Low-Power and High-Performance 1-Bit CMOS Full-Adder Cell

Keivan Navi and Omid Kavehei
Shahid Beheshti University, Tehran, Iran
Email: *{navi, kavehei}@sbu.ac.ir*

Mahnoush Ruholamini, Amir Sahafi, Shima Mehrabi and Nooshin Dadkhahi
Researches and Sciences Center of Hesarak, Tehran, Iran

***Abstract*—In this paper a new low power and high performance adder cell using a new design style called "Bridge" is proposed. The bridge design style enjoys a high degree of regularity, higher density than conventional CMOS design style as well as lower power consumption, by using some transistors, named bridge transistors. Simulation results illustrate the superiority of the resulting proposed adder against conventional CMOS 1-bit full-adder in terms of power, delay and PDP. We have performed simulations using HSPICE in a 90 nanometer (*nm*) standard CMOS technology at room temperature; with supply voltage variation from 0.65v to 1.5v with 0.05v steps.**

***Index Terms*—CMOS Circuit, VLSI, Full adder, Bridge style**

I. Introduction

Moore's law as we know will no longer exist in a near future, and one can already see the phenomenon of reduction of clock frequency due to excessive power dissipation. The reason is very simple: physical limits of silicon, since it is not possible to shrink atoms. Therefore, new technologies that will completely or partially replace silicon are arising.

According to the ITRS roadmap [1], these technologies have a high level of density and are slow, or the opposite: can achieve high speeds but with a huge area overhead even when comparing to future CMOS technology. Hence, traditional systems will suffer from the same problems that embedded systems suffer today: the necessity of increasing performance with severe area and power constraints. Additionally, traditional high performance architectures as the diffused superscalar machine are also achieving their limits, and recent increases in performance occurred mainly thanks to boosts in clock frequency.

As an example, the clock frequency of Intel's Pentium 4 processor only increased from 3.06 to 3.2 GHz between 2002 and 2003 [2]. This way, the frequency increase rate reduction, together with the foreseen slow technologies are new architectural challenges to be dealt with. In most VLSI applications, arithmetic operations play an important role. Commonly used operations are addition, subtraction, multiplication and accumulation, and the 1-bit Full Adder (FA) cell is the building block for most implementations of these operations.

Obviously, enhancing the building block performance is critical for enhancing overall system performance [3]-[6]. The vast use of this operation in arithmetic functions attracts many researchers to this field. In recent years several variants of different logic styles have been proposed to implement 1-bit adder cells [6]-[23]. They commonly aimed to reduce power consumption and increase speed.

With the increasing demand for battery-operated portable applications such as cell phones, PDAs and laptop computers, as well as low-intensity applications such as distributed sensor networks, the need for power sensitive design has grown significantly. It has been shown that reducing the supply voltage is the most direct means of reducing dissipated power [5], [6], and operating CMOS devices in the subthreshold region is considered to be the most energy-efficient solution for low-performance applications [5]. These papers have investigated different approaches realizing adders using CMOS technology; each has its own pros and cons. To summarize, some performance criteria are considered in the design and evaluation of adder cells.

After introducing a novel design methodology, we will present a high-speed CMOS 1-bit adder cell. The paper is organized as follows. Section II explores conventional CMOS design style. In section III a new style, called bridge, is described for CMOS logic circuits, and then based on the idea of bridge style a new 1-bit adder has been proposed. Section IV shows the simulation results in a 0.18-μm standard CMOS process technology, and finally some conclusions are given.

II. Power Consumption in digital CMOS

The average power dissipated in a generic digital CMOS gate is given by [4], [5]:

$$P_{avg} = P_{dynamic} + P_{short-circuit} + P_{static}$$
$$= V_{DD} \cdot f_{clk} \cdot \sum_i \left(V_{iswing} \cdot C_{iload} \cdot \alpha_i \right) \quad (1)$$
$$+ V_{DD} \cdot \sum_i I_{isc} + V_{DD} \cdot I_l$$

Where f_{clk} denotes the system clock frequency, $V_{i\ \mathrm{swing}}$ is the voltage swing at node i (ideally equal to V_{DD}), $C_{i\ \mathrm{load}}$ is the load capacitance at node i, α_i is the activity factor at node i, and $I_{i\ \mathrm{sc}}$ and I_l are the short circuit and leakage currents, respectively.

When operating CMOS devices in the subthreshold region, the power supply voltage is kept lower than the absolute of the devices' threshold voltage. This ensures that the transistor channel is never fully inverted, but is operated in weak or moderate inversion while the transistor is in its'on' state. According to [8], subthreshold logic gates have a near ideal voltage transfer characteristic, due to the exponential I-V relationship. We model the I-V relationship of the saturated device in weak inversion through the EKV model [9], [10]:

$$I_{sub} = I_S \cdot e^{\frac{V_G - V_{T0} - n \cdot V_S}{n \cdot U_T}} \qquad V_{DS} \rangle 4 \cdot U_T \quad (2)$$

Where I_S is the specific current defined by the model:

$$I_S = 2 \cdot n \cdot \eta_n(p) \cdot C_{ox} \cdot \frac{W}{L} \cdot U_T^2 \quad (3)$$

Note that all potentials are referred to the local substrate. U_T is the thermal voltage kT/q, n is the subthreshold swing parameter (slope factor), V_{T0} is the zero-biased threshold voltage, $\mu_n(p)$ is the carrier mobility for n or p channel devices, C_{ox} is the oxide capacitance, and W and L are the effective width and length of the channel.

III. Conventional CMOS Style

A basic cell in digital computing systems is the 1-bit full adder which has three 1-bit inputs (A, B, and C_{in}) and two 1-bit outputs (*sum* and *carry*). The relations between the inputs and the outputs are expressed as:

$$sum = \left(a \oplus b \right) \oplus c \quad (4)$$

$$carry = a \cdot b + c \cdot \left(a \oplus b \right) \quad (5)$$

The above Boolean expressions may be rearranged as:

$$sum = \bar{c} \left(a + b + c \right) + a \cdot b \cdot c \quad (6)$$

$$carry = a \cdot b + c \cdot \left(a + b \right) \quad (7)$$

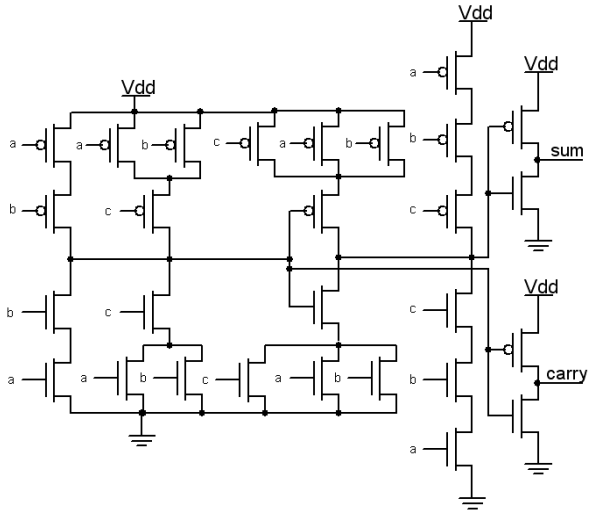

Figure 1. Conventional CMOS full adder.

The 1-bit conventional CMOS full adder cell is shown in Fig. 1. The 1-bit full adder cell has 28 transistors. Different logic styles can be investigated from different points of view. Evidently, they tend to favor one performance aspect at the expense of others. In other words, it is different design constraints imposed by the application that each logic style has its place in the cell library development. Even a selected style appropriate for a specific function may not be suitable for another one. For example, static approach presents robustness against noise effects, so automatically provides a reliable operation. The issue of ease of design is not always attained easily. The CMOS design style is not area-efficient for complex gates with large fan-ins. Thus, care must be taken when a static logic style is selected to realize a logic function.

Pseudo NMOS technique is straightforward, yet it compromises noise margin and suffers from static power dissipation. Pass transistor logic style is known to be a popular method for implementing some specific circuits such as multiplexers and XOR-based circuits, like adders. On the other hand, dynamic logic facilitates the realization of fast, small and complex gates. However, this advantage is gained at the expense of parasitic effects such as load sharing, which makes the design process hazardous.

Charge leakage necessitates frequent refreshing, reducing the operational frequency of the circuit. In general, none of the mentioned styles can compete with CMOS style in robustness and stability [3]-[5].

The CMOS structure combines PMOS pull-up and NMOS pull-down networks to produce considered outputs. In this style all transistors (either PMOS or NMOS) are arranged in completely separate branches, each may consist of several sub-branches. Mutually exclusiveness of pull-up and pull-down networks is of a great concern. Fig. 1 shows the conventional CMOS 28-transistor adder [6].

IV. Bridge Style CMOS Circuits

Bridge circuits are circuits that create a conditional conjunction between two circuit nodes. Using this kind of circuits the classical circuits can be implemented faster and smaller than the conventional. Since one of the important parameters in circuit design is the chip area, the proposed style might reduce area or increase density of transistors in unit of area. In this section a novel and efficient style of CMOS circuits would be introduced.

If a function has 2^{n-1} logical '0' and 2^{n-1} logical '1', the function could be implemented by a fully symmetric style using the proposed structure. For instance, we focus on Sum and Carry functions. The first one could be implemented by XOR3 (XOR circuit with three inputs) function and the second one with majority function. The equations of Sum (XOR3) and Carry (majority) are given above as (4) and (5). If these relations have been extended, they could change to the equ. (8) and (9).

$$sum = xor3 = F = a\cdot b\cdot \bar{c} + a\cdot \bar{b}\cdot c + \bar{a}\cdot b\cdot c + a\cdot b\cdot c \quad (8)$$

$$carry = majority = G = a\cdot \bar{b}\cdot \bar{c} + \bar{a}\cdot b\cdot \bar{c} + \bar{a}\cdot \bar{b}\cdot c + a\cdot b\cdot c \quad (9)$$

According to the CMOS circuit behavior the output of these circuits would be inverted. Based on this knowledge the output of *F* is NAND and the output of *G* is XNOR. If *a*=0 the relations (8) and (9) change to equ. (10) and (11).

$$F = b\cdot c \quad (10)$$

$$G = b\cdot \bar{c} + \bar{b}\cdot c \quad (11)$$

In the same manner if *a*=1 in the relations (8) and (9) those equations alter to the (12) and (13).

$$F = b\cdot \bar{c} + \bar{b}\cdot c + b\cdot c = b + c \quad (12)$$

$$G = \bar{b}\cdot \bar{c} + b\cdot c \quad (13)$$

Based on this knowledge the output of *F* is NOR and the output of *G* is XOR. These equations prove that we are able to implement some of the important arithmetic functions. The bridge style circuits can be categorized into two structures. One of them is fully-symmetric style and another one is semi-symmetric. This categorization is based on the similarities amount of P and N networks implementation. If the implementation of P and N is fully-similar then the style of circuit is fully-symmetric, and if those implementations are not similar then we could say it is semi-symmetric. For instance, implementation of the majority function is illustrated in Fig. 2. According to notified rules, implementation of majority circuits is fully-symmetric (Fig. 2).

As another example the structure of $F = \overline{\bar{A}\bar{B}\bar{C} + \bar{A}BC + A\bar{B}\bar{C} + ABC}$ is demonstrated in Fig. 3.

Based on Fig. 3, there is only one difference between P and N networks in the circuit that occurs in the order of *C* variable.

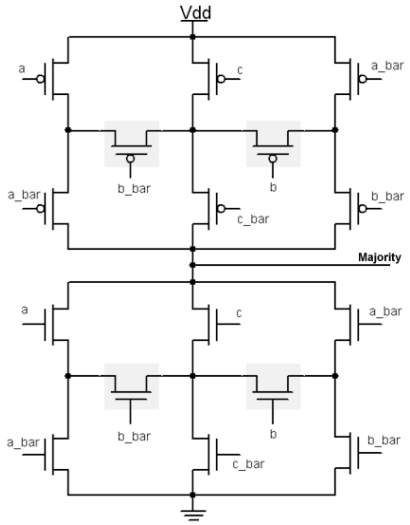

Figure 2. Bridge Implementation of Majority Function (Fully-Symmetric Style).

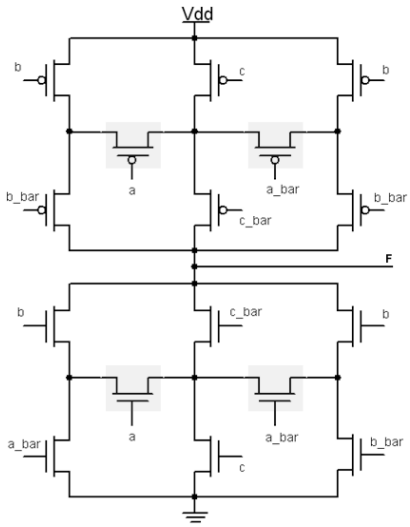

Figure 3. Bridge Implementation of $F = \overline{\bar{A}\bar{B}\bar{C} + \bar{A}BC + A\bar{B}\bar{C} + ABC}$ Function (Semi-Symmetric Style).

According to above information, we can say the proposed style could be able to implement CMOS circuits in a symmetric manner which is very useful for VLSI layout design, placement and routing. This improvement can be implement mentioned circuits in smaller area.

V. Proposed Full-Adder

In this section we introduce a new design style, entitled "bridge". As mentioned before, conventional CMOS design style performs realizations by organizing some different branches. Each branch provides a path from supply lines to an output, whereas bridge design style focuses its attention to meshes, and connects each two adjacent mesh by a transistor, named "bridge transistor". Bridge transistors provide the possibility of sharing transistors of different paths to create a new path from supply lines to an output.

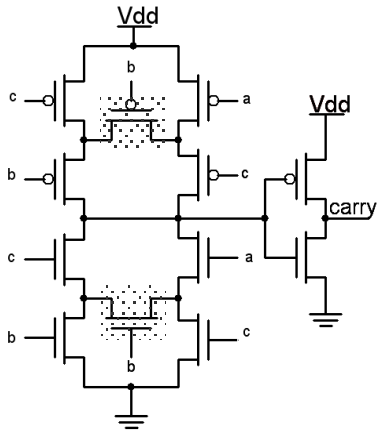

Figure 4. Carry generator (majority function) circuit.

These transistors must be arranged in such a way that not only validate the correctness of the circuit, but also preserve pull-up and pull-down networks mutually exclusive. In general, bridge style can lead to very structured designs. In this style control signals can be applied to the sides of meshes. Fig. 2 illustrates a bridge implementation of the majority function (carry generator).

The equation (7) Boolean expressions may be rearranged as:

$$carry = a \cdot b \cdot b + c \cdot a + c \cdot b \qquad (14)$$

The transistors in grey boxes (Fig. 4) are bridge transistors. This structure uses 12 transistors to implement the relevant function. With respect to high flexibility of bridge methodology, variants of logic circuits can be easily realized. Fig. 5 shows a three input XOR (sum) cell, which exploits 4 bridge transistors.

Inputs must be applied to the gates of bridge transistors such that the possibility of simultaneous activation of two bridge transistors is vanished. The design of Fig. 5 requires 16 transistors to implement the circuit (without considering the inverters which will be discussed later).

Since complementary inputs are fed to the circuit of Fig. 5, it is better to follow the same strategy to implement the majority function; instead of using an inverter at the output node. Thus, the inverter can be omitted from the last stage of the circuit. The equation (6) and (7) Boolean expressions for *sum* may be rearranged as:

$$sum = a \cdot b \cdot c + a \cdot \bar{b} \cdot \bar{c} + \bar{a} \cdot b \cdot \bar{c} + \bar{a} \cdot \bar{b} \cdot c \qquad (15)$$

$$carry = a \cdot b \cdot c + a \cdot b \cdot \bar{c} + \bar{a} \cdot b \cdot c + a \cdot \bar{b} \cdot c \qquad (16)$$

Fig. 6 shows the implementation using the above idea. As seen, this circuit uses two bridge transistors like previous design. A complete fast full adder can be built by placing the sum circuit of Fig. 5 and the carry generator of Fig. 6 together as a whole circuit (according to the simulation results in section IV). The circuit requires 16 transistors to generate the sum signal, 10 transistors to produce the carry signal and 6 for complementary inputs, resulting in a device count of 32. In consequence, it uses 4 transistors more than conventional CMOS adder.

The increase in transistor count, in addition to some other reasons such as using more inverters, causes more power consumption of the considered design as compared to conventional CMOS adder (with respect to simulation results).

However, the improvement in speed over conventional CMOS adder is eminent and drastically leads to a better power-delay product. The simulation results are presented in the next section. We have also considered some other adder cells either in voltage or current mode, but conventional CMOS adder is chosen for being analogous to our proposed adder.

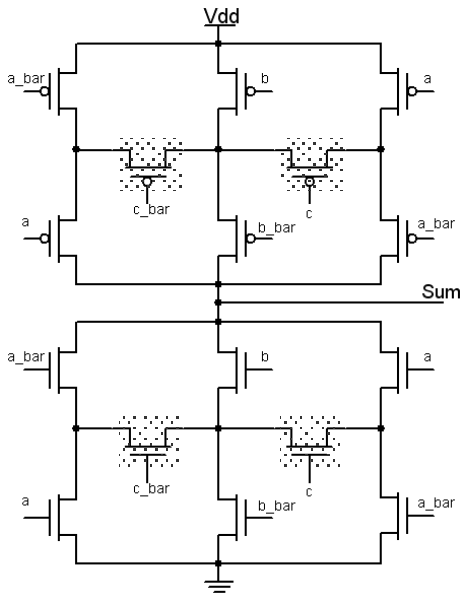

Figure 5. Three inputs XOR (sum function) circuit.

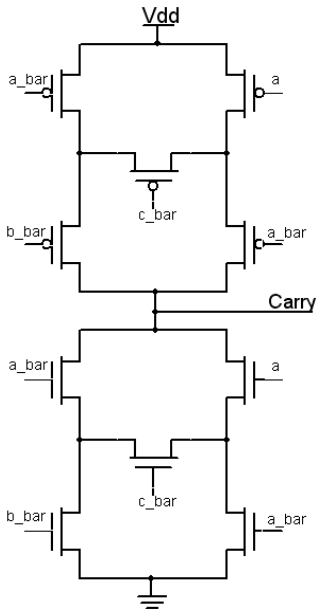

Figure 6. Carry generator circuits (Three Inputs Majority Function).

VI. Simulation Results

We have performed simulations using HSPICE in a 90 nanometer (*nm*) standard CMOS technology at room temperature; with supply voltage ranges from 0.65v to 1.5v with 0.05v steps. The simulation structure used is shown in Fig. 7 [21], [23]. Data analysis of the carry and sum signals has been performed by Cosmos scope software. Also, the average power consumption of the simulated circuits is calculated by the expression (1).

To establish an impartial testing environment each of 8 input patterns have been carried out (Fig. 8). It can be extracted from Fig. 8 that all simulated designs have properly responded to different input patterns. In the correctness evaluation process of circuit's functionality, Vdd is assumed 1.5 volt. The average power consumption curves for both simulated designs are presented in Fig. 9. The proposed adder uses 14% more transistors than the conventional CMOS full-adder.

As illustrated in Fig. 9, the suggested adder shows 13.8% (@ Vdd=0.65 volt) to 31.5% (@ Vdd=1.5 volt) degradation in terms of power consumption than conventional CMOS design. In contrast, the delay and average power consumption curves (Fig. 10) depict impressive improvement of 41.5% (@ Vdd=0.65 volt) to 0.37% (@ Vdd=1.5 volt) and 60% (@ Vdd=0.65 volt) to 10.2% (@ Vdd=1.5 volt) respectively over conventional CMOS counterpart; resulting in a better power-delay product (Fig. 11). Overall, the design presented remarkably improves speed (Table I).

VII. Conclusion

In this paper a novel design methodology, entitled bridge style, for CMOS full adder, is presented, and afterwards a new 1-bit adder is proposed based on the idea of bridge and compared to its conventional CMOS contender. Our new design style uses some transistors, called bridge transistors, sharing transistors of different paths to generate new paths from supply lines to circuit outputs. The suggested bridge adder shows better performance in delay as compared to conventional CMOS adder. According to HSPICE simulation in 90 *nm* CMOS process technology at room temperature, and under given conditions, an improvement of 41.5% (@ Vdd=0.65 volt) to 0.37% (@ Vdd=1.5 volt) in speed over conventional CMOS adder is achieved. In addition, the suggested adder shows 13.8% (@ Vdd=0.65 volt) to 31.5% (@ Vdd=1.5 volt) degradation in terms of power consumption than conventional CMOS design. In addition, the delay and average power consumption results depict impressive improvement of 60% (@ Vdd=0.65 volt) to 10.2% (@ Vdd=1.5 volt) over conventional CMOS counterpart in power-delay product.

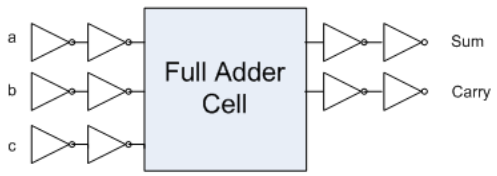

Figure 7. Test Bench.

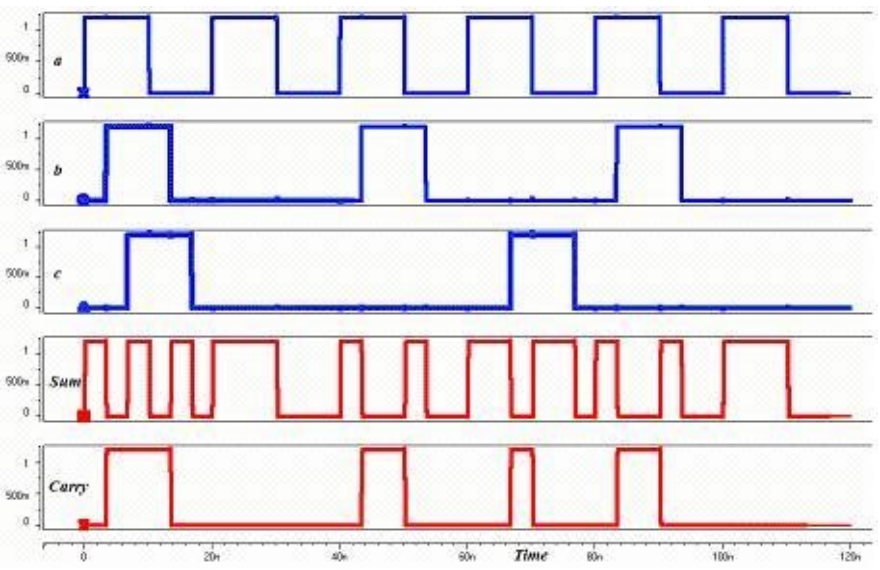

Figure 8. Simulation input patterns and output signals form.

Table I.
Improvement in Speed, Power Consumption, PDP Quantity.

	MAX	min
Speed	41.5% (@ Vdd=0.65 volt)	0.37% (@ Vdd=1.5 volt)
Power	13.8% (@ Vdd=0.65 volt)	31.5% (@ Vdd=1.5 volt)

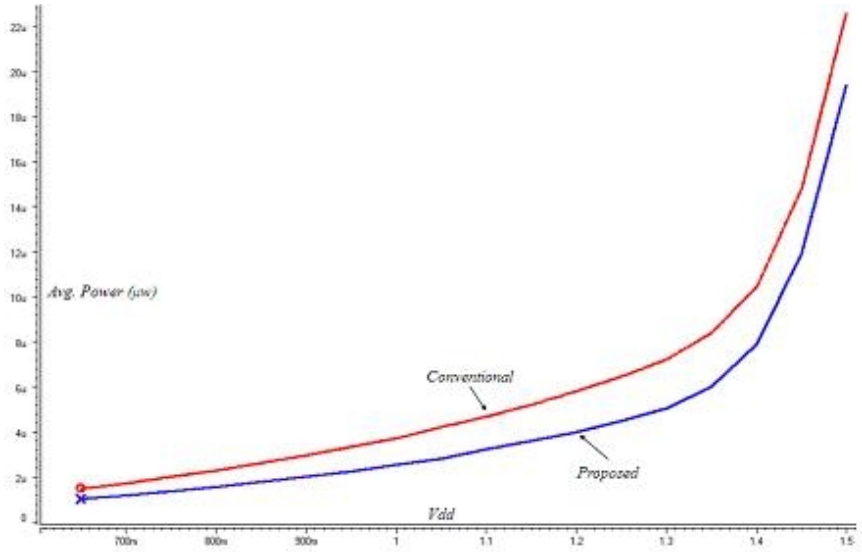

Figure 9. Average power consumption curves.

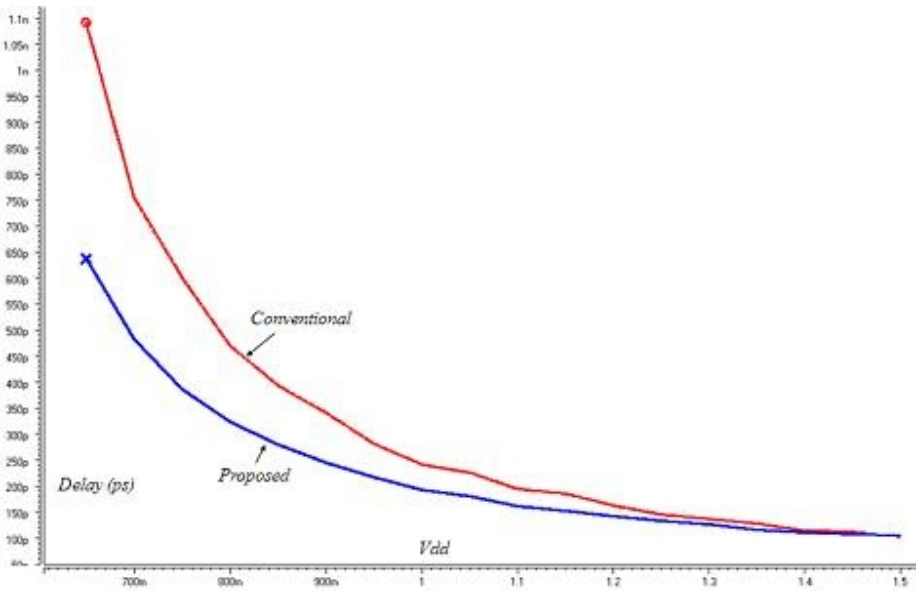

Figure 10. Delay curves.

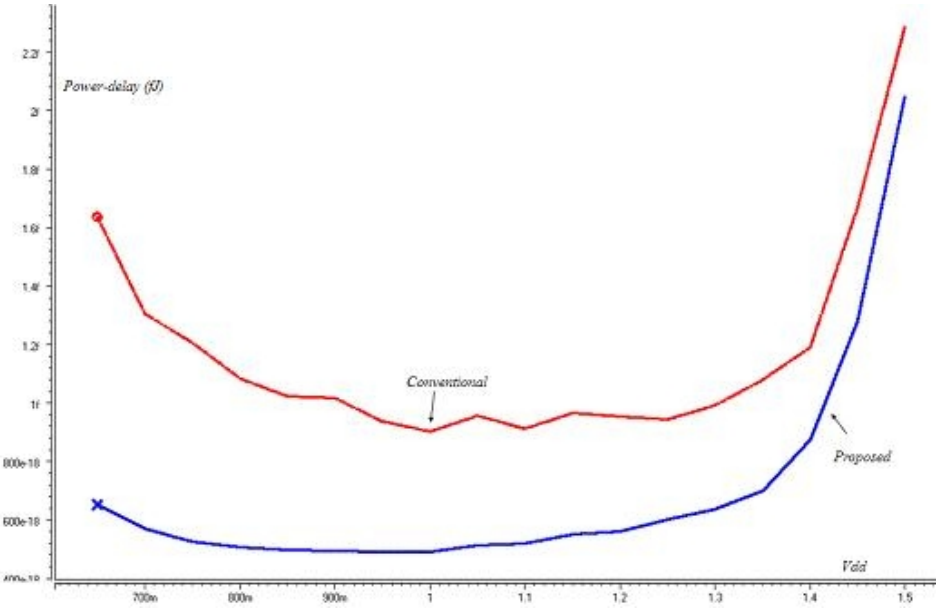

Figure 11. Power delay product (PDP) curves.

REFERENCES

[1] International Technology Roadmap for Semiconductors, 2004.

[2] Flynn, M.J.; Hung, P. "Microprocessor design issues: thoughts on the road ahead". *IEEE Micro*, Vol. 25, Issue 3, May-June 2005 pp. 16 - 31.

[3] Rabaey J. M., A. Chandrakasan, B. Nikolic, *Digital Integrated Circuits, A Design Perspective*, 2nd 2002, Prentice Hall, Englewood Cliffs, NJ.

[4] Uyemura J., *CMOS Logic Circuit Design,* Kluwer, 1999, ISBN 0-7923-8452-0.

[5] N. Weste, K. Eshragian, Principles of CMOS VLSI Design: A Systems Perspective, Addison-Wesley, 1993.

[6] R. Zimmermann, W. Fichtner, "Low-power logic styles: CMOS versus pass-transistor logic," *IEEE J. Solid-State Circuits*, Vol. 32, pp. 1079–1090, July 1997.

[7] H.T. Bui, A.K. Al. Sheraidah, Y. Wang., "Design and Analysis of 10-transistor Full Adders Using Novel XOR-XNOR Gates," *Technical Report*, Florida Atlantic University, October 1999.

[8] N. Zhuang, H. Ho, "A new design of the CMOS full adder," *IEEE. J. of Solid-State Circuits*, Vol. 27, No. 5, pp. 840- 844, May 1992.

[9] A. Chandrakasan, Low Power Digital CMOS Design, Ph.D. Thesis, University of California at Berkeley, Memorandum No. UCB/ERL M94/65, August 1994.

[10] H.T. Bui, Y. Wang, Y. Jiang, "Design and analysis of low-power10-transistor full adders using novel XOR–XNOR gates," *IEEE Transactions on Circuits & Systems II*, Vol. 49, pp. 25–30, Jan. 2002.

[11] Ko U., P. T. Balsara, W. Lee, "Low-Power Design Techniques for High Performance CMOS Adders," *IEEE Transactions on VLSI Systems*, Vol. 3, No. 2, pp. 327-333, June 1995.

[12] S. F. Hsiao, M. R. Jiang, J.S. Yeh, "Design of high-speed low-power 3-2 counter and 4-2 compressor for fast multipliers," *ElectroN. Letter*, Vol.34, No. 4, pp. 341–343, 1998.

[13] D. Radhakrishnan, A.P. Preethy, "Low-power CMOS pass logic 4-2 compressor for high-speed multiplication," *Proc. 43rd IEEE Midwest Symp. Circuits & Systems*, Vol. 3, pp. 1296–1298, 2000.

[14] A.M. Shams, M.A. Bayoumi, "A structured approach for designing low-power adders," *Proc. 31st Asilomar Conf. Signals, Systems Computers*, Vol. 1, pp. 757–761, 1997.

[15] A.M. Shams, M.A. Bayoumi, "A novel high-performance CMOS 1-bit full-adder cell," *IEEE Transactions on Circuits & Systems. II*, Vol. 47, pp. 478–481, May 2000.

[16] A.M. Shams, T.K. Darwish, M.A. Bayoumi, "Performance analysis of low-power 1-bit CMOS full adder cells," *IEEE Transactions on VLSI Systems*, Vol. 10, pp. 20–29, Jan. 2002.

[17] Abu-Shama, M.A. Bayoumi, "A new cell for low power adders," *Proc. lot. Midwest Symp. Circuits & Systems*, pp. 1014-1017, 1995.

[18] A.A. Fayed, M.A. Bayoumi, "A Low Power 10-Transistor Full Adder Cell for Embedded Architectures," *Proc. IEEE Symp. Circuits & Systems*, Vol. 4, pp. 226-229, Sydney, Australia, May 2001.

[19] D. Radhakrishnan, "Low-voltage low-power CMOS full adder," *Proc. Inst. Elect. Eng., Circuits Devices Systems*, Vol. 148, No. 1, pp. 19–24, 2001.

[20] O. Kavehie, K. Navi, "A Novel 54×54-bit Scalable Multiplier Architecture," *13th Iranian Conference on Electrical Engineering*, pp. 367-371, May 2005.

[21] C. Chang, J. Gu, M. Zhang, "Ultra Low-Voltage Low-Power CMOS 4-2 and 5-2 Compressors for Fast Arithmetic Circuits," *IEEE Transactions on Circuits & Systems*, Vol. 51, No. 10, pp. 1985-1997, Oct. 2004.

[22] K. Navi, A. Kazeminejad, D. Etiemble, "Performance of CMOS Current Mode Full Adders", *IEEE Proc. Int'l. Symp. Multiple Valued Logic*, pp. 27-34, May 1994.

[23] K. Navi, O. Kavehie, M. Rouholamini, A. Sahafi, S. Mehrabi, "A Novel CMOS Full Adder," *20th International Conference on VLSI Design held jointly with 6th International Conference on Embedded Systems (VLSID'07)*, pp. 303-307, Jan. 2007, Bangalore, India.