International Journal of Scientific Engineering and Research (IJSER)
www.ijser.in
ISSN (Online): 2347-3878, Impact Factor (2015): 3.791

Design and Implementation of Four Bit Binary Array Multiplier

Yasmeen Saundatti

Assistant Professor, Department of Electronic & Communication Engineering, SECAB.I.E.T, Vijayapura, Karnataka, India

Abstract: ***Recently, several experimental systems based on programmable logic have been designed and implemented which are programmed using a hardware design methodology. One necessary component of the software environment will be a library of standard macro cells corresponding to commonly used arithmetic and logical operations. In this paper Array multiplier is designed especially for programmable logic. This multiplier is cellular, highly pipelined and uses only of local interconnections. The design is particularly carried out for a 4- bit multiplier.***

Keywords: CMOS, VLSI, adder, CSA, Array multiplier, micro wind tool

1. Introduction

Today, complex circuits are described in high-level description languages, like VHDL or Verilog, and synthesized to gate-level. A core operation in actual circuits, especially in digital signal processing such as Filtering, Modulation, or Video Processing or Satellite Communication or Graphics or Control systems etc, is multiplication. The computational performance of a DSP system is limited by its multiplication performance. This paper presents fundamental of some multiplication array multiplier and its implementation details at CMOS level and the results thereof. Hardware multiplier implementation will have better speed than implementing the same using sequential statements in any higher level language[1].Traditionally shift and add algorithm has been implemented to design, however this is not suitable for VLSI implementation and also from delay point of view. Some of the important algorithm proposed in literature for VLSI implementable fast multiplication is Booth multiplier, array multiplier and Wallace tree multiplier. This paper presents the array multiplier fundamental technical aspects behind this approach.

2. Multiplication Algorithm

The multiplication algorithm for an N bit multiplicand by N bit multiplier is shown below:
Y= Yn-1 Yn-2 .....Y2 Y1 Y0 Multiplicand
X= Xn-1 Xn-2..... X2 X1 X0 Multiplier

Example 1101 4-bits
1101 4-bits

1101
0000
1101
1101

10101001

AND gates are used to generate the Partial Products, PP, If the multiplicand is N-bits and the Multiplier is M-bits then there is N* M partial product. The way that the partial products are generated or summed up is the difference between the different architectures of various multipliers [3]. Multiplication of binary numbers can be decomposed into additions.

The equation for the addition is:

$$P(m+n) = A(m)B(n) = \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} a_i b_j 2^{i+j}$$

Array multiplier is well known due to its regular structure. Multiplier circuit is based on add and shift algorithm. Each partial product is generated by the multiplication of the multiplicand with one multiplier bit. The partial product are shifted according to their bit orders and then added. The addition can be performed with normal carry propagate adder. N-1 adders are required where N is the multiplier length.

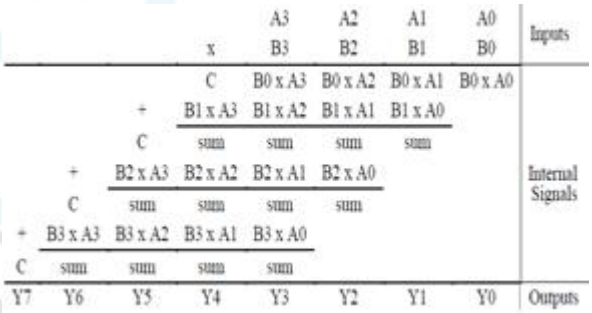

Figure 1: The general structure of a 4x4 array

Although the method is simple as it can be seen from the example as shown in figure 1, the addition is done serially as well as in parallel. To improve on the delay and area the Carry Save Adders are used, in which every carry and sum signal is passed to the adders of the next stage. Final product is obtained in a final adder .In array multiplication we need to add, as many partial products as there are multiplier bits. This arrangement is shown in the figure 2 below.

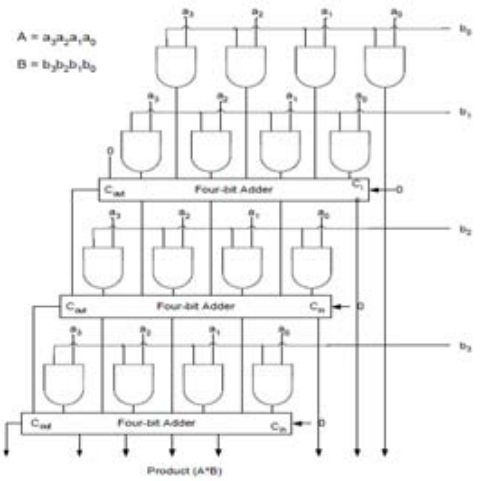

Figure 2: The general gate level arrangement of a 4x4 array

The general structure of a 4x4 array is as shown in figure 3. This scheme calculates the bit products ai.bj using AND gates[1]. The product bits are formed using adders in each column. The adders are arranged in a carry save chain as can be seen by nothing that the carry-out bits are fed to the next available adder in the column to the left. The array multiplier accepts all of the input bits simultaneously. The longest delay in the calculation of the product bits depends on the speed of the adders. The carry-chain in P7 that originates from the P1 column and propagates through the P2-P6 quantities would be an obvious problem. Input buffers may be added to synchronize the dataflow as shown in figure 5 in general, an array multiplier for n-bit words requires n (n-2) full adders, n half-adders, and n2 AND gates. The gate count allows an estimate of the required area based on the library entries.

3. Schematic Diagrams

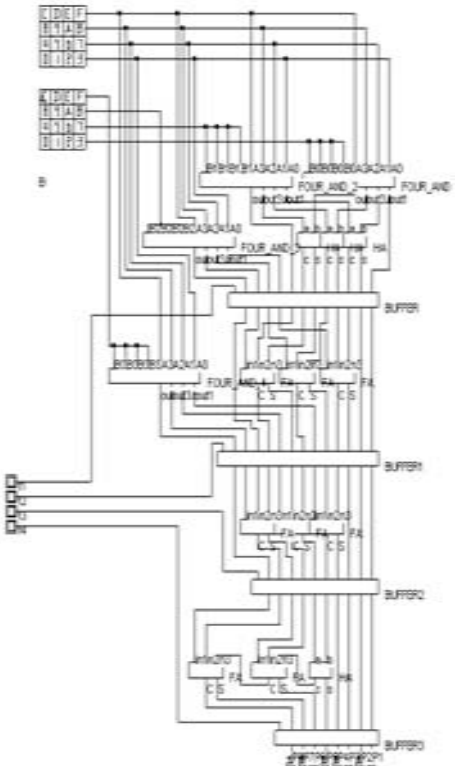

Figure 3: The general structure of a 4x4 array

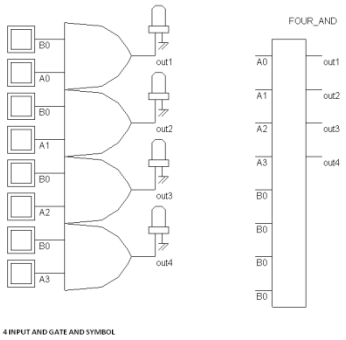

Figure 4: Input and gate and symbol

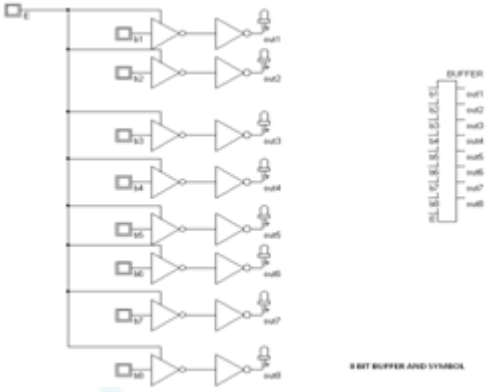

Figure 5: Bit buffer and symbol

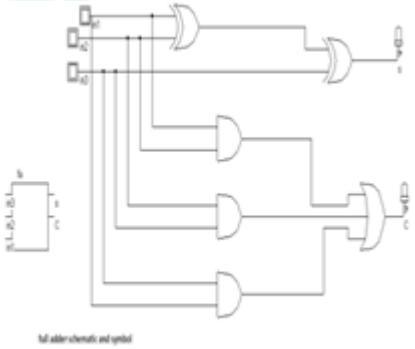

Figure 6: Full adder schematic and symbol

4. Carry- Save Adder

Carry save adders are based on the idea that a full- adder really has three inputs and produces two outputs as shown in figure 6. While we usually associate the third input with a carry-in, it could well be used as a regular value. The full adder is used as a 3-2 reduction network where it starts with bits from 3 words, adds them, and then has an output that is 2 bits wide. We can build an n-bit carry save adder by using n separate adders [4]. The name carry save arises from the fact that we save the carry-out word instead of using it immediately to calculate a final sum. CSA's are useful in situations where we need to add more than two numbers, since the design automatically avoids the delay in the carry-out bits, a CSA chain may be faster than using standard adders.

5. Simulation results

The simulation of the 4 bit binary array multiplier is presented in this section. For simulation Digital schematic of micro wind tool is used. The 4x4 array multiplier is as shown in simulation waveforms.

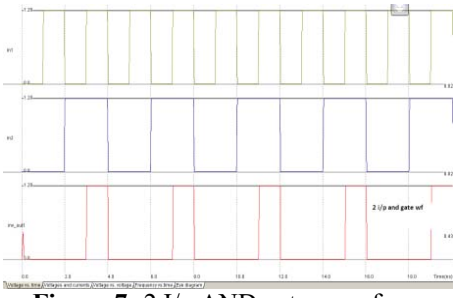

Figure 7: 2 I/p AND gate waveform

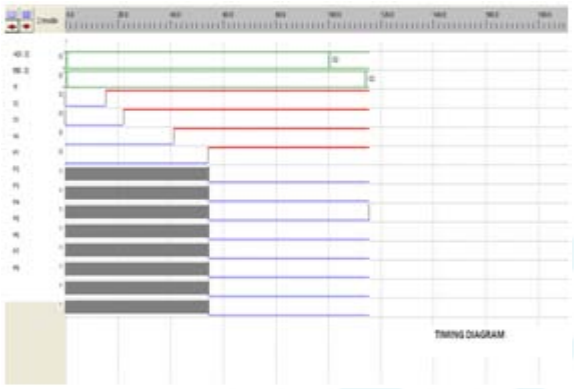

Figure 8: Timing diagram of the general structure of a 4x4 array

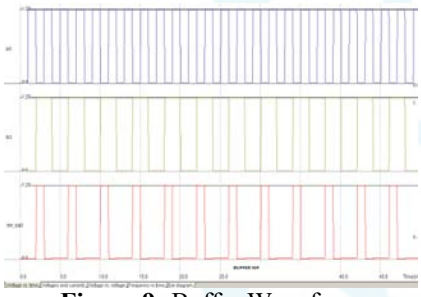

Figure 9: Buffer Waveform

6. Conclusion and Future scope

This paper presents a basic multiplier for Multiplication of 4 bit binary numbers which is suitable For CMOS implementation with partial product generation Technique. From the results of simulation it can be concluded that Array Multiplier gives efficient result with reduced delay. However Array Multiplier gives optimum Power consumption. This work is performed on 4x4 array bit array multiplier. Therefore, it can be extended for higher bit array multiplication.

References

[1] JOHN P UYEMURA," Introduction to VLSI circuits and system".
[2] Raminder Preet Pal Singh, Parveen Kumar, Balwinder Singh"Performance Analysis of 32-Bit Array Multiplier with a Carry Save Adder and with a Carry-Look-Ahead Adder", International Journal of Recent Trends in Engineering, Vol 2, No. 6, November 2009.
[3] Mohammed Hasmat Ali, Anil Kumar Sahani."Study, Implementation and Comparison of Different Multipliers based on Array, KCM and Vedic Mathematics Using EDA Tools", International Journal of Scientific and Research Publications, Volume 3, Issue 6, June 2013 1 ISSN 2250-3153.
[4] Divya Bora, Dr. U.M. Gokhale", Review on Design of High Speed Array Multiplier Using BICMOS Logic", International Research Journal of Engineering and Technology (IRJET), Volume: 02 Issue: 09 | Dec-2015, p-ISSN: 2395-0072.
[5] Sumit Vaidya1 and Deepak Dandekar, " Delay-Power Performance Comparison of Multipliers In VLSI Circuit Design", International Journal of Computer Networks & Communications (IJCNC), Vol.2, No.4, July 2010.
[6] N. Ravi, Dr. T. S. Rao, Dr. T. J. Prasad, "Performance Evaluation of Bypassing Array Multiplier with Optimized Design", International Journal of Computer Applications (0975 – 8887) Volume 28– No.5, August 2011