International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 10, October - 2013

A Comparative Analysis of Different 8-Bit Adder Topologies at 45nm Technology

Suryasnata Tripathy [1] L B OM Prakash [2] B. S. Patro [3] Sushanta K. Mandal [4]
School of Electronics
KIIT University
Bhubaneswar, Odisha, India

***Abstract*— Adders form an indispensable part of digital integrated design due to their extensive application in efficient implementation of basic binary arithmetic. The pre- requisites of a basic adder topology are undoubtedly faster operating speed, acceptable power consumption and effectively lower on chip area. The present paper provides an in depth comparative analysis of various contemporary adder topologies. The detailed study of the adder architectures presented in this paper is intended to facilitate the trade-off between area, propagation delay and power dissipation while selecting an adder topology for any digital design. Five different adder topologies namely ripple carry adder, carry save adder, carry bypass adder, linear carry select adder and square root carry select adder are compared exhaustively on the basis of several design metric and performance parameters in this work using Cadence EDA tool in 45nm CMOS process technology. The robustness of the designs is analysed using corner analysis at various process corners.**

***Keywords*— Carry bypass adder, Carry save adder, CMOS, Corner analysis, Linear carry select adder, Ripple carry adder, Square root carry select adder**

I. INTRODUCTION

Adders are the most significant data path elements in a digital design given their prominence in implementing basic digital operations such as addition, subtraction, multiplication and division. The extensive use of different adder topologies for executing the basic binary arithmetic indicates that the accuracy, speed and overall performance of any digital system are largely influenced by the resident adder blocks. Different architectures for the binary addition process have been proposed [1-3, 6] over the years and rigorous attempts are still going on to improve the design and performance bottlenecks in the existing topologies. The design constraints can be summarised with respect to area and transistor count whereas the performance parameter includes factors such as delay and power consumption. Among the existing adder topologies, the ripple carry adder is by far the simplest one. The N-bit ripple carry addition process requires the cascading of N full adder blocks leading to a delay proportional to the operand size (i.e. the number of bits, N). The overall area for such an adder topology is a strong function of the operand size. The carry look-ahead adder topology certainly comes as an improvement over the ripple carry architecture in terms of delay, yet it suffers from irregular layout issues. This topology also suffers from fan-in and fan-out issues to ensure minimal performance degradation. Topologies such as carry bypass adder and carry save adder are evolved in order to facilitate fast addition operations. The carry select adder architecture provides a significant improvement in terms of the speed; however it inculcates a large area overhead. The improvement in speed is ensured by pre-computing the sum and carry values in any bit addition stage for all possible carry-in values ('0' and '1') from the previous stage and then selecting the appropriate results on the basis of the actual carry input bit. However, the fan out issues are critical for such topologies as the number of multiplexer units (required to select the appropriate sum value) to be driven by the carry input bit increases exponentially. [8]

The present paper focuses on comparing various design metrics of the different binary adder topologies. An 8-bit addition process is considered which is implemented successfully using the different architectures and the performance is compared based upon the results obtained. The functionality and performance analysis is carried out using Cadence EDA tool (virtuoso platform). Performance metric such as delay (with and without physical parasitics), power consumption, power delay product and design metric such as area and transistor count are analysed thoroughly. Corner analysis for individual topologies is performed to ensure their robustness. A basic cmos mirror topology for the one bit full adder block is used throughout the different architectures as the basic building block. A transmission gate based (6T) exclusive or design and a transmission gate based multiplexer design is used across the adder topologies.

The rest of the paper is organised as follows. Section 2 throws light on the various adder topologies and the issues concerning them. Section 3 summarises the analysis work involved with the implemented designs based upon the different performance metrics. The last section concludes the paper.

II. ADDER TOPOLOGIES

The present section of the paper presents different adder topologies that are used in this study. They are:

A. Ripple carry adder
B. Carry bypass adder
C. Carry save adder
D. Linear carry select adder
E. Square-root carry select adder

A. *Ripple carry adder(RCA)*

An N-bit binary ripple carry adder is constructed by cascading N full adder circuits in series where each full adder circuit in the architecture performs one bit stage addition operation. The carry-in (Cin) bit ripples from one bit stage to the next before reaching at the full adder stage responsible for summing up the most significant bits of the operands. The propagation delay encountered in such a topology is a strong function of the number of bit stages (N) as the addition operation at one bit stage is dependent upon the rippling of the carry-in bit through the previous stages. The type of input vector applied to the circuitry also determines the extent of rippling. The worst case delay for a ripple carry configuration happens when the carry generated at the least significant bit stage propagates all the way through the intermediate bit addition stages to be consumed at the most significant bit position[9]. The propagation delay for an n-bit addition process [8] can be approximated by:

$$\mathbf{t_{add} = (n\text{-}1)\ t_{carry} + t_{sum}}$$

where t_{add}, t_{carry} and t_{sum} are the total propagation delay for an n-bit adder, the delay from the carry-in bit to the carry-out bit in one bit stage and the delay from the carry-in bit to the sum bit in one bit stage respectively. The propagation delay for a N-bit ripple carry adder, as discussed above, is linearly proportional to the number of bit stages that is O(N).

B. *Carry bypass adder(CBA)*

The linear dependence of the speed of a binary adder topology on the number of bit stages involved necessitates the use of several logic optimization approaches in the architectural level. The impracticality in the use of basic ripple carry adders with a relative large word length leads to the topology level modifications to design faster adders.

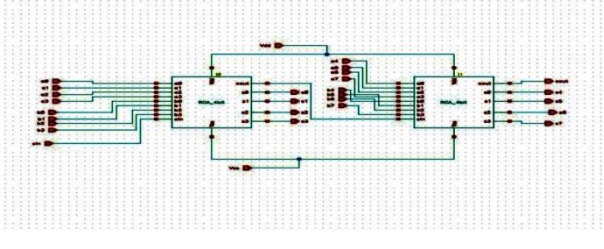


Fig 1: RCA 8-bit schematic (Generated using Cadence EDA tool)

Carry bypass topology [7] is one such modified adder topology that incorporates a carry bypass option rather than the basic rippling action, subjective to certain conditions. An incoming carry bit to a chain of adder stages propagates through the configuration if the propagate signals (for any bit stage addition with Ai and Bi be the inputs, the corresponding propagate signal would be given as Ai xor Bi) corresponding to all the individual stages are logic high. This is the necessary and sufficient condition to enable carry bypass across an adder chain. A basic multiplexer based architecture is developed incorporated with the basic cascaded full adder design to implement the carry bypass topology. For an N-bit addition operation, an approximate expression for the total propagation delay [8] can be written as:

$$\mathbf{T_p = t_{setup} + M\ t_{carry} + (N/M - 2)\ t_{bypass} + (M\text{-}1)\ t_{carry} + t_{sum}}$$

where $\mathbf{T_p}$**:** the total propagation delay, $\mathbf{t_{setup}}$: the fixed overhead time to create the generate and propagate signals, $\mathbf{t_{carry}}$: the propagation delay through a single bit stage, $\mathbf{t_{bypass}}$: the propagation delay through the bypass multiplexer for a single bypass stage, $\mathbf{t_{sum}}$: the time to generate the sum at the final stage. M, in the above expression, stands for the number of bits per stage and thus N/M indicates the number of addition stages used. The improvement in performance becomes significant with larger values of N; however, for smaller word length applications, the carry bypass topology fails to provide any significant advantage over the basic ripple carry binary adder architecture.

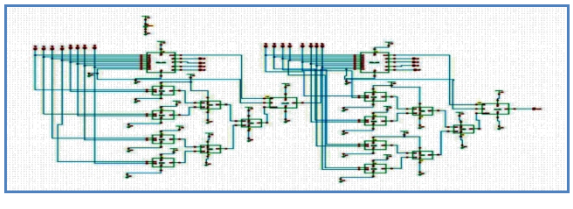


Fig 2: Carry bypass adder (8-bit) Schematic (Generated using Cadence EDA tool)

C. *Carry save adder(CSA)*

The need of speed optimization in addition operation for high speed datapaths led to the development of fast adder architectures. The carry save adder [5] topology is one such fundamental fast adder design available to the digital designers. The basic principle of this architecture is to save and forward the carry bits from individual bit adder stages (of one adder chain) to the next appropriate bit adder stage (of the next adder chain) instead of rippling it across the same stage. The idea of saving the carry at one stage and utilising it at the next can be continued indefinitely without any immediate carry propagation. These adder

stages can be arranged in a binary tree like structure where the cumulative delay depends largely upon the number of inputs to be added, rather than the number of bits per input. The ultimate addition stage of such a carry save topology is a vector merging adder that incorporates the carry propagation (rippling) among the constituent full adders of the stage.

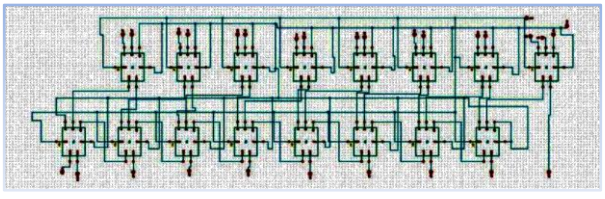


Fig 3: 8-bit Carry save adder schematic (Generated using Cadence EDA tool)

D. Linear carry select adder(LCA)

The latency introduced in to the addition process due to rippling of carry bits across adder blocks from the least significant position to the most significant stage is the main design bottleneck for binary adder design. This linear dependency among bit adder stages can be compromised by anticipating all the possible carry input values from the previous stage in advance and perform the addition for all those values independently and then selecting the appropriate result depending upon the actual carry input from the previous stage. This procedure is the basic principle behind the linear carry select adder architecture [4, 10]. The extra hardwire overhead used to evaluate an addition stage with all possible carry input values is compensated by the improvement of speed introduced by them. The selection of the appropriate result depending upon the actual carry input is performed by a chain of multiplexers. The use of extra hardwire for the carry select topology results in a large silicon area overhead and a comparatively higher power dissipation. However, the optimization is obtained in terms of overall propagation delay reduction, which provides the design trade off. This adder topology has a gate level depth of O(√N) for an N-bit addition operation. The first order model for the worst case propagation delay [8] corresponding to an N-bit carry select adder can be written as:

$$T_{add} = t_{setup} + M\, t_{carry} + (N/M)\, t_{mux} + t_{sum}$$

where, t_{setup}, t_{sum} **and** t_{mux} are fixed delays corresponding to the fixed time overhead for setting up all addition stages, generating the sum in the last addition stage and the time delay introduced by each multiplexer stage respectively. The t_{carry} is the time delay of the carry through a single one bit full adder stage. M stands for the number of bits per adder stage; hence, (N/M) represents the total number of stages involved in the N-bit adder design.

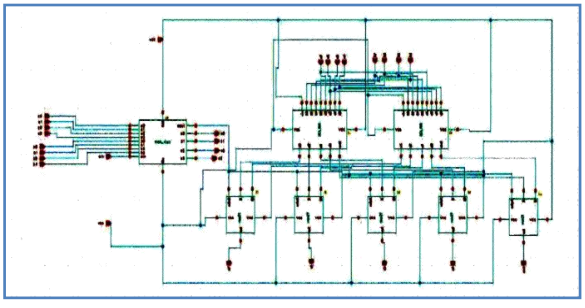


Fig 4: 8-bit linear carry select adder schematic (Generated using Cadence EDA tool)

E. Square root carry select adder(SqrtCA)

The linear carry select design splits the N-bit addition to several stages of addition with each stage having same number of operand bits to deal with, that is, each stage incorporates the same length of full adder chain. However, this design criterion introduces a severe speed limitation as the latency introduced due to the fact that higher level adder blocks still have to wait for the lower level addition and carry selection process to complete. This is caused because of the mismatch between the arrival times of the carry-in signals at different adder blocks. The square root carry select adder [8] is an improved carry select topology that removes these limitations associated with the linear carry select adder design. The basic approach to do away with the latency is to remove the mismatch between the adder stages and the multiplexer stages. One of the ways to address this issue is to equalize the delay through both paths by progressively adding more bits to subsequent adder stages instead of biding for an equal partition of operand bits for each stage. This implies the first stage can be a 2-bit adder structure, the second stage a 3-bit adder structure, the third stage a 4-bit adder structure and so on. In convention, an N-bit addition process results in a propagation delay proportional to √N and for larger values of N, the propagation delay experienced by a square root adder design becomes mostly constant. The propagation delay for such a topology [8] can be expressed as:

$$T_{add} = t_{setup} + M\, t_{carry} + (\sqrt{N})\, t_{mux} + t_{sum}$$

where the symbols have the same meaning as explained in the linear carry select adder section.

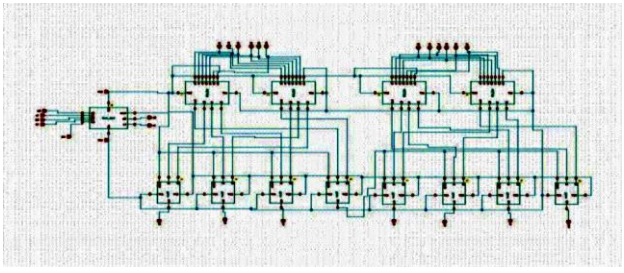


Fig 5: 8-bit Square root carry select adder (Generated using Cadence EDA tool)

III. PERFORMANCE ANALYSIS

To evaluate the performance metrics associated with the different adder topologies discussed above, 8-bit adder configurations for all the architectures are implemented in **45nm** technology using **Cadence EDA tool** in **Virtuoso** platform. The schematics of the different topologies are converted to their corresponding physical layouts using the layout XL editor of the tool and DRC and LVS analysis are performed to ensure authenticity of the designs. Both the schematic and av-extracted (layout with the physical parasitics) form of each of the topologies are subjected to repeated verification to ensure functionality match. A common test vector is applied to both the forms in a config view platform to estimate the effects of the parasitics associated with each design. The power consumptions associated with the schematic and the physical layout of a design are estimated using the same test vector, with an input power supply of 1 volt. Corner analysis is performed for all the topologies to see the performance metric variation and to validate the robustness of each design. The propagation delay is also analysed at different temperatures and voltage supplies to establish the relation between them. The various results for the performance analysis are tabulated in table I-VIII.

TABLE I: AREA AND TRANSISTOR COUNT OF THE ADDER TOPOLOGIES

Name of the topology	**Transistor count**		**Total area (in μm^2)**
	Nmos count	Pmos count	
Ripple carry adder	112	112	837.71
Carry bypass adder	160	160	2232.22
Carry save adder	224	224	1234.25
Linear carry select adder	183	183	3097.89
Sq. root carry select adder	220	220	2222.18

Table I summarizes the silicon area overhead of the adder topologies. The 8-bit ripple carry adder happens to be the most area efficient design. The other adder architectures seem to consume greater on chip area because of the design complexity introduced to obtain performance optimization.

TABLE II: PROPAGATION DELAY ASSOCIATED WITH THE ADDER TOPOLOGIES

Name of the topology	**Propagation delay (pS)**	
	Schematic	Av_extracted
Ripple carry adder	0151	0.374
Carry bypass adder	0.161	0.320
Carry save adder	0.137	0.139
Linear carry select adder	0.131	0.253
Square root carry select adder	0.112	0.119

Table II enlists the propagation delays associated with the different adder topologies. The schematic column includes the worst case delay between the carry-in and carry-out bits for the corresponding schematic of the architecture whereas the Av_extracted column highlights the post layout delay which incorporates the physical parasitic effects. The square root carry select adder proves to be the most efficient design with respect to timing and speed constraints. The carry bypass and ripple carry adder configurations seem to have similar results. Since for lower order adders, the carry bypass adder topology nearly resembles the ripple carry configuration as far as performance is concerned. The power dissipation associated with the different adder designs are tabulated

in Table III. Both pre layout and post layout power dissipation values are given to illustrate the physical parasitic effects. The carry select topologies prove to be the least power efficient configurations due to the high hardwire complexity associated with the designs.

Table III: Power Dissipation Associated with the Adder Topologies

Name of the topology	**Power dissipation (µW)**	
	Schematic	Av_extracted
Ripple carry adder	0.502	0.504
Carry bypass adder	0.602	2.172
Carry save adder	0.603	1.704
Linear carry select adder	0.835	3.112
Square root carry select adder	0.952	3.391

Table IV: Power-Delay Product of the Adder Topologies

Name of the topology	**Power-Delay product (watt. sec)**
Ripple carry adder	0.076 E -15
Carry bypass adder	0.096 E -15
Carry save adder	0.082 E -15
Linear carry select adder	0.109 E -15
Square root carry select adder	0.107 E -15

The power delay products for the individual adder topologies are listed in Table IV. The square root carry select adder architecture and the linear carry select design are the ones to have a relatively larger power delay product for the 8-bit configuration presented in the paper. The ripple carry adder demonstrates a fairly low power delay product among all the topologies discussed in the present work.

A. Corner Analysis

In semiconductor manufacturing, a process corner is a design-of-experiments technique which corresponds to a variation in fabrication parameters used in applying an integrated circuit design to a semiconductor wafer. A process corner represents a three or six sigma variation from nominal doping concentrations and other such fabrication parameters that allow a designer to test any design against various adverse fabrication conditions in order to verify the robustness. In a convention, three process corners are said to exist, namely typical, fast and slow, where the first one corresponds to nominal fabrication conditions, while the other two correspond to conditions with higher and lower carrier mobility (with respect to nominal value) respectively. Thus, in general, five such process corners are possible, notably ff (fast-fast), fs (fast-slow), sf (slow-fast), ss (slow-slow) and tt (typical-typical; same as normal process corner) where the first letter in the naming convention indicates the NFET condition and the second letter corresponds to the PFET condition. The adder topologies under analysis in this paper are subjected to corner analysis to verify the robustness. Corner analysis is performed for both schematic and layout stages of the designs against all the five process corner paradigms. The results of the pre layout and post layout corner analysis are listed in Table V and Table VI respectively.

B. Temperature Analysis for the propagation delay

The variation of the propagation delay parameter with respect to temperature is illustrated in Table V. The adder architectures are simulated at two different temperatures, one being close to room temperature and the other being an adversely high value, with the same test vectors, to show the relationship between temperature and propagation delay.

TABLE V: CORNER ANALYSIS FOR PRE-LAYOUT POWER DISSIPATION(PD) AND PROPAGATION DELAY(PRLD)

Name of the topology	**Design parameter**	**Corners**				
		Tt	**ff**	**fs**	**sf**	**ss**
RCA	PrLD (ns)	0.151	0.111	0.174	0.136	0.206
	PD (uwatt)	0.500	0.544	0.495	0.498	0.455
CSA	PrLD (ns)	0.137	0.107	0.128	0.147	0.186
	PD (uwatt)	0.603	0.829	0.643	0.577	0.526
CBA	PrLD (ns)	0.161	0.125	0.152	0.130	0.207
	PD (uwatt)	0.602	0.668	0.595	0.600	0.547
LCA	PrLD (ns)	0.131	0.123	0.104	0.142	0.166
	PD (uwatt)	0.835	0.899	0.826	0.831	0.770
SqrtCA	PrLD (ns)	0.112	0.089	0.101	0.123	0.139
	PD (uwatt)	0.952	1.052	0.948	0.949	0.882

TABLE VI: CORNER ANALYSIS FOR POST-LAYOUT POWER DISSIPATION AND PROPAGATION DELAY (PoLD)

Name of the topology	**Design parameter**	**Corners**				
		tt	**Ff**	**fs**	**sf**	**Ss**
RCA	PoLD (ns)	0.360	0.273	0.399	0.341	0.497
	PD (uwatt)	0.504	0.547	0.496	0.501	0.457
CSA	PoLD (ns)	0.139	0.113	0.150	0.132	0.181
	PD (uwatt)	1.704	1.837	1.721	1.686	1.598
CBA	PoLD (ns)	0.320	0.230	0.301	0.341	0.587
	PD (uwatt)	2.172	2.318	2.167	2.177	2.072
LCA	PoLD (ns)	0.253	0.249	0.201	0.267	0.341
	PD (uwatt)	3.112	3.286	3.107	3.118	2.984
SqrtCA	PoLD (ns)	0.119	0.096	0.130	0.126	0.146
	PD (uwatt)	3.391	3.584	3.388	3.395	3.249

With increase in temperature, the propagation delay increases, however, the variation in the propagation delay is not large enough. This table speaks about the robustness of different adder topologies with respect to temperature.

C. Supply voltage vs. propagation delay

The variation of performance of the adder configurations with respect to variation in the power supply values is highlighted in Table VIII. With lower supply voltage, the performance seems to degrade a bit in this 45 nm technology environment. The performance seems to improve with the increase in power supply value.

D. *Summary*

The above tabulation and relevant discussion summarizes the robustness of the adder topologies against different design and performance metrics such as area, propagation delay and power overhead. Fig 6 gives a clear idea about the propagation delay associated with various adder topologies under analysis. The square root carry select architecture appears to be the most time aware design. The carry bypass adder matches the ripple carry adder performance as far as propagation delay is concerned because of smaller word length analysis. For higher order operations i.e. with larger values of N (word length), it is expected to provide noticeable advantage over the basic ripple carry binary adder in terms of overall propagation delay and hence, the speed of operation.

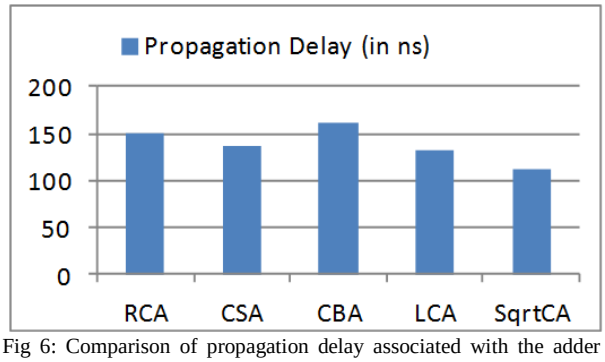


Fig 6: Comparison of propagation delay associated with the adder topologies

Fig 7 signifies the differences in power dissipation among the various adder topologies. The ripple carry adder seems to be the most power efficient design; however, it fails to deliver adequate speed of operation. The square root carry select architecture accounts for the maximum power dissipation among all the topologies under scanner. This power overhead is apprehended by the additional hardwires present in the circuitry to provide the performance optimization.

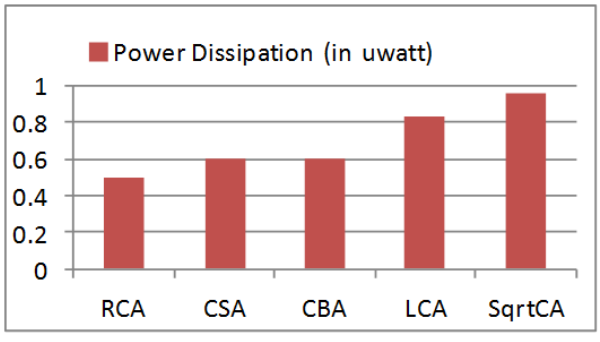


Fig 7: Comparison of power dissipation associated with the adder topologies

Fig 8 displays the area overhead associated with the different adder designs. The ripple carry adder seems to have the most compact design while the linear carry select topology acquires the largest on chip area. Carry save adder configuration occupies moderate silicon area while the other topologies fall in between the two extremes.

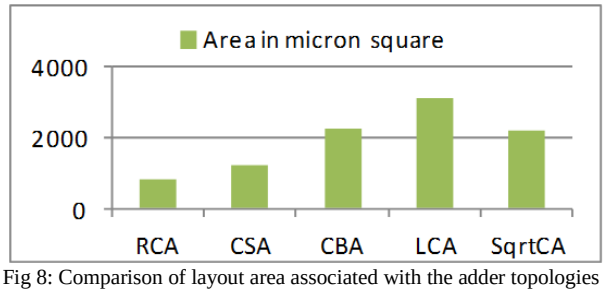


Fig 8: Comparison of layout area associated with the adder topologies

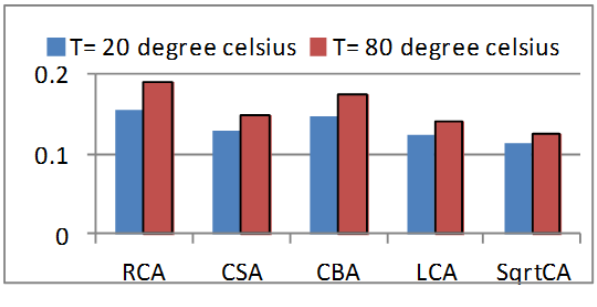


Fig 9: Pre layout propagation delay variation with temperature

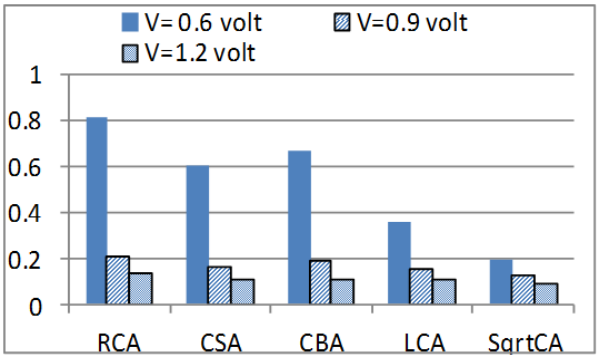
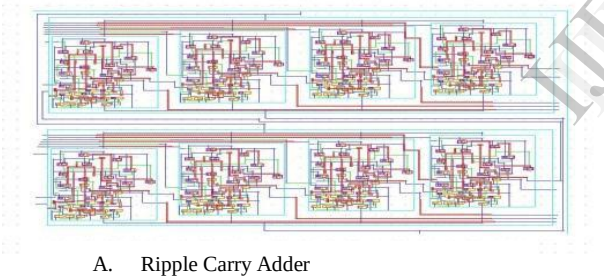
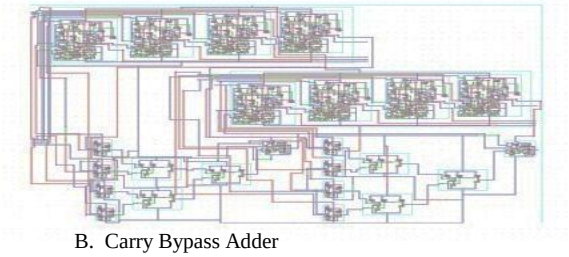


Fig 10: Pre layout propagation delay variation with supply voltage

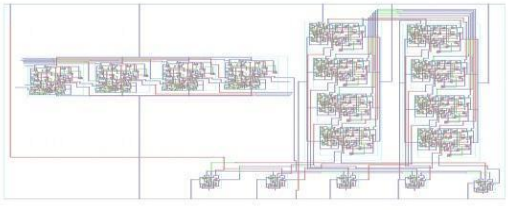
The variation of the pre layout propagation delay with environmental conditions such as temperature and supply voltage is displayed in Fig 9 and Fig 10 respectively. Similar results can be shown by plotting the post layout delays against such varying environmental conditions. The design robustness and effectiveness against differential environmental conditions have been analysed here for all the adder topologies. The layouts of the different designs are shown in figure 11.



A. Ripple Carry Adder

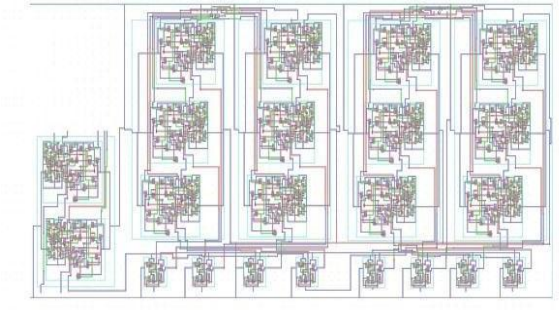


B. Carry Bypass Adder



C. Linear Carry Adder

D. Carry Save Adder



E. Square root Carry Adder

Fig 11: Layouts of various adder topologies.

Table VII: Propagation delay variation with temperature

Name of the topology	Design parameter	Temperature (in degree Celsius)	
		20	80
RCA	PrLD (ns)	0.156	0.190
	PoLD(ns)	0.332	0.426
CSA	PrLD (ns)	0.129	0.148
	PoLD(ns)	0.248	0.297
CBA	PrLD (ns)	0.148	0.174
	PoLD(ns)	0.328	0.341
LCA	PrLD (ns)	0.123	0.142
	PoLD(ns)	0.232	0.280
SqrtCA	PrLD (ns)	0.106	0.119
	PoLD(ns)	0.114	0.125

Table VIII: Propagation delay variation with power supply

Name of the topology	Design parameter	Supply Voltage (in volt)		
		0.6	0.9	1.2
RCA	PrLD (ns)	0.810	0.204	0.132
	PoLD(ns)	3.304	0.467	0.249
CSA	PrLD (ns)	0.604	0.164	0.106
	PoLD(ns)	1.329	0.315	0.218
CBA	PrLD (ns)	0.666	0.190	0.107
	PoLD(ns)	3.172	0.539	0.239
LCA	PrLD (ns)	0.354	0.149	0.103
	PoLD(ns)	1.043	0.289	0.211
SqrtCA	PrLD (ns)	0.194	0.125	0.089
	PoLD(ns)	0.197	0.127	0.091

IV. CONCLUSION

In the present work, an exhaustive analysis of five different adder topologies implementing an 8 bit addition process in 45 nm CMOS process technology has been performed using Cadence EDA tool. The different performance metric and design parameters are compared and the results are tabulated. The impacts of physical parasitics associated with the layouts on the performance of the adder topologies are thoroughly analysed. The topologies are tested for design robustness through process corner analysis. The temperature and power supply dependence of the propagation delay of individual architectures is also studied. All the simulation results and performance analysis results are tabulated in the previous sections. The results suggest that ripple carry architecture happens to be the most compact and power efficient design while the square root carry select adder topology provides a reasonable improvement in speed at the cost of silicon area and power overhead. The adder topologies that correspond to smaller silicon area and transistor count are to be preferred for designs that require compactness and are constraint with respect to on chip area. Digital designs that demand high speed operation must prefer faster topologies (such as square root carry select adder) though the improvement in speed would come at the cost of larger on chip area and higher power dissipation.

ACKNOLEDGEMENT

We would like to express our gratitude to Kalinga Institute of Industrial Technology, Bhubaneswar, Odisha for providing CADENCE EDA Tool, used to carry out this work.

Reference

[1] Reza Faghih Mirzaee, Mohammad Hossein Moaiyeri, Keivan Navi, "High Speed NP-CMOS and Multi-Output Dynamic Full Adder Cells", World Academy of Science, Engineering and Technology 39, 2010.

[2] Padma Devi, Ashima Girdher, Balwinder Singh, **"**Improved Carry Select Adder with Reduced Area and Low Power Consumption"**,** International Journal of Computer Applications (0975 – 8887), Vol 3 – No.4, June 2010.

[3] R Uma, Vidya Vijayan, M Mohanapriya, Sharon Paul, "Area, delay and power comparison of adder topologies", International journal of VLSI design and Communication Systems, Vol 3, No 1, February 2012.

[4] I-Chyn Wey, Cheng-Chen Ho, Yi-Sheng Lin, and Chien-Chang Peng, "An Area-Efficient Carry Select Adder Design by Sharing the Common Boolean Logic Term", proceedings of the international multi conference of engineers and computer scientists,2012, Vol II, IMECS 2012, March 14-16, 2012, Hong Kong.

[5] Chakib Alaoui, "Design and Simulation of a Modified Architecture of Carry Save Adder", International Journal of Engineering (IJE), Vol 5: Issue (1): 2011.

[6] Andy Goyal, Ashish Kumar, "Comparative analysis of energy-efficient low power 1-bit full adders at 120nm technology", International Journal of Advances in Engineering & Technology, July 2012.

[7] Pak K Chan, Matrine D F Schlag, Clark D. Thomborson, Vojin G.Oklobdzija, "Delay optimization of carry skip adders and block carry look ahead adders using multidimensional dynamic programming", IEEE transactions on computers, vol. 4i, no. 8, august 1992.

[8] Jan M Rabaey, A Chandrakasan, B Nikolic, "Digital Integrated Circuits- A design perspective" 2nd Ed. , Pearson Education, 2003.

[9] M. Morris Mano, Michael D. Ciletti, "Digital design", fourth edition. Pearson Education India, 2008.

[10] Ohsang Kwon, Earl E. Swartzlander Jr., Kevin J. Nowka, "A fast hybrid carry-lookahead/carry-select adder design". ACM Great Lakes Symposium on VLSI 2001: 149-152.

Authors

Mr.Suryasnata Tripathy

He received his B.Tech in Electronics & Telecommunication Engineering from C.V. Raman College of Engineering, Bhubaneswar under Biju Patnaik University & Technology, Odisha. Currently he is an M.Tech student in VLSI design & Embedded System in School of Electronics Engineering at KIIT University, Bhubaneswar, Odisha. His main researches focus on low power, high speed digital circuit data-path design.

Mr. L.B.Omprakash

He received his B.Tech in Electronics & Telecommunication Engineering from Konark Institute of Science Technology, Bhubaneswar under Biju Patnaik University & Technology, Odisha. Currently he is pursuing his M.Tech in VLSI design & Embedded System at School of Electronics Engineering under KIIT University, Bhubaneswar, Odisha. His main researches focus on low voltage, low power, high speed digital circuit design and reversible based logical design.

Mr. B. Shivalal Patro

He has completed his B.Tech from Trident Academy of Technology in Electronics & Telecommunication Engineering in 2010, Odisha. He then completed M.Tech in Communication Systems from KIIT University in 2012. Currently, he is a Ph.D scholar at KIIT University, Bhubaneswar, Odisha. His area of research includes high speed, low power analog and mixed signal IC design and optimization.

Dr. Sushanta K. Mandal

He received his B.E. degree in Electrical Engineering from Jalpaiguri Govt. Engineering College, West Bengal, India in 1993, MS and Ph.D degree from Indian Institute of Technology (IIT), Kharagpur, India in 2002 and 2008 respectively. He has 12 years of teaching and industry experience. He was actively associated with Advanced VLSI Design Laboratory at IIT. Kharagpur from 2002-2007. From August 2007 to June 2010, he worked as an Assistant Professor at Dhirubhai Ambani Institute of Information and Communication Technology (DA-IICT), Gandhinagar. Currently he is a Professor at KIIT University, Orissa since July 2010. He has published number of research papers in reputed journals and conferences. His research interests include modelling and synthesis of on-chip passive devices, digital, analog and mixed-signal CMOS VLSI design, high-level synthesis and optimization of analog circuits, soft computing applications in VLSI.