Trade-offs in Designing High-Performance Digital Adder based on Heterogeneous Architecture

Raminder Preet Pal Singh
Dept. ECE, Arni University
Kathgarh, H.P, India

Ashish Chaturvedi
Dept. CSE, Arni University,
Kathgarh, H.P, India

Onkar Singh
Dept. ECE, Arni University,
Kathgarh, H.P, India

ABSTRACT
To design an efficient integrated circuit in terms of Area, Power and speed is one of the challenging task in modern VLSI design field. In the past decade numbers of research have been carried out to optimize design based on area, speed and power utilization. In this paper performance analysis of different available adder architectures has been carried out and then we proposed a Heterogeneous architecture, which composed of four different sub adders (Ripple Carry, Carry Look Ahead, Carry Skip and Carry Select Adder) to design an adder unit in order to demonstrate trade-offs between performance parameters i.e. Area, Power and speed. We consider area optimization under delay constraint, area optimization under power constraint and finally power optimization under delay constraint.
All the adders are design using VHDL. To get power, delay and area report, we use XILINX 9.1 i as synthesis tool and Modelsim XE III 6.2g for simulation. FPGA-Spartan III is used for implementation.

Keywords
Adder, Ripple carry adder, Lookahead carry adder, VHDL simulation.

1. INTRODUCTION
Adders are most commonly used in various electronic applications e.g. Digital signal processing in which adders are used to perform various algorithms like FIR, IIR etc. In past, the major challenge for VLSI designer is to reduce area of chip by using efficient optimization techniques. Then the next phase is to increase the speed of operation to achieve fast calculations like, in today's microprocessors millions of instructions are performed per second. Speed of operation is one of the major constraints in designing DSP processors. Now, as most of today's commercial electronic products are portable like Mobile, Laptops etc. that require more battery back up. Therefore, lot of research is going on to reduce power consumption. Therefore, there are three performance parameters on which a VLSI designer has to optimize their design i.e. Area, Speed and Power. It is very difficult to achieve all constraints for particular design, therefore depending on demand or application some compromise between constraints has to be made.

2 PRIOR WORK
In 1990, modified Carry-Skip Adders was presented by reducing first block delay with carry-lookahead adders using multidimensional dynamic programming [12]. In 1996, transistor-level simulation of the adders using HSPICE is done for area, time and power trade-off between different fast adders [6]. In 2002, a new concept of hybrid adders is presented to speed up addition process by *Wang et al*. that gives hybrid carry look-ahead/carry-select adders design [7]. In 2007, a new 54×54-bit multiplier is designed using high-speed carry-look-ahead adder and has been fabricated by CMOS technology [4]. In 2008, low power multipliers based on new hybrid full adders is presented [5]. In 2008, *Hasan Krad et al* worked on the performance analysis for a 32-Bit Multiplier with a Carry-Look-Ahead Adder and a 32-bit Multiplier with a Ripple Adder using VHDL [3].

3 HOMOGENEOUS ADDERS

3.1. Ripple Carry Adder (RCA)
Ripple carry adder can be designed by cascading full adder in series i.e. carry from previous full adder is connected as input carry for the next stage. Full adder is a basic building block of Ripple carry adder. Therefore, to design n-bit parallel adder, it requires n full adders. This kind of adder is called a *ripple-carry adder*, since each carry bit "ripples" to the next full adder. The layout of a ripple-carry adder is simple, which allows for fast design time; however, the ripple-carry adder is relatively slow, since each full adder must wait for the carry bit to be calculated from the previous full adder.

In our design we use 16 full adders to design a 16-bit parallel adder. The major limitation of Ripple carry adder is that as the bit length goes on increases, delay also increases. Therefore, Ripple carry adder is not suitable if large number bits are to be added.

The major element that causes delay is carry propagation; therefore it is important to calculate carry delay from input to output. For n-bit Ripple carry adder, Delay for carry can be calculated as: -

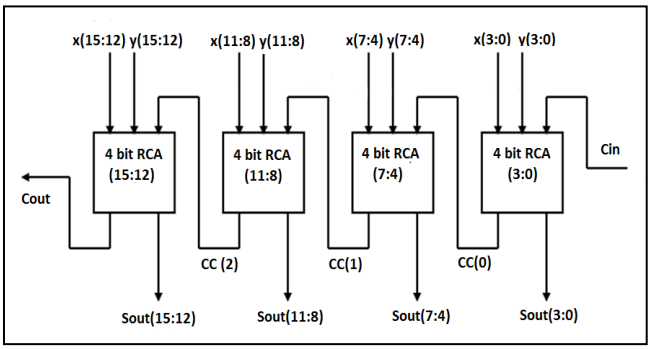

FIGURE 1: Schematic block diagram of 16-bit ripple carry adder

$$T_C = T_{FA}((x0, y0) \text{ to } c0) + (n-2) * T_{FA}(\text{cin to cout})) + T_{FA}(\text{cin to sout } (n-1)) \quad (1)$$

Where T_{FA} (input to output) represent the delay of full adder on the path between it's specified input and output.

3.2 Carry Lookahead Adder (CLA)

Lookahead carry algorithm speed up the operation to perform addition, because in this algorithm carry for the next stages is calculated in advance based on input signals. Carry lookahead depends on two things:Calculating, for each digit position, whether that position is going to propagate a carry if one comes in from the right and Combining these calculated values to be able to deduce quickly whether, for each group of digits, that group is going to propagate a carry that comes in from the right.

If X and Y are two inputs, "ci" is initial carry, "sout" and "cout" are output sum and carry respectively, then Boolean expression for calculating next carry and addition is:

Pi = xi xor yi --- Carry Propagation (2)

Gi = xi and yi --- Carry Generate (3)

C_{i+1} = Gi or (Pi and C_i)--Next Carry (4)

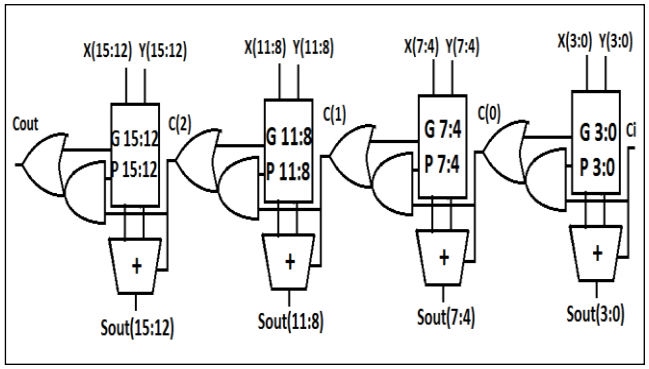

FIGURE 2: Schematic block diagram of 16-bit carry look ahead adder divided into 4 blocks [2]

The delay of N-bit carry look-ahead adder based on 4-bit look-ahead blocks is:

$T_{CLA} = 4 \log_4 N + 1$ gate levels (5)

3.3 Carry Skip Adder (CSKA)

In case of N-bit Ripple carry adder, carry has to propagate through all N stages, which results in large delay in performing binary addition. In contrast, it is possible to skip carry over group of n-bits in case of Carry Skip Adder. This result in less delay as compare to ripple carry adder. The worst-case carry propagation delay in a N-bit carry skip adder with fixed block width *b*, assuming that one stage of ripple has the same delay as one skip, can be derived:

T_{CSKA} = (b -1) + 0.5+ (N/b-2) + (b -1) (6)

= 2b + N/b – 3.5 Stages (7)

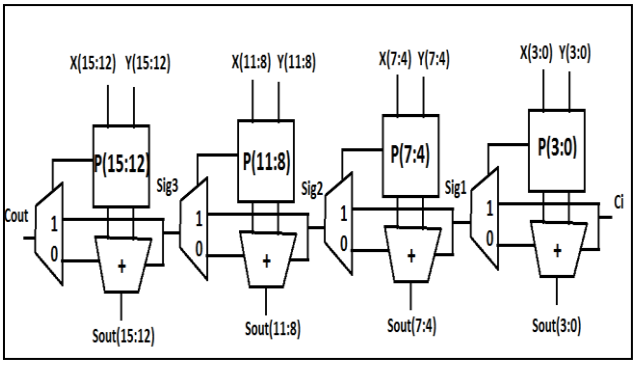

FIGURE 3: 16-Bit Carry-Skip Adder [1].

3.4 Carry Select Adder (CSLA)

Carry select adder is based on the principle to calculate sum that is based on assuming input carry from previous stage. One adder calculates the sum assuming input carry of 0 while the other calculates the sum assuming input carry of 1. Then, the actual carry triggers a multiplexer that selects the appropriate sum [2]. Fig.4 shows the schematic block diagram of 16-bit Carry select adder consists of 4-blocks each of 4-bit Look ahead carry adder [11]. Carry output of each block is fed into next block as input carry.

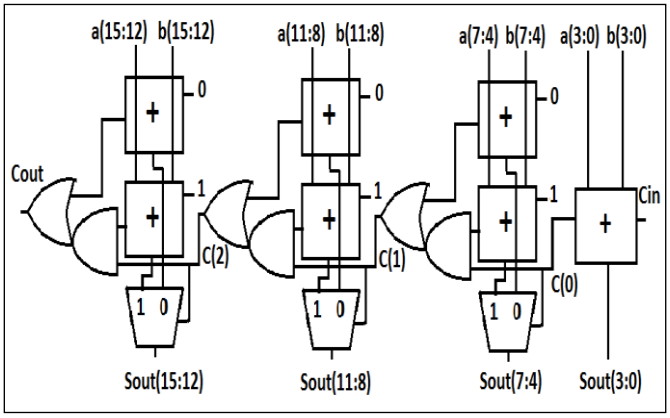

FIGURE 4: Schematic block diagram of 16-bit Carry select adder [2].

TABLE 1: AREA, POWER AND DELAY REPORT FOR ADDERS

ADDERS	CLB'S	Delay (ns)	Delay (ns)	Power (mW)	Power (mW)
Block Size = 4-Bit		**Sum**	**Carry**	**Dynamic**	**Static**
Ripple Carry	24	24.1	23.5	7.6	218.7
Look Ahead	26	20.9	20.6	13.3	219.4
Carry Select	27	17.1	17.7	10.1	219
Carry Skip Adder	30	15.4	12.1	13.9	219.4

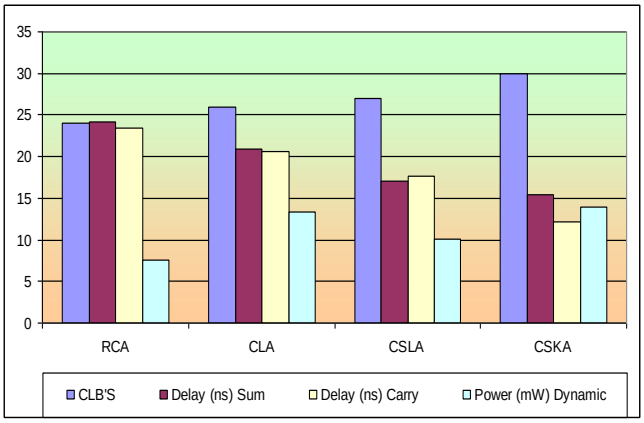

FIGURE 5: Area, Speed and Power comparison of Homogeneous Adder

4 HETEROGENEOUS ADDER

16-bit Heterogeneous adder proposed in this paper, consists of four sub adders SA1 (Ripple carry adder), SA2 (Carry skip adder), SA3 (Carry select) and SA4 (Carry look ahead adder). Initially, we choose equal bit-width for each sub-adder i.e. 4-bit as shown in figure 7. All these four sub adders are concatenates to form a 16-bit heterogeneous adder.

The order i.e. bit-width of each sub adder has an impact on the performance of a heterogeneous adder. Now, the main point of consideration is that how to select number of bits for each sub-adder? Bit-width selection for each sub-adder can be done on the basis of requirements (i.e. Area, Speed and Power constraints) of particular application where the design is to be implemented. For example, in order to achieve high speed of operation will cost large area as per the characteristic curve of delay-area shown in figure 6. Dark black dots shown in fig. 6 represent delay and area performance for different homogeneous adder. However, with the help of heterogeneous adder architecture, any desired performance can be achieved by adjusting bit-width of sub-adders.

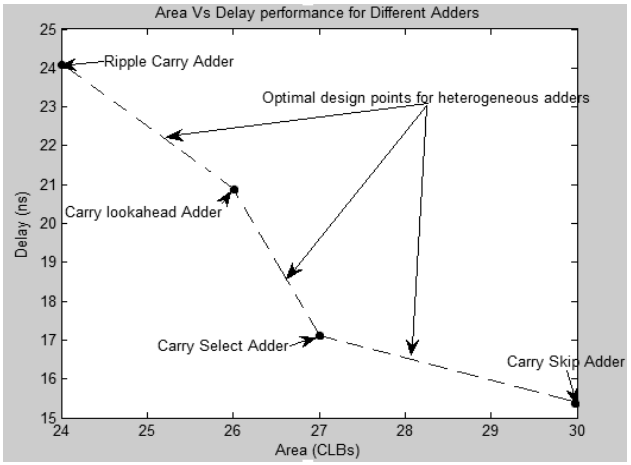

FIGURE 6: Conceptual design space for adder design

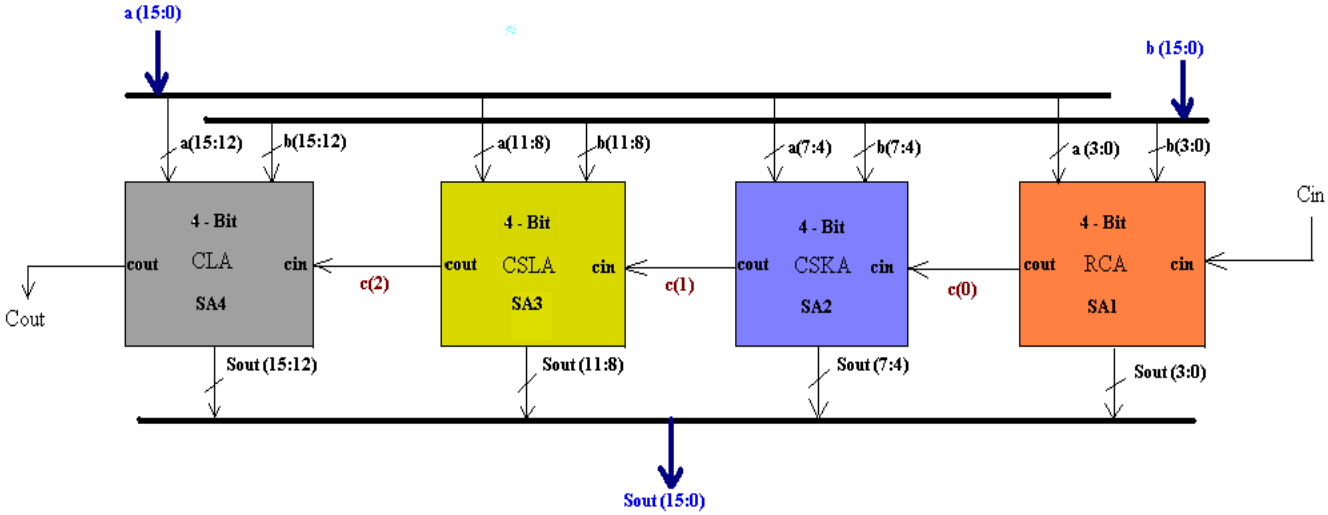

FIGURE 7: Block Diagram of Proposed Heterogeneous Adder

5 RESULTS AND DISCUSSION

Table 1 shows the result of different homogeneous adder, it is observed that every adder has its own advantages and limitations e.g. ripple carry adder cover small chip area and dynamic power consumption but at the cost of large operation delay whereas carry skip adder gives high speed of operation but at the cost of large surface area coverage and more dynamic power consumption. Therefore, in order to optimize adder design as per requirement, heterogeneous adder architecture shown in figure 7, gives flexibility in design so as to get desired performance.

Figure 8, shows trade-off between area and speed of operation i.e. larger the gate count for adder design lower will be the speed of operation. Figure 9, shows trade-off between area and power consumption i.e. gate count (area) is directly proportional to dynamic power consumption. Figure 10, shows trade-off between power consumption and speed of operation i.e. dynamic power consumption is inversely proportional to delay time to get final output.

TABLE 2: AREA, POWER AND DELAY REPORT FOR ADDERS

CONFIGURATIONS	Label	Area	Delay
ADDERS		Gate Count	(ns)
16 bit RCA	A	192	24.1
8 RCA + 8 CLA	B	222	22.1
8 RCA + 4 CSLA + 4 CLA	C	234	21.5
16 bit CLA	D	237	20.7
5 RCA + 4 CSKA + 4 CSLA + 3 CLA	E	252	19.2
1 RC + 8 CSKA + 4 CSLA + 3 CLA	F	270	17.2
8 CLA + 4 CSKA + 4 CSLA	G	276	16.9
4 CLA + 4 CSKA + 8 CSLA	H	282	16.5

TABLE 3: AREA, POWER AND DELAY REPORT FOR ADDERS

CONFIGURATIONS	Label	Area	Power (mW)
ADDERS		Gate Count	Dynamic
4 CSKA + 12 CSLA	A	321	4.87
4 CLA + 12 CSLA	B	303	4.86
8 CSKA + 8 CSLA	C	300	4.68
8 CLA + 8 CSLA	D	294	4.48
1 RC + 4 CSKA + 8 CSLA + 3 RCA	E	285	4.3
1 RC + 4 CSKA + 8 CSLA + 3 CLA	F	282	4.16
4 CLA + 4 CSKA + 8 CSLA	F	282	4.13
1 RC + 8 CSKA + 4 CSLA + 3 RCA	G	264	3.57
8 CLA + 8 CSKA	H	258	3.42
12CSKA + 4 CLA	J	255	3.16
4 RCA + 12 CSKA	K	246	2.63
8 RCA + 4 CSLA + 4 CLA	L	234	2.49
12 RCA + 4 CSKA	M	210	1.78
16 bit RCA	N	192	1.76

TABLE 4: AREA, POWER AND DELAY REPORT FOR ADDERS

CONFIGURATIONS	Label	Power (mW)	Delay
		Dynamic	(ns)
4 CSKA + 12 CSLA	A	4.87	16.4
8 CSKA + 8 CSLA	B	4.68	16.5
1 RC + 8 CSKA + 4 CSLA + 3 CLA	C	4.34	17.2
1 RC + 4 CSKA + 8 CSLA + 3 CLA	D	4.16	17.7
2 RCA + 4 CSKA + 4 CSLA + 6 CLA	E	3.78	18.7
8 CLA + 8 CSKA	F	3.42	19.4
4 RCA + 8 CSLA + 4 CLA	G	2.86	20.4
16 bit CLA	H	2.64	20.7
8 RCA + 4 CSLA + 4 CLA	J	2.49	21.5
16 bit RCA	K	1.76	24.1

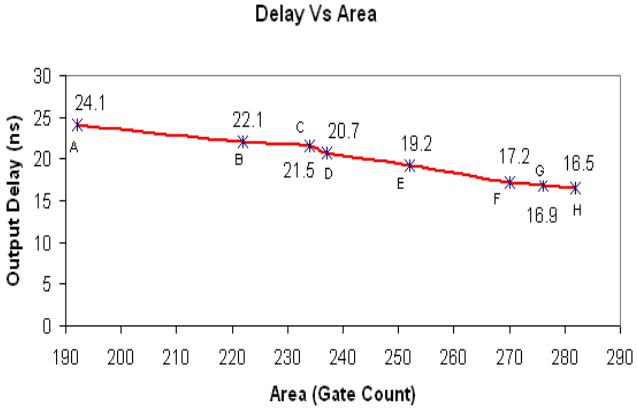

FIGURE 8: Performance comparison of adders

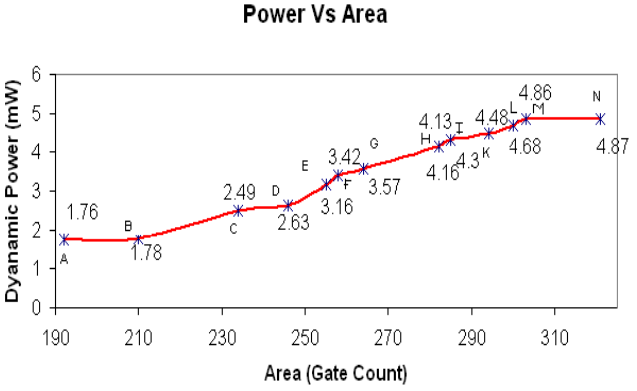

FIGURE 9: Performance comparison of adders

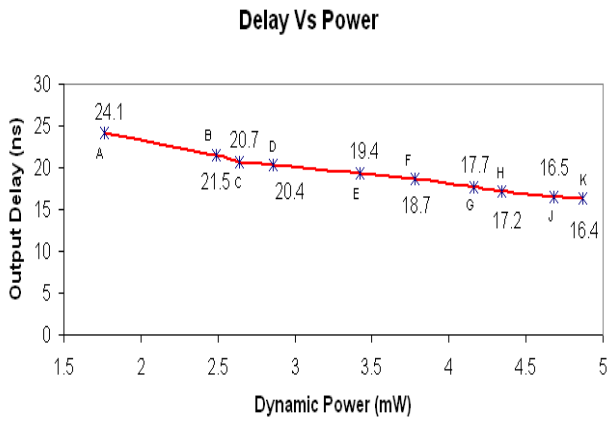

FIGURE 10: Performance comparison of adders

6 CONCLUSION

To analyze trade-offs in designing digital adder, we proposed a heterogeneous adder architecture, which consists of sub-adders of various sizes and carry propagation schemes. The proposed architecture allows more design trade-offs and hence provide flexibility to optimize design as per application demand.

7 REFERENCES

[1] B. Parhami, 2000, Computer Arithmetic, Algorithm and Hardware Design, Oxford University Press, New York, pp. 91-119.

[2] Pong P. Chu "RTL Hardware Design Using VHDL: coding for Efficiency, Portability and Scalability" Wiley-IEEE Press, New Jercy, 2006

[3] Hasan Krad and Aws Yousif Al-Taie, "Performance Analysis of a 32-Bit Multiplier with a Carry-Look-Ahead Adder and a 32-bit Multiplier with a Ripple Adder using VHDL", Journal of Computer Science 4 (2008): 305-308.

[4] Asadi, P. and K. Navi "A novel high-speed 54-54-bit multiplier", Am. J. Applied Sci., 4 (9) (2007): 666-672.

[5] Z. Abid, H. El-Razouk and D.A. El-Dib, "Low power multipliers based on new hybrid full adders", Microelectronics Journal, Volume 39, Issue 12, 2008, 1509-1515.

[6] Nagendra, C.; Irwin, M.J.; Owens, R.M.,"Area-time-power tradeoffs in parallel adders", Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on Volume 43, Issue 10, (1996): 689 – 702.

[7] Wang, Y.; Pai, C.; Song, X., "The design of hybrid carry look-ahead/carry-select adders, Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on Volume 49, Issue 1, (Jan 2002) : 16-24.

[8] May Phyo Thwal, Khin Htay Kyi, and Kyaw Swar Soe, "Implementation of Adder-Subtracter design with VerilogHDL", International Journal of Electronics, Circuits and Systems Volume 2 number 3, 2008.

[9] Min Cha and Earl E. Swartzlander, Jr, "Modified Carry Skip Adder for reducing first block delay", Proc. 43rd IEEE Midwest Symp. on Circuits and Systems, Lansing MI, 2000: 346-348.

[10] Yu Shen Lin, "Delay Efficient 32-bit Carry-Skip Adder", Electronics, Circuits and Systems, 2006, ICECS '06, pp. 506-509.

[11] Behnam Amelifard, Farzan Fallah, Massoud Pedram, "Closing the gap between Carry Select Adder and Ripple Carry Adder: A new class of Low-power and High-performance Adders", Proceedings of the Sixth International Symposium on Quality Electronic Design (ISQED'05) , 2005.

[12] Jin-Fu Li, Jiunn-Der Yu, Yu-Jen Huang, "A Design Methodology for Hybrid Carry-Lookahead/Carry-Select Adders with Reconfigurability", 0-7803-8834-8/05/$20 ©2005 IEEE, 2005.

[13] Pak K. Chan, et al, Delay Optimization of Carry-Skip Adders and Block Carry-Lookahead Adders Using Multidimensional Dynamic Programming, IEEE Transactions on Computers, vol. 41, No. 8 (Aug. 1992): 920-93.

[14] S. Shanthala, C.P. Raj, "Design and VLSI Implementation of Pipelined Multiply Accumulate Unit", International Conference on ETET-09, pp 381-386, 2009.

[15] Sarabdeep Singh and dilip Kumar, "Design of Area and Power Efficient Modified Carry Select Adder", International Journal of Computer Application (November 2011), vol. 33 – No.3.

[16] Mariano Aguirre-Hernandez and Monico Linares-Aranda, "CMOS Full-Adders for Energy-Efficient Arithmetic Applications", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol.19, No. 4, April 2011

[17] Deepa Sinha, Tripti Sharma, k.G.Sharma, Prof.B.P.Singh, "Design and Analysis of low Power 1-bit Full Adder Cell",IEEE, 2011.

[18] H. S. Dhillon , and A. Mitra , "A Reduced-Multiplication Algorithm for Digital Arithmetic ," International Journal of Computational and Mathematical Sciences, pp. 64-69, Spring 2008.

[19] Shubin.V.V, "Analysis and Comparison of Ripple Carry Full Adders by Speed", Micro/Nano Technologies and Electron Devices (EDM),2010, International Conference and Seminar on, pp.132- 135,2010.