Journal of Graphic Era University
Vol. 4, Issue 1, 49-56, 2016
ISSN: 0975-1416 (Print)

Implementation and Performance Analysis of different Multipliers

Pooja Karki, Subhash Chandra Yadav*

Department of Electronics and Communication Engineering
Graphic Era University, Dehradun, India
*Corresponding author: subhash.yadav775@gmail.com

Abstract
Multipliers are very important and are used in various applications. This paper presents analysis of different multipliers design such as array multiplier, row bypassing multiplier and column bypassing multiplier. The multipliers are implemented using verilog HDL and the simulation is done in Modelsim simulator. The multipliers are compared in terms of delay and area. It is observed that delay in column bypassing multiplier is 22.48% less than array multiplier and 21.25% less than row bypassing multiplier. It is also observed that the column bypassing multiplier has less area than araay multiplier and row bypassing multiplier.

Keywords-Array multiplier, Row Bypassing Multiplier, Column Bypassing Multiplier, Switching Activity.

1. Introduction

Multiplier is important in digital signal processing and various other applications. With advancement in technology, multiplier which offers high speed, low power consumption, less area etc. are been designed. Digital multipliers are amid the most significant arithmetic functional units in many applications, such as the Fourier transform, discrete cosine transform and digital filtering. Multiplication is most commonly used operations in various applications like math processor and in various scientific applications (Goyal and Kaur, 2012).

2. Different Types of Multiplier

2.1 Array Multiplier

Array multiplier is well known due to its regular structure. Multiplier circuit is basically based on add and shift algorithm. Each partial product is generated by the multiplication of multiplicand with multiplier bit. The partial products are then shifted according to their bit orders and then added respectively. The addition can be performed with carry propagate adder. In multiplication, partial product is the sum of the number of bits of the two operands i.e. multiplicand and multiplier (Devi et al., 2015).

Let us consider the multiplication of 1111 * 0001 bit. In this, we have 4-bit multiplicand bit i.e. 1111 and 4-bit multiplier bit i.e. 0001. The multiplicand bit is multiplied with multiplier bit respectively. The half adder and full adder are used for addition of various bit. Various bits are added respectively and partial product is generated. Here we have a partial product of 8-bit i.e. 00001111.

Figure 1 shows the design of 4-bit array multiplier. Here, various half adder and full adder blocks are used for addition. The multiplication of multiplicand bit and multiplier bit is done and the partial products are generated.

2.2 Row Bypassing Multiplier

In bypassing technique, the idle part of the circuit which is not operating is shut down to save the power. In the row bypassing multiplier, if the multiplier bit is zero, the addition operation corresponding to that row is disabled which reduces the switching activity and hence the reduction in power dissipation can be achieved. In this technique, whenever certain multiplier bit b_j is zero, then the addition operation in the j-th row is disabled. This technique totally depends on the number of zeroes in the multiplier. In this, the multiplier bits are b_j can be used as the selection line of the multiplexer (Nujum and Cheriyan, 2013).

Let us consider the multiplication of 1100 * 1010 bit. In this, the multiplier bit consists of two zero bit. During multiplication, the corresponding adders of 1^{st} and 3^{rd} row will be disabled and the previous sum is considered as the current sum. In case of 8*8 row bypassing multiplier, of 16*16 row bypassing multiplier etc, more reduction in power dissipation can be achieved.

Figure 2 shows the design of 4-bit row bypassing multiplier.

2.3 Column Bypassing Multiplier

Column bypassing technique means turning off various columns in the multiplier array each time certain multiplicand bits are zero. In column bypassing multiplier technique, some diagonal columns are turned off using the bypassing technique whenever certain multiplicand bits are zero to save the power (Sahu and Meena, 2013). This technique completely depends on the number of zeroes in the multiplicand bit. In this, the multiplicand bits are a_i can be used as the selection line of the multiplexer.

Let us consider the, multiplication of 1100 * 1101. In this, the multiplicand bit consists of two zero bit, so the corresponding column of full adder in 1^{st} and 3^{rd} diagonals will be disabled and the output carry of full adder is 0 and the sum is simply equal to the third bit which is the summation output of its upper full adder (Lin et al., 2015).

Figure 3 shows the design of 4-bit column bypassing multiplier.

3. Simulation Results

The design is coded using verilog HDL and the simulation is done using Modelsim simulator. The designs are synthesized and implemented in Xilinx ISE 14.7. Xinilx ISE is a design environment for FPGA product and it is primarily used for circuit synthesis and design. While ModelSin logic simulator is used for system level testing.

Journal of Graphic Era University
Vol. 4, Issue 1, 49-56, 2016
ISSN: 0975-1416 (Print)

3.1. Array Multiplier

Figure 4 shows the block diagram of an array multiplier. It is coded in verilog HDL. Here, a bit and b bit is input bit and p bit is partial product (Ramesh, 2011).

Figure 5 shows the output waveform of an array multiplier in which 1111 bit is multiplied with 0001 and the output is 00001111.

3.2. Row Bypassing Multiplier

Figure 6 shows the block diagram of a row bypassing multiplier. It is coded in verilog HDL. Here, a bit and b bit is input bit and p bit is partial product.

Figure 7 shows the output waveform of 4*4 row bypassing multiplier in which multiplicand bit 0101 is multiplied with 0011 and the output is 00001111.

3.3. Column Bypassing Multiplier

Figure 8 shows the block diagram of a 4*4 column bypassing multiplier. It is coded in verilog HDL. Here, a bit and b bit is input bit and p bit is partial product.

Figure 9 shows the output waveform of 4*4 column bypassing multiplier in which multiplicand bit 1111 is multiplied with 0101 and the output is 00000110 (Chirde and Jadhav, 2015).

4. Comparison

Comparison of delay and area of array multiplier, row bypassing multiplier and column bypassing multiplier is done.

4.1. Comparison of Time Delay

By comparing the time delay obtained in synthesis report, we can decide the high sped multiplier. Table gives the comparison results of array multiplier, row bypassing multiplier and column bypassing multiplier.

From the table 1, it can be seen that the array multiplier has the maximum time delay followed by row bypassing multiplier and column bypassing multiplier.

4.2. Comparison of Area Utilization on the Basis of Number of Slices

The number of slices in column bypassing multiplier is 15 followed by array multiplier that has 17 and then row bypassing multiplier that has 23. It can be observed that column bypassing multiplier has the least area followed by array multiplier and then row bypassing multiplier (Figure 10).

Journal of Graphic Era University
Vol. 4, Issue 1, 49-56, 2016
ISSN: 0975-1416 (Print)

5. Result Analysis

From the table, it is understood that the time delay for column bypassing multiplier is better than any other multiplier. Hence, it can be used for high speed bypassing multiplier. The multipliers are compared in terms of delay and area. It is observed that the delay in column bypassing multiplier is 22.48% less than array multiplier and 21.25% less than row bypassing multiplier. It is also observed that the column bypassing multiplier has less area than array multiplier and row bypassing multiplier.

S. No.	Array multiplier	Row bypassing multiplier	Column bypassing multiplier
1. Time Delay (ns)	15.101	14.865	11.706
2. No. of slices (out of 920)	17 (1%)	23 (2%)	15 (1%)

Table 1. Delay and area comparison of different multipliers

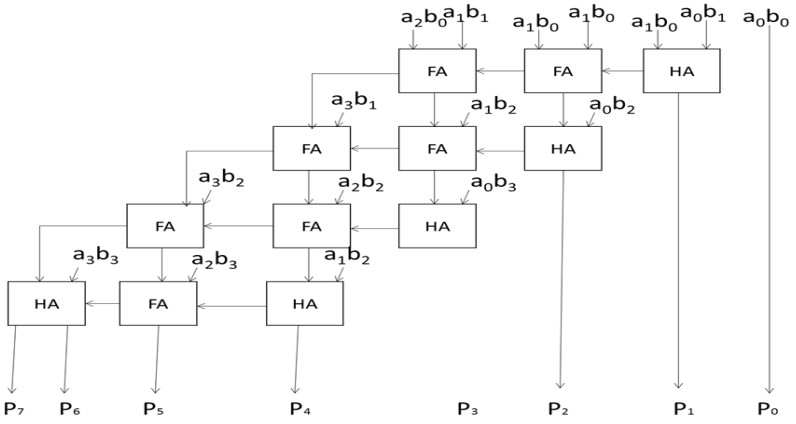

Figure 1. Design of array multiplier

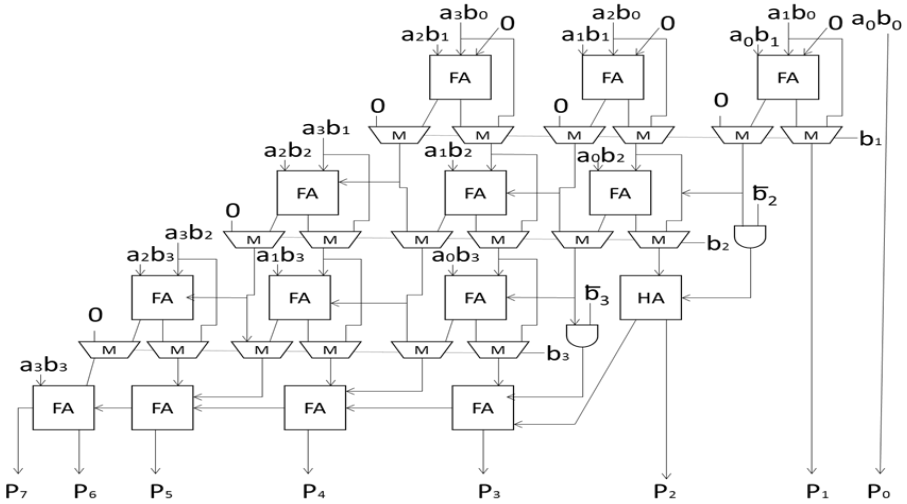

Figure 2. Design of 4*4 row bypassing multiplier

Journal of Graphic Era University
Vol. 4, Issue 1, 49-56, 2016
ISSN: 0975-1416 (Print)

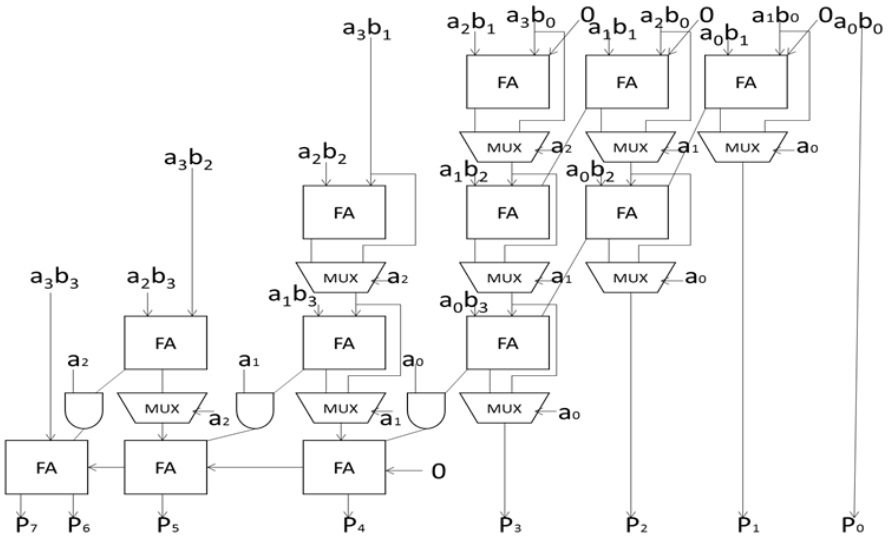

Figure 3. Design of 4*4 column bypassing multiplier

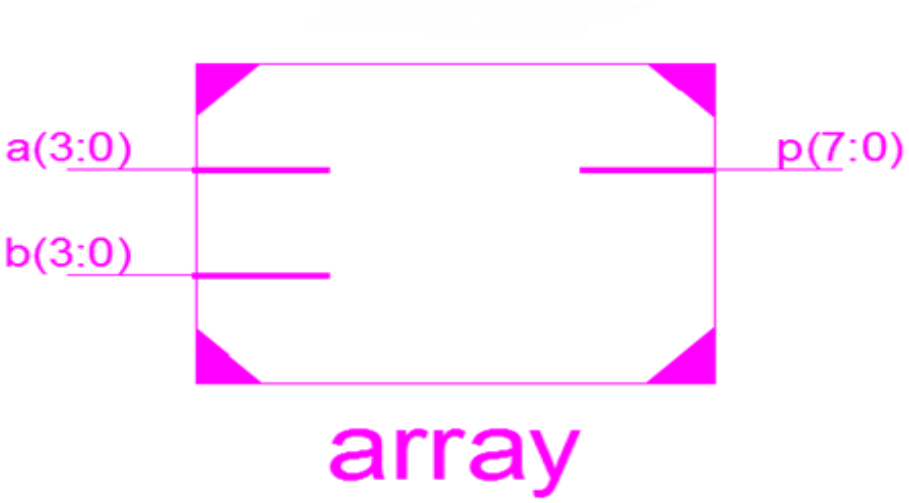

Figure 4. Block diagram of array multiplier

Figure 5. Output waveform of array multiplier

Journal of Graphic Era University
Vol. 4, Issue 1, 49-56, 2016
ISSN: 0975-1416 (Print)

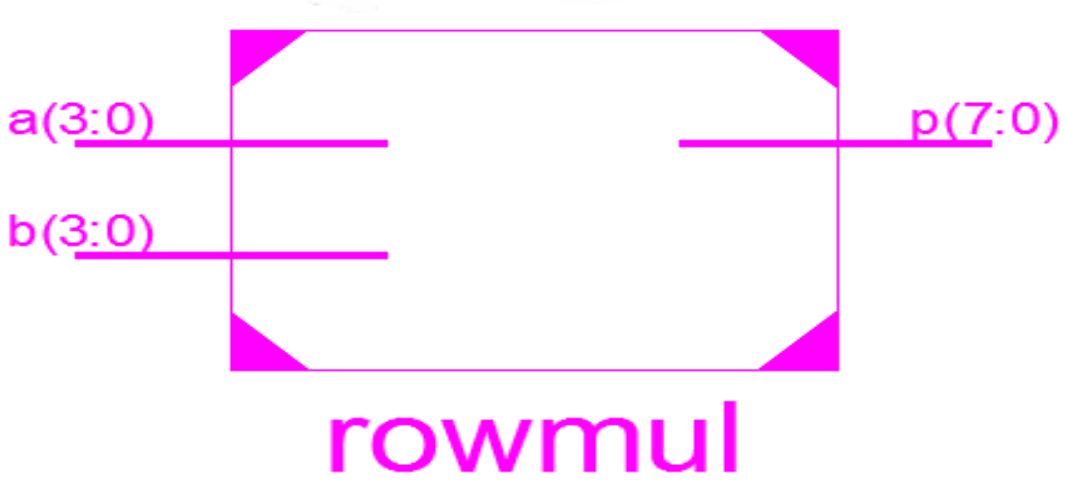

Figure 6. Block diagram of 4*4 row bypassing multiplier

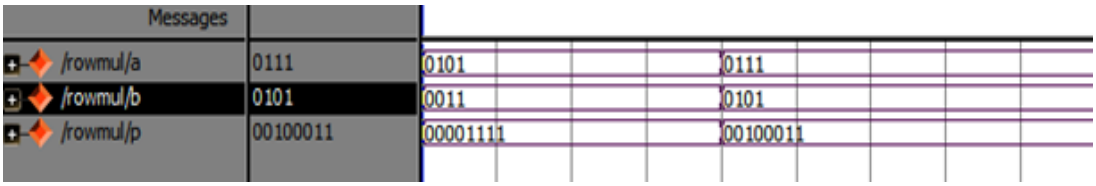

Figure 7. Output waveform of 4*4 row bypassing multiplier

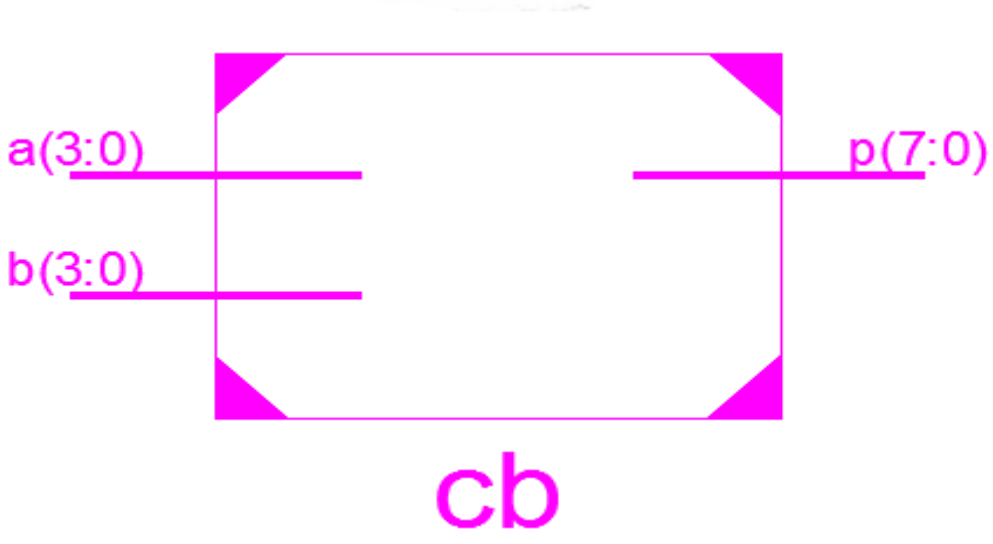

Figure 8. Block diagram of 4*4 column bypassing multiplier

Journal of Graphic Era University
Vol. 4, Issue 1, 49-56, 2016
ISSN: 0975-1416 (Print)

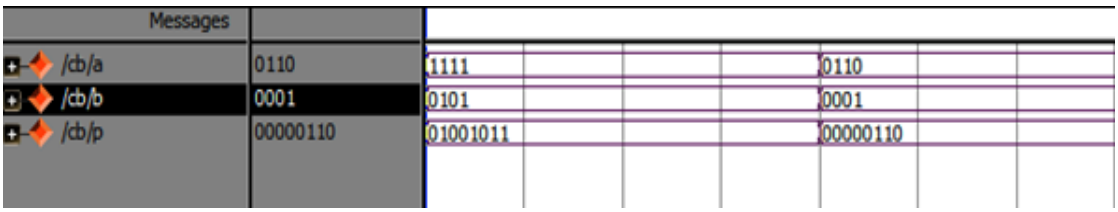

Figure 9. Output waveform of 4*4 column bypassing multiplier

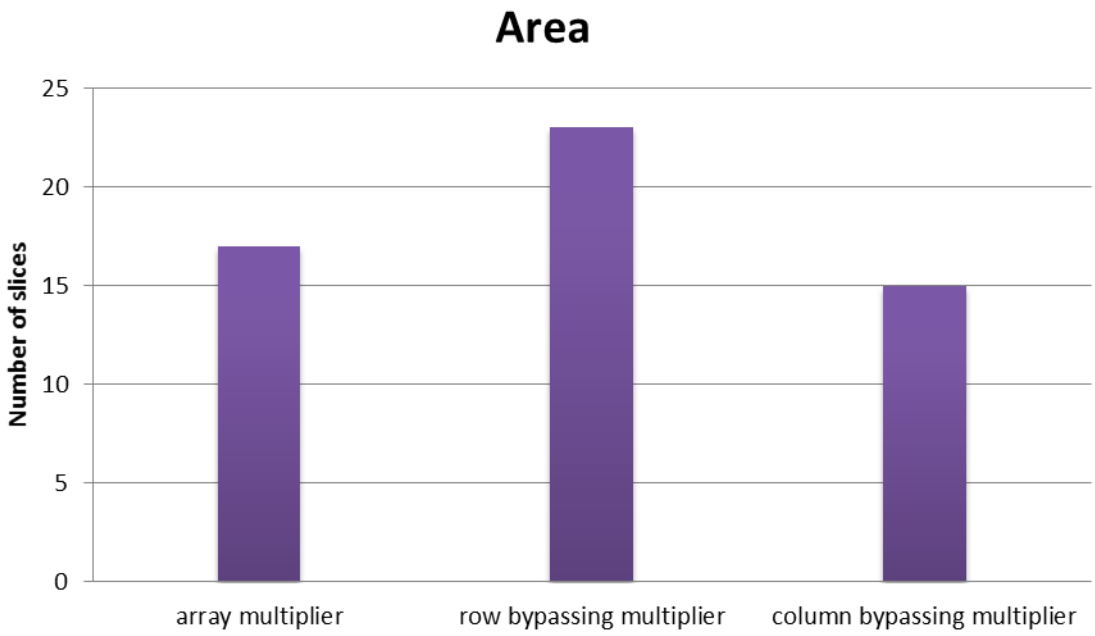

Figure 10. Comparision of area utilization in terms of number of slices of different multipliers

References

Chirde, V. S., & Jadhav, U. (2015, September). Design of a multiplier with Adaptive Hold Logic (AHL) circuit to reduce aging effects. In Computer, Communication and Control (IC4), 2015 International Conference on (pp. 1-5). IEEE.

Devi, T. M., Renganayaki, M. G., & Tech, M. (2015). Design of Low Power Vedic Multiplier Using Adaptive Hold Logic. International Journal of Emerging Technology in Copmuter Science and Electronics (IJETCSE), 248-252.

Goyal, C., Kaur, G. P., (2012). Comparative analysis of low power 4-bit multipliers using 120 nm CMOS technology. International journal of engineering research and applications (IJERA), 551-555.

Lin, C., Cho, Y. H., & Yang, Y. M. (2015). Aging-aware reliable multiplier design with adaptive hold logic. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 23(3), 544-556.

Nujum, R., Cheriyan, J. (2013) Low Power Variable Latency Multiplier with AH Logic. International Journal of Science and Research (IJSR), 2319-7064.

Ramesh, A. P. (2011). Implementation of dadda and array multiplier architectures using TANNER tool. International Journal of Computer Science and Engineering Technology (IJCSET), 2(3), 28-41.

Sahu, P. K., Meena, N. (2013). Comparative Study of different multiplier architectures. International Journal of Engineering and Technology, 4(10), 4293- 4297.