International Journal of Applied Engineering Research ISSN 0973-4562 Volume 15, Number 10 (2020) pp. 960-965
© Research India Publications. http://www.ripublication.com

Energy Efficient Approximate Multiplier Using Adiabatic Logic

M. Giridhar Reddy[1], A.V.N.Tilak[2], P. Prathyusha[3]

*Department of Electronics & Communication Engineering, Gudlavalleru Engineering College,
Gudlavalleru, Krishna District, A.P. – 521356, India*

Abstract

Multipliers play a vital role in optimized digital signal processors. Applications such as multimedia, machine learning, recognition, and data mining are inherently error-tolerant and do not require perfect accuracy in computation. For these applications, approximate circuits may play an important role as a promising alternative for reducing area, power, and delay in digital systems that can tolerate some loss of accuracy, thereby achieving better performance in energy efficiency. The power consumption can be further reduced by employing low power techniques such as adiabatic logic. Accurate and approximate 8-bit array multipliers are designed using CMOS and ECRL adiabatic logic and implemented in Mentor Graphics. The ECRL circuits give rise to power savings of about 59% as compared to CMOS logic. The results indicate the suitability of ECRL circuits for realizing battery operated portable systems where power dissipation is a critical parameter.

Keywords: Approximate multiplier, adiabatic logic, CMOS, ECRL, power dissipation.

1. INTRODUCTION

Multipliers play a vital role as essential building blocks of microprocessors, digital signal processors, embedded systems, etc. However, multipliers are characterized by intricate logic design and consume much of the energy in these systems. Power consumption is one of the most important characteristics of battery-powered hand-held devices. Over the years, a lot of efforts have been put into reducing the power consumption of systems at different design levels. One of the recent techniques for power reduction is trading accuracy for power consumption. Different designs have been proposed in this regard. Some error-resilient applications, like multimedia processing, machine learning, fuzzy logic, neural networks, wireless communications, data mining, and recognition [1-4] do not require exact computation. In such applications, approximate computing [5] can be carried out to gain substantial performance improvement in terms of complexity of the system architecture, power consumption, and delay, at the cost of a reduction in the accuracy. This level of error tolerance is used to design approximate arithmetic circuits for artificial intelligence (AI) and digital signal processing (DSP) applications, which effectively take advantage of the inability of the human eye to detect variation in finer details within images and videos and does not affect their normal operation.

Approximations can be introduced in any of the basic building blocks of the multiplier, viz., partial products generation, partial products reduction, and carry-propagate addition. A well-established approximation technique is the truncation of the partial products in which some of the partial products are not formed, and the truncation error is reduced with the help of suitable correction functions. Power dissipation is one of the main obstacles to attain high performance of digital circuits and is increasingly being investigated. Approximate computer arithmetic circuits based on CMOS technology have been extensively studied, and designs of approximate adders, multipliers, and dividers for both fixed-point and floating-point formats have been proposed [6-9].

An approximate adder and an approximate multiplier based on the broken-array multiplier (BAM) technique was proposed by [3] and demonstrated their benefits in terms of delay and area when exploited to implement face recognition neural network and defuzzification block of a fuzzy processor. In [10], an approximate multiplier to filter an image, consisting of some 2 x 2 inaccurate building blocks were proposed that could save power between 31.8% and 45.4% over an accurate multiplier. The approximate result of the error-tolerant multiplier (ETM) [5] was computed by dividing multiplication into one accurate and one approximate part. Power saving of more than 50% was reported for a 12-bit multiplier.

The power consumption can be further reduced by employing adiabatic logic style circuit design. The Adiabatic circuit reduces the charging and discharging speed and lowers the power consumption by storing charges in the parasitic capacitance and giving it back to the supply. It makes use of the AC power source instead of the DC power supply to store the charging energy in the circuit and is moved back to the power source during the discharging process. This charging and discharging process that happens in both pull-up and pull-down circuits can maintain energy in the power source and reduces the power dissipation drastically. Several kinds of adiabatic circuits have been designed and discussed in many previously published works [11-16]. They vary in a number of operation clocks, single or dual-rail style, charging/ discharging path, reversible-/irreversible-logic style [17]. In the proposed work, efficient charge recovery logic (ECRL) based adiabatic approximate multiplier is designed and analyzed for finding its power consumption.

The remaining part of the paper is organized as follows. The background is given in section 2. The performance evaluation of accurate and approximate array multipliers is carried out in section 3, and section 4 concludes the paper.

2. BACKGROUND

A. Array Multiplier

An array multiplier is a digital combinational circuit used for multiplying two binary numbers by employing an array of full adders and half adders. This array is used for nearly the simultaneous addition of various product terms involved. An array of AND gates is used before the adder array to generate various product terms. The multiplication of two binary numbers can be done with one micro-operation by using a combinational circuit that forms the product bits all at once. The design space of these multipliers is explored based on their power, area, delay, and error. Being able to use approximate computing provides the designer with the ability to make trade-offs between accuracy and speed as well as power/energy consumption**.** Implementation of multiplier comprises three steps, viz., generation of partial products, partial product reduction, and finally, a vector merges addition to produce the final product from the sum and carry rows generated from the reduction tree.

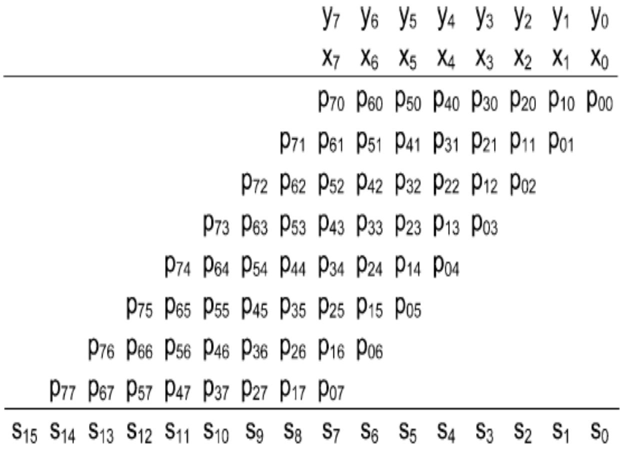
Multiplication of two unsigned 8-bit numbers ($X = x_7x_6x_5x_4x_3x_2x_1x_0$) and ($Y = y_7y_6y_5y_4y_3y_2y_1y_0$) based on add and shift algorithm is depicted in figure 1. Each partial product is generated by the multiplication of the multiplicand with one multiplier bit. The partial products (p_{00} to p_{77}) are positioned according to their bit orders and then added to get the sum bits as s_0 to s_{15}.

Figure 2 shows the block diagram of an 8-bit array multiplier with its regular structure. It uses two-input AND logic for the generation of partial products. Each row of partial products is shifted according to the position of the bit of the multiplier, forming what is commonly called the partial-product array. Finally, partial products that are in the same column are summed together, forming the result.

B. Approximate Multiplier

One of the techniques to achieve approximate multiplication for reduction of power consumption is the partial product perforation method in which the generation of partial products is skipped, thereby reducing the accumulation of the operands. Consider two n-bit numbers, X and Y. The result of their multiplication X × Y is obtained after summing all the partial products Xy_i, where y_i is the i^{th} bit of Y. A partial perforated product is not inserted in the accumulation tree, and hence n full adders can be eliminated.

Figure 3 illustrates the partial product reduction process for the multiplication of two 8-bit numbers for both accurate and approximate arrays. The dots represent the bits of the partial products, and the stages represent the delay of the reduction process. The dashed boxes with four dots are 4:2 compressors, those with three are full adders, and those with two are either full- or half-adders. The order of a partial perforated product indicates the error in the final result, higher the order, greater will be the error. When more than one partial product is perforated, the total error results from the addition of the errors produced from the perforation of each partial product separately.



								y_7	y_6	y_5	y_4	y_3	y_2	y_1	y_0
								x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0
								p_{70}	p_{60}	p_{50}	p_{40}	p_{30}	p_{20}	p_{10}	p_{00}
							p_{71}	p_{61}	p_{51}	p_{41}	p_{31}	p_{21}	p_{11}	p_{01}	
						p_{72}	p_{62}	p_{52}	p_{42}	p_{32}	p_{22}	p_{12}	p_{02}		
					p_{73}	p_{63}	p_{53}	p_{43}	p_{33}	p_{23}	p_{13}	p_{03}			
				p_{74}	p_{64}	p_{54}	p_{44}	p_{34}	p_{24}	p_{14}	p_{04}				
			p_{75}	p_{65}	p_{55}	p_{45}	p_{35}	p_{25}	p_{15}	p_{05}					
		p_{76}	p_{66}	p_{56}	p_{46}	p_{36}	p_{26}	p_{16}	p_{06}						
	p_{77}	p_{67}	p_{57}	p_{47}	p_{37}	p_{27}	p_{17}	p_{07}							
s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1	s_0

Figure 1: Illustration of an unsigned 8-bit multiplication

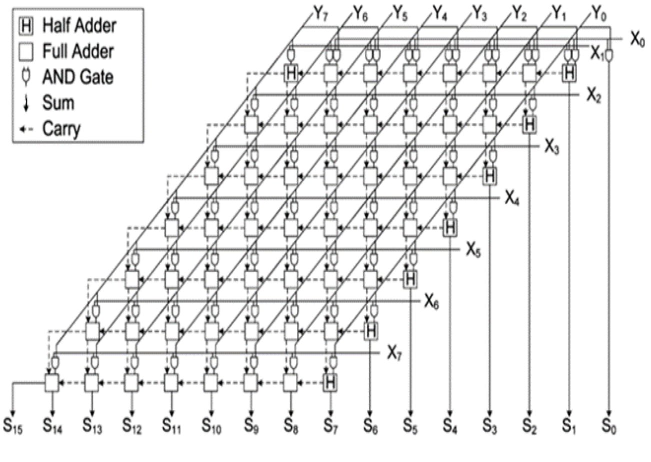
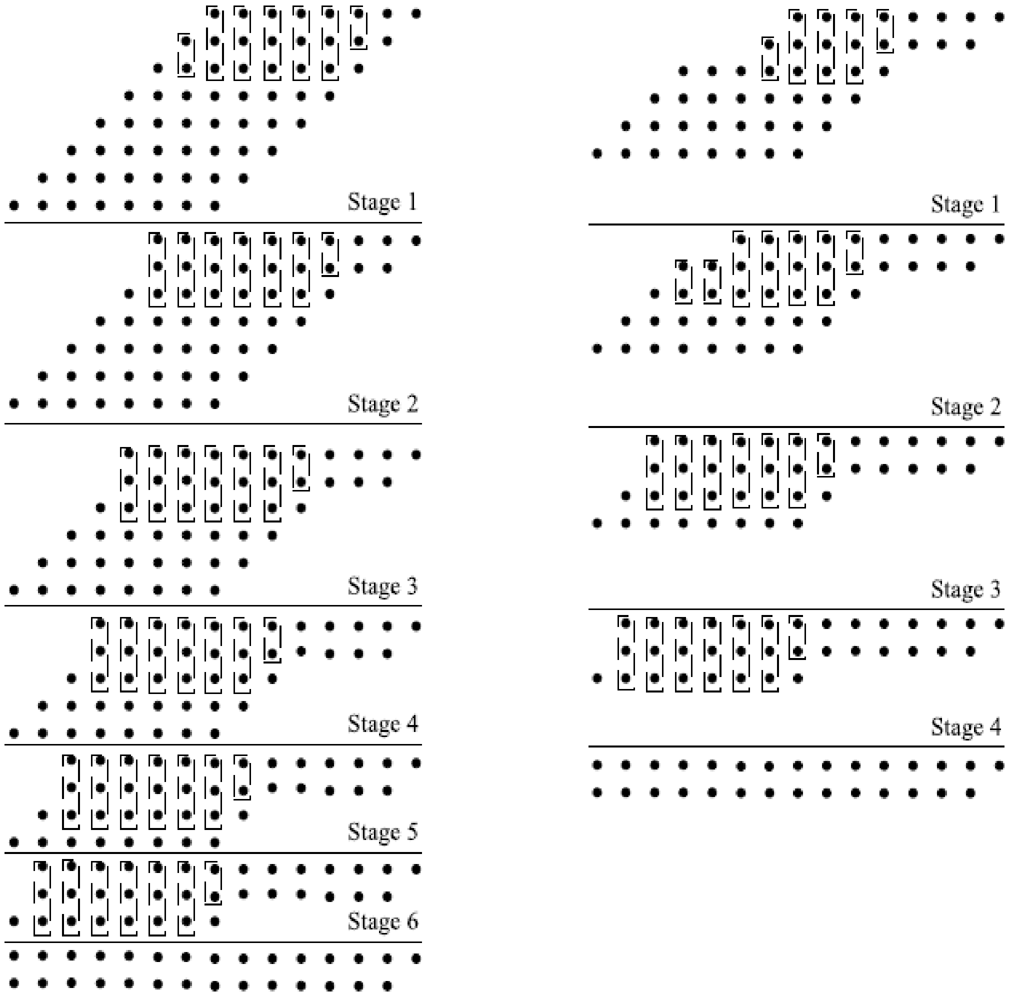


Figure 2: Block diagram of 8-bit array multiplier



(a) (b)

Figure 3: Partial product reduction process for 8 x 8 multiplication with (a) accurate array and (b) approximate array

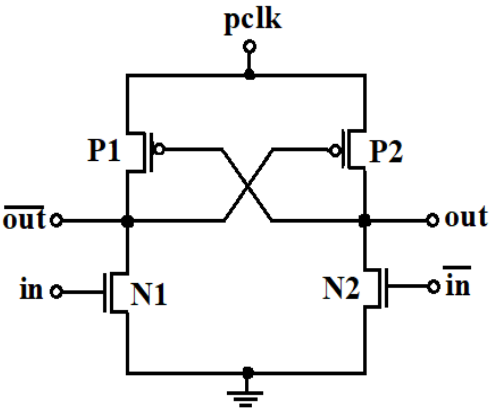


Figure 4: ECRL buffer/inverter

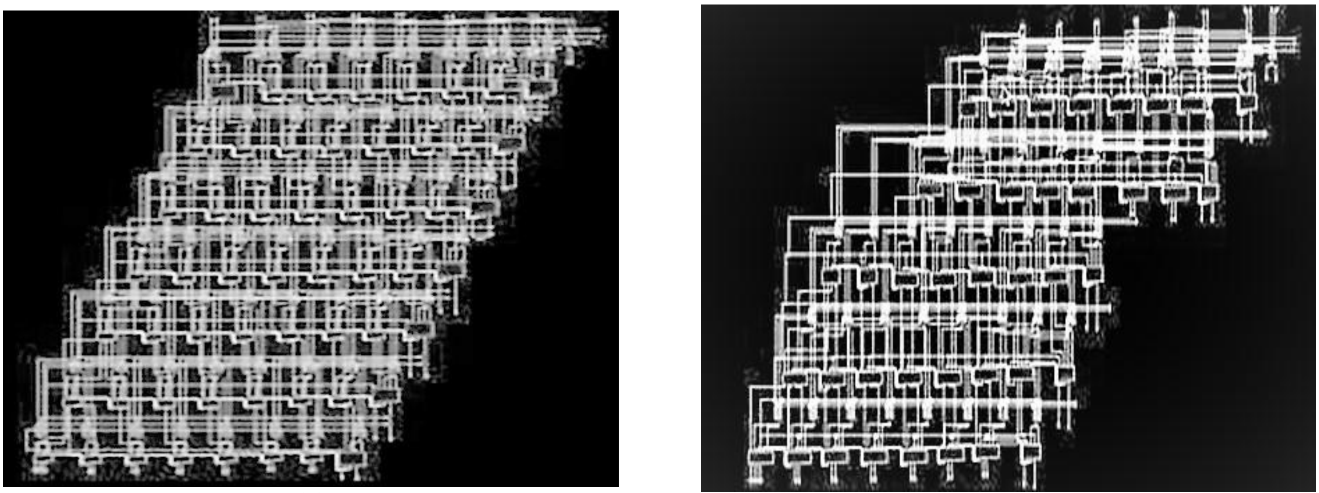
C. Efficient Charge Recovery Logic (ECRL)

ECRL buffer/inverter shown in figure 4 consists of two cross-coupled PMOS transistors in the pull-up section, whereas the pull-down section is constructed with a tree of NMOS transistors. Their source terminals are connected to the power clock, and the gate of each one is connected to the drain of the other. These nodes form the complementary output signals. The logic function in the functional block can be realized with only NMOS transistors in the pull-down section. An AC power supply 'pclk' is used for ECRL gates, to recover and reuse the supplied energy. Both out and $\overline{\text{out}}$ are generated so that the power clock generator can always drive a constant load capacitance independent of the input signal.

3. PERFORMANCE EVALUATION OF ACCURATE AND APPROXIMATE ARRAY MULTIPLIERS

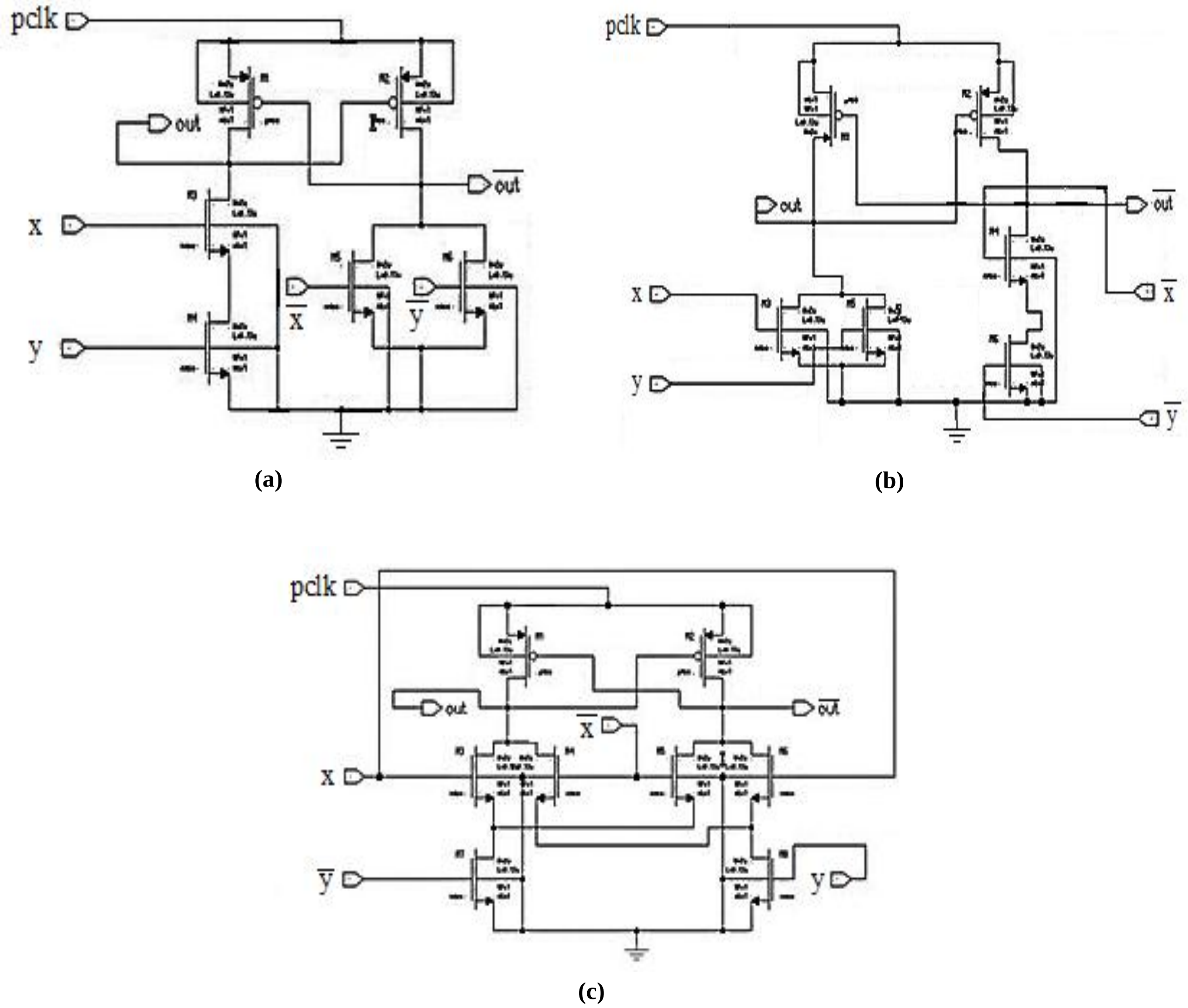
To explore the performance of array multiplier in terms of power consumption, 8-bit accurate and approximate array multipliers are designed using CMOS and ECRL adiabatic logic and implemented in Mentor Graphics. Figure 5 shows the schematics of 8-bit accurate and approximate array multipliers.

The ECRL NAND/AND, NOR/OR, and XNOR/XOR gates required to implement multiplier circuits of figure 5 are shown in figures 6(a), 6(b), and 6(c) respectively.



(a) (b)

Figure 5: Schematic diagrams of 8-bit (a) accurate and (b) approximate array multipliers

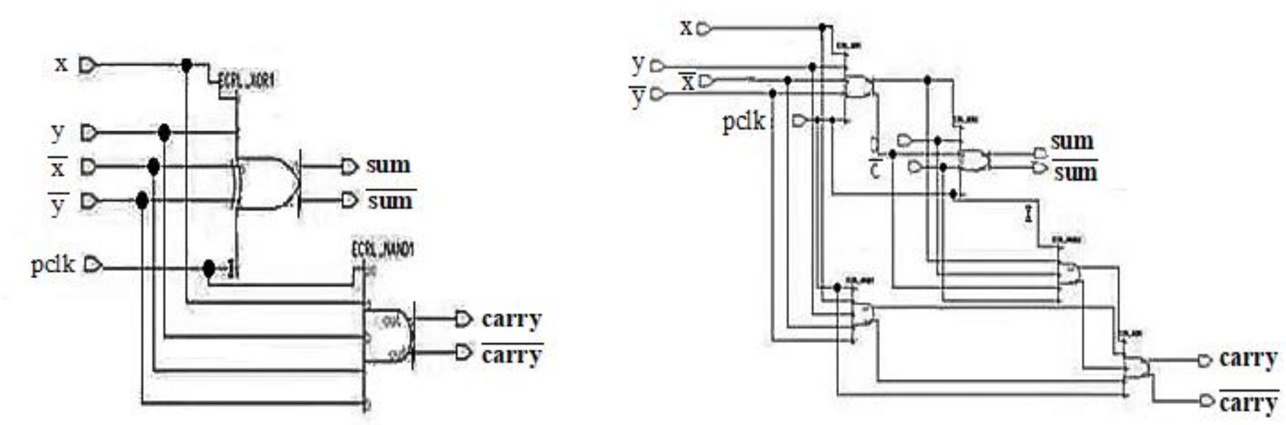


(a) (b)

(c)

Figure 6: ECRL (a) NAND/AND, (b) NOR/OR, and (c) XNOR/XOR gates

Figures 7(a) and 7(b) respectively shows the half adder and full adder circuits using ECRL logic.



(a) **(b)**

Figure 7: (a) Half adder and (b) full adder circuits using ECRL

The designed circuits are simulated using Mentor Graphics tool, taking the inputs as

$$X = x_7x_6x_5x_4x_3x_2x_1x_0 = 00110101$$

and

$$Y = y_7y_6y_5y_4y_3y_2y_1y_0 = 11111111$$

Power analysis is carried out for both CMOS and ECRL based accurate and approximate 8-bit array multipliers. Table 1 gives a comparison of the power dissipation of the circuits. For both accurate and approximate array multipliers, the power dissipated by the ECRL circuits is considerably lower than that of CMOS circuits. Thus, the ECRL logic offers considerable savings in power and is thus suitable for the design of energy efficient systems.

Table 1: Comparison of power dissipation of 8-bit array multiplier circuits using CMOS and ECRL logic

Parameter	CMOS		ECRL	
	Accurate array multiplier	Approximate array multiplier	Accurate array multiplier	Approximate array multiplier
Power (µW)	7.5155	4.6310	3.094	1.908

4. CONCLUSION

In this work, 8-bit accurate and approximate array multipliers are designed using CMOS and ECRL adiabatic logic. The circuits are simulated for functional verification using the Mentor graphics tool. The power analysis carried out indicates 58.84% and 58.79% power reduction with ECRL logic for accurate and approximate multipliers, respectively, as compared to CMOS circuits. Hence, it can be concluded that ECRL logic is best suitable for realizing portable systems where power dissipation is a critical parameter.

REFERENCES

[1] Vinay K. Chippa, Srimat T. Chakradhar, Kaushik Roy, and Anand Raghunathan, 2013, "Analysis and characterization of inherent application resilience for approximate computing," Proc. 50th ACM/EDAC/IEEE Des. Automat. Conf. (DAC), Austin, TX, USA, pp. 1–9.

[2] Q. Xu, T. Mytkowicz, and N. S. Kim, 2016, "Approximate computing: A survey," IEEE Des. Test, 33(1), pp. 8–22.

[3] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, 2010, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I, Reg. Papers, 57(4), pp. 850–862.

[4] Rangharajan Venkatesan, Amit Agarwal, Kaushik Roy, and Anand Raghunathan, 2011, "MACACO: Modeling and analysis of circuits for approximate computing," Proc. IEEE/ACM Int. Conf. on Comput.-Aided Des. (ICCAD), pp. 667–673.

[5] J. Han, and M. Orshansky, 2013, "Approximate computing: An emerging paradigm for energy-efficient design," Proc. 18th IEEE Eur. Test Symp., pp. 1–6.

[6] Soheil Hashemi, R. Iris Bahar, and Sherief Reda, 2015, "DRUM: A Dynamic Range Unbiased Multiplier for approximate applications," Proc. IEEE/ACM Int. Conf. on Comput. Des., pp. 418 - 425.

[7] Vaibhav Gupta, Debabrata Mohapatra, Anand Raghunathan, and Kaushik Roy, 2013, "Low-power digital signal processing using approximate adders," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., 32(1), pp.124-137.

[8] Semeen Rehman, Walaa El-Harouni, Muhammad Shafique, Akash Kumar, and Jorg Henkel, 2016, "Architectural-space exploration of approximate multipliers," Proc. Int. Conf. Comput.-Aided Des., pp.1-6.

[9] Amir Momeni, Jie Han, Paolo Montuschi, and Fabrizio Lombardi, 2015, “Design and analysis of approximate compressors for multiplication,” IEEE Trans. Comput., 64(4), pp. 984 - 994.

[10] Parag Kulkarni, Puneet Gupta, and Milos Ercegovac, 2011, “Trading accuracy for power with an under designed multiplier architecture,” Proc. Int. Conf. VLSI Design, pp. 346–351.

[11] Joonho Lim, Dong-Gyu Kim, and Soo-Ik Chae, 1999, “A 16-bit carry-lookahead adder using reversible energy recovery logic for ultra-low-energy systems,” IEEE J. Solid-State Circuits, 34(6), pp. 898–903.

[12] A.G. Dickinson, and J.S. Denker, 1995, “Adiabatic dynamic logic,” IEEE J. Solid-State Circuits, 30(3), pp. 311–315.

[13] A. Kramer, J.S. Denker, S.C. Avery, A.G. Dickinson, and T.R. Wik, 1994, “Adiabatic computing with the 2n-2n2d logic family,” Proc. IEEE Symp. VLSI Circuits, pp. 25–26.

[14] D. Maksimovic, V.G. Oklobdzija, B. Nikolic, and K.W. Current, 2000, “Clocked CMOS adiabatic logic with integrated single-phase power clock supply,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 8(4), pp. 460–463.

[15] F. Liu and K. T. Lau, 1998, “Pass-transistor adiabatic logic with NMOS pull-down configuration,” Electron. Lett., 34(8), pp. 739–741.

[16] Yibin Ye, and K. Roy, 2001, “QSERL: quasi-static energy recovery logic,” IEEE J. Solid-State Circuits, 36(2), pp. 239–248.

[17] Cihun-Siyong Alex Gong, Muh-Tian Shiue, Ci-Tong Hong, and Kai-Wen Yao, 2008, "Analysis and design of an efficient irreversible energy recovery logic in 0.18μm CMOS." IEEE Trans. Circuits Syst. I, Reg. Papers, 55(9), pp. 2595-2607.