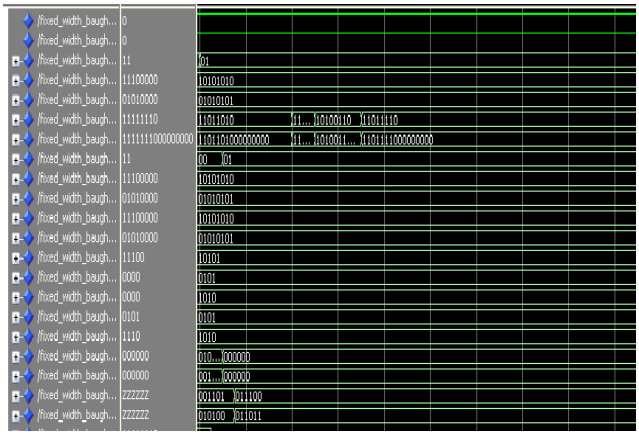
As mentioned, the multiplications of CM2, CM3, and CM4 are of power-inefficient because they invoke all hardware resource to compute. It is desirable to apply low-power schemes such that the proposed reconfigurable fixed-width multiplier possesses power-efficient capability. We apply low-power schemes including clock gating and zero input techniques to achieve power saving. In the fig 5: actually in previous architecture [58] five pipes lined stages are used. But here I used to eliminate one pipe line which is at the output of the and gate. So the it will not affect any value at the output stage. By eliminating the pipeline stage, sothat we can reduce the power consumption at this stage. The power saving may be look a little bit but if we consider more no of stages in Dsp and image processing applications this may give a lot of power saving. and if one stage is reduced the speed of the multiplier is improved

## *VI.Simulation Results*

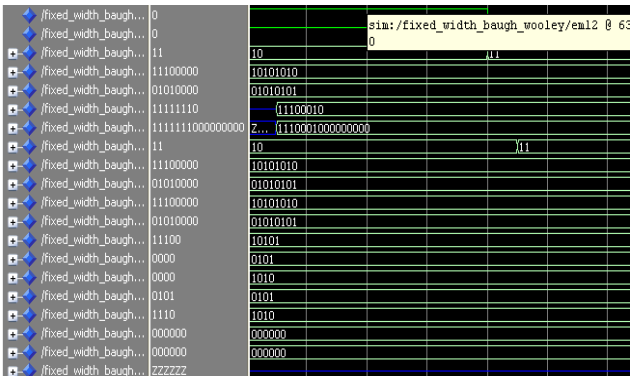
*A. simulation Result for CM1*



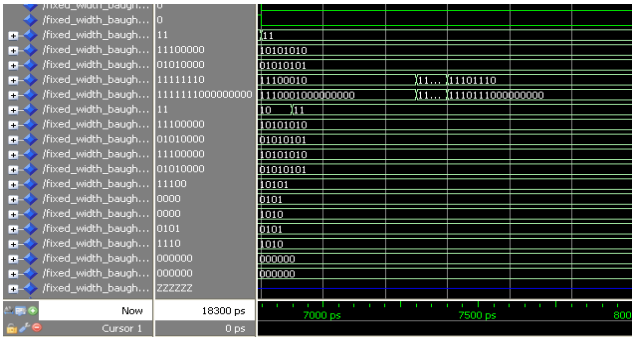
*B.simmulation result forCM2*



*C.simmulation result forCM3*



*D.simmulation result for CM4*




## VII .CONCLUSIONS

This paper presents a framework for the pipelined reconfigurable fixed-width Baugh-Wooley multiplier to generate a family of fixed-width and full-precision multipliers includingCM1, CM2, CM3, and CM4. We make use of low-power schemes including gated clock and zero input techniques to achieve power reduction of 0.81, 12.46,17.93,and23.2 percent, on average, compared with the non-reconfigurable multiplier for n ¼ 8; 16; 24, and 32, respectively And by using the comparator at the first stage we can reduces the rounding error.

## REFERENCES

[1] C.R. Baugh and B.A. Wooley, "A Two's Complement Parallel Array Multiplication Algorithm," IEEE Trans. Computers, vol. 22, no. 12, pp. 1045-1047, Dec. 1973.
[2] A.D. Booth, "A Signed Binary Multiplication Techniques," Quarterly J. Mechanics and Applied Math., vol. 4, pp. 236-240, 1951.
[3] O.L. MacSorley, "High-Speed Arithmetic in Binary Computer," Proc. Conf. Institute of Radio Engineers (IRE '61), vol. 49, pp. 67-91,1961.
[4] K. Hwang, Computer Arithmetic: Principles, *Architecture, and Design.N John-Wiley, 1979*
*[5] F. Cavanagh, Digital Computer Arithmetic: Design and Implementation. McGraw-Hill, 1984.*
*[6] M.D. Ercegovac and T. Lang, Digital Arithmetic. Morgan and Kaufmann, 2004.*
*[7] S.L. Freeny, "Special-Purpose Hardware for Digital Filtering," Proc. IEEE, vol. 63, no. 4, pp. 633-647, Apr. 1975.*
*[8] Y.C. Lim, "Single-Precision Multiplier with Reduced Circuit Complexity for Signal Processing Applications," IEEE Trans.Computers, vol. 41, no. 10, pp. 1333-1336, Oct. 1992.*
*[9] M.J. Schulte and E.E. Swartzlander Jr., "Truncated Multiplication with Correction Constant," Proc. Workshop Very Large Scale Integration (VLSI) Systems Signal Processing, VI, pp. 388-396, 1993.*
*[10] S.S. Kidambi, F. El-Guibaly, and A. Antoniou, "Area-Efficient Multipliers for Digital Signal Processing Applications,"* IEEE Trans. Circuits and Systems, vol. 43, no. 2, pp. 90-94, Feb. 1996.