Area, Delay and Power Comparison of Adder Topologies

[1]R.UMA,Vidya Vijayan[2], M. Mohanapriya[2], Sharon Paul[2]

[1]Research Scholar,
Department of Computer Science
Pondicherry University, Pondicherry,India
uma.ramadass1@gmail.com
[2]UG Students,
Department of Electronics and Communication Engineering
Rajiv Gandhi College of Engineering and Technology
Pondicherry, India

Abstract

Adders form an almost obligatory component of every contemporary integrated circuit. The prerequisite of the adder is that it is primarily fast and secondarily efficient in terms of power consumption and chip area. This paper presents the pertinent choice for selecting the adder topology with the tradeoff between delay, power consumption and area. The adder topology used in this work are ripple carry adder, carry look-ahead adder, carry skip adder, carry select adder, carry increment adder, carry save adder and carry bypass adder. The module functionality and performance issues like area, power dissipation and propagation delay are analyzed at 0.12µm 6metal layer CMOS technology using microwind tool.

Keywords : *Ripple Carry Adder, Carry Save Adder, Carry Increment Adder, Carry Select Adder.*

I. INTRODCUTION

Cell-based design techniques, such as standard-cells and FPGAs, together with versatile hardware synthesis are rudiments for a high productivity in ASIC design. In the majority of digital signal processing (DSP) applications the critical operations are the addition, multiplication and accumulation. Addition is an indispensable operation for any digital system, DSP or control system. Therefore a fast and accurate operation of a digital system is greatly influenced by the performance of the resident adders. Adders are also very significant component in digital systems because of their widespread use in other basic digital operations such as subtraction, multiplication and division. Hence, improving performance of the digital adder would extensively advance the execution of binary operations inside a circuit compromised of such blocks. Many different adder architectures for speeding up binary addition have been studied and proposed over the last decades. For cell-based design techniques they can be well characterized with respect to circuit area and speed as well as suitability for logic optimization and synthesis.

Ripple Carry Adder (RCA)[1][2] is the simplest, but slowest adders with $O(n)$ area and $O(n)$ delay, where n is the operand size in bits. Carry Look-Ahead (CLA)[3][4] have $O(n \cdot \log(n))$ area and $O(\log(n))$ delay, but typically suffer from irregular layout. On the other hand, Carry

DOI : 10.5121/vlsic.2012.3113

Skip Adder(CSA)[5][6],carry increment[7][8]and carry select[9][10] have O(n) area and $O(n^{l+2/l+1})$ delay provides a good compromise in terms of area and delay, along with a simple and regular layout. Carry save adder have O(n) area and O(log n) delay. CLA adders can be realized in two gate levels provided there is no limit on fan in/out. The carry select adders (CSelA) reduce the computation time by pre-computing the sum for all possible carry bit values (ie '0' and '1'). After the carry becomes available the correct sum is selected using multiplexer. Carry Select Adder are in the class of fast adders, but they suffer from fan-out limitation since the number of multiplexers that need to be driven by the carry signal increases exponentially. In the worst case, a carry signal is used to select n/2 multiplexers in an n-bit adder. When three or more operands are to be added simultaneously using two operand adders, the time consuming carry propagation must be repeated several times. If the number of operands is 'k', then carries have to propagate (k-1). The existing adder topology is presented in Figure (1).

In the present work, the design of an 8-bit adder topology like ripple carry adder, carry look-ahead adder, carry skip adder, carry select adder, carry increment adder, carry save adder and carry bypass adder are presented. The functionality and performance analysis are done using microwind. Since Microwind integrates traditionally separated front-end and back-end chip design into an integrated flow, accelerating the design cycle and reduced design complexities. It tightly integrates mixed-signal implementation with digital implementation, circuit simulation, transistor-level extraction and verification. Performance issues like area, power dissipation and propagation delay for all the adders are analyzed at 0.12µm 6metal layer CMOS technology using microwind tool.

The remainder of this paper is organized as follows. Section II explains the topology detail of 8-bit adders. Section III presents the performance analysis. Section IV presents the simulation results implemented in 0.12-µm CMOS technology. Section V discusses summary and the final section presents the conclusion.

II. ADDER TOPOLOGIES

This section presents the design of adder topology. In this work the following adder structures are used:

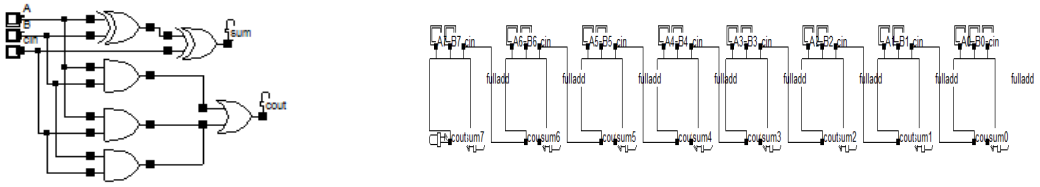
- Ripple Carry Adder
- Carry Save Adder
- Carry Look-Ahead Adder
- Carry Increment adder
- Carry Skip Adder
- Carry Bypass Adder
- Carry Select Adder

A. Ripple Carry Adder (RCA)

The ripple carry adder is constructed by cascading full adders (FA) blocks in series. One full adder is responsible for the addition of two binary digits at any stage of the ripple carry. The carryout of one stage is fed directly to the carry-in of the next stage. Even though this is a simple adder and can be used to add unrestricted bit length numbers, it is however not very efficient when large bit numbers are used. One of the most serious drawbacks of this adder is that the delay increases linearly with the bit length. The worst-case delay of the RCA is when a carry signal transition ripples through all stages of adder chain from the least significant bit to the most significant bit, which is approximated by:

$$t = (n-1)t_c + t_s \qquad \text{Eq (1)}$$

where t_c is the delay through the carry stage of a full adder, and t_s is the delay to compute the sum of the last stage. The delay of ripple carry adder is linearly proportional to n, the number of bits, therefore the performance of the RCA is limited when n grows bigger. The advantages of the RCA are lower power consumption as well as compact layout giving smaller chip area. The design schematic of RCA is shown in Figure (2). The simulation result is shown in Figure (3a).



a. Full adder design **b. Design of Ripple Carry Adder**

Figure 2 Schematic of RCA

B. Carry Save Adder (CSaA)

The carry-save adder [11][12]reduces the addition of 3 numbers to the addition of 2 numbers. The propagation delay is 3 gates regardless of the number of bits. The carry-save unit consists of *n* full adders, each of which computes a single sum and carries bit based solely on the corresponding bits of the three input numbers. The entire sum can then be computed by shifting the carry sequence left by one place and appending a 0 to the front (most significant bit) of the partial sum sequence and adding this sequence with RCA produces the resulting *n* + 1-bit value. This process can be continued indefinitely, adding an input for each stage of full adders, without any intermediate carry propagation. These stages can be arranged in a binary tree structure, with cumulative delay logarithmic in the number of inputs to be added, and invariant of the number of bits per input. The main application of carry save algorithm is, well known for multiplier architecture is used for efficient CMOS implementation of much wider variety of algorithms for high speed digital signal processing .CSA applied in the partial product line of array multipliers will speed up the carry propagation in the array. The design schematic of Carry Save Adder is shown in Figure (4). The simulation result is shown in Figure (3b).

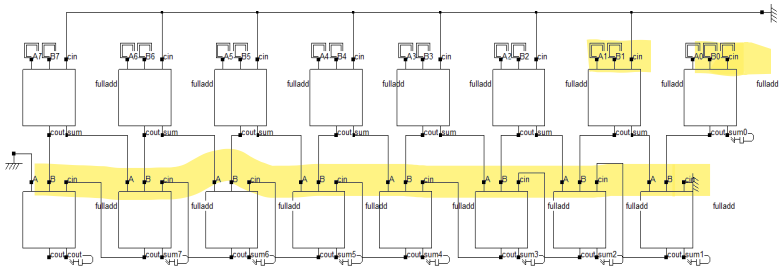


Figure 4 Schematic of Carry Save Adder

C. Carry Look-Ahead Adder

Carry look-ahead adder is designed to overcome the latency introduced by the rippling effect of the carry bits. The propagation delay occurred in the parallel adders can be eliminated by carry look ahead adder. This adder is based on the principle of looking at the lower order bits of the

augends and addend if a higher order carry is generated. This adder reduces the carry delay by reducing the number of gates through which a carry signal must propagate. Carry look ahead 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. The net effect is that the carries start by propagating slowly through each 4-bit group, just as in a ripple-carry system, but then moves 4 times faster, leaping from one look ahead carry unit to the next. Finally, within each group that receives a carry, the carry propagates slowly within the digits in that group

This adder consists of three stages: a propagate block/ generate block, a sum generator and carry generator. The generate block can be realized using the expression

$$G_i = A_i.B_i \qquad \text{for i=0,1,2,3} \qquad \text{Eq (2)}$$

Similarly the propagate block can be realized using the expression

$$P_i = A_i \oplus B_i \qquad \text{for i=0,1,2,3} \qquad \text{Eq (3)}$$

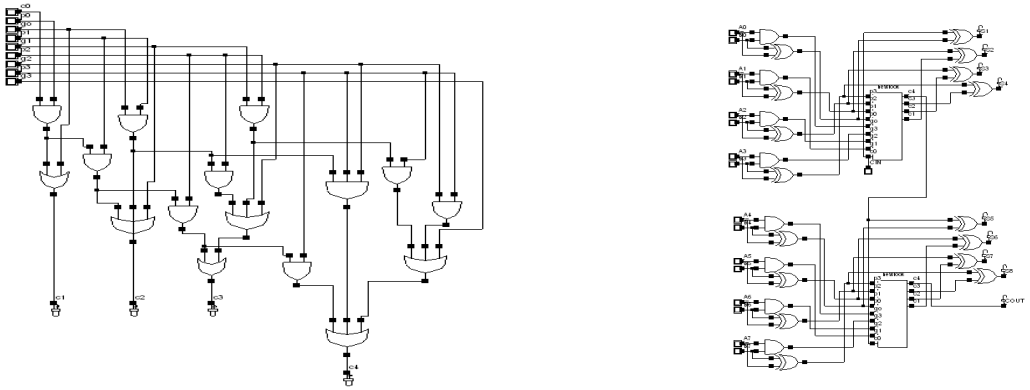
The carry output of the (i-1)th stage is obtained from

$$C_i(cout) = G_i + P_i C_{i-1} \quad \text{for i=0,1,2,3} \quad \text{Eq (4)}$$

The sum output can be obtained using

$$S_i = A_i \oplus B_i C_{i-1} \quad \text{for i=0,1,2,3} \qquad \text{Eq (5)}$$

An 8 bit look ahead adder using two four bit look ahead block is shown in Figure (5) the COUT of the 4-bit CLA is given as the CIN for the second 4-bit CLA. The simulation result is shown in Figure (3c)



a. Look-Ahead Block **b. 8-bit Look-Ahead Adder**

Figure 5 Schematic of Carry Look-Ahead Adder

Carry Increment Adder (CIA)

An 8-bit increment adder includes two RCA (Ripple carry adder) of four bit each. The first ripple carry adder adds a desired number of first 4-bit inputs generating a plurality of partitioned sum and partitioned carry. Now the carry out of the first block RCA is given to CIN of the conditional increment block. Thus the first four bit sum is directly taken from the ripple carry output. The second RCA block regardless of the first RCA output will carry out the addition operation and will give out results which are fed to the conditional increment block. The input CIN to the first RCA block is given always low value. The conditional increment block consists of half adders. Based on the value of cout of the 1st RCA block, the increment operation will take place. Here the half adder in carry increment block performs the increment operation. Hence the output sum

of the second RCA is taken through the carry increment block. The design schematic of Carry Increment Adder is shown in Figure (6). The simulation result is shown in Figure (3d).

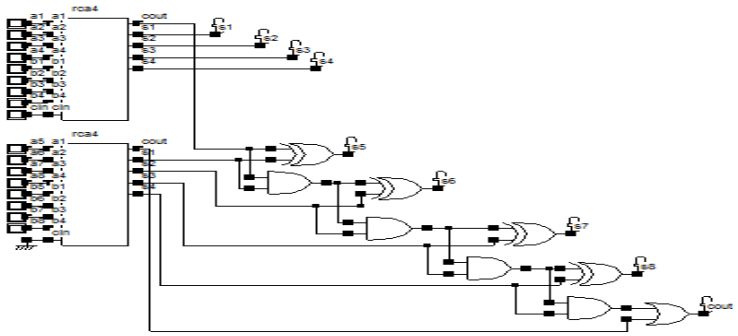


Figure 6 Schematic of Carry Increment Adder

D. Carry Skip Adder (CSkA)

A carry-skip adder consists of a simple ripple carry-adder with a special speed up carry chain called a skip chain. Carry skip adder is a fast adder compared to ripple carry adder when addition of large number of bits take place; carry skip adder has $O(\sqrt{n})$ delay provides a good compromise in terms of delay, along with a simple and regular layout This chain defines the distribution of ripple carry blocks, which compose the skip adder. A carry-skip adder is designed to speed up a wide adder by aiding the propagation of a carry bit around a portion of the entire adder. Actually the ripple carry adder is faster for small values of N. However the industrial demands these days, which most desktop computers use word lengths of 32 bits like multimedia processors, makes the carry skip structure more interesting. The crossover point between the ripple-carry adder and the carry skip adder is dependent on technology considerations and is normally situated 4 to 8 bits. The carry-skip circuitry consists of two logic gates. The AND gate accepts the carry-in bit and compares it to the group propagate signal

$$p_{[i,i+3]} = p_{i+3} \bullet p_{i+2} \bullet p_{i+1} \bullet p_i \qquad \text{Eq (6)}$$

using the individual propagate values. The output from the AND gate is ORed with cout of RCA to produce a stage output of

$$carry = c_{i+4} + p_{[i,i+3]} \bullet c_i \qquad \text{Eq (7)}$$

If $p_{[i,i+3]}$=0, then the carry-out of the group is determined by the value of c_{i+4}. However, if $p_{[i,i+3]}$=1 when the carry-in bit is c_i=1, then the group carry-in is automatically sent to the next group of adders. The design schematic of Carry Skip Adder is shown in Figure (7). The simulation result is shown in Figure (3e).

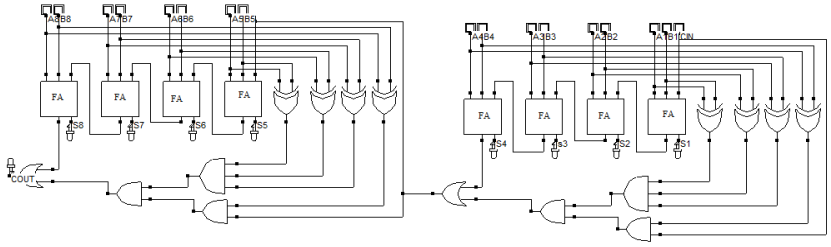


Figure 7 Schematic of Carry Skip Adder

E. Carry Bypass Adder (CByA)

As in a ripple-carry adder, every full adder cell has to wait for the incoming carry before an outgoing carry can be generated. This dependency can be eliminated by introducing an additional bypass (skip) to speed up the operation of the adder. An incoming carry Ci,0=1 propagates through complete adder chain and causes an outgoing carry C0,7=1 under the conditions that all propagation signals are 1. This information can be used to speed up the operation of the adder, as shown in Figure (8). When BP = P0P1P3P4P5P6P7P8 = 1, the incoming carry is forwarded immediately to the next block through the bypass and if it is not the case, the carry is obtained via the normal route. If (P0P1P3P4P5P6P7 = 1) then C0,7 = Ci,0 else either Delete or Generate occurred. Hence, in a CBA the full adders are divided into groups, each of them is "bypassed" by a multiplexer if its full adders are all in propagate. The simulation result is shown in Figure (3f).

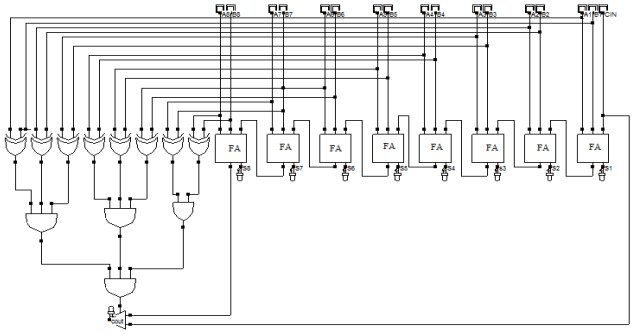


Figure 8 Schematic of Carry Bypass Adder

F. Carry Select Adder (CSelA)

A carry-select adder is divided into sectors, each of which – except for the least-significant – performs two additions in parallel, one assuming a carry-in of zero, the other a carry-in of one. A four bit carry select adder generally consists of two ripple carry adders and a multiplexer. The carry-select adder is simple but rather fast, having a gate level depth of $O(\sqrt{n})$. Adding two n-bit numbers with a carry select adder is done with two adders (two ripple carry adders) in order to perform the calculation twice, one time with the assumption of the carry being zero and the other assuming one. After the two results are calculated, the correct sum, as well as the correct carry, is then selected with the multiplexer once the correct carry is known. The design schematic of Carry Select Adder is shown in Figure (9). A carry-select adder speeds 40% to 90%faster than RCA by performing additions in parallel and reducing the maximum carry path. The simulation result is shown in Figure (3g).

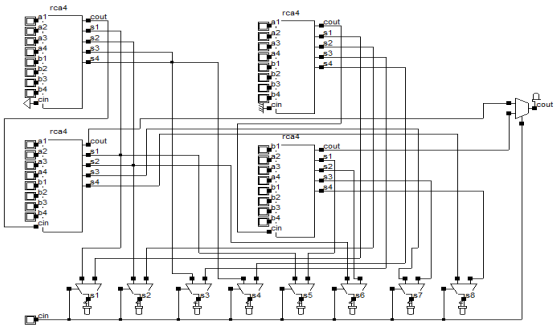


Figure 9 Schematic of Carry Select Adder

III. PERFORMANCE ANALYSIS

To evaluate performance; the adder structures discussed in this paper was designed using 0.12µm CMOS technology using Microwind. The microwind tool integrates traditionally separated front-end and back-end chip design into an integrated flow, accelerating the design cycle and reduced design complexities. It tightly integrates mixed-signal implementation with digital implementation, circuit simulation, transistor level extraction and verification. All simulations are carried out at nominal conditions: VDD=1.2 V, I/O supply voltage:2.5 V and room temperature= 27 ^{o}C. The device model used in this simulation is empirical level 3, monte-carlo (normal dist. 20%) with the following MOSFET model parameter:

*n-Mos Model
*low leakage
Model N1 NMOS level = 3 VTO =0.40 UO = 600.000 TOX = 2.0E-9
+LD = 0.000 THETA = 0.500 GAMMA = 0.400
+PHI =0.200 KAPPA = 0.060 VMAX = 120.00K
+CGSO = 100.0p CGDO =100.0
+CGBO = 60.0p CJSW = 240.0P
*p-Mos Model
*low leakage
Model P1 NMOS level = 3 VTO =0.45 UO = 200.000 TOX = 2.0E-9
+LD = 0.000 THETA = 0.300 GAMMA = 0.400
+PHI =0.200 KAPPA = 0.060 VMAX = 110.00K
+CGSO = 100.0p CGDO =100.0
+CGBO = 60.0p CJSW = 240.0P

Table 1 Area, Delay and Power Dissipation of Adders

Adder topology	Gate count			Power dissipation (mW)	Area µm^2	Delay (ns)
	nMOS	pMOS	Total			
RCA	144	144	288	0.206	2214	4.208
CSaA	288	288	576	1.082	5904	2.924
CLA	136	136	272	0.312	2160	3.1
CIA	171	171	342	0.261	2793	2.880
CSkA	194	194	388	0.603	3486	3.022
CByA	186	186	372	0.459	3116	3.01
CSelA	300	300	600	1.109	6201	2.75

Table 2 AT, AT^2 and PD values of Adders

Adder topology	AT	AT^2	PD
RCA	9316.512	39203.88	0.866848
CSaA	17263.3	50477.88	3.163768
CLA	6696	20757.6	0.9672
CIA	8043.84	23166.26	0.75168
CSkA	10534.69	31835.84	1.822266
CByA	9379.16	28231.27	1.38159
CSelA	17052.75	46895.06	3.04975

Table3 Energy Delay Parameters of Adders

Adder topology	IDD MAX(mA)	IDD AVG(mA)	Rise and Fall delay (ns)
RCA	3.32	0.369	0.043
CSaA	4.820	0.657	0.024
CLA	2.389	0.200	0.04
CIA	4.453	0.355	0.020
CSkA	4.712	0.482	0.031
CByA	6.262	0.941	0.024
CSelA	7.765	1.151	0.017

Table4 Parasitic Extraction of Adders

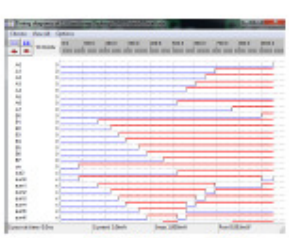
Adder topology	Capacitance (fF)	Resistance (OHM)	Inductance (nH)
RCA	144.07	10442	1.22
CSaA	204.84	11659	2.11
CLA	114.12	7235	1.09
CIA	189	10685	1.509
CSkA	270.26	15640	2.81
CByA	265	16488	2.76
CSelA	438.45	20538	3.333

Table 1 presents the performance analysis of different adder topologies. Table 2 presents the parameters of AT, AT^2 and PD values of adders. Table 3 and 4 presents the energy delay and parasitic extraction values. All the adders are simulated with multiple design corners (TT, FF, FS, and SS) to verify that operation across variations in device characteristics and environment. To establish an unbiased testing environment, the simulations have been carried out using a comprehensive input signal pattern, which covers every possible transition for all the adders. The frequencies have been chosen in the range from 10 to 500MHz and its input and output capacitance is set to 10pf.

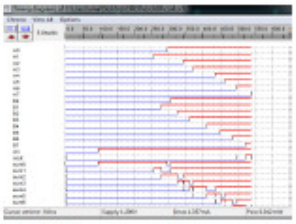
IV. SIMULATION RESULT

This section presents the simulated results of adder topologies. The above adder topologies are simulated using Microwind DSCH 3.1. Functional testing and timing analysis were carried out for the entire adder module used in this work. The MICROWIND software is dedicated to the training in sub micron CMOS VLSI design, consisting in a layout editor, electrical circuit extractor and a fast online analog simulator. The technology library used in this work is CMOS 6-metal layers 0.12µm technology, consequently lambda is 0.06µm (60nm). The microwind simulation provides two environments like logic editor and simulator. They are DSCH and MW used to validate logic design simulation with delay analysis and physical circuit extraction. All the adders used in this work are simulated using DSCH. First the simulation is performed using schematic entry and its corresponding test patterns are generated and it's functionally is verified. After verification the schematic file is converted to VERILOG file. Secondly using MW environment the VERILOG file is imported using the command "compile verilog file" so that the schematic of the logic design will be converted into physical layout. Using this physical layout

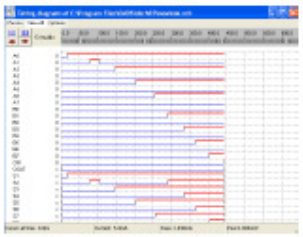
the parasitic values like resistance, capacitance, node voltage and current can be estimated. When the design is converted into physical layout the MW tool will automatically generate the spice netlist providing the information regarding the transistor model used, its temperature condition and transistor second order values. An extraction of spice netlist for full adder is shown in Figure 14. The simulation result of adder topologies is shown in Figure (3).



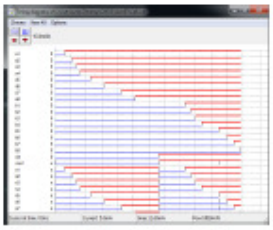
a. Ripple Carry Adder



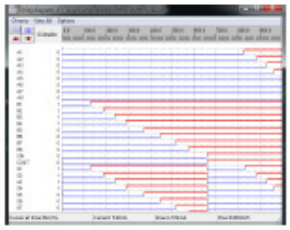
h. Carry Save Adder



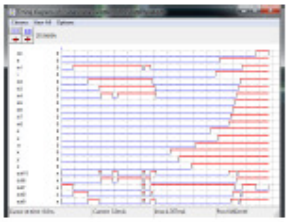
C. Carry Look-Ahead Adder



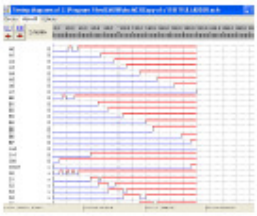
d. Carry Increment Adder



e. Carry Skip Adder



f. Carry Bypass Adder



g. Carry Select Adder

Figure 3 Simulation Result of Adder Topologies

V. SUMMARY

In this work, the performances of adder topologies are tested for robustness against area, delay and power dissipation. They are selected for this work since they have been commonly used in many applications. Addition is an indispensable operation for any high speed digital system, digital signal processing or control system**.** Therefore pertinent choice of adder topologies is an essential importance in the design of VLSI integrated circuits for high speed and high performance CMOS circuits. The operating frequency of adder topologies are set at 500MHz and its power dissipation and delay are observed. The graph in Figure (10a) shows the distribution of power dissipation values of different adder topology. Figure (10b, c, d) represents the area distribution, transistor count and delay distribution of adders.

From the power distribution graph it is observed that the maximum power dissipation occurs for carry select adder and next comes the carry save adder. The least power dissipation occurs for ripple carry adder and carry increment adders. From the area distribution and gate count the carry select and carry save adders occupies more area and gate count, ripple carry and carry increment occupies less area and gate count.

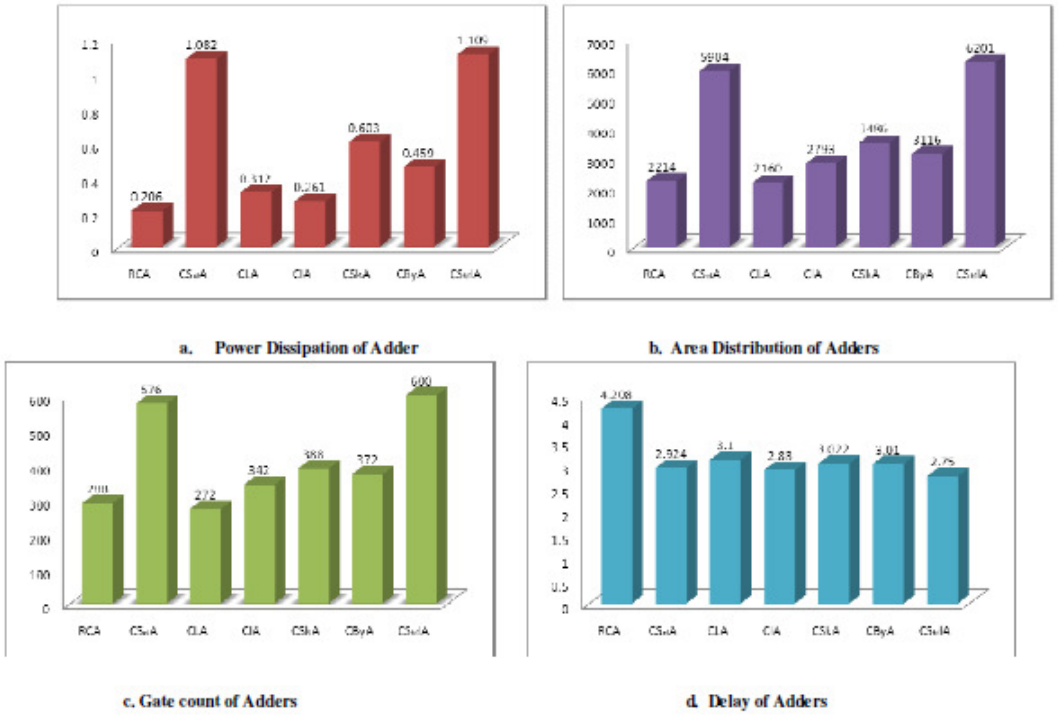


Figure 10 Comparison of adders in terms of area, delay and power dissipation

From the delay comparison it is observed that the maximum delay occurs for ripple carry adder. The minimum delay occurs for carry select, carry increment and carry save adders. The overall comparison presents the tradeoff between area, power dissipation and delay.

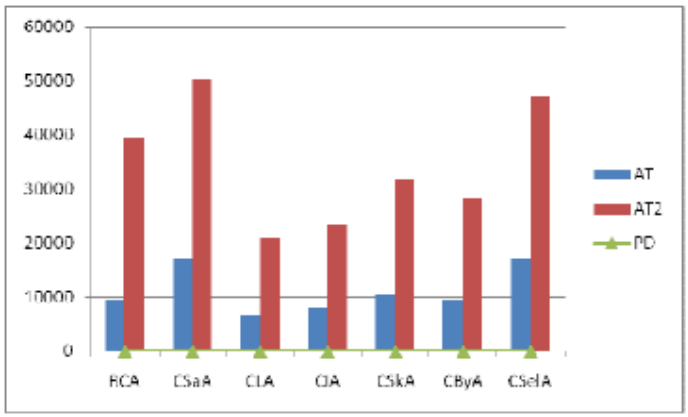


Figure 11 Comparison of adders in terms AT, AT^2 and PD

Figure (11) presents the comparison of adders in terms of AT, AT^2 and PD values. Carry look-ahead adders and carry increment adders have low AT, AT^2 and PD values. Figure (12) shows the automated layout generated using microwind MW03. All data for area, delay and power dissipation are obtained by microwind tool and simulations performed at the 0.12µm technology with power calculated using Predictive Technology Model (PTM). The granularity of transistor size is set to the minimum width of 1.02µm and the minimum length of 0.12µm for NMOS and

the minimum width of 1.98 and the minimum length 0.12for PMOS. The simulated result for the maximum and average drain current IDDMAX and IDDAVG is shown in Figure. 13.

VI CONCLUSION

In this work, an exhaustive analysis of adder topologies in 0.12μm CMOS technologies has been carried out. The comparison has been performed with area, delay and power dissipation. The impact of layout parasitics has been included in the transistor-level design phase. The Performance analysis, simulation result and comparison are reported in section III, IV and V. According to the presented results, the adder topology which has the best compromise between area, delay and power dissipation are carry look-ahead and carry increment adders and they are suitable for high performance and low-power circuits. The fastest adders are carry select and carry save adders with the penalty of area. The simplest adder topologies that are suitable for low power applications are ripple carry adder, carry skip and carry bypass adder with least gate count and maximum delay.

REFERENCES

[1] Animul islam, M.W. Akram, S.D. pable ,Mohd. Hasan, "Design and Analysis of Robust Dual Threshold CMOS Full Adder Circuit in 32 nm Technology", International Conference on Advances in Recent Technologies in Communication and Computing,2010.

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

[3] Nabihah Ahmad, Rezaul Hasan, "A new Design of XOR-XNOR gates for Low Power application", International Conference on Electronic Devices,Systems and Applications(ICEDSA) ,2011.

[4] R.Uma, "4-Bit Fast Adder Design: Topology and Layout with Self-Resetting Logic for Low Power VLSI Circuits", International Journal of Advanced Engineering Sciences and Technology, Vol No. 7, Issue No. 2, 197 – 205.

[5] David J. Willingham and izzet Kale, "A Ternary Adiabatic Logic (TAL) Implementation of a Four-Trit Full-Adder,IEEE, 2011.

[6] Padma Devi, Ashima Girdher and Balwinder Singh, "Improved Carry Select Adder with Reduced Area and Low Power Consumption", International Journal of Computer Application,Vol 3.No.4, June 2010 .

[7] B.Ramkumar, Harish M Kittur, P.Mahesh Kannan, "ASIC Implementation of Modified Faster Carry Save Adder", European Journal of Scientific Research ISSN 1450-216X Vol.42 No.1, pp.53-58,2010.

[8] Y. Sunil Gavaskar Reddy and V.V.G.S.Rajendra Prasad, "Power Comparison of CMOS and Adiabatic Full Adder Circuits", International Journal of VLSI design & Communication Systems (VLSICS) Vol.2, No.3, September 2011

[9] 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.

[10] Ning Zhu, Wang Ling Goh, Weija Zhang, Kiat Seng Yeo, and Zhi Hui Kong, "Design of Low-Power High-Speed Truncation-Error-Tolerant Adder and Its Application in Digital Signal Processing", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 18, No. 8, August 2010.

[11] Sreehari Veeramachaneni, M.B. Srinivas, "New Improved 1-Bit Full Adder Cells", IEEE, 2008.

[12] Tripti Sharma, k.G.Sharma, Prof.B.P.Singh, Neha Arora, "High Speed, Low Power 8T Full Adder Cell with 45% Improvement in Threshold Loss Problem", Recent Advances in Networking, VLSI and Signal Processing.

[13] G.Shyam Kishore, "A Novel Full Adder with High Speed Low Area", 2nd National Conference on Information and Communication Technology (NCICT) 2011 Proceedings published in International Journal of Computer Applications® (IJCA).

[14] Shubhajit Roy Chowdhury, Aritra Banerjee, Aniruddha Roy, Hiranmay Saha, "A high Speed 8 Transistor Full Adder Design using Novel 3 Transistor XOR Gates", International Journal of Electrical and Computer Engineering 3:12 2008.

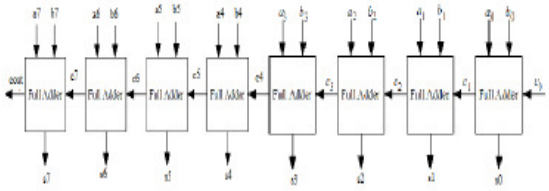
[15] Romana Yousuf and Najeeb-ud-din, "Synthesis of Carry Select Adder in 65 nm FPGA", IEEE.

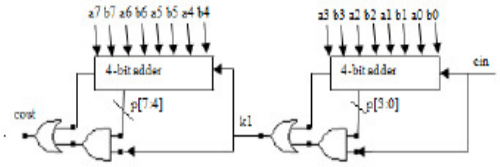
[16] Shubin.V.V, "Analysis and Comparison of Ripple Carry Full Adders by Speed", Micro/Nano Technologies and Electron Devices(EDM),2010, International Conference and Seminar on, pp.132-135,2010.

[17] Pudi. V, Sridhara., K, "Low Complexity Design of Ripple Carry and Brent Kung Addersin QCA",Nanotechnology,IEEE transactions on,Vol.11,Issue.1,pp.105-119,2012.

[18] Jian-Fei Jiang; Zhi-Gang Mao; Wei-Feng He; Qin Wang, "A New Full Adder Design for Tree Structured Aritmetic Circuits", Computer Engineering and Technology(ICCET),2010,2nd International Conference on,Vol.4,pp.V4-246-V4- 249,2010.

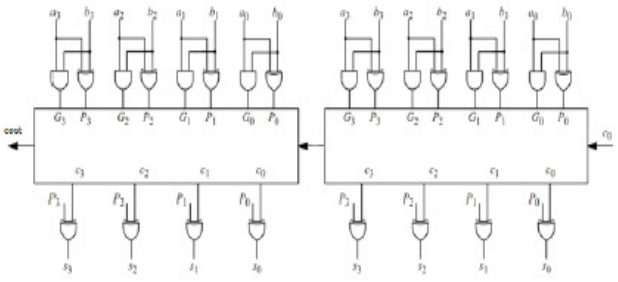
Existing adder Topology

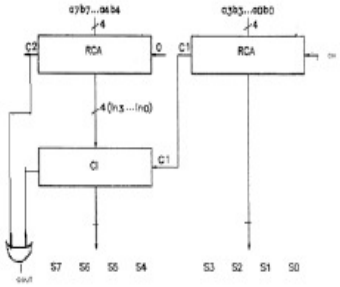




a. 8-bit Ripple Carry Adder (RCA)

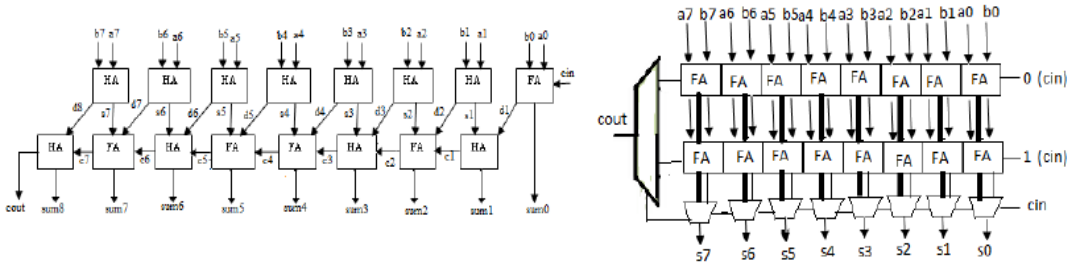
b. 8-bit Carry Skip Adder (CSkA)





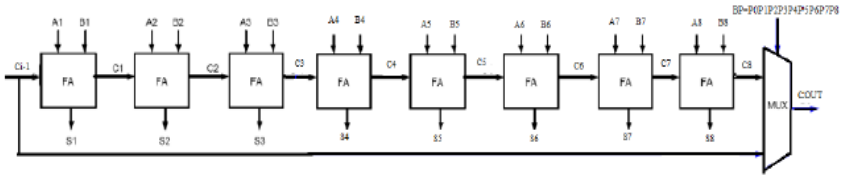
c. 8-bit Carry Look-Ahead Adder (CLA)

d. Carry Increment Adder (CIA)



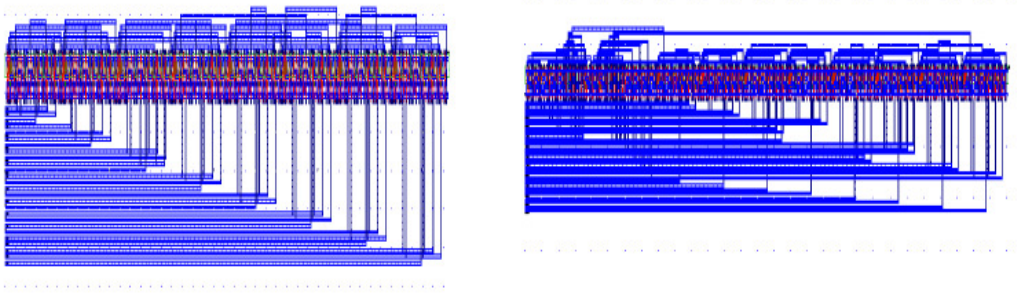
b. Carry Save Adder (CSaA)

f. Carry Select Adder (CSelA)



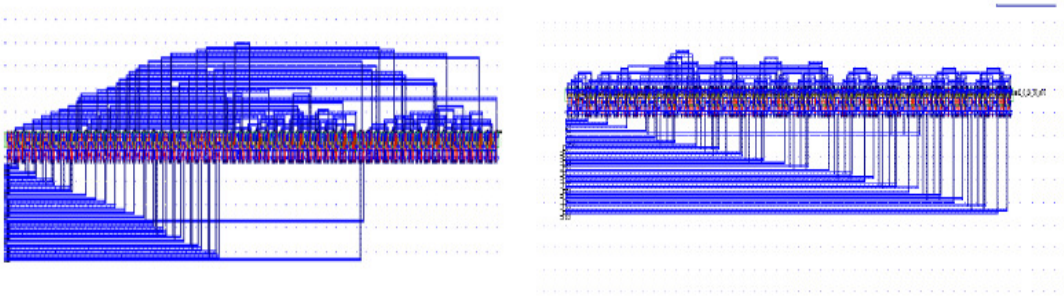
g.Carry Bypass Adder (CByA)

Figure 11 Existing Adder Topologies



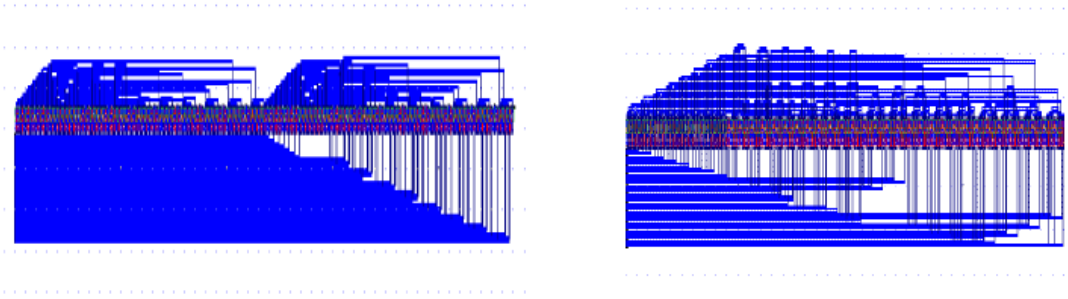
a. Layout of 8-bit Ripple Carry Adder (RCA)

b. Layout of 8-bit Carry Skip Adder (CSkA)



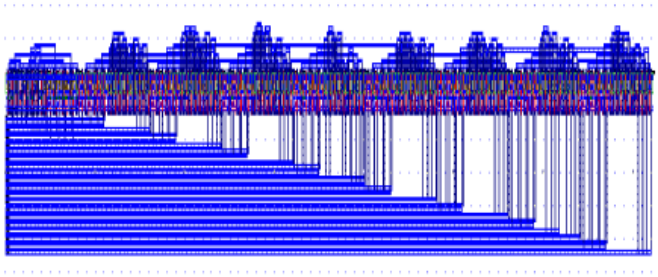
. c. Layout of 8-bit Carry Look-Ahead Adder (CLA)

d. Layout of 8-bit Carry Increment Adder (CIA)



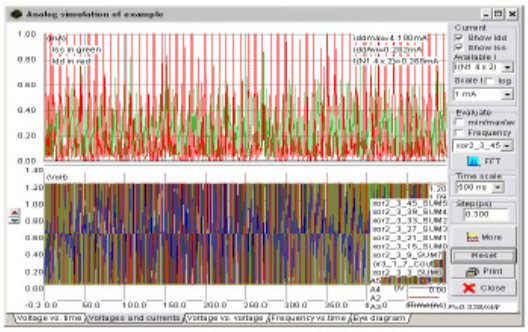
e. Layout of Carry Save Adder (CSaA)

f. Layout of Carry Select Adder (CSelA)

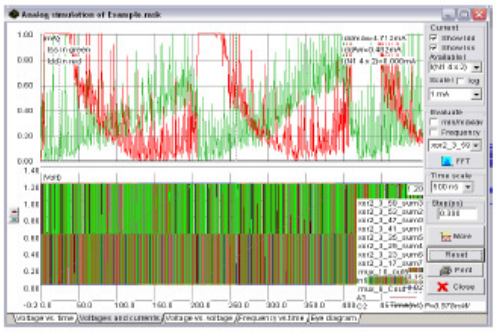


g. Layout of Carry Bypass Adder (CByA)

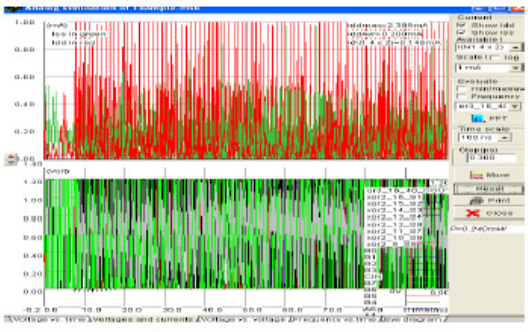
Figure 12 Layout of Existing Adder Topologies



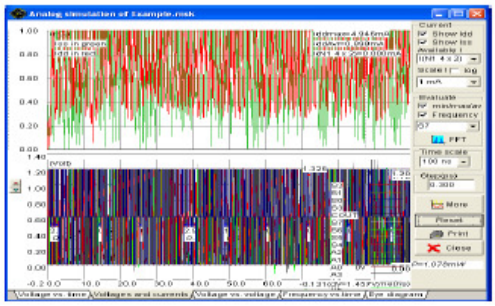
a. IDD and IDDMAX of Ripple Carry Adder (RCA)



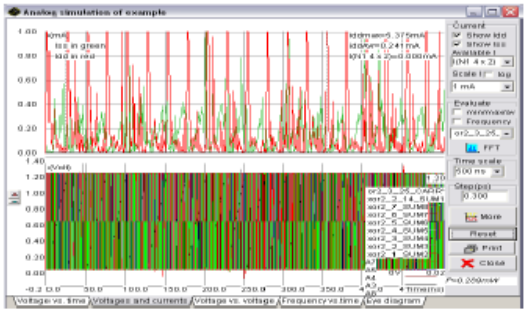
b. IDD and IDDMAX of 8-bit Carry Skip Adder (CSkA)



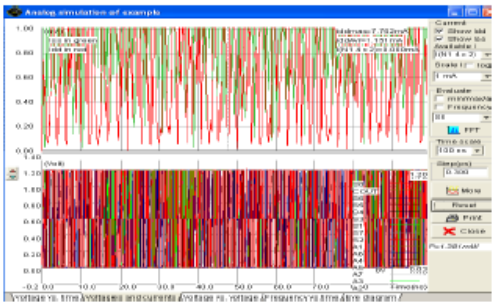
c. IDD and IDDMAX of Carry Look-Ahead Adder (CLA)



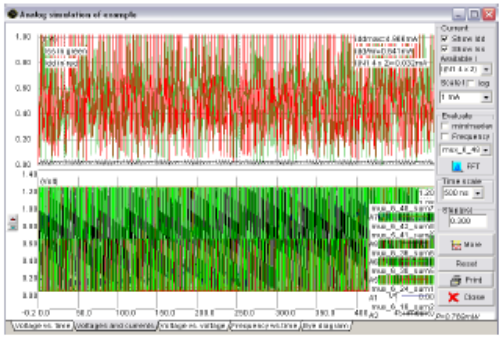
d. IDD and IDDMAX of 8-bit Carry Increment Adder



e IDD and IDDMAX of Carry Save Adder (CSaA)

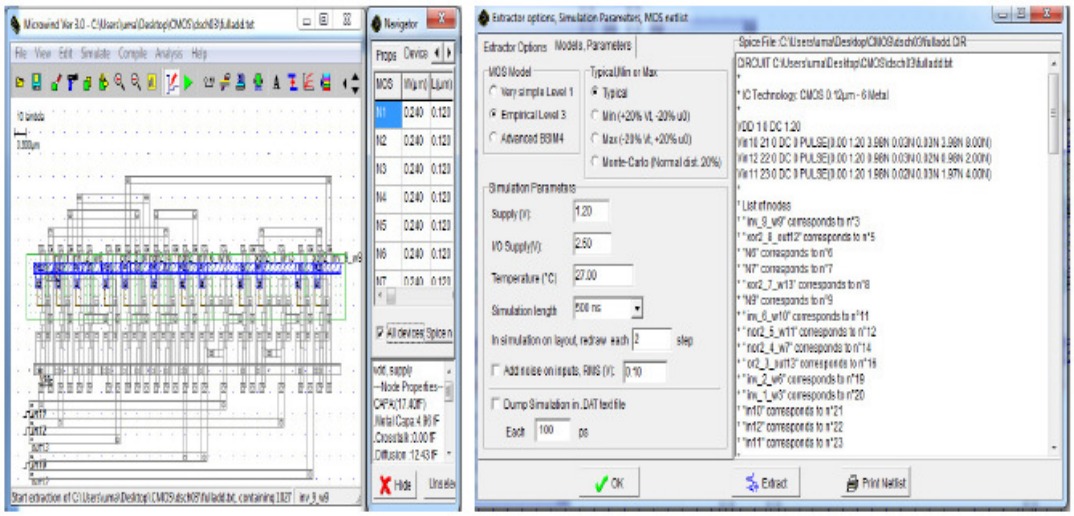


f. IDD and IDDMAX of Carry Select Adder (CSelA)



g. IDD and IDDMAX of Carry Bypass Adder (CByA)

Figure 13 IDD and IDDMAX of Existing Adder Topologies



MOS devices

MN1 0 8 3 0 N1 W= 0.24U L= 0.12U MN2 0 6 5 0 N1 W= 0.24U L= 0.12U MN3 8 22 6 0 N1 W= 0.24U L= 0.12U

MN4 0 8 7 0 N1 W= 0.24U L= 0.12U MN5 0 9 8 0 N1 W= 0.24U L= 0.12U MN6 21 23 9 0 N1 W= 0.24U L= 0.12U

MN7 0 21 10 0 N1 W= 0.24U L= 0.12U MN8 0 21 11 0 N1 W= 0.24U L= 0.12U MN9 12 20 0 0 N1 W= 0.24U L= 0.12U

```
MN10 0 11 12 0 N1 W= 0.24U L= 0.12U MN11 14 19 0 0 N1W= 0.24U L= 0.12U MN12 0 3 14
0 N1 W= 0.24U L= 0.12U
MN13 0 17 16 0 N1 W= 0.24U L= 0.12U MN14 17 24 0 0 N1 W= 0.24U L= 0.12U N15 0 14 17
0 N1 W= 0.24U L=
0.12U MN16 0 22 19 0 N1 W= 0.24U L= 0.12U MN17 0 23 20 0 N1 W= 0.24U L= 0.12U MP1
1 8 3 1 P1 W= 0.72U L=
0.12U MP2 1 6 5 1 P1 W= 0.72U L= 0.12U MP3 7 22 6 1 P1 W= 0.72U L= 0.12U MP4 1 8 7 1
P1 W= 0.72U L= 0.12U
MP5 1 9 8 1 P1 W= 0.72U L= 0.12U MP6 10 23 9 1 P1 W= 0.72U L= 0.12U MP7 1 21 10 1 P1
W= 0.72U L= 0.12U
MP8 1 21 11 1 P1 W= 0.72U L= 0.12U MP9 13 20 12 1 P1 W= 0.72U L= 0.12U MP10 1 11 13 1
P1 W= 0.72U L= 0.12U
MP11 15 19 14 1 P1 W= 0.72U L= 0.12U MP12 1 3 15 1 P1 W= 0.72U L= 0.12U MP13 1 17 16
1 P1 W= 0.72U L=
0.12U MP14 18 24 17 1 P1 W= 0.72U L= 0.12U MP15 1 14 18 1 P1 W= 0.72U L= 0.12U MP16
1 22 19 1 P1 W= 0.72U
L= 0.12U MP17 1 23 20 1 P1 W= 0.72U L= 0.12U *
C2 1 0 17.396fF C3 3 0 1.524fF C5 5 0 1.246fF C6 6 0 0.760fF C7 7 0 0.582fF C8 8 0 1.756fF
C9 9 0 0.760fF C10 10 0
0.582fF C11 11 0 1.128fF C12 12 0 0.631fF C13 13 0 0.186fF C14 14 0 1.282fF C15 15 0
0.186fF C16 16 0 0.836fF
C17 17 0 0.843fF C18 18 0 0.186fF C19 19 0 1.183fF C20 20 0 1.234fF C21 21 0 1.478fF C22
22 0 1.325fF C23 23 0
1.173fF C24 24 0 0.349fF
*
* n-MOS Model 3 :
* low leakage
.MODEL N1 NMOS LEVEL=3 VTO=0.40 UO=600.000 TOX= 2.0E-9
+LD =0.000U THETA=0.500 GAMMA=0.400
+PHI=0.200 KAPPA=0.060 VMAX=120.00K
+CGSO=100.0p CGDO=100.0p
+CGBO= 60.0p CJSW=240.0p
*
* p-MOS Model 3:
* low leakage
.MODEL P1 PMOS LEVEL=3 VTO=-0.45 UO=200.000 TOX= 2.0E-9
+LD =0.000U THETA=0.300 GAMMA=0.400
+PHI=0.200 KAPPA=0.060 VMAX=110.00K
+CGSO=100.0p CGDO=100.0p
+CGBO= 60.0p CJSW=240.0p
*
* Transient analysis
*
* (Winspice)
.options temp=27.0
.control
tran 0.1N 500.00N
print V(23) V(22) V(16) V(21) V(5) > out.txt
plot V(23) V(22) V(16) V(21) V(5)
.endc
.END
```

Figure 14 Spice Netlist of full adder circuit