AN EFFICIENT BIT RATE PERFORMANCE OF SERIAL-SERIAL MULTIPLIER WITH 1'S ASYNCHRONOUS COUNTER

P.Rajesh[1] and B.Gopinath[2]
[1]Department of ECE, SNS Tech, Coimbatore City, T.N., India
[2]Assistant professor, SNS Tech, Coimbatore, T.N., India

ABSTRACT

Traditional Serial-Serial multiplier addresses the high data sampling rate. It is effectively considered as the entire partial product matrix with n data sampling cycle for n×n multiplication function instead of 2n cycles in the conventional multipliers. The existing Serial-Serial multiplier is the first bit serial structure. Newly developed serial-serial multiplier design is capable of processing input data at (GBs) without buffering and with reduced total number of computational cycle. This multiplication of partial products by considering two series inputs among which one is starting from LSB the other from MSB. Using this feed sequence and accumulation technique it takes only n cycle to complete the partial products. It achieves high bit sampling rate by replacing conventional full adder and highest 5:3 counters. Here asynchronous 1's counter is presented. This counter takes critical path is limited to only an AND gate and D flip-flops. Accumulation is integral part of serial multiplier design. 1's counter is used to count the number of ones at the end of the nth iteration in each counter produces. The proposed multiplier consists of a serial-serial data accumulator module and carries save adder that occupies less silicon area than the full carry save adder. In this paper we proposed model address for the 8bit 2's complement implementing the Baugh-wooley algorithm and unsigned multiplication implementing the proposed architecture for 8×8 Serial-Serial unsigned multiplication. We can able to extend the 16bit multiplication.

KEYWORDS: *Binary multiplication, Serial-Serial multiplication, Serial-Parallel multiplication; Asynchronous counter; Serial-Link bus architecture.*

I. INTRODUCTION

Serial-serial multiplication techniques have been in use for many years . The proposed a structure based on 1's counters , which computed the N bits of the product of an NxN multiplication in N bits clock cycles, using N processing cells. One cell was dedicated to the computation of each output bit, In this paper, the design of a bit serial-serial multiplier which utilises a bit serial, least significant bit first format for both input and output based on particular methods. The design and implementation approaches of multipliers contribute substantially to the area, speed and power consumption of computational intensive VLSI systems. Often the delay of multipliers dominates the critical path of this System and due to issues reliability and portability.

Power consumption is a critical criterion for applications that demand low-power as its primary metric. While low power and high speed multiplier circuits are highly demanded, it is not always possible to achieve both criteria simultaneously. Therefore, a good multiplier design requires some tradeoffs between speed and power consumption. The main objective of designing bit-parallel multipliers which makes them different from bit-serial and digit-serial multipliers. Bit-serial multipliers provide the lowest possible area complexity.

In the literature, the bit-serial algorithms have been studied for the polynomials basis and two different algorithms have been proposed, namely the Least Significant Bit (LSB) first and the Most Significant Bit (MSB) first bit-serial Algorithms. Fast arithmetic circuits are key elements of high performance computers and data processing systems. In the majority of these applications, multipliers have been a critical and obligatory component in dictating the overall circuit performance when constrained by power consumption and computation speed. Compressors are a critical component of the multiplier circuit, which greatly influence the overall multiplier speed. The authors propose two novel high performance 5:2 compressor architectures. The main objective of their designs is to limit the carry propagation to a single stage, thereby reducing the overall propagation delay.

Based on the partial product formation multipliers are classified into two types. They are serial and parallel multiplier. Parallel multipliers are popular for their high speed operation but for long word length they are limited by hardware cost and power consumption of the applications. Serial multipliers are useful in low cost and low power applications. Several architectures have been reported in the literature for the implementation of serial multipliers but do not find any significant work in the reduction of partial product formation. Each architecture produce the partial product in 2N cycles for an N x N multiplication.

A new approach to form the entire partial product matrix in just N sampling cycles for an N x N multiplication instead of at least 2N cycles in the conventional serial-serial multipliers. A new approach to serial accumulation of data by using asynchronous counters is suggested here which essentially count the number of 1's in respective input sequences. The counters effectively replace the full adders in the accumulation circuit. In this project the partial products are formed in serial manner and they are accumulated by the counters and finally the product can be obtained by adding the accumulated output by the ripple carry adder. Serial multipliers are popular for their low area and power. They are broadly classified into two categories, namely serial-serial and serial-parallel multiplier. In a serial-serial multiplier both the operands are loaded in a bit-serial fashion, reducing the data input pads. To reduce the number of computational cycles from 2n to n in an n×n serial multiplier, several serial-parallel multipliers.

II. RELATED WORK

A fast serial-parallel multiplier based on the Baugh-Wooley algorithm is proposed. It is shown that sign extension of the sum or carry bit, produced during the addition of the bit product rows, can be avoided, and a more efficient multiplier can be obtained. We present a two's-complement FSP multiplier that is based on the well known Baugh-Wooley algorithm. In the proposed multiplier, there is no need to sign extend the sum or carry bits of the partial product sums.

The technique described can be extended to two's complement multiplication. The sign of the two's complement number is extended either directly, as the extension of the sign bit of the bit-product sum, or created as a result of a carry generated by previous row addition. Main contribution of our wok will follow,

2.1. Major participation of our work

- An FSP multiplier based on the Baugh-Wooley algorithm has been developed. It has been shown that there are 2's complement sign extensions of the sum or the carry bits produced during the addition of two-bit product row.
- 1's counter is used to count the number of ones at the end of the nth iteration in each counter produces. The proposed multiplier consists of a Serial-Serial data accumulator module and carries save adder that occupies less silicon area than the full carry save adder.
- Unsigned multiplier is based on the proposed architecture for 8×8 serial-serial unsigned multiplication.
- It is effective multiplier for a time, area and delay analysis technique.

2.2. Compressor architecture

In conventional method consist of the several architectures and designs of low-power 4-2 and 5-2 compressors capable of operating at ultra low supply voltages. These compressor architectures are anatomized into their constituent modules and different static logic styles based on the same deep sub micrometer CMOS process model are used to realize them. Different configurations of each architecture, which include a number of novel 4-2 and 5-2 compressor designs, are prototyped and simulated to evaluate their performance in speed, power dissipation and power-delay product. The newly developed circuits are based on various configurations of the novel 5-2 compressor architecture with the new carry generator circuit, or existing architectures configured with the proposed circuit for the exclusive OR (XOR) and exclusive NOR (XNOR) [XOR–XNOR] module. The proposed new circuit for the XOR–XNOR module eliminates the weak logic on the internal nodes of pass transistors with a pair of feedback PMOS–NMOS transistors. Driving capability has been considered in the design as well as in the simulation setup so that these 4-2 and 5-2 compressor cells can operate reliably in any tree structured parallel multiplier at very low supply voltages. Two new simulation environments are created to ensure that the performances reflect the realistic circuit operation in the system to which these cells are integrated. Simulation results show that the 4-2 compressor with the proposed XOR–XNOR module and the new fast 5-2 compressor architecture are able to function at supply voltage as low voltage.

2.2.1. 4-2 Compressor architectures

A 4-2 compressor has five inputs and three outputs, as shown in “Fig. 1”, The four inputs x1,x2 ,x3 , and x4 and the output sum have the same weight. The output carry is weighted one binary bit order higher.

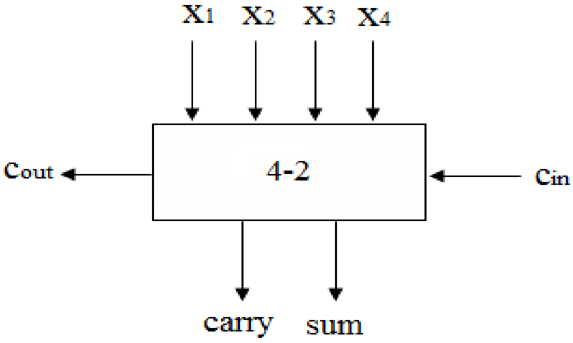


Figure.1 4-2 compressor

The conventional implementation of a 4-2 compressor is composed of two serially connected full adders, at gate level, high input compressors are anatomized into XOR gates and carry generators normally implemented by multiplexers (MUX). Therefore, different designs can be classified based on the critical path delay in terms of the number of primitive gates. Since the difference between the delays of widely used XOR gate and carry generator is trivial in an optimized design, the delay of the compressor is more commonly specified as (m+n)Δ. The throughput rate has been increased by judicially changing the way the partial products are accumulated. Main benefit of this project is it occupies less area and fast operation. It considers the 16 bit operation also it must be very efficient. The bit consideration is very important for any multiplication. Each iteration required single n cycle. The two carry signals carry and Cout is generated from both the XOR and XNOR functions of the input signals. The Sum output is generated by several two-input XOR circuits, some internal signals of which can be used to generate the two carries. “Fig.2”, shows the logic decomposition of this 4-2 compressor architecture. It is mainly composed of six modules, four of which are XOR circuits and the other two are 2-1 MUX. Three special XOR–XNOR modules marked with “X-OR” generate both the XOR and XNOR signals simultaneously to other modules driven by them.

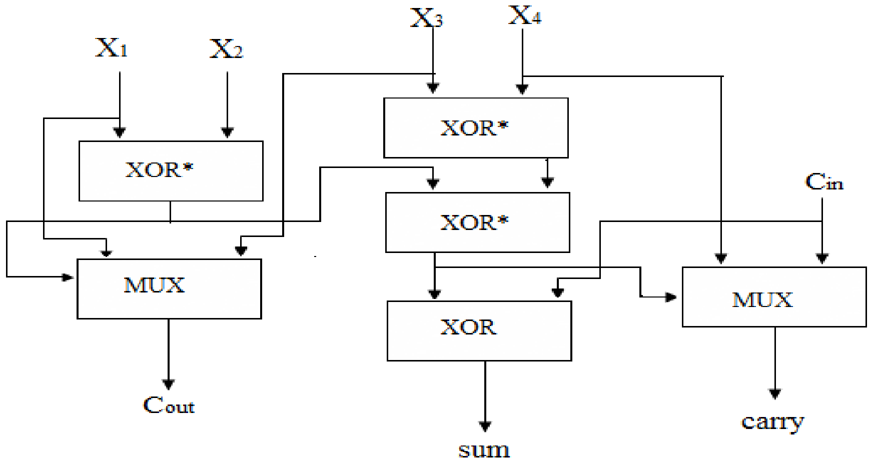


Figure.2 Logic decomposition of 4-2 compressor

2.2.2. 5-2 Compressor architectures

The 5-2 compressor is another widely used building block for high precision and high speed multipliers. The block diagram of a 5-2 compressor is shown in "Fig.3", which has seven inputs and four outputs.

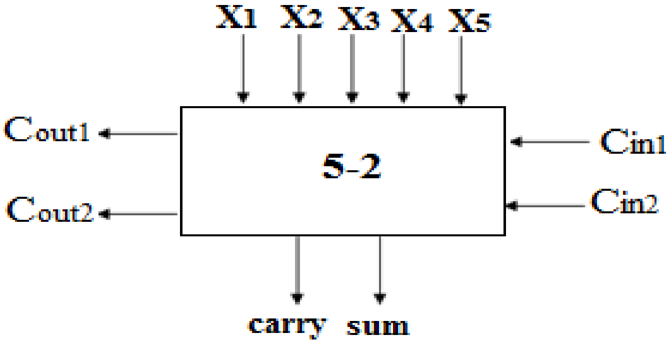


Figure.3 5-2 compressor

Five of the inputs are the primary inputs x1, x2, x3, x4 and x5, and the other two inputs, Cin1 and Cin2 receive their values from the neighbouring compressor of one binary bit order lower in significance. All the seven inputs have the same weight. The 5-2 compressor generates an output of the same weight as the inputs, and three outputs Carry, Cout1, Cout2 and weighted one binary bit order higher.

A third widely used compressor of significant importance is the 5-2 compressor. Its block diagram is shown in "Fig.3", it has seven inputs of which five are direct inputs and two are carry-in bits from a previous stage. Similarly, there are four outputs of which two are carry-out bits to the next stage and the other two are sum and carry bits. A simple implementation of the 5-2 compressor is to cascade three full adders in a hierarchical structure, as shown in "Fig.4", which has a critical path delay of 6Δ.

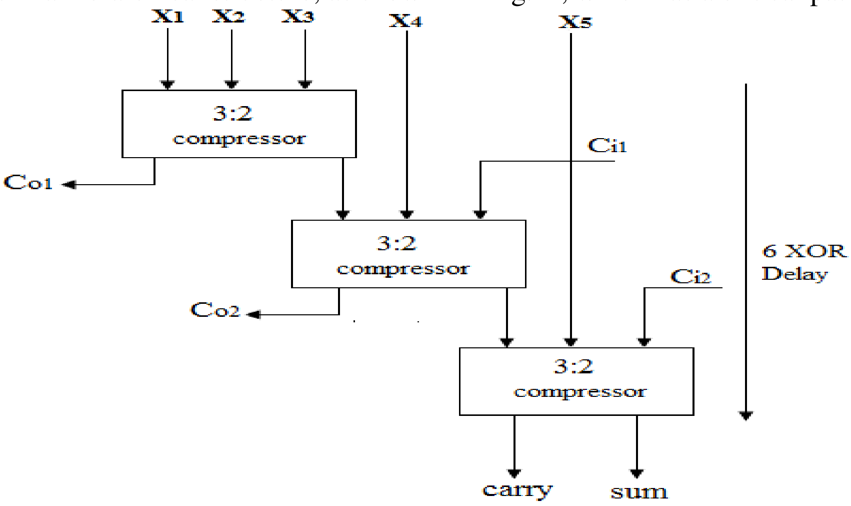


Figure.4 implementing using 3:2 compressors

In this simplest form, a 5:2 compressor can be designed by cascading three 3:2 compressors as shown in Fig. 5. This structure has a delay of 6 XORs and is slower than the 6:2 compressor presented in delayed convention which has a delay of only five XORs. A faster implementation of the 5:2 compressors with 5 XOR delays is presented in [7]. The main objective in our designs is to limit the carry propagation. In the conventional implementation of the 5:2 compressors using 3:2 compressors shown in Fig.5, the first compressor generates Co1, the second generates Co2 and the third generates the Sum and Carry. The second compressor receives inputs x4, ci1 and the output from the first compressor. Hence, Co2 is a function of ci1, thereby propagating ci1 across the compressor. These types of compressors require many numbers of inputs and outputs. So it is complex structured than 1's counter. It can possible to count the number of 1's.

2.3. Review of serial multiplier

In a serial-serial multiplier both the operands are loaded in a bit-serial fashion, reducing the data input pads to two serial multipliers are popular for their low area and power. Bit-serial processing can result in efficient communications, both within and between VLSI chips, because of the reduced number of interconnections required. Serial multiplier designs which are particularly suitable for applications where input data are sequentially presented .The operating speeds are determined mainly by the propagation delays along the critical path within the processing elements. A major advantage offered by bit serial processors is when the operands are available only one bit at a time, the processing speed can be improved using bit-serial arithmetic elements. The structures that use this approach can achieve moderate speeds with comparatively small area.

2.3.1. Implementation stages for serial- serial multiplier

This operation consists of three stages

i) The generation of partial products.

ii) The reduction of partial products.

iii) Final carry-propagation addition.

The partial products can be generated either in parallel or serially, depending on the target application and the availability of input data. The partial products are generally reduced by carry-save adders using an array or a tree structure. Carry propagation addition is inevitable when the number of partial products is reduced to two rows. This final adder can be a simple ripple carry adder for low power or a carry look-ahead adder for high speed. It is highly desirable to reduce the number of partial products before the carry-save adder’s stage. The drawback is that and higher order compressors are slower and consumes more power than the full adders. An accumulator is an adder which successively adds the current input with the value stored in its internal register.

2.3.2. Serial multiplier with counter based accumulator

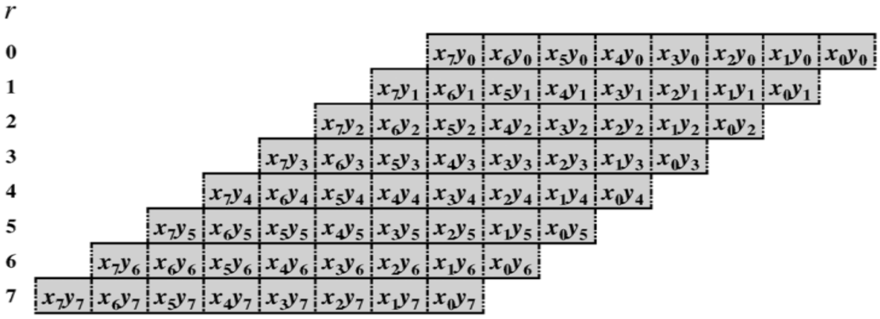
In this type of multipliers the partial products are formed in a serial manner. Instead of using full adders counters are used to count the number of 1’s in a column in this structure. This method provides power minimization from the fact that the counter output will not toggle unless a ‘1’ is present at its input. The counter output is applied to a ripple carry adder to obtain the final product.By using this counter based accumulator the partial products are formed in N cycles for an NXN multiplication. A new approach to serial/parallel multiplier design by using parallel 1’s counters to accumulate the binary partial product bits. The 1’s in each column of the partial product matrix due to the serially input operands are accumulated using a serial T-flip flop (TFF) counter.

2.4. Proposed serial-serial multiplier

Accumulation is an integral part of serial multiplier design. A typical accumulator is simply an added that successively adds the current input with the value stored in its internal register. Generally, the adder can be a simple RCA but the speed of accumulation is limited by the carry propagation chain. The accumulation can be speed up by using a CSA with two registers to store the intermediate sum and carry vectors, but a more complex fast vector merged adder is needed to add the final outputs of these registers. A new approach to serial accumulation of data by using asynchronous counters is suggested here which essentially count the number of 1’s in respective input sequences.

2.4.1. 8 and 16 bit word length for unsigned multiplier

A new technique of generating the individual row of partial products by considering two serial inputs, one starting from the LSB and the other from MSB. It takes only n cycles to complete the entire partial product generation.



r															
0								x_7y_0	x_6y_0	x_5y_0	x_4y_0	x_3y_0	x_2y_0	x_1y_0	x_0y_0
1							x_7y_1	x_6y_1	x_5y_1	x_4y_1	x_3y_1	x_2y_1	x_1y_1	x_0y_1	
2						x_7y_2	x_6y_2	x_5y_2	x_4y_2	x_3y_2	x_2y_2	x_1y_2	x_0y_2		
3					x_7y_3	x_6y_3	x_5y_3	x_4y_3	x_3y_3	x_2y_3	x_1y_3	x_0y_3			
4				x_7y_4	x_6y_4	x_5y_4	x_4y_4	x_3y_4	x_2y_4	x_1y_4	x_0y_4				
5			x_7y_5	x_6y_5	x_5y_5	x_4y_5	x_3y_5	x_2y_5	x_1y_5	x_0y_5					
6		x_7y_6	x_6y_6	x_5y_6	x_4y_6	x_3y_6	x_2y_6	x_1y_6	x_0y_6						
7	x_7y_7	x_6y_7	x_5y_7	x_4y_7	x_3y_7	x_2y_7	x_1y_7	x_0y_7							

Figure.5 conventional partial product formation

The PP generation of an 8×8 multiplier for two unsigned numbers X and Y. "Fig.5", shows the conventional partial product formation and "Fig. 6", shows the generation sequence of the PPs. Row r generated in cycle r , for r=0,1……..n-1.

The PPs in "Fig.7", are generated in such an unconventional way in order to facilitate their accumulation on-the-fly by the proposed counter-based accumulation technique. A PP bit corresponding to the middle column of the PP is produced by the centred AND gate.

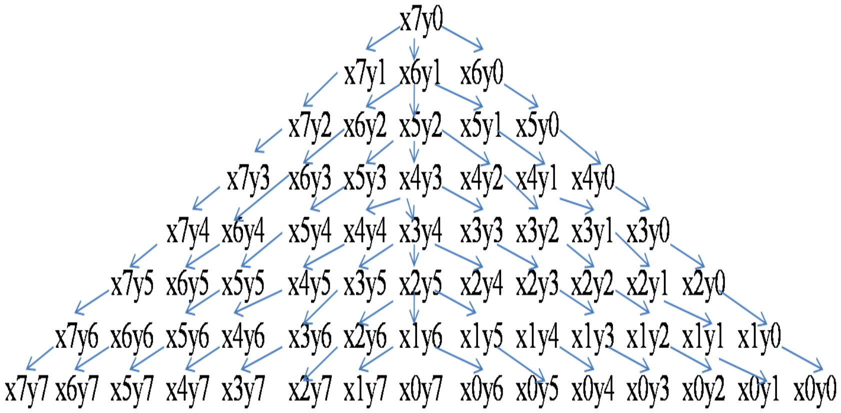

x7y0
x7y1 x6y1 x6y0
x7y2 x6y2 x5y2 x5y1 x5y0
x7y3 x6y3 x5y3 x4y3 x4y2 x4y1 x4y0
x7y4 x6y4 x5y4 x4y4 x3y4 x3y3 x3y2 x3y1 x3y0
x7y5 x6y5 x5y5 x4y5 x3y5 x2y5 x2y4 x2y3 x2y2 x2y1 x2y0
x7y6 x6y6 x5y6 x4y6 x3y6 x2y6 x1y6 x1y5 x1y4 x1y3 x1y2 x1y1 x1y0
x7y7 x6y7 x5y7 x4y7 x3y7 x2y7 x1y7 x0y7 x0y6 x0y5 x0y4 x0y3 x0y2 x0y1 x0y0

Figure.6 Proposed partial product generation

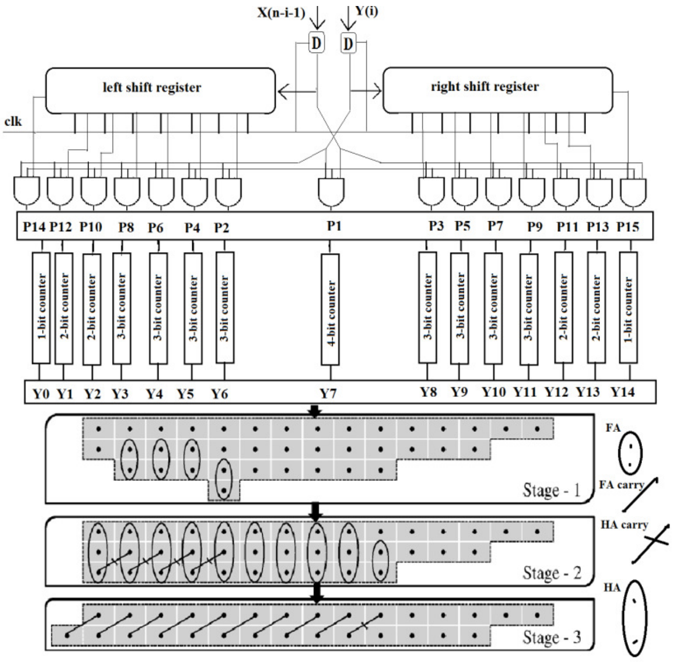


Figure.7 Proposed architecture for 8 × 8 serial-serial unsigned multiplication

The latched outputs are wired to the correct FAs and HAs (half adders) according to the positional weights of the output bits produced by the counters. From "Fig.7", it is observed that the column height has been reduced from 8 to 4 and the final product, can be obtained with two stages of CSA tree and a final RCA. Similarly, for 16×16, 32×32 and 64× 64 multipliers the column heights are reduced logarithmically from 16, 32, and 64 to 5, 6, and 7, respectively. This drastic reduction in column height leads to a much simpler CSA tree, and hence reducing the overall hardware complexity and power consumption.

The latching register between the counter and the adder stages not only makes it possible to pipeline the serial data accumulation and the CSA tree reduction, but also prevents the spurious transitions from propagating into the adder tree.

2.4.2. 8 and 16 bit word length for 2's complement numbers

The Most digital systems operate on signed numbers commonly represented in 2's complement. 2's complement numbers using the Baugh–Wooley algorithm. The architecture of the proposed 2's complement serial-serial multiplier is depicted below structure. The addition of the term raises the height of CSA tree by only two bits regardless of the word length of the operands.

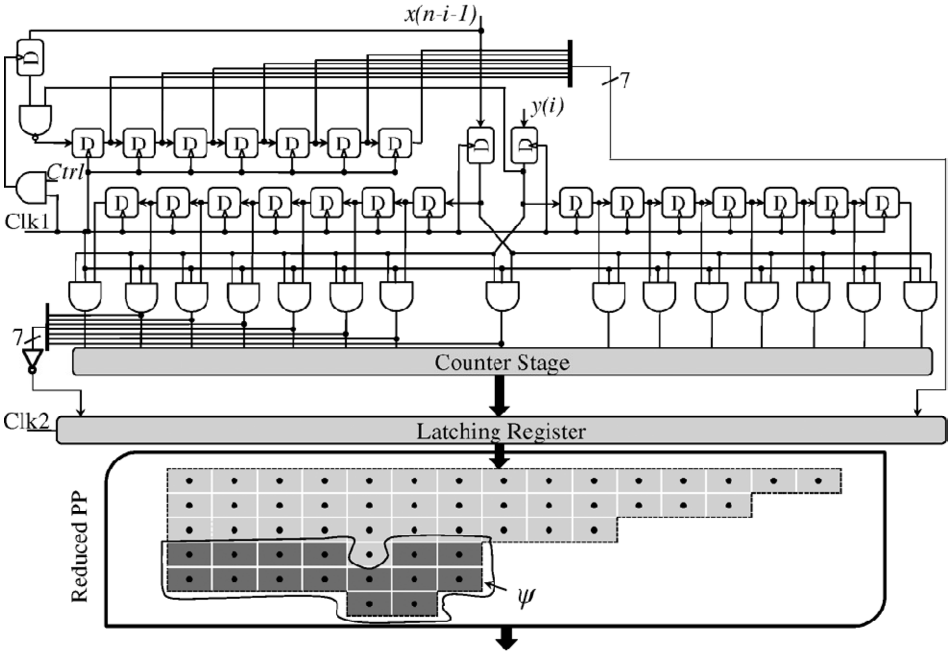


Figure.8 Proposed architecture for 8×8 serial-serial 2's complement multiplication

2.5. Simulation result.

Table 1: Proposed delay cycle of the serial-serial multiplication

Method	Bit rate	Operating Mode	Cycles
Proposed	8×8	Unsigned	N
Proposed	8×8	2's complement	N
Proposed	16×16	Unsigned	N
Proposed	16×16	2's complement	N

The both 8 bit and 16 bits are proposed based on their respective algorithms and architecture. 2's complement is using to implement in baugh-wooley algorithm. Unsigned multipliers are using to implement in proposed architecture. These different word lengths are support for the 1's counter. Both 8 bit and 16 bit outputs are getting 8 clock pulses(n cycles) so required time is vey minimum. In previous session did only 8bit multiplication here we completed 16 bit also.

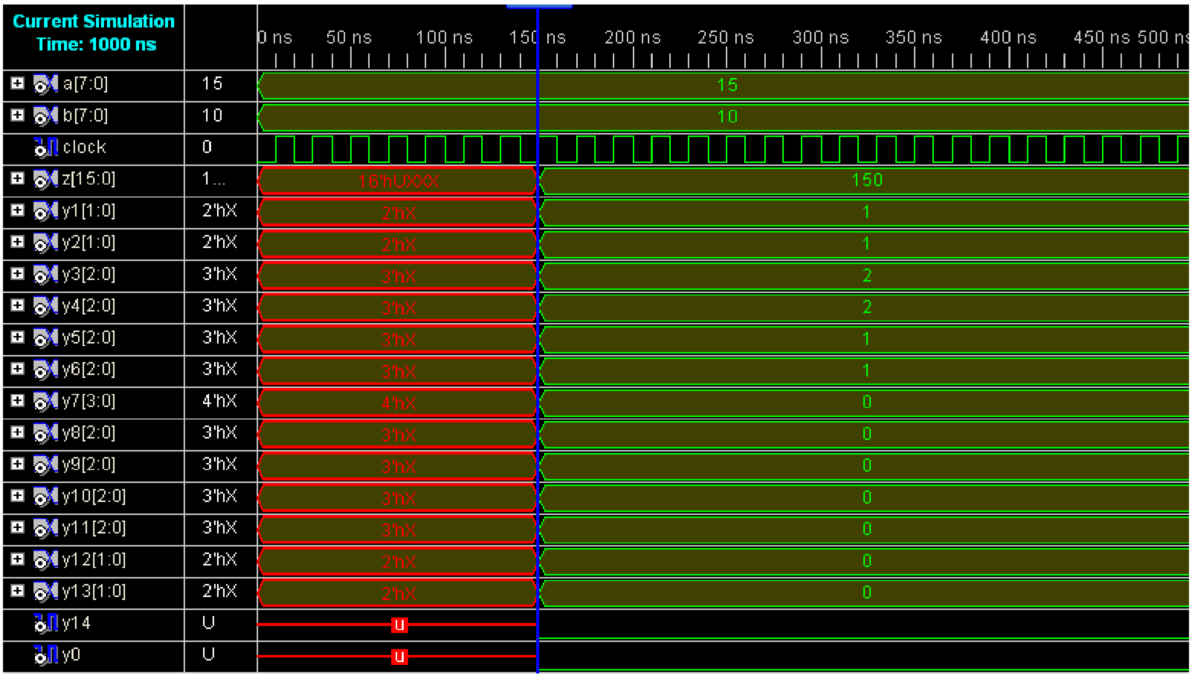


Figure.9 Snapshot of the output sequence of the proposed multiplier for 8 bit unsigned numbers

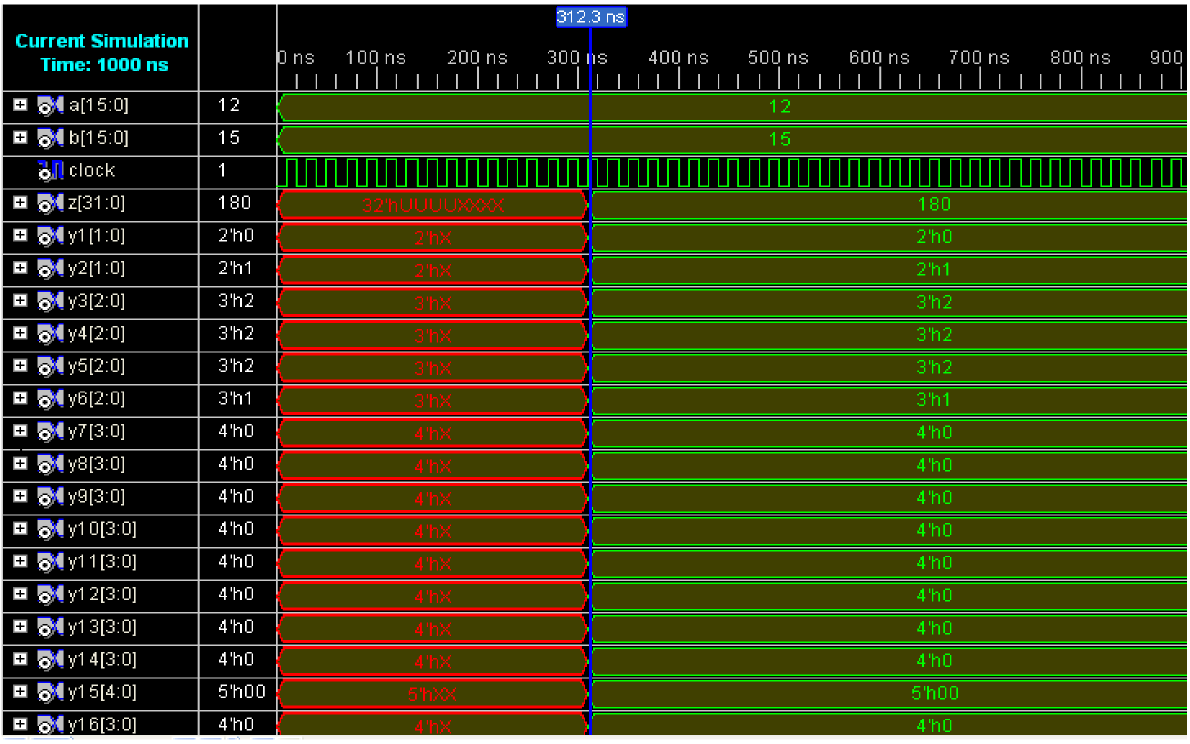


Figure.10 Snapshot of the output sequence of the proposed multiplier for 16 bit multiplier

III. Conclusion

In this paper, we have studied for computing serial-serial multiplication is introduced by using low complexity asynchronous counters. By exploiting the relationship among the bits of a partial product matrix, it is possible to generate all the rows serially in just n cycles for an n×n multiplication. Employing counters to count the number of 1's in each column allows the partial product bits to be generated on-the-fly and partially accumulated in place with a critical path delay of only an AND gate and a DFF. The counter-based accumulation reduces the PP height logarithmically and makes it possible to achieve an effective reduction rate. The proposed counter-based multiplier performs many serial-serial and serial-parallel multipliers in speed but its hybrid architecture does carry an area overhead.

REFERENCES

[1] A High Bit Rate Serial-Serial Multiplier with On-the-Fly Accumulation by Asynchronous Counters', in proc. IEEE Conf. 2010.

[2] Aggoun.A, Ashur.A, and Ibrahim,M.K (2000),'Area-time efficient serial-serial multipliers,' in Proc. IEEE Conf. Circuits Syst. (ISCAS), Geneva, Switzerland, pp. 585–588.

[3] Almiladi.A, Ibrahim.M.K,Al-Akidi.M, and Aggoun.A (2007),'High performance scalable bidirectional mixed radix-serial serial multipliers,'IET Proc-Comput. Digit. Tech., vol.1, no. 5, pp. 632–639.

[4] Bi.G and Jones.E.V, 'High-performance bit-serial adders and Multipliers (1992),' IEEE Proc G-Circuits Devices Systs, vol. 139, no. 1, pp.109–113.

[5] Dobkin.R, Moyal.M, Kolodny.A, and Ginosar.R (2010), 'Asynchronous current mode serial communication,' IEEE Trans. Very Large Scale Integer. (VLSI) Syst., vol. 18, no. 7, pp. 1107–1117.

[6] Ghoneima.M, Ismail.Y, Khellah.M, Tschanz.J, and De.V (2009),'Serial link bus: A low-power on-chip bus architecture,' IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 56, no. 9, pp. 2020–2032.

[7] Gnanasekaran.R (2006), 'On a bit-serial input and bit-serial output multiplier,' IEEE Trans. Compute., vol. C-32, no. 9, pp. 878–880.

[8] Menon.R and Radhakrishnan.D (2006), 'High performance 5:2 compressor architectures,' IEEE Proc-Circuits Devices Syst., vol. 153, no. 5, pp. 447–452.

[9] Nibouche.O, Bouridarie.A, and Nibouche.M (2001), 'New architectures for serial-serial multiplication,' in Proc. IEEE Conf. Circuits Syst. (ISCAS), Sydney, Australia, vol. 2, pp. 705–708.

[10] Saleh.H.I, Khalil.A.H, Ashour.M.A, and Salama.A.E (2001),'Novel serial-parallel multipliers,' IEEE Proc-Circuits Devices Syst., vol. 148, no. 4, pp. 183–189.

[11] Sunder.S, El-Guibaly.F, and Antoniou.A (1995), 'Two's-complement fast serial-parallel multiplier,' IEEE Proc.-Circuits Devices Syst, vol. 142, no. 1, pp. 41–44.

Authors

Rajesh received the B.E degree from P.S.R College of technology, Tamilnadu, India, in 2009. Now doing M.E degree for VLSI design in S.N.S College of technology, Tamilnadu, India. Am atttened many conferences for both national and international journals.Now doing my research work.