www.ijaceeonline.com ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

ASIC Implementation of Processing Unit Using Radix Decimal Multiplier

K. Saranya
Assistant Professor, Department of Electrical and Electronics Engineering,
Dr. Mahalingam College of Engineering and Technolgy, Pollachi, India.
Email: saranya@drmcet.ac.in

R. Baby Janagam
Assistant Professor, Department of Electronics and Instrumentation Engineering,
Dr. Mahalingam College of Engineering and Technolgy, Pollachi,India.
Email: jana@drmcet.ac.in

Abstract: *Decimal Multipliers finds its applications commercially and financially in vast domains. Decimal hardware multiplier unit is used as important unit in general purpose processing units where iterative multiplication is involved. We develop the Binary Coded Decimal fully parallel multiplier that uses the redundant BCD codes. The Partial Products Generation is done in parallel using a new radix-10 recoding method. This recoding method helps to reduce the power delay product .It act as efficient multiplicand multiples generating unit. The conventional binary carry-save adder is adapted to perform final addition. To show the effectiveness of our design, RTL synthesis is done. The comparative analysis of radix-5 and radix-10 BCD multiplication method using the CADENCE EDA Tool in 180nm technology is performed. Processing unit is implemented by using proposed radix-10 multiplier to perform the addition, subtraction, multiplication and accumulation.*

Keyword: *Carry Save Adder; Decimal Multiplier; Parallel Multiplier Radix multiplication; Recoder; Redundant Codes.*

1. INTRODUCTION

BCD codes are 4 bit decimal codes and it is used in wide areas and enormous applications to handle decimal data. Computation required for arithmetic operation is reduced by usage of BCD. For the storage, shifting and almost all the arithmetic operations to simple operation of decimal data, BCD remains as a suitable encoding format to use in real time. In the recent situation, many devices are performing financial and commercial operations in real time.

The BCD multiplier algorithm that inherits the properties of two different redundant Binary codes is proposed in [1]. The architecture redundant of BCD excess-3 codes (XS-3) are used. It reduces significantly the latency and area of multipliers. Partial products are generated concurrently using a signed digit radix recoding of the BCD multiplier. This encoding has numerous advantages. As it is a self-complementing the negative multiplicand multiple can be obtained by merely inverting the bits of the corresponding positive one using XOR gates. Also, this allows a fast and simple generation of multiplicand multiples without generating the carry. The usage of Ex-3 coding is eliminated in the proposed system and simple recoding method is employed to achieve the performance optimization.

The architectures of two parallel decimal multipliers that performs radix recoding of the multiplier and a simplified set of multiplicand multiples is implemented in [2]. The simplification of partial products is implemented on a multiple operand type carry-save addition tree algorithm that uses non BCD decimal-coded number systems. The design methodology combines all these techniques to obtain efficient reduction with different area and delay trade-offs that is independent of number of partial products generated. All the needed decimal multiplicand multiples, except the 3X multiple, are obtained. It uses few steps for simplification of combinational logic .It involves 4221 weighted binary and 5211weighted binary for performing left shifts in the bit vector representation of operands. This paper uses too many recoders to

Cite this paper:

K.Saranya, R.Baby Janagam, "ASIC Implementation of Processing Unit Using Radix Decimal Multiplier", International Journal of Advances in Computer and Electronics Engineering, Vol. 2, No. 10, pp. 15-22, October 2017.

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

transform the multiplicand into4221,5211 etc thus increasing the delay. The 3X multiple generation is also critical in this method.

A combinational type decimal multiplication is developed in [3]. It uses the pipelining concept to achieve the needed throughput. Though the different implementations of decimal multiplication structures are available, the unit uses the parallel method and implements simpler recoding of the operands to reduce the number of partial product computations. The concept of carry independent accumulation of the partial products is realized using carry-save adders (CSA).It add a carry save operand plus another BCD operand to produce a carry save result.

The multiplier is implemented by accumulation of partial products in [4].It is performed by multiplying the whole multiplicand by a weighted digit of the multiplier. It uses the combinational logics. It is delay optimized and also area-optimized BCD digit multipliers. It has eight input bits and eight output bits. A general direct approach for decimal partial product generation is prior computation of all the possible 10 multiples (0,1X,2X,3X,4X,5X,6X,7X,8X,9X) that multiplicand needs at the outset of the multiplication process. The appropriate partial product is picked by implementing a 10-way selector. It uses double BCD representation at the end of reduction process, instead of the BCDCS method of encoding. Though this has speed as advantage cost of area consumption is more.

The novel designs for fixed-point decimal multiplication that makes use of decimal carry-save addition to decrease the critical path delay in [5]. The multiplicands multiples are stored and uses decimal carry-save addition iteratively. Then it uses decimal (4:2) compressors and decimal carry-propagate addition to produce the final product. The use of compressors and adders increase the delay as well as the area in this method, Decimal arithmetic operations have become essential in the computation of many business and commercial applications, in order to maintain decimal rounding and precision. 4-bit Decimal codes satisfy that the sum of the weights of the bits are from 0 to 9. The 10 4-bit combinations from 0000 to 1001 represent a decimal digit in [0:9]. These codes have been used in the fast processing of decimal multiple operand addition and multiplication operation [2], [6], [7].

IEEE 854,IEEE 754-2008 Standards are universally followed for decimal floating-point arithmetic [13] [14], [15]. Digit wise arithmetic multiplication and division are performed recurrently leading to very low performance. Area and power dissipation are the optimization parameters considered in recent Decimal Floating Point Units. Hardware implementations use Binary Coded Decimal instead of weighted binary for decimal fixed-point manipulation and for conversion between machine and human representation [12], [11]. The same data path width is maintained and used to increase the speed and decrease the delay caused by redundancy in 4 bit encoding methodology. BCD multiplication can be done in parallel and sequential manner. Sequential BCD multipliers are smaller than equivalent parallel implementations, but have higher latency and reduced throughput thus parallel multipliers are mostly frequently used.

Section 2 explains the concepts used in the Tomas Lang and Alberto Nannarelli (2006) [3] multiplier and Vazquez (2014) [1] Radix-10 multiplier architecture and its implementation. Section 3 explains the concepts used in the proposed Radix-10 multiplier.The processing unit and its implementation based on the proposed method are explained in Section 4. Section 5& 6 shows the experimental setup and simulation results and performance analysis of various architectures.

2. PRIOR WORK

2.1 Radix 5 Multiplier

Tomas Lang and Alberto Nannarelli (2006) [3] performs the multiplication of BCD numbers in parallel. For the multiplication P = M ×N, both the multiplicand M and the multiplier N are sign magnitude N-digit numbers in BCD format. The multiplication process comprises of three major steps. First step is the generation of intermediate products, Reduction, and Addition. First, the positive multiplicand multiples are generated. They are 5M, 10M, 2M, 1M. The negative multiplicand multiple is obtained by simply doing the bit inversion of positive multiplicand multiple. The negative multiplicand multiples are obtained by using XOR gates that invert the bits of positive multiplicand multiples. The multiplier digits in recoded form are used for selection of 5 M, 10M, 2M, 1M. The recoding digits are obtained by radix- 5 technique as shown in Table 1. To avoid complicated multiples of M, this method recode Ni =NH+NL with NH ∈ {0, 5, 10} and NL∈ {-2, -1, 0, 1, 2}.For M=7,N_{5i} in NH and N_{2i} in YL are enabled,YL=2.Similarly M=8 ,N_{10i} in NH Y2i in YL are enabled and it involves the usage of complement subtraction.

The radix-5 is represented as {0, 1, 2, 3, 4}. The SD-Radix-5 is represented as {-2,-1, 0, 2, 1}. In this architecture, the YL has the sign bit which is used for selecting the negative multiplicand multiples. The sign bit is not required for YH. The selected multiplicand multiples are used to produce the partial products. The accumulation of the partial products is done using carry-save adders that avoids carry accumulation .It add a carry save operand with another BCD operand to produce a final result. Final addition is performed by using 4-bit BCD adder. The detector circuit is used to check whether the sum is valid or not. The detector circuit also verifies the carry propagation. It will decide the correction factor as 0 or 6.

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

2.2. Radix 10 Multiplier

Vazquez et al (2014)[1] Radix-10 Multiplier uses three stages to perform a decimal multiplication in multipliers. The efficient techniques are adopted to improve the computation. The intermediate product generation is performed by using radix-10 based recoded multiplicand multiples generated as shown in Table 1. The multiples are generated in Excess-3 format are positive and it perform addition eliminating the carry. The Radix-10 recoded values are processed by using 10 base recoder. The Y5i, Y4i, Y3i, Y2i, Y1i are the digits used for recoding and act as the selection bits of multiplicand multiples. Sign bit (Ysi) is used to select the negative multiplicand multiples as shown in the figure 1. The intermediate products are generated by using selected multiplicand multiples. For X=7 and 2 the selection is same except the sign bit. But the result is converted to Excess 3 and processed in final adder and reconverted back to obtain the final output thus increasing the delay and area.

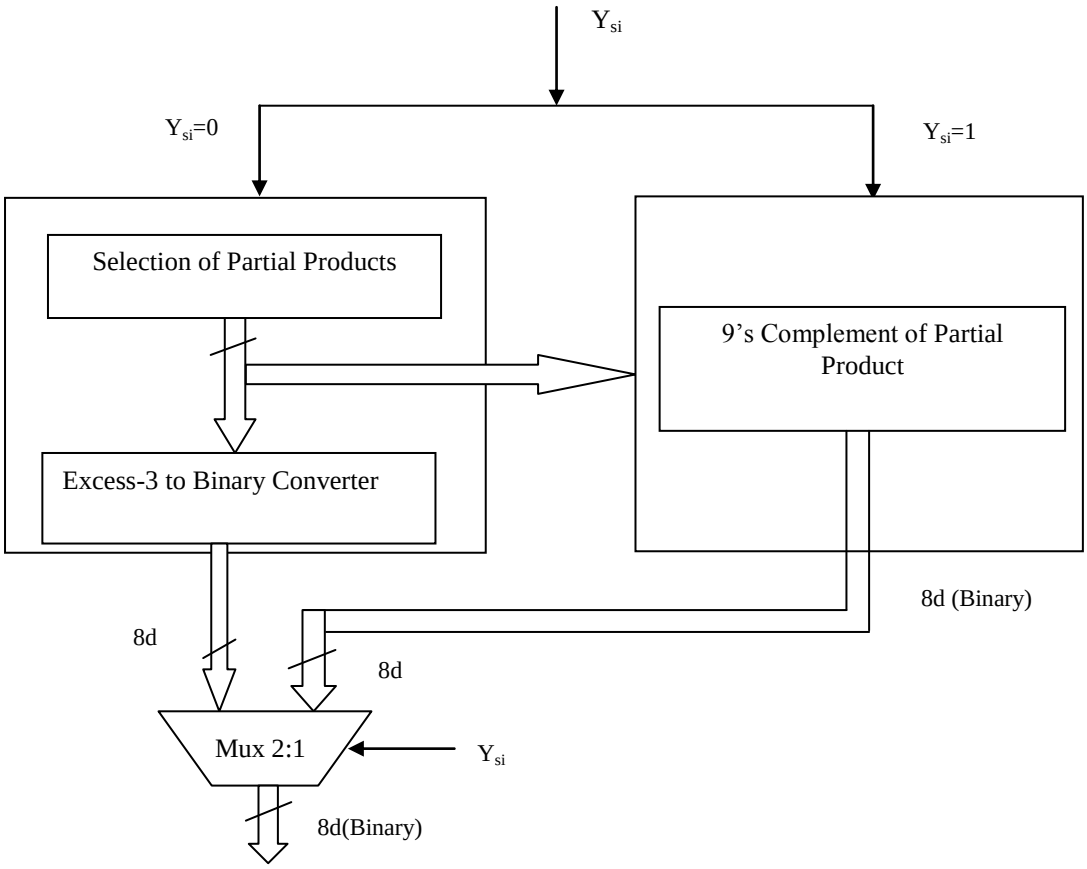

Figure 1 Recoder of Vazquez et al (2014)[1]

3. PROPOSED WORK

The multiplication process of proposed Radix-10 decimal multiplier is same as the existing radix-x multipliers except the recoding techniques. Though the product generation is by same steps the internal process of these two units are varied from existing multipliers. The multiplicand multiples unit is generates five sets 1X, 2X, 3X,4X,5X in BCD format directly.1X,2X,4X are generated just by shifting the input X.3X and 5X are generated by adding 4X and 1X respectively as shown in figure 2. These sets are used for partial product generation. In the Radix-10 recoder, the sign bit is not generated. Negative multiplicand multiples are not used in this design. Only the positive multiplicand multiples are used for selection of multiplicand multiple as available in Figure 3.

The novel recoding method eliminates the use of Excess-3 and Signed Digit other BCD encoding used like 4211,5211 etc.The recoder is designed to operate in two different forms .For X<5 the direct selection of output and X>5 the multiplicand multiples are selected based on the parity detector circuit. The recoding digits represented in Table 1 are used as a select signals of multiplicand multiples. The selection of multiplicand multiples unit is done with the multiplexers .The selected multiples are considered as the intermediate partial product. Parity checker is used for processing the partial product. If parity bit is enabled,

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

the partial product is taken out directly. Parity bit is not enabled, the partial products are taken and BCD addition process is carried out as shown in figure 4. The need for sign bit and Excess 3 encoding is eliminated when compared to the work done by Vazquez et al (2014) [1].

This also overcomes the use of Yi =YH+YL with YH ∈ {0, 5, 10} and YL∈ {-2, -1, 0, 1, 2} [3].BCD adder used will be enabled only then the BCD input is greater than 5. Only during that case the BCD addition is required. Otherwise the output is taken directly. The selection of multiples is same for 0&5, 1&6, 2&7, 3&8, 4&9 except the Y5i which is enabled for BCD input greater than 5.

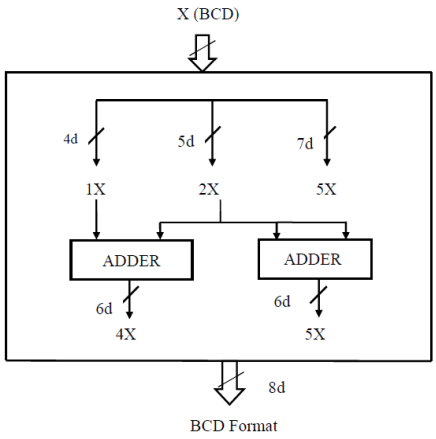

Figure 2 Generation of Multiplicand Multiples in proposed system

4. PROCESSING UNIT

The architecture of processing element is shown in Figure 5. The Proposed Radix-10 multiplier is used in 4-bit processing element and is extended up to 16 bit.

The processing unit performs four different operations addition, subtraction, multiplication and accumulation. In this design, four Control Signals are used to perform the operations of processing unit. The selection of operations in processing unit is done by using 2 control signals.00 for addition, 01 for subtraction,10 for multiplication and 11 for multiply accumulate unit. Multiplication process consumes more power and time compared to addition and two's complement subtraction. In this implementation radix-10 decimal multiplier is used which reduces the power and time of the processing unit.

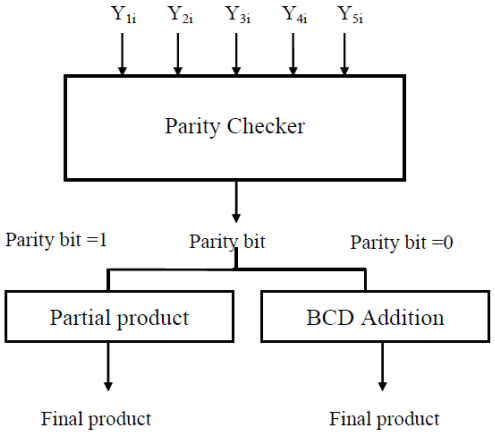

Figure 4 Implementation of Parity checker in the proposed system

Multiplication and accumulate computes the product of two numbers and adds to an accumulator. Modern computers may contain a dedicated MAC, consisting of a multiplier implemented, adder and an accumulator register that stores the result.

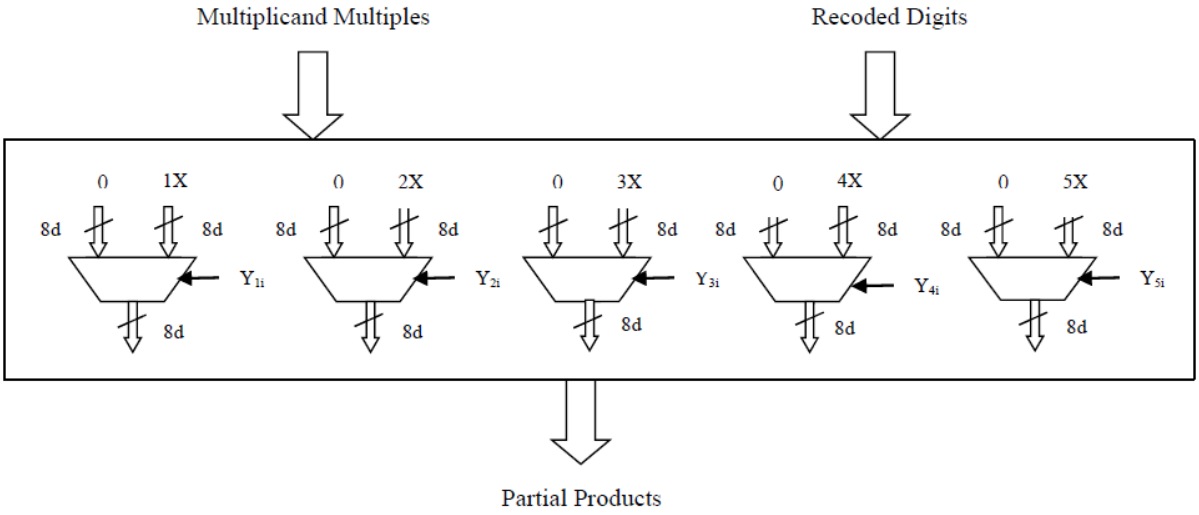

Figure 3 Selection of Multiplicand Multiples Unit in proposed system

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

To perform an accumulation operation, the output of multiplier from the previous execution stored in the register is sent back through accumulation control multiplexers controlled through select signal and added up to the current output. In this design, the proposed radix-10 multiplier is slightly modified to perform the additional operations. The control signal and register are included in proposed radix-10 multiplier to implement the processing element.

5. EXPERIMENTAL SETUP & SIMULATION RESULT

The latest version of the NClaunch simualtor called IES supports Verilog, VHDL, and System Verilog etc is used. This tool is used for logical verification. Encounter RTL synthesis tool with well-defined and optimized algorithms perform true top-down global RTL design synthesis that can be used for direct silicon realization.

It helps in simultaneous optimization of parameters like timing, area, and power. The final output of the Proposed Radix-10 decimal multiplieris shown in Figure 5.7. m11 and m12 are taken as the BCD inputs. Output P is considered as the final partial product. m11 is taken as 00110111 and m12 is taken as 10000110. The output for the corresponding input is 0011000110000010 is shown in figure 5.

TABLE I REPRESENTATION OF RECODING DIGITS FOR PROPOSED RADIX-10 MULTIPLIER

	Ref[3]					Ref[1]						Proposed				
	Y_{10i}	Y_{5i}	Y_{2i}	Y_{1i}	Y_{Si}	Y_{si}	Y_{5i}	Y_{4i}	Y_{3i}	Y_{2i}	Y_{1i}	Y_{5i}	Y_{4i}	Y_{3i}	Y_{2i}	Y_{1i}
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
2	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0
3	0	1	1	0	1	0	0	0	1	0	0	0	0	1	0	0
4	0	1	0	1	1	0	0	1	0	0	0	0	1	0	0	0
5	0	1	0	0	0	1	1	0	0	0	0	1	0	0	0	0
6	0	1	0	1	0	1	0	0	1	0	0	1	0	0	0	1
7	0	1	1	0	0	1	0	0	0	1	0	1	0	0	1	0
8	1	0	1	0	1	1	0	0	0	0	1	1	0	1	0	0
9	1	0	0	1	1	1	0	0	0	0	0	1	1	0	0	0

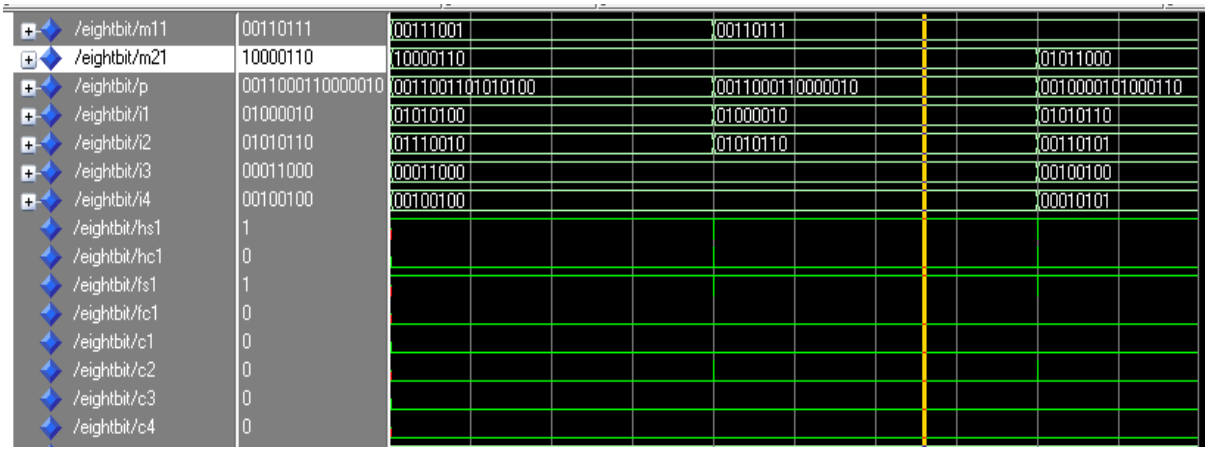

Figure 5 Simulation Waveforms of Proposed Radix-10 Decimal Multiplier

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

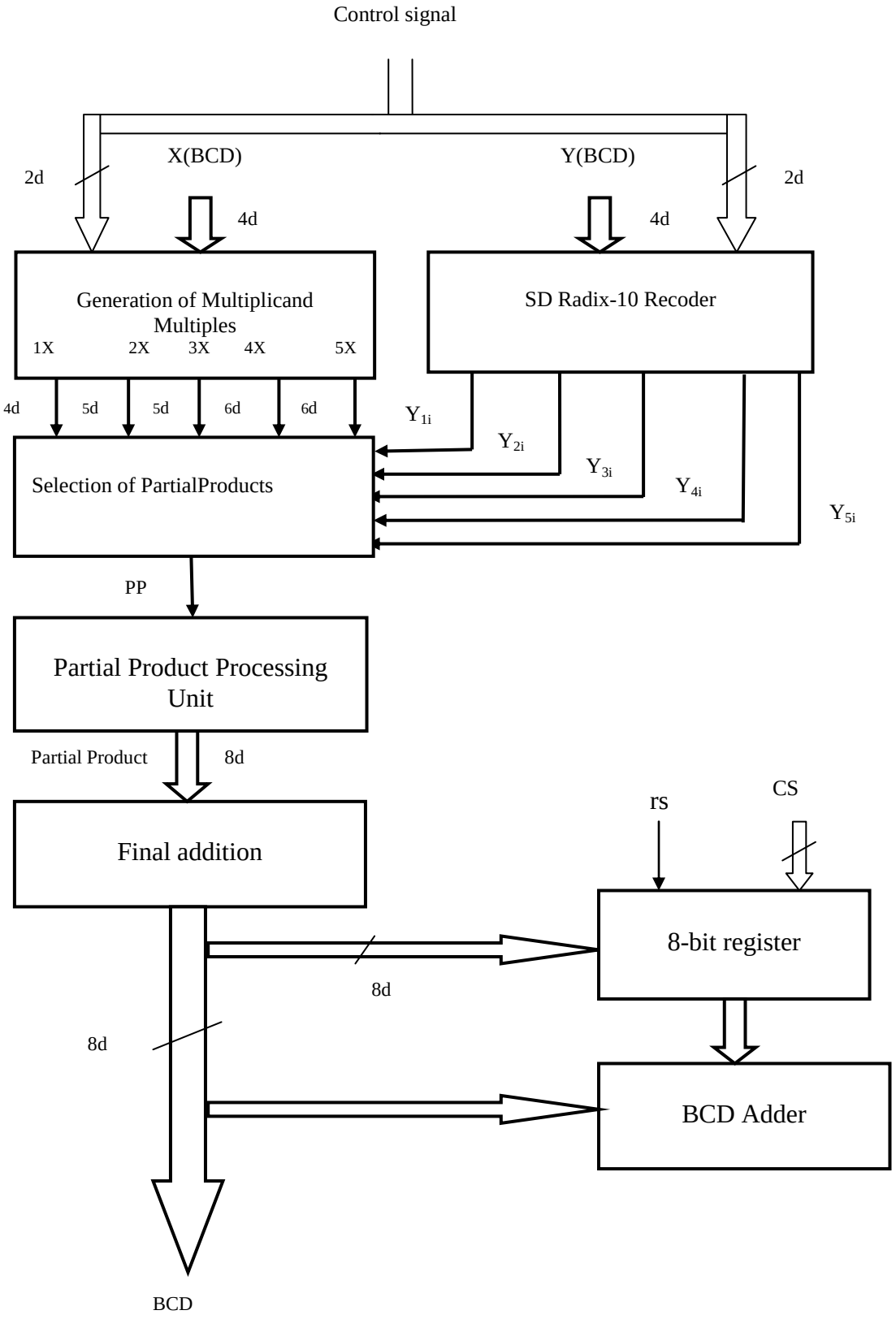

Figure 6 Block Diagram of Processing unit incorporating the Radix Multiplier

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

6. PERFORMANCE ANALYSIS

The decimal multiplier architectures viz., Tomas Lang & Alberto Nannarelli (2006) Radix-5 multiplier, Vazquez et al (2014) Radix-10 multiplier and proposed Radix-10 multiplier are designed using VHDL, simulated using NcLaunch Tool and synthesised using Encounter RTL Compiler. The synthesis is done by mapping to 180nm technology file using RTL synthesizer and power, delay and area estimation is done. It incorporates Encounter Digital Implementation that helps in providing real physical timing to logic structuring, mapping, optimization, and analysis with supply voltage 1.8 V and scaling technology 180 nm. The measured values are provided in table.6.1

TABLE 6.1 COMPARISONS OF AREA, POWER, DELAY PDP AND ADP

Ref	Area μm^2	Power nW	Delay Ps	PDP 10^{-15} Joules	ADP *10^-12 m^2s
Ref [1]	2087.54	32094.512	10031	321.93	20.93
Ref [3]	2598	40897	12085	494.24	31.39
Proposed	1768	29784	9195	273.86	16.25

From Table 5.1, it seen that area and power of proposed radix-10 decimal multiplier is 18.07% & 7.75% and 46.9% & 37.31% lower compared to Vazquez et al (2014)[1] Radix-10 Multiplier and Tomas Lang & Alberto Nannarelli (2006) [3]multiplier. From the figure 7 it is inferred that the proposed multiplier shows better results when compared to the Ref[1] and Ref[3].Power Area and Delay are reduced due to the new recoder and in turn helps in implementation of efficient processing unit.

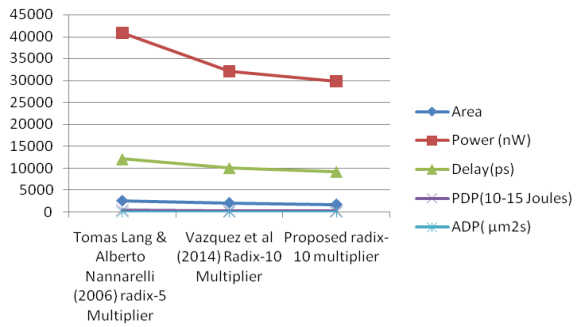

Figure 7 Performance Analysis of Proposed system with Ref [1] and Ref [3]

7. CONCLUSION

In this study, efficient BCD multiplication and processing unit designs are proposed. By evaluations of the proposed design, it is shown that the area power, delay, ADP, PDP is significantly reduced compared to prior designs. It is shown that the average Power and Power Delay Product of proposed Radix-10 decimal multiplier is 7.75%, 37.31% and 17.5%, 80.47% less compared to Ref [3] and [1]. Processing unit is implemented using proposed radix-10 multiplier which is suitable for high speed applications. Thus the proposed radix-10 decimal multiplier is suitable for high speed and portable applications.

REFERENCES

[1] Alvaro Vazquez, Elisardo Antelo, and Javier D. Bruguera," Fast Radix-10 Multiplication Using Redundant BCD Codes" IEEE TRANSACTIONS ON COMPUTERS, VOL. 63, NO. 8, PP:1902-1914 Aug.2014

[2] Alvaro Vazquez, Elisardo Antelo and Paolo Montuschi "Improved Design of High-Performance Parallel Decimal Multipliers" IEEE TRANSACTIONS ON COMPUTERS, VOL. 59, NO. 5,PP: 679-693,May 2010.

[3] Tomas Lang and Alberto Nannarelli, (Nov. 2009)" A Radix-10 Combinational Multiplier "IEEE Explore Signals, Systems and Computers, ACSSC '06, December PP313-317,2006.

[4] Ghassem Jaberipur and Amir Kaivani, "Improving the speed of parallel decimal multiplication," IEEE TRANSACTIONS ON COMPUTERS, VOL. 58, NO. 11,PP 1539–1552,2006

[5] M. A. Erle and M. J. Schulte, "Decimal multiplication via carrysave addition," in Proc. IEEE Int. Conf Appl.-Specific Syst., Arch., Process., pp. 348–358, Jun. 2003.

[6] A. Vazquez, E. Antelo, and P. Montuschi,"A new family of highperformance parallel decimal multipliers," in Proc. 18th IEEE Symp. Comput. Arithmetic, pp. 195–204,June 2007

[7] A. Vazquez and E. Antelo, "Multi-operand decimal addition by efficient reuse of a binary carry-save adder tree," in Proc. 44th ASILOMAR Conf. Signals, Syst. Comput., pp. 1685–1689,Nov 2010.

[8] L. Han and S. Ko,"High speed parallel decimal multiplication with redundant internal encodings," IEEE Trans. Comput., vol. 62, no. 5, pp. 956–968May 2013.

[9] E. M. Schwarz, J. S. Kapernick, and M. F. Cowlishaw, "Decimal floating-point support on the IBM System z10 processor," IBM J. Res. Develop., vol. 51, no. 1, pp. 4:1–4:10,Feb 2009.

[10] L. Dadda and A. Nannarelli,"A variant of a Radix-10 combinational multiplier," in Proc. IEEE Int. Symp. Circuits Syst., pp. 3370–3373,May 2008.

[11] H. Schmid, Decimal Computation. Hoboken, NJ, USA: Wiley, 1974.

[12] R. K. RichardsArithmetic Operations in Digital Computers. New York, NY, USA: Van Nostrand, 1955, Aug. 2012.

[13] A. Aswal, M. G. Perumal, and G. N. S. Prasanna, "On basic financial decimal operations on binary machines," IEEE Trans. Comput., vol. 61, no. 8, pp. 1084–1096.

[14] M. F. Cowlishaw, E. M. Schwarz, R. M. Smith, and C. F. Webb, "A decimal floating-point specification," in Proc. 15th IEEE Symp. Comput. Arithmetic, pp. 147–154,June 2001.

[15] IEEE Standard for Floating-Point Arithmetic, IEEE Std 754(TM)-2008 IEEE Comput. Soc., Aug. 2008.

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 2, Issue 10, October 2017, pp. 15–22

Authors Biography

K. Saranya, is a Assistant Professor, Department of EEE in Dr. Mahalingam College of Engineering & Technology, Pollachi, Tamil Nadu. India. She completed her BE in department of EEE at SKCET, Coimbatore under Anna University. She completed her M.E in department of Applied Electronics at Government College of Technology, Coimbatore. Her research interests are VLSI Design, Analog Circuits, Reversible Circuits, ASIC Implementation.

R. Baby Janagam, is a Assistant Professor, Department of EIE in Dr. Mahalingam College of Engineering & Technology, Pollachi, Tamil Nadu. India. She completed her BE in department of EIE at Annamalai University, Chidambaram. She completed her M.E in department of Applied Electronics at Maharaja Engineering College, Avinashi, Coimbatore. Her research interests are VLSI Design.

Cite this paper:

K.Saranya, R.Baby Janagam, "ASIC Implementation of Processing Unit Using Radix Decimal Multiplier", International Journal of Advances in Computer and Electronics Engineering, Vol. 2, No. 10, pp. 15-22, October 2017.