**FIXED WIDTH MULTIPLIER**

Low Power-Delay Product

3/25/2018

# **BY:**

# Navya Jyothi Vyshnavi

# Surya Shivesh

Contents

# *Abstract*

# *Introduction*

# *Basics of Baugh-Wooley Multiplier*

# *Block Diagrams Of Different Multiplicaion Modules*

# *Other Blocks Used*

# *VHDL Code*

# *Simulation Results*

# *Conclusions*

1. **ABSTRACT:**

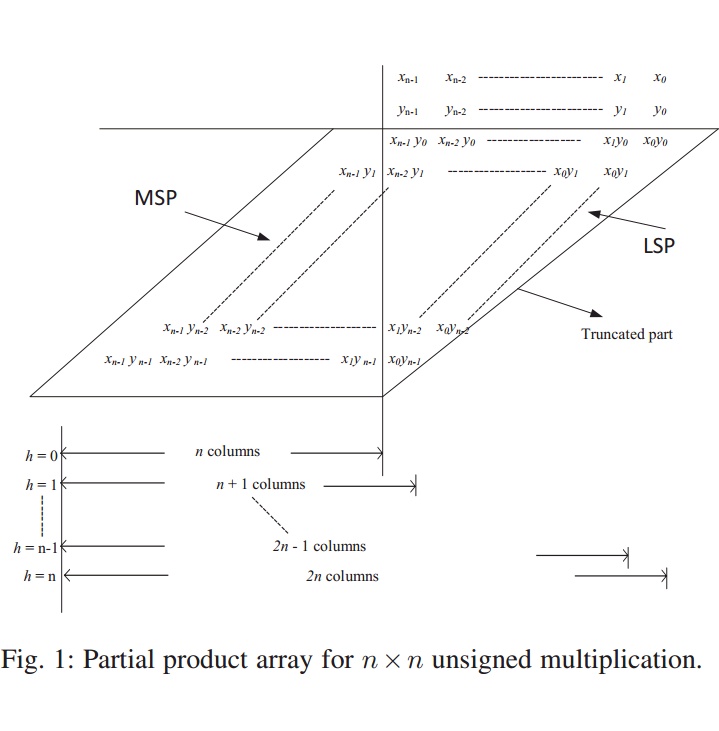
*This paper tells about the fixed width multiplier having low power-delay product ,it receives 2 n bit-numbers and produces an n bit product .In order to obtain this three multiplication modules are used which work as independent smaller precision multipliers. To sum up the partial products obtained from those modules ,carry propagate adder and Brent-Kung adders are used so that the performance is still improved. This process of using these multiplication modules is nothing but Baugh-Wooley Algorithm .So this, designed multilplier has low power-delay product compared to other mutlipliers.*

# ***2) Introduction***:

*With advances in technology ,many researchers have come up with various multipliers such as Baugh-Wooley , Dadda, Wallace tree and Booth multiplier .But of them ,the power-efficient and high speed multiplier plays an important role of very large-scale integration (VLSI) systems. By directly truncating n-bit Least Significant Bit (LSB) output a fixed-width multiplier (single precision) can be achieved, that produces n-bit output product with n-bit multiplier and n-bit multiplicand. However, truncating the LSB part of the multiplication product leads to large truncation errors (sum of the reduction and rounding errors). In order to mitigate this truncation error, many error compensation circuits are designed with less area.*

*However, in today’s applications one of the major challenges for high-performance DSP applications is the power consumption, both static and dynamic. Therefore,instead of targeting them independently, there is a need to find an optimum between speed and power. This is represented by the average energy dissipated for one switching event which is known as power-delay product*

*In this paper, we propose a new approach that uses three multiplication modules and parallel adders, to improve the speed and power-delay product of a fixed-width multiplier. The rest of the paper is organized as follows: The Baugh-Wooley array multiplier is briefly reviewed in Section 3. In Section 3, a parallel fixed-width multiplication engine with three multiplication modules is presented. The results of delay reduction, power consumption and for n = 8 and h = 2 are presented in Section 7. Finally, Section 8 presents a conclusion.*

.

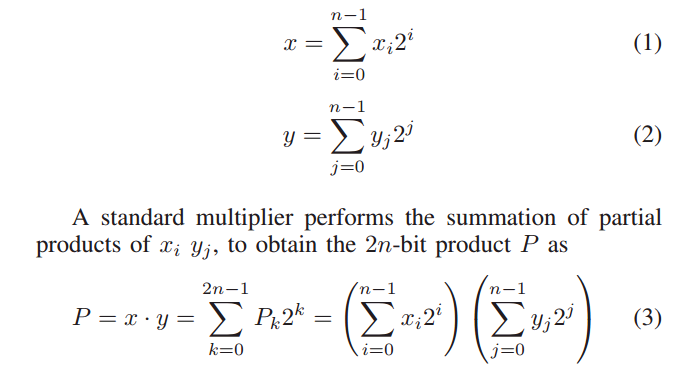
*Here MSP is most significant part and LSP is Least Significant Part also called as truncation part .Because by removing this we are going to design the required multiplier.*

**3)Basics of Baugh-Wooley Multiplier:**

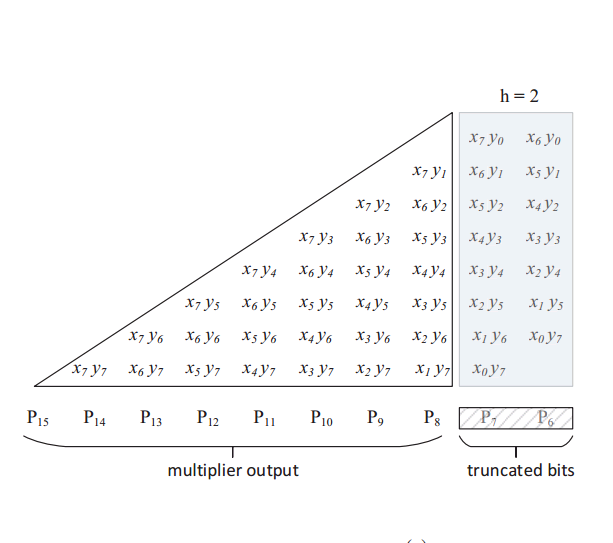
*The full-length n-bit unsigned Baugh-Wooley partial product array can be divided into*

*two subsets of most significant part (MSP) and less significant part (LSP) as shown in*

*Fig. 1.*

*Considering the Baugh-Wooley array multiplier with two unsigned n-bit inputs x and y, we can respectively, represent an n-bit multiplicand x and an n-bit multiplier y as :*

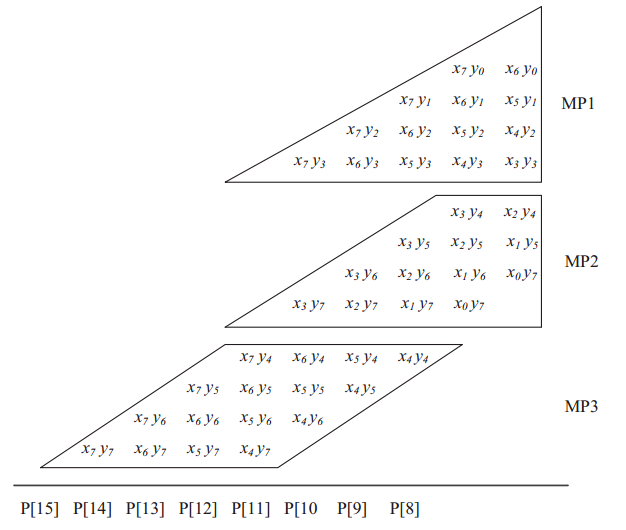
*The product of two 8-bit numbers x and y can be represented as follows*

*Fig:2(a)*

*It is known that from the above discussion, there are various variable correction techniques that have been widely used in order to reduce error and area cost. Therefore, we chose n = 8 and the multiplier is confined to h = 2 as shown in Fig. 2(a). The partial products are decomposed into three multiplication modules which are denoted by MP1, MP2 and MP3 as shown in Fig. 2(b). The design structure of these modules is as follows.*

**4)DifferenT Multiplication Modules:**

*The basic partial product array can be divided into three multiplication modules as shown below:* Fig: 2(b)

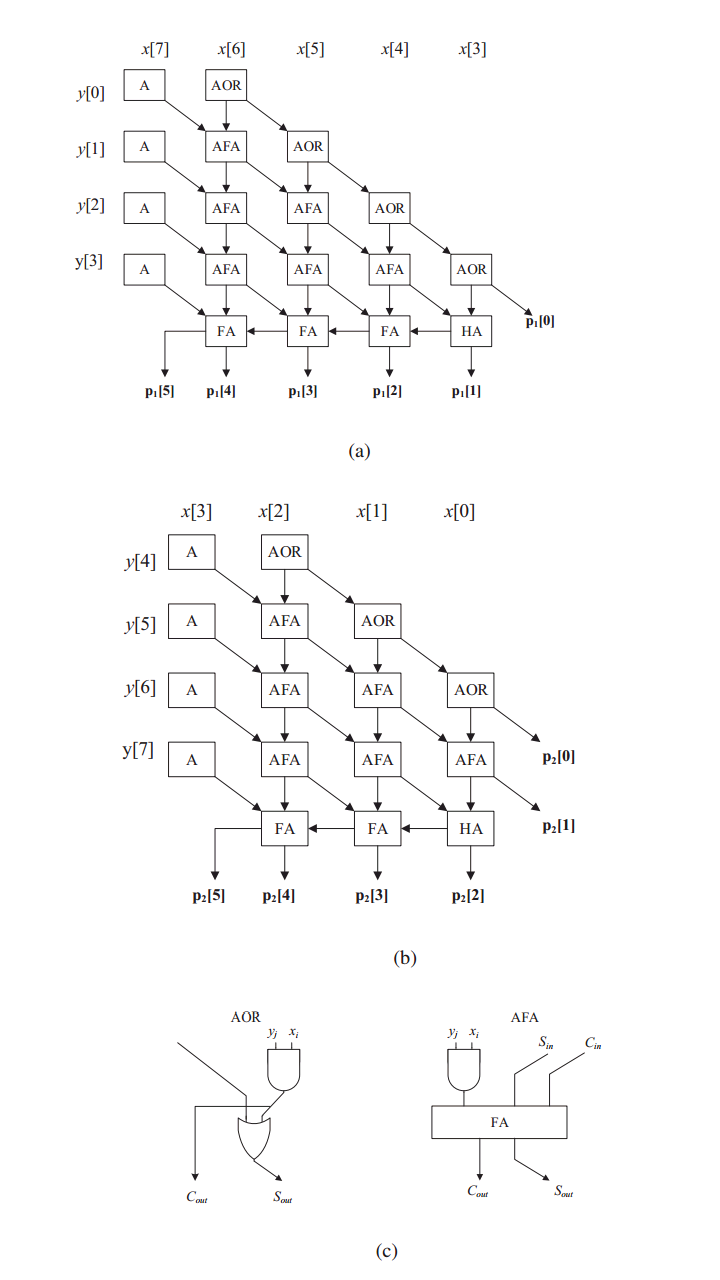
**

*We can observe that the truncation bits are neglected ,which helps in the lower power consumption .The architecture of these blocks is mentioned below:*

Multiplication modules mp1 and mp3:

*These multiplication modules are designed using Bough - Wooley algorithm.*

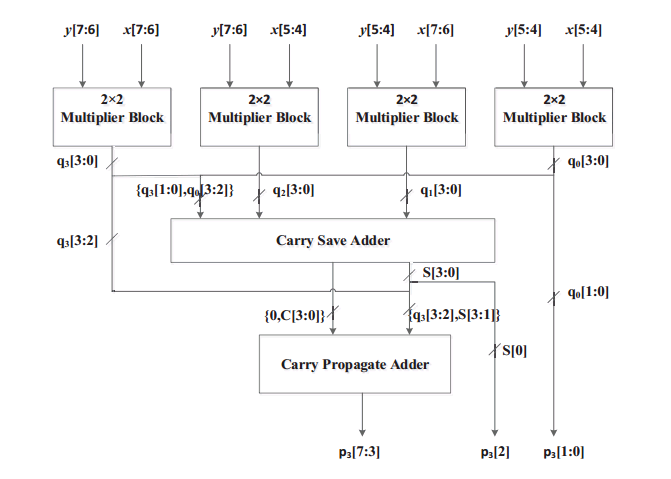
*The detailed diagram of the corresponding modules MP1 and MP2 are exposed in Fig. (a) and Fig. (b) respectively, where A, HA, and FA denote an AND gate, a half adder and a full adder, the logic diagram of AOR and AFA are depicted in Fig. (c). For MP1 module the inputs are x[7 : 3] and y[3 : 0] that generates partial product output as p1[5 : 0]. Similarly, for MP2 module the inputs are x[3 : 0] and y[7 : 4] that generates partial product output as p2[5 : 0]. These outputs are generated independently by using MP1 and MP2, which can improve the speed of the design.*

**

multiplication module mp3:

*The input bits x[7 : 4] and y[7 : 4] of the module MP3 are partitioned into two independent 2-bit operands as xh=x[7 : 6] and xl=x[5 : 4] for x and yh=y[7 : 6] and yl=y[5 : 4] for y, where h and l represents higher and lower order bits of x and y. These operands are computed in parallel using four 2 × 2 multiplier blocks that can generates four partial product outputs q0[3 : 0], q1[3 : 0], q2[3 : 0] and q3[3 : 0].*

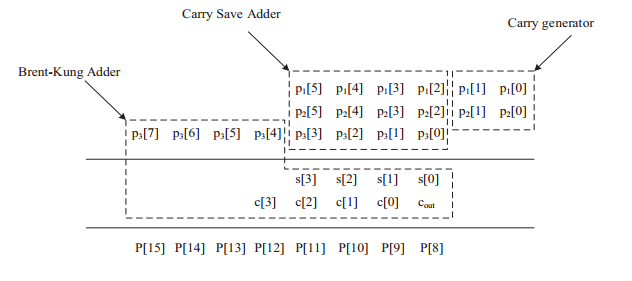
*The architecture of multiplication block is as shown below:*

**

Final Summation of Partial Products:

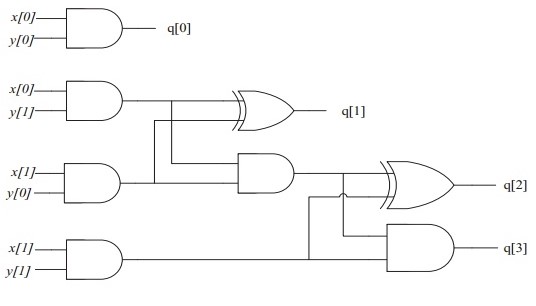
*The partial product outputs of three multiplication modules MP1, MP2 and MP3 are added as shown in the below fig. The sum of partial product bits p1[1 : 0] and p2[1 : 0] generates final product bits P6 and P7 with a carry cout. But, for a fixedwidth multiplier the product bits P6 and P7 are truncated and only carry has to propagate to the next level. To propagate the carry the logic diagram of the carry generator is designed.*

*A carry save adder is used to sum the partial products p1[5 : 2], p2[5 : 2] and p3[3 : 0] that generates sum and carry outputs as s[3 : 0] and c[3 : 0] respectively. Then, finally the sum and carry bits generated using carry save adder, the carry bit cout from carry generator and partial product bits p3[7 : 4] are summed up using Brent-Kung adder to get the final product P[15 : 7] of the fixed-width multiplier*

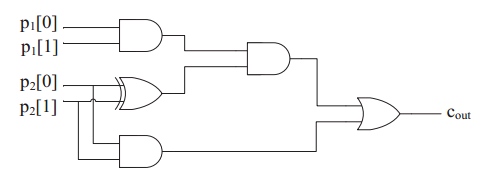
**

5) Other Blocks Used:

1. *The main component used in the multiplication block MP3 is 2\*2 multiplier block*.
2. *The logic diagram of 2\*2 multiplier is as shown*



1. *The carry generation circuit used in summing up the partial products is shown below:*

**

1. *The major adders used in this design are carry propagate adder(or carry look ahead adder) and Brent-Kung adder.*
2. *Using these adders there will be more improvement in the delay and so they also play a role in obtaining lower power delay product.*