

Performance Analysis of Low-Power 1-Bit CMOS Full Adder Cells

Ahmed M. Shams, *Member, IEEE*, Tarek K. Darwish, *Student Member, IEEE*, and Magdy A. Bayoumi, *Fellow, IEEE*

Abstract—A performance analysis of 1-bit full-adder cell is presented. The adder cell is anatomized into smaller modules. The modules are studied and evaluated extensively. Several designs of each of them are developed, prototyped, simulated and analyzed. Twenty different 1-bit full-adder cells are constructed (most of them are novel circuits) by connecting combinations of different designs of these modules. Each of these cells exhibits different power consumption, speed, area, and driving capability figures. Two realistic circuit structures that include adder cells are used for simulation. A library of full-adder cells is developed and presented to the circuit designers to pick the full-adder cell that satisfies their specific applications.

Index Terms—Addition, arithmetic, full adder, low power, performance analysis.

I. INTRODUCTION

ADDITION is one of the fundamental arithmetic operations. It is used extensively in many VLSI systems such as application-specific DSP architectures and microprocessors. In addition to its main task, which is adding two binary numbers, it is the nucleus of many other useful operations such as subtraction, multiplication, division, address calculation, etc. In most of these systems the adder is part of the critical path that determines the overall performance of the system. That is why enhancing the performance of the 1-bit full-adder cell (the building block of the binary adder) is a significant goal.

Recently, building low-power VLSI systems has emerged as highly in demand because of the fast growing technologies in mobile communication and computation. The battery technology doesn't advance at the same rate as the microelectronics technology. There is a limited amount of power available for the mobile systems. So designers are faced with more constraints: high speed, high throughput, small silicon area, and at the same time, low-power consumption. So building low-power, high-performance adder cells is of great interest.

In this paper, a structured approach for analyzing the adder design is introduced. It is based on decomposing the full adder into smaller modules. Each of these modules is implemented, optimized, and tested separately. Several full-adder cells are composed by connecting these modules. The remainder of the paper is organized as follows. In Section II, some low power considerations, to be taken into account when designing a VLSI

system are reviewed. In Section III, the building modules of the full-adder cell are presented. In Section IV, different designs of each of these modules are implemented, analyzed, and compared. Then twenty different full-adder cells are developed in Section V, using different combinations of designs of each of the building modules. Finally, the full-adder cells simulation results are presented in Section VI. The adder cells are compared based on power consumption, speed, power delay product, area, and driving capability.

II. POWER CONSIDERATIONS

Designing systems aiming for low power is not a straightforward task, as it is involved in all the IC design stages beginning with the system behavioral description and ending with the fabrication and packaging processes. In some of these stages there are guidelines that are clear and there are steps to follow that reduce power consumption, such as decreasing the power-supply voltage. While in other stages there are no clear steps to follow, so statistical or probabilistic heuristic methods are used to estimate the power consumption of a given design [1], [2].

There are three major components of power dissipation in complementary metal-oxide-semiconductor (CMOS) circuits.

- 1) *Switching Power*: Power consumed by the circuit node capacitances during transistor switching.
- 2) *Short Circuit Power*: Power consumed because of the current flowing from power supply to ground during transistor switching.
- 3) *Static Power*: Due to leakage and static currents.

The first two components are referred to as dynamic power. Dynamic power constitutes the majority of the power dissipated in CMOS VLSI circuits. It is the power dissipated during charging or discharging the load capacitances of a given circuit. It depends on the input pattern that will either cause the transistors to switch (consume dynamic power) or not to switch (no dynamic power consumed) at every clock cycle. It is given by the following in [3]:

$$P_{\text{dynamic}} = \sum_i (C_{i \text{ load}} \cdot V_{i \text{ swing}} \cdot \alpha_i) \cdot f_{\text{clk}} \cdot V_{dd} + \sum_i (K_i \cdot \alpha_i) (V_{dd} - 2V_t)^3 \cdot f_{\text{clk}} \text{ and}$$

$$K_i = \frac{\beta_t}{12}$$

Manuscript received May 21, 2001. This work was supported by the United States Department of Energy (DoE) EETAPP program, DE-97ER12220.

T. Darwish and M. Bayoumi are with the Center of Advance Computer Studies, University of Louisiana, Lafayette LA 70504 USA.

A. A. Shams is with Intel Corporation, Hillsboro, OR 97124 USA.

Publisher Item Identifier S 1063-8210(02)00796-5.

where

$C_{i \text{ load}}$	load capacitance at node i ;
$V_{i \text{ swing}}$	voltage swing;
α_i	switching activity factor;
f_{clk}	system clock frequency;
V_{dd}	power supply voltage;
V_t	transistor threshold voltage;
β	gain factor of the transistor;
τ	rise or fall time of the signal.

The summation is over all the nodes of the circuit. Reducing any of these components will end up with lower-power consumption, although, it is of equal importance to increase the system-clock frequency for faster operation.

Estimating the power of a large circuit is a complex task. Heuristic algorithms, statistical, and probabilistic methods are used to generate random-input patterns to test the switching activity of the circuit. These methods become less accurate when the size of the circuit increases. It is better to decompose the large circuit into smaller modules and then use these methods to estimate the power consumption of each module. When the decomposed modules are small enough, exact methods can be used to optimize their performance. CAD tools and simulators could be used to build the circuit layout, simulate it, and estimate its power dissipation. Following this strategy, the best design of a given module is found and then by connecting the modules together the bigger circuit is formed, which will be optimized for low-power dissipation.

III. FULL ADDER BUILDING BLOCK

The full-adder function can be described as follows: Given the three 1-bit inputs A , B , and C_{in} , it is desired to calculate the two 1-bit outputs sum and C_{out} , where

$$\begin{aligned} sum &= (A \oplus B) \oplus C_{in} \\ C_{out} &= A \cdot B + C_{in} \cdot (A \oplus B). \end{aligned}$$

There are standard implementations for the full-adder cell that will be used as basis for comparison in this paper. Among these adders there are the following:

- 1) The transmission-gates CMOS adder (TG-CMOS) [4], it is based on transmission gates and has 20 transistors.
- 2) The transmission function full-adder (TFA) cell [5] is based on the transmission function theory and it has 16 transistors.
- 3) The low power implementation of the full-adder cell that has only 14 transistors (14T) [6]. It is based on the low power XOR design and transmission gates.
- 4) The complementary pass-transistor (CPL) full adder [7], [12], it has 32 transistors and uses the CPL logic family.
- 5) The CMOS full adder (CMOS) [7] has 28 transistors and is based on the regular CMOS structure (pull-up and pull-down networks).

These full adders are shown in Fig. 1. Although they all perform the same function, their styles of generating the intermediate nodes and the outputs are different, the loads on the inputs

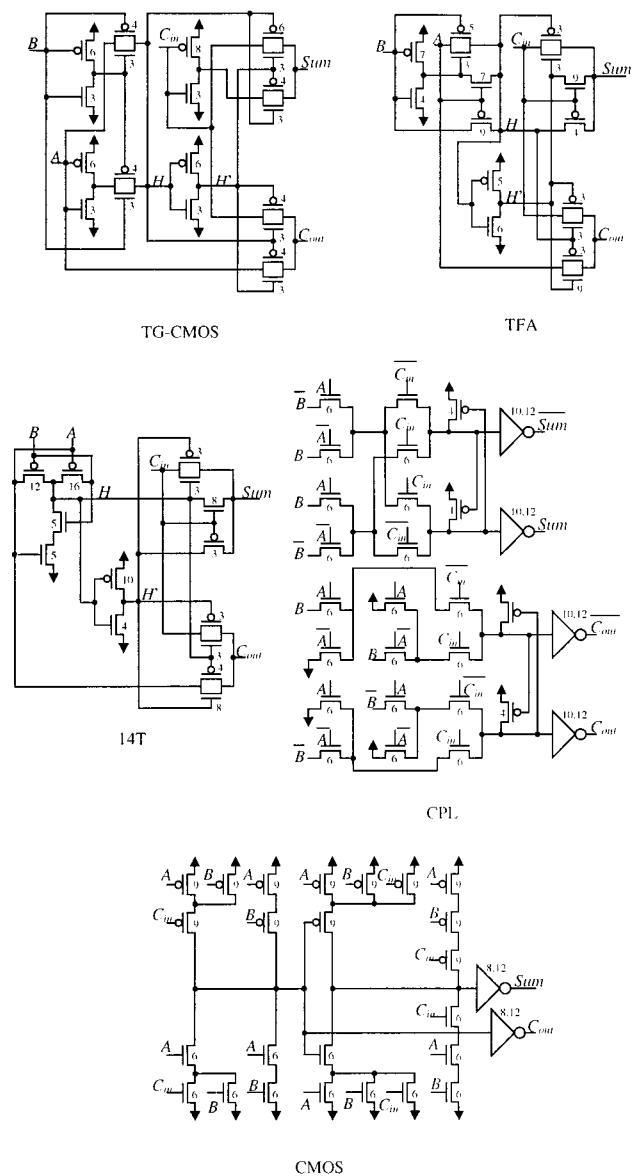


Fig. 1. Standard full adder cells. The transmission-gates CMOS adder (TG_CMOS) [4]. The transmission function adder (TFA) [5]. The 14-transistors adder (14T) [6]. The conventional CMOS adder (CMOS) [7]. The complementary CMOS logic adder (CPL) [7].

and intermediate nodes are different, and the transistor count varies significantly. For example, TG-CMOS, TFA, and 14T generate $A \oplus B$ and use it and its complement as a select signal to generate the outputs; while CMOS generates C_{out} through a single static CMOS gate and finally CPL generates many intermediate nodes and their complement in order to generate the final outputs. Having a signal and its complement produce a guaranteed switching activity that may occur with every change in any of the inputs. Another problem, which is overloading the inputs (especially with oversized transistor gates), produces high capacitance values for these nodes. This problem is clear with CPL and CMOS, and less with TG-CMOS, TFA, and 14T. Another problem that is unique in CMOS is that it generates the sum using the C_{out} signal as an input, which produces an unwanted additional delay. The other adder cells try to balance the generation of both signals.

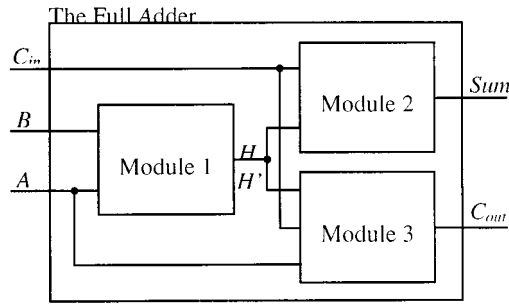


Fig. 2. The adder cell divided into three main modules.

It is clear, from an analytical perspective, that CPL is not a good candidate for low power due its high transistor count, its high switching activity of intermediate nodes, and overloading of its inputs. But, it is shown, through simulation, that CPL is better than CMOS for the studied circuit conditions [7]. So both circuits will be considered for comparison in this paper.

It is worth mentioning, that the full adders, TG-CMOS, TFA, and 14T have the advantages of lower-transistor count, lower loading of the inputs and intermediate nodes, and balanced generation of sum and C_{out} signals. These full adders have better performance than CMOS and CPL ones.

The full-adder cell equations can be written as

$$\begin{aligned} sum &= H \oplus C_{in} = H \cdot C'_{in} + H' \cdot C_{in} \\ C_{out} &= A \cdot H' + C_{in} \cdot H \end{aligned}$$

where H is the half sum ($A \oplus B$). It is clear that H and its complement H' are the key variables in both adder equations. If the generation of H and H' is optimized, this could greatly enhance the performance of the full-adder cell. A special module should be dedicated to the generation of these two signals. Another module is needed to generate the sum using H , H' and C_{in} . A third module is needed to generate C_{out} given H , H' , A , and C_{in} . Dividing the full adder in this way enables the analysis, enhancement, optimization, and testing of each module separately [8]. A block diagram of the full-adder cell and its building blocks is shown in Fig. 2.

IV. ANALYSIS OF THE FULL ADDER MODULES

A. First Module

1) *Design Options:* The first module is required to generate both the XOR and XNOR functions. One way of doing this is to generate the XOR function, then use an inverter to generate the XNOR function. Another option is to try to generate both of them simultaneously, but generally more transistors will be needed.

Five different designs of the first module are shown in Fig. 3, designs Fig. 3(a)–(c) use the first option, while the rest use the second option. A minimum of six transistors are used (the least known to the authors), while maximum of ten transistors is imposed because it is believed that designs with more than this figure will not be competitive for low power. Design Fig. 3(a) is composed of two-transmission gates and three inverters. It is the one used by TG-CMOS. Design Fig. 3(b) uses eight transistors and is based on the transmission function theory. This

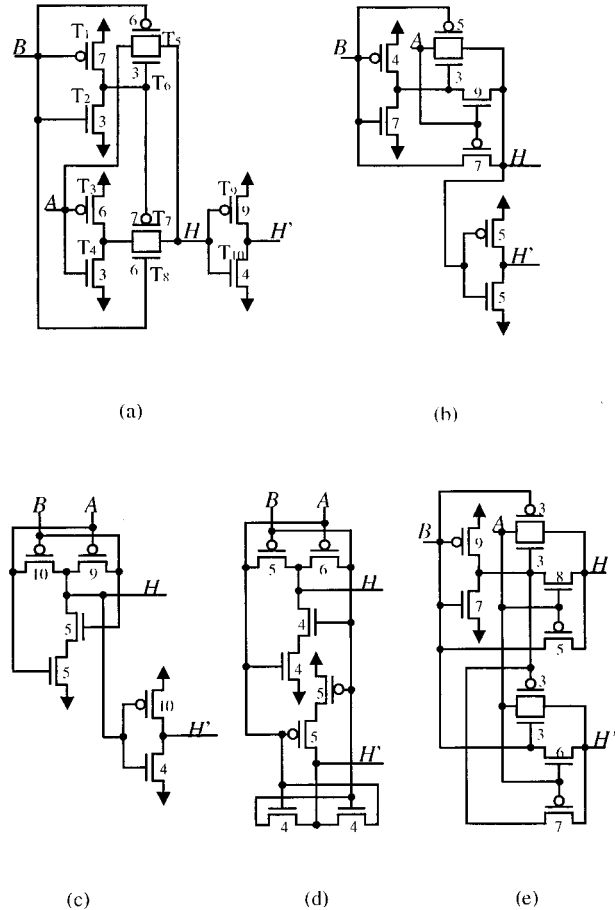


Fig. 3. Five different designs of the first module.

design is used in TFA. Design Fig. 3(c) is the one presented in [9], and used by [6] to build 14T. It uses only six transistors. Design Fig. 3(d) has eight transistors, it uses the same XOR design as in design Fig. 3(c), and the four transistors XNOR design [9] to generate H and H' simultaneously. Design Fig. 3(e) is similar to design Fig. 3(b) and has ten transistors.

The layouts of all these designs are prototyped in 0.35μ CMOS technology, and simulated using Hspice [10] with level 13 BSIM transistor models.

2) *Transistor Sizing:* Sizing of the transistors for this module is done in an iterative manner by the following steps.

- 1) Set all the transistors (N and P) to the minimum size.
- 2) Simulate the circuit with all possible input-pattern-to-input-pattern transitions (16 transitions).
- 3) Figure the transition with the highest delay (H or H'), and mark the transistors that are involved.
- 4) Size one of the transistors in this critical path.
- 5) Repeat Steps 2), 3), and 4) until the power-delay product for the cell continues to increase.
- 6) Record the transistor sizes corresponding to the minimum power-delay product.

This methodology guarantees that only the right transistors (the ones in the critical path) are sized, and in a proper way. No oversizing or undersizing will be incurred, which makes it optimal for power-delay product performance. Although, this is a lengthy process, it is guaranteed to give excellent transistor

TABLE I
SIMULATION RESULTS FOR FIVE DIFFERENT
DESIGNS OF THE FIRST MODULE

Fig. 3	Tr. #	Power (e^{-4} W)	Delay (e^{-10} sec)	Pwr*Dly
A	10	0.2883	1.1720	0.3320
B	8	0.1869	0.9753	0.1822
C	6	0.1664	0.9599	0.1597
D	8	0.1494	0.6879	0.1027
E	10	0.3622	0.7437	0.2693

sizing results, especially for small circuits. Following the same methodology with larger circuits will take much longer time.

Taking for example, the circuit of Fig. 3(a), Step 1) is to set all transistors to minimum sizes ($W = 3$). Step 2) is to simulate the circuit with all the 16-input transitions. In Step 3), the highest delay ($1.695e^{-10}$ s) is found to be associated with the transition $[(A, B): \text{from } (0,1) \text{ to } (0,0)]$ at the H' output. The power is $0.225e^{-4}$ Watts, thus giving an initial power-delay product of $0.3814e^{-14}$ Watts \cdot s. T_1 , T_5 , T_6 , and T_9 are the active transistors in this highest-delay transition. In step 4), transistor T_9 is chosen for sizing. In Step 5), a new iteration, starting from the simulation part of Step 2), is initiated. Now, Step 3), the highest delay is found to be $1.654e^{-10}$ s and it is associated with the transition $[(A, B): \text{from } (0,0) \text{ to } (0,1)]$ at the H' output also. The power and power-delay product are $0.229e^{-4}$ Watts and $0.3787e^{-14}$ Watts \cdot s, respectively. Transistors T_2 , T_7 , T_8 , and T_{10} are active in this new transition, and T_{10} is chosen for sizing in Step 4). This procedure is repeated until no more sizing can make any improvements. The results for this module, which are listed in Table I, are obtained after 34 iterations.

3) *Input Patterns and Output Loading*: The input signals A and B are designed to produce all the different transitions from an input pattern to another (for example: 00 to 00, 00 to 01, 00 to 10, and so on). A finite input pattern with 16-clock cycles that has all the transitions is developed. Having a finite input pattern is an important supporting factor in the above-described iterative methodology. The Hspice inputs A and B are defined as piecewise-linear signals with rise and fall times equal to 10% of the duty cycle of the fastest input. These inputs are applied through buffers (two cascaded inverters), which then loads the first module with more realistic inputs regarding slope and driving strength.

The outputs of the first module (nodes H , and H') will load the inputs of the second and third modules. This load is composed of gates and sources/drains of transmission gates. The average load is calculated from the actual designs used for the second and third modules. Both H and H' have to drive an average load of three transistor gates and one transistor source/drain, so the load is set to this average. An illustration of the circuit used to simulate the first module is shown in Fig. 4.

4) *Simulation Results*: The results of the simulation are shown in Table I. Regarding power consumption, design Fig. 3(d) is the best, although it does not have the least-transistor count. It has no internal direct path between the power supply and the ground rails, which eliminates direct short circuit

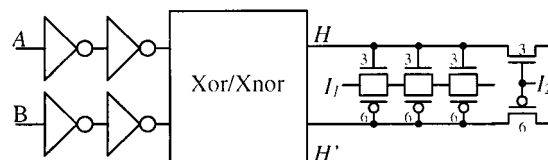


Fig. 4. Circuit structure used for simulating the first module.

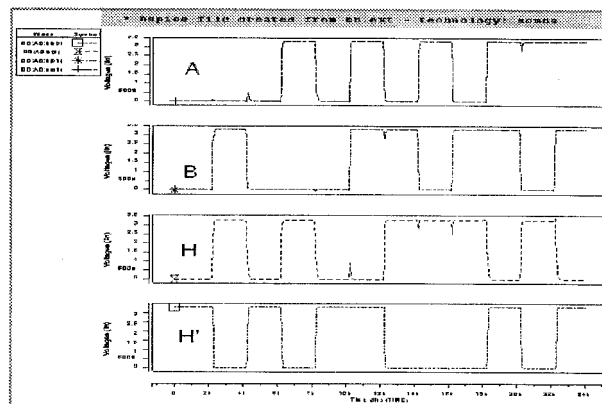


Fig. 5. Example waveforms for the first module.

current (sneak paths from previous driving stage output still exist, which is present in all other designs, as well). It has incomplete voltage swing at H when $(A = 0, B = 0)$ and incomplete voltage swing at H' when $(A = 1, B = 1)$ which account for less dynamic power consumption at those nodes. Also, it has less load capacitance at node H , since it is driving fewer loads than all other designs, which provides additional savings in dynamic power. The disadvantage is that it may not be suitable for VLSI circuits with low voltage supply, as the incomplete voltage swing is not desirable in such circuits [7]. Design Fig. 3(c) comes second, although it has the lowest transistor count. It has more capacitive load at the H node with high-switching activity and one inverter that introduces short circuit power component. These two reasons account for consuming a little more power than expected. It has an incomplete swing at the H node for the input pattern $(A = 0, B = 0)$. Design Fig. 3(b) comes next due to having low capacitive loads at its circuit nodes. Designs Fig. 3(a) and (e) have ten transistors each, which account for having more power than the other designs.

Considering delay, it is clear that the designs that generate H and H' simultaneously are superior. Eliminating the inverter from the critical path account for the speed gain. Finally regarding the power-delay product as expected, design Fig. 3(d) is the best. Some example waveforms for the inputs and outputs of the first module are shown in Fig. 5.

Designs with the same number of transistors exhibit different power consumption figures. Depending on the physical connections of each design, different capacitances are formed at each of the internal nodes leading to different dynamic power components. Also, if the design uses more number of inverters it probably ends with more power consumption; due to more short circuit power component and increased capacitances at their input gates.

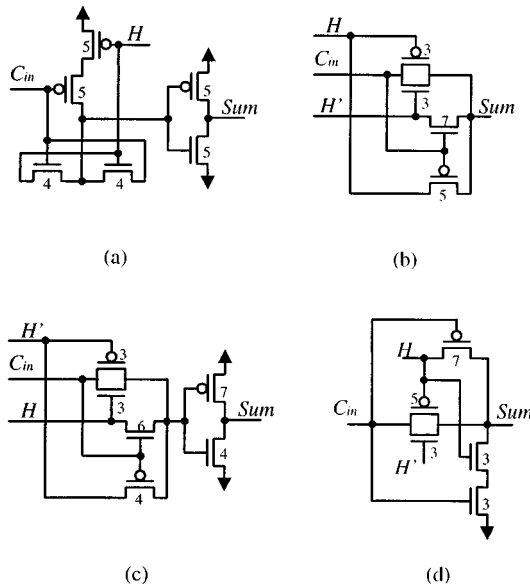


Fig. 6. Four different designs for the second module.

B. Second Module

This module is required to generate the sum given the inputs H , H' (generated by the first module) and C_{in} . It is an XOR function too, so most of the designs given here are the same ones used in the first module. An important requirement of this module is to provide enough driving power to the following gates. Four different designs of the second module are shown in Fig. 6. Design Fig. 6(a) uses the four-transistor XNOR design [9], followed by an inverter. It uses H and C_{in} only to generate the sum , it is the only design that does not need H' as an input (less H' capacitance). Design Fig. 6(b) is the transmission function implementation of the XOR gate, it does not need inverters since both H and H' are available. Design Fig. 6(c) generates sum' using the transmission function implementation of the XNOR function, then uses an inverter to generate the sum . This is primarily for providing more driving capability for the sum signal, but this leads to increasing the transistor count to six [same as design Fig. 6(a)]. Finally design Fig. 6(d) uses five transistors to generate the sum [11].

The inputs of this module are driven by the outputs of the first module (H and H'), which may not be clean signals for some cells. Therefore, for trying to achieve accurate simulation results, it was decided to use these actual outputs to drive the module's inputs. The results of the simulation are shown in Table II.

Design Fig. 6(b) has the lowest average power consumption, since it is the only design that has no ground- or power-supply rails (no short-circuit current), and has the lowest transistor count. Design Fig. 6(d) comes next, with a slight difference. Design Fig. 6(c) is ranked third and design Fig. 6(a) comes last as expected due to having an inverter and more transistor count, but they have the best sum output signal. For designs Fig. 6(b) and (d), the sum output signals are fairly good, but are not expected to drive bigger loads. They can be used efficiently in designs where the adder cell is followed by a buffer or a latch. Their delays are also less than the inverted output designs, due

TABLE II
SIMULATION RESULTS FOR FOUR DIFFERENT
DESIGNS OF THE SECOND MODULE

Fig. 6	Tr. #	Power (e^{-4} W)	Delay (e^{-10} sec)	Pwr*Dly
A	6	0.3319	0.5545	0.2256
B	4	0.1678	0.4604	0.0465
C	6	0.2353	0.5246	0.1497
D	5	0.1740	0.4709	0.0737

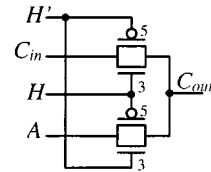


Fig. 7. The only adopted design for the third module.

TABLE III
SIMULATION RESULTS FOR THE THIRD MODULE

Fig. 7	Tr. #	Power (e^{-4} W)	Delay (e^{-10} sec)	Pwr*Dly
	4	.0170	0.443	0.0753

to having one-less stage (no inverter). They have outstanding power-delay product, which makes them perfect for low-power and high-performance designs.

C. Third Module

The third module is required to generate C_{out} , given H , H' , A (or B) and C_{in} as inputs. An important requirement, same as the second module, is to provide enough driving power for loading the following gates. All commonly used adders, as well as the new designs shown in [4] and [9] use the same approach to generate the C_{out} , which is a multiplexer passing either A , (or B), or C_{in} , according to the value of H . It seems that it is the only design known so far to generate C_{out} using only four transistors, given these inputs. Other designs will need the complement of C_{in} : i.e., two more transistors, which will end up with six transistors. Other designs require eight transistors. So it was decided to use only this design for the third module. It is shown in Fig. 7 and its simulation results are shown in Table III. The driving power of the C_{out} signal depends on the input signals A and C_{in} , since either of them will pass. Also, it depends on the transistor sizes of the transmission gates used. If A or C_{in} are outputs of a previous cascaded adder cell, these signals will decay and consequently the C_{out} signal will lack driving power. So, it is recommended to use this design in circuits where a latch or a buffer follows the adder cell's outputs.

V. BUILDING THE 1-BIT FULL ADDER CELLS

Twenty different adder cells can be built (most of them are novel circuits) using the various designs of each module. The following convention will be used for naming the adder cells. An adder cell will be referred to by two letters, the first letter denotes the first module design shown in Fig. 3, and the second letter denotes the second module design shown in Fig. 6. Two

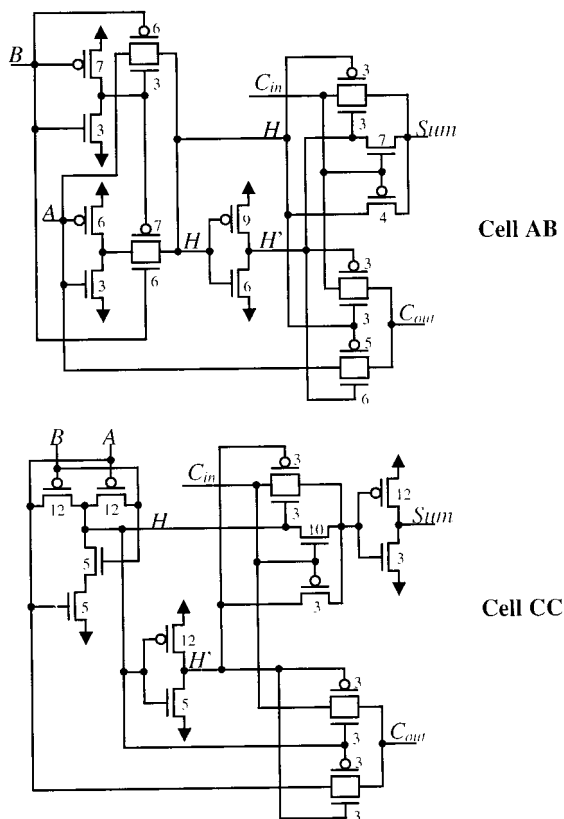


Fig. 8. Two examples of full adder cells.

of these cells are shown in Fig. 8, as an example. The first one is cell AB (uses designs Figs. 3(a), 6(b), and 7). While the second is cell CC. The selected adder cells have a range from 14–20 transistors. Cell CB is the only design with 14 transistors. It is in fact the fourteen-transistors adder cell shown in Fig. 1. Another combination of designs forms the transmission function full adder shown in Fig. 1: cell BB. Full-adder cells with a maximum of 20 transistors are also formed, such as cells AA, AC, EA, and EC. All the other adder cells have transistor counts ranging from 15–19 transistors. This is a good range for comparison between different designs of adder cells targeting low-power consumption, and low-power-delay product. A total of 23 adders, the 20 new adders, and the TG-CMOS, CMOS, and CPL adders, have been designed and prototyped. Each adder exhibits its own figures for power consumption, delay, area, and driving capability. The transistor count for adders used in this paper is much less than the ones used in [11], which range from 24–48 transistors and the ones used in [12], which range from 26–54 transistors.

VI. PERFORMANCE EVALUATION

A. Input Patterns

To compare these cells, input patterns that fairly test all the cases should be applied. An input pattern, which maximizes the power consumption for a given cell, could exhibit less power for another, while another input pattern could have the reverse situation due to different distribution of capacitances in both circuits. For example, Fig. 9 shows a portion of the SPICE files generated for two different adder cells: AA and BB.

Cell AA	Cell BB
C0 x5/a_12_2 GND 3.0fF	C0 x5/a_12_2 GND 3.0fF
C1 x4/a_12_7 GND 3.0fF	C1 x4/a_12_2 GND 3.0fF
C2 clk GND 11.1fF	C2 clkbar GND 11.1fF
C3 x3/a_12_2 GND 3.0fF	C3 x3/a_12_2 GND 3.0fF
C4 x1/Cout GND 3.6fF	C4 x1/Cin GND 5.4fF
C5 x1/a_108_39 GND 3.0fF	C5 x1/Hb GND 5.9fF
C6 x1/a_7_8 GND 3.2fF	C6 x1/a_8_13 GND 3.1fF
C7 x1/Hb GND 3.2fF	C7 x1/A GND 7.5fF
C8 x1/B GND 9.0fF	C8 x1/B GND 4.9fF
C9 x1/H GND 12.5fF	C9 x1/H GND 10.0fF
C10 x1/A GND 4.2fF	C10 x2/a_12_7 GND 3.0fF
C11 x1/Cin GND 6.6fF	C11 x1/Cout GND 4.1fF
C12 x2/a_12_2 GND 3.0fF	C12 Vdd GND 2.2fF
C13 clkbar GND 11.2fF	C13 x0/a_12_7 GND 3.0fF
C14 Vdd GND 2.8fF	C14 x1/S GND 3.1fF
C15 x0/a_12_7 GND 3.0fF	C15 clk GND 11.0fF
C16 x1/S GND 3.1fF	

Fig. 9. Portion of Hspice files for two-adder cells.

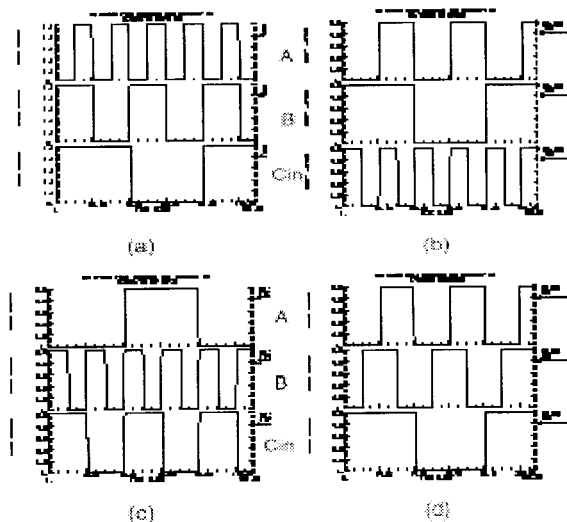


Fig. 10. Input patterns used to evaluate the performance of the 23-adder cells.

Cell AA has more capacitance at input B than the capacitance at input A , while cell BB has the reverse situation. An input pattern having higher frequency at input A than at input B will lead to an unfair comparison. While another input pattern having higher frequency at input B will be unfair too. In addition, an input pattern with the high frequency at both inputs will not cause much switching at the cell's intermediate nodes. A good input pattern for comparing power consumption of adder cells should alternate the high frequency at the input and intermediate nodes. A good example is the concatenation of the four patterns, as shown in Fig. 10.

Regarding speed, the input patterns should have all the required input-pattern-to-input-pattern transitions. In the case of three inputs (A , B , and C_{in}), a total of 64 different transitions exist. The delay of the cell should be measured for each of these 64 transitions. The input pattern used for the simulation process is a concatenation of the four-input patterns shown in Fig. 10, plus the 64 transitions. Again this will produce a finite input pattern, which will be beneficial for our iterative transistor sizing methodology.

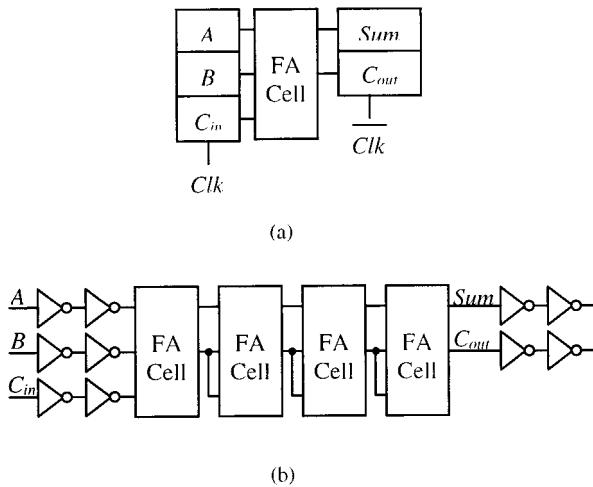


Fig. 11. Circuit structures used for simulation of FA cells.

The transistor-sizing methodology is similar to the one used in sizing the individual modules and is described in the following steps:

- 1) Set the transistor sizes to the ones obtained while sizing the module itself.
- 2) Simulate the circuit with the above-described input pattern.
- 3) Figure the transition with the highest delay (sum or C_{out}) and mark the transistors that are involved.
- 4) Size one of the transistors in this critical path.
- 5) Repeat steps 2, 3, and 4 until the power-delay product of the cell continues to increase.
- 6) Record the transistor sizes corresponding to the minimum power-delay product.

The effort spent in this process is reduced by sizing the individual modules first and by the proper selection of the loads used to test the individual modules. These two factors help to reduce the number of iterations considerably. The number of iterations to reach a satisfactory power-delay product varies from one adder cell to another. And the final transistor sizes, even for the same module, vary from one cell to another.

B. Simulation Circuit Structures

The choice of the circuit structures for simulating the adder cells is made based on the use of the adder cell in bigger structures. Examples of bigger structures are pipelined multipliers, regular multipliers, and binary adders. In pipelined multipliers, one pipeline stage consists of full-adder cells working in parallel followed by latches [13]–[15]. The full-adder cell is the nucleus of such applications and its performance determines the overall performance of the system. So the first structure for simulation is based on those applications and it is illustrated in Fig. 11(a). The inputs are fed to the adder cells from latches and the outputs are latched (C^2 MOS noninverting latches). Full-adder cells in such structures need not to have high-driving power, or even have full-swing outputs since the latches will act as a buffer between adjacent pipeline stages and will pull up or down any nonfull swing or weak signals. In regular multipliers and binary

adders that use full-adder cells as the building block, a cascade of full-adder cells is usually utilized. In such cases, the driving power of the adder cell is a must in order to provide the next cell with clean inputs. The second-circuit structure used to compare the adder cells is based on this concept and is illustrated in Fig. 11(b). A cascade of four-adder cells is utilized, the inputs are fed from buffers (two cascaded inverters) to give more realistic signals and the outputs are loaded with buffers to give proper loading conditions. Full-adder cells that perform well regarding the first structure, may not do so for the second due to the difference in the requirements of both.

C. First Circuit Structure Simulation Results

The simulation results for the 23 full-adder cells using the first-circuit structure are shown in Table IV. Results are sorted by low-power consumption. The values of power shown are for the adder cell only, i.e., excluding the power consumed by the latches. Also, the delay values are measured from the moment the clock reaches the adder inputs till the latest of the sum and C_{out} values reaches the output latches. These results show, that the designs that exhibit the lowest-power consumption for the first and second modules [Fig. 3(d), and Fig. 6(b)] when combined together give the cell with the lowest-power consumption. But cells following in the ranking are not sorted by performance of individual module designs presented in Tables I–III. Designs Fig. 3(b)–(d), and Fig. 6(b) and (d) are good candidates for low-power full-adder cells. Results show that some adder cells outperform existing standard implementations; two new cells outperform 14T (CB), three new cells outperform TFA (BB), and seven new cells outperform TG-CMOS cell. While CMOS and CPL show at the end of the table. The best cell (DB) consumes 14% less power than CB, 15% less power than BB, and 25% less than TG-CMOS. Cell DB and its simulation waveforms are shown in Fig. 12. Cells using design Fig. 6(c) have an average power consumption performance, while providing a clean-sum-output signal.

It is shown that the ranking is not necessarily related to the transistor count. But this happens only to a certain extent; adder cells with higher-transistor count occupy the bottom of the table. The authors believe that the distribution and magnitude of the capacitances found in the circuit are a good measure of the power consumption. But for larger designs it is hard to use this measure. The transistor count and their activity factors provide a good heuristic measure in this case. It should be pointed out that these results are for a 3.3 Volts power supply. When the power-supply voltage is reduced, other factors may play a role in changing the ranking of the adder cells. For example, having incomplete voltage swing at some internal nodes may lead to a constant current drain, which in turn increases the power consumption of the cell than usual [11].

The same results are sorted by speed and presented in Table V. The cell with the lowest-delay value is cell ED. Designs Fig. 3(d), (e) and Fig. 6(b) and (d) are good candidates for high-speed adder cells. Six cells outperform 14T, eight cells outperform TFA, and nine cells outperform TG-CMOS. Cell ED is 13% faster than 14T, 17% faster than TFA, and 26% faster than TG-CMOS. It is worth to note that generating H

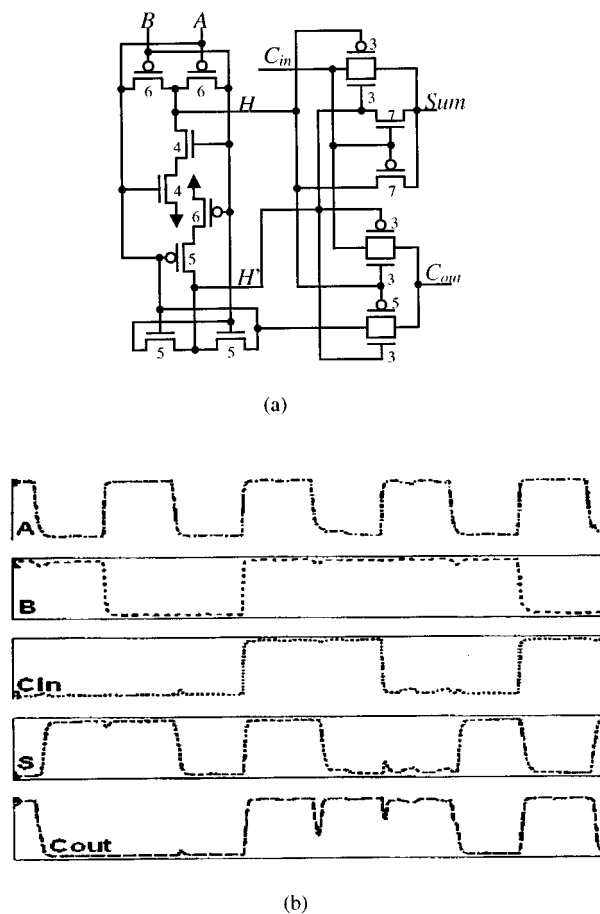


Fig. 12. Cell DB transistor circuit and waveforms.

and H' simultaneously in the first module tends to give better performance results; the best five cells regarding speed use this option.

Considering the power-delay product, which is a compromise between speed and power consumption; two cells outperform 14T, six cells outperform TFA, and nine cells outperform TG-CMOS, Tables IV and V.

Driving power of the cells was not effectively tested by this circuit structure and this is the main reason that a second one is introduced. Downsizing of transistors regarding the first-circuit structure is a recommended choice for targeting low-power adder cells, since eventually the latches will take care of enhancing the signal strength and swing.

D. Second Circuit Structure Simulation Results

For applications using a cascade of full-adder cells, driving power of the cell is a must. One or more of the following ways can enhance driving power:

- 1) Extra sizing of cell's transistors.
- 2) Inserting buffers after each cell, or after every other cell to enhance weak signals.
- 3) Using adder cells with buffered outputs (sum and C_{out} are output of inverters).

In order for the first option to provide acceptable driving power, major transistor sizing is needed for the adder cells presented in this paper, which are based on transmission gates, pass

TABLE IV
SIMULATION RESULTS FOR THE 23 FA CELLS
SORTED BY POWER CONSUMPTION

Cell	Tr. #	Power (e^{-4} W)	Delay (e^{-10} sec)	Pwr#Dly
DB	16	0.569	1.499	0.853
DD	17	0.590	1.581	0.933
CB	14	0.661	1.661	1.098
BD	17	0.665	1.712	1.138
BB	16	0.670	1.752	1.174
CD	15	0.688	1.650	1.135
BC	18	0.731	2.172	1.588
EB	18	0.743	1.488	1.106
AB	18	0.749	1.763	1.320
TG-CMOS	20	0.759	1.964	1.490
AD	19	0.762	1.689	1.287
ED	19	0.787	1.448	1.140
EC	20	0.802	1.595	1.280
CC	16	0.809	2.380	1.925
AC	20	0.840	2.016	1.693
DA	18	0.840	3.374	2.834
BA	18	0.865	2.152	1.861
DC	18	0.924	3.349	3.094
AA	20	0.962	2.257	2.171
EA	20	1.022	2.289	2.340
CA	16	1.047	2.896	3.032
CMOS	28	1.641	2.273	3.730
CPL	32	2.797	4.474	12.514

TABLE V
SIMULATION RESULTS FOR THE 23 FA CELLS SORTED BY DELAY

Cell	Tr. #	Power (e^{-4} W)	Delay (e^{-10} sec)	Product
ED	19	0.787	1.448	1.140
EB	18	0.743	1.488	1.106
DB	16	0.569	1.499	0.853
DD	17	0.590	1.581	0.933
EC	20	0.802	1.595	1.280
CD	15	0.688	1.650	1.135
CB	14	0.661	1.661	1.098
AD	19	0.762	1.689	1.287
BD	17	0.665	1.712	1.138
BB	16	0.670	1.752	1.174
AB	18	0.749	1.763	1.320
TG-CMOS	20	0.759	1.964	1.490
AC	20	0.840	2.016	1.693
BA	18	0.865	2.152	1.861
BC	18	0.731	2.172	1.588
AA	20	0.962	2.257	2.171
CMOS	28	1.641	2.273	3.730
EA	20	1.022	2.289	2.340
CC	16	0.809	2.380	1.925
CA	16	1.047	2.896	3.032
DC	18	0.924	3.349	3.094
DA	18	0.840	3.374	2.834
CPL	32	2.797	4.474	12.514

transistors, and the four-transistors implementation of XOR and XNOR. It is more efficient, from power consumption point of view, to increase the transistor count using the second or third option than to have huge transistors.

The second option is used to simulate selected adder cells from the 23-cell library. Buffers are inserted wherever the signal is weak. Each of the designs of the second and third module need separate investigation regarding its signal strength, which is fed to the next cell. After examining the output signals of each of these designs, the following strategy is used for inserting buffers after the *sum* signal:

- 1) Adder cells using design Fig. 6(a) and (c) and do not need any change.
- 2) Adder cells using designs Fig. 6(b) and (d) need a buffer after every other cell to enhance the *sum* signal. This is equivalent to increasing the cell's transistor count by two.

While for the C_{out} signal provided by the mux shown in Fig. 7, a buffer is needed after every other cell. This is equivalent to increasing each cell's transistor count by two. Table VI shows the effective increase in the transistor count for each cell using this method.

The following cells are selected for simulation:

- 1) Cells 14T, TFA, COMS, CPL, and TG-CMOS as standard reference cells.
- 2) Cells DB and DD for expected low-power performance.
- 3) Cell ED and EB for expected speed performance.
- 4) Cells EC and BC for expected high-driving power.

Simulation results of the selected cells are shown in Table VII, which are sorted by power consumption. The power consumption value is for the four cascaded adder cells, in addition to the intermediate buffers. While the delay is measured from the moment the inputs are applied to the first cell, until the latest of the *sum* and C_{out} signals of the fourth cell is produced.

As expected, cell DB is still the best regarding power consumption, while cell DD is still good as well; 14T is also superior. Cells using design Fig. 6(b), or Fig. 6(d) provide good candidates for low-power applications. They also produce good signals and have high speed. TG-CMOS, CMOS, and CPL have the worst power performance regarding this circuit structure.

The same results are sorted by speed and shown in Table VIII. Cell EB, that was expected to offer high-speed performance, failed to do so. Cell EC is the best, this is because there are no added buffer delays in the *sum* signal critical path, as most other cells have. This shows the effectiveness of using buffered outputs cells in cascaded structures, since they provide clean outputs.

This leads us to the third option discussed earlier, which is using adder cells with *sum* and C_{out} signals produced from inverters. The only design used to generate C_{out} in Fig. 7 is not buffered. If C'_{out} is generated, followed by an inverter to get C_{out} , A , A' , and C'_{in} signals will be needed. This technique will add four-six transistors to the cells discussed in this paper, which is greater than the option of adding intermediate buffers to C_{out} . So the authors believe that adding intermediate buffers is the best solution for cascading adder cells discussed in this paper.

TABLE VI
EFFECTIVE INCREASE IN THE TRANSISTOR COUNT OF
FA CELLS DUE TO BUFFER INSERTION

Cells	Effective increase in tr#
AA, BA, CA, DA, EA AC, BC, CC, DC, EC	2
AB, BB, CB, DB, EB AD, BD, CD, DD, ED	4

TABLE VII
SIMULATION RESULTS FOR THE SELECTED FA CELLS SORTED BY
POWER CONSUMPTION. (THE TRANSISTOR COUNT INCLUDES
THE USED BUFFERS WITH EACH CELL)

Cell	Tr. #	Power (e^{-4} W)	Delay (e^{-10} sec)	Pwr*Dly
DB	20	4.296	8.759	37.62
CB	18	4.341	9.175	39.82
DD	21	4.364	11.09	48.39
BB	20	4.625	9.743	45.06
BC	20	4.634	9.654	44.73
ED	23	4.983	9.606	47.86
EB	22	5.014	10.88	54.55
EC	22	5.159	7.116	36.71
TG-CMOS	24	5.307	9.966	52.889
CMOS	28	10.23	7.817	79.96
CPL	32	14.12	9.979	140.9

TABLE VIII
SIMULATION RESULTS FOR THE SELECTED FA CELLS SORTED BY SPEED.
(THE TRANSISTOR COUNT INCLUDES THE USED BUFFERS WITH EACH CELL)

Cell	Tr. #	Power (e^{-4} W)	Delay (e^{-10} sec)	Pwr*Dly
EC	22	5.159	7.116	36.71
CMOS	28	10.23	7.817	79.96
DB	20	4.296	8.759	37.62
CB	18	4.341	9.175	39.82
ED	23	4.983	9.606	47.86
BC	20	4.634	9.654	44.73
BB	20	4.625	9.743	45.06
TG-CMOS	24	5.307	9.966	52.889
CPL	32	14.12	9.979	140.9
EB	22	5.014	10.88	54.55
DD	21	4.364	11.09	48.39

VII. CONCLUSION

An extensive performance analysis of 1-bit full-adder cells has been presented. The adder cell has been divided into three constituting modules. Different designs for each of these modules have been implemented, simulated, analyzed, and compared. Twenty full-adder cells (most of them are novel circuits) are formed from combinations of these modules. Each adder cell exhibits its own figures of power consumption, delay, area, and driving power. Adder cells are implemented and simulated using two different circuit structures in which they

are commonly used. Performance of adder cells regarding the first-circuit structure is different from their performance in the second due to different requirements of both circuits. Adders are ranked, based on simulation results, according to power consumption, delay, and power delay product for each of the circuit structures. Some novel adder cells outperformed existing standard designs in each of these performance parameters. A library of adder cells is presented to the designers to pick the adder cells that satisfy their system design requirements. An analysis is presented of how to increase the driving power of adder cells and the most suitable method for adders presented in this paper; which is intermediate buffer insertion employed during the simulation of the second circuit structure.

From the previous analysis of adder cells, it is concluded that there is no perfect adder cell that can be used by all types of applications. Design constraints enforced by each application provide different requirements needed from the adder cell. Based on these requirements designers can choose an adder cell that satisfies their needs. The work presented in this paper gives more insight and deeper understanding of constituting modules of the adder cell to help the designers in making their choices.

ACKNOWLEDGMENT

We thank the anonymous reviewers for their remarkable notes that added more depth to this work.

REFERENCES

- [1] G. M. Blair, "Designing low-power CMOS," *Inst. Elect. Eng. Electron. Commun. Eng. J.*, vol. 6, pp. 229–236, Oct. 1994.
- [2] S. Devadas and S. Malik, "A survey of optimization techniques targeting low-power VLSI circuits," in *Proc. 32nd ACM/IEEE Design Automation Conf.*, San Francisco, CA, June 1995, pp. 242–247.
- [3] H. J. M. Veendrick, "Short-Circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits," *IEEE J. Solid-State Circuits*, vol. SSC-19, pp. 468–73, Aug. 1984.
- [4] N. Weste and K. Eshraghian, *Principles of CMOSVLSI Design, A System Perspective*. Reading, MA: Addison-Wesley, 1993.
- [5] N. Zhuang and H. Wu, "A new design of the CMOS full adder," *IEEE J. of Solid-State Circuits*, vol. 27, pp. 840–844, May 1992.
- [6] E. Abu-Shama and M. Bayoumi, "A new cell for low-power adders," in *Proc. Int. Midwest Symp. Circuits Syst.*, 1995.
- [7] R. Zimmermann and W. Fichtner, "Low-power logic styles: CMOS versus pass-transistor logic," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1079–90, July 1997.
- [8] A. Shams and M. Bayoumi, "A structured approach for designing low-power adders," in *ASILOMAR Conf. Signals, Syst., and Comput.*, Pacific Grove, CA, June 1997.
- [9] J.-M. Wang, S.-C. Fang, and W.-S. Feng, "New efficient designs for XOR and XNOR functions on the transistor level," *IEEE J. Solid-State Circuits*, vol. 29, pp. 780–786, July 1994.
- [10] *HSPICE User's Manual Version H9002*, Meta-Software Inc, Campbell, CA, 1992.
- [11] H. Lee and G.E. Sobelman, "A new low-voltage adder circuit," in *Proc. 7th Great Lakes Symp. VLSI*, Urbana, IL, 1997.
- [12] S. Issam, A. Khater, A. Bellaouar, and M. I. Elmasry, "Circuit techniques for CMOS low-power high performance multipliers," *IEEE J. Solid-State Circuits*, vol. 31, pp. 1535–1544, Oct. 1996.
- [13] F. Lu and H. Samulei, "A 200-MHz CMOS pipelined multiplier-accumulator using a quasidomino dynamic full-adder design," *IEEE J. Solid-State Circuits*, vol. 28, pp. 123–132, Feb. 1993.
- [14] S.-J. Jou, C.-Y. Chen, En-Chung, and C.-C. Su, "A pipelined multiplier-accumulator using a high-speed, low-power static and dynamic full adder design," *IEEE J. Solid-State Circuits*, vol. 32, pp. 114–118, Jan. 1997.
- [15] A. M. Shams, W. M. Badawy, and M. A. Bayoumi, "An enhanced low-power pipelined multiplier-accumulator for DSP applications," in *Proc. 1998 Int. Conf. Microelectronics*, Monastir, Tunisia, Dec. 1998.



Ahmed M. Shams (M'98) received the B.S. degree from the Department of Computer Science and Automatic Control, Alexandria University, Egypt, and the M.S. and Ph.D. degrees in Computer Engineering from the University of Louisiana at Lafayette, in 1991, 1997, and 2000, respectively.

Currently, he is with Intel Corporation, Hillsboro, OR, as a Senior Component Design Engineer. His research interests include VLSI circuit design, computer arithmetic, and architectures for digital video processing.



Tarek K. Darwish (S'00) received the B.S. and the M.S. degrees in computer engineering from the University of Balamand, Lebanon, and the M.S. degree, also in computer engineering, from the University of Louisiana at Lafayette, in 1996, 1998, and 2001, respectively. He is currently a Ph.D. candidate with the Center for Advanced Computer Studies (CACS) at the University of Louisiana.

Since 2000, he has been a Research Assistant with the CACS, in the VLSI Research group of M. Bayoumi, University of Louisiana. His research interests

include low power VLSI system design and CAD-tools.



Magdy Bayoumi (S'80–M'84–SM'87–F'99) received the B.Sc. and M.Sc. degrees in electrical engineering from Cairo University, Egypt. He received the M.Sc. degree in computer engineering from Washington University, St. Louis, MO, and the Ph.D. degree in electrical engineering from the University of Windsor, ON, Canada.

He is currently Director of the Center for Advanced Computer Studies (CACS) and Department Head of the Computer Science Department, University of Louisiana, Lafayette. He is also the Edmiston

Professor of Computer Engineering and Lamson Professor of Computer Science at the Center for Advanced Computer Studies, University of Louisiana at Lafayette, where he has been a faculty member since 1985. He is editor or coeditor of three books in the area of VLSI Signal Processing. His research interests include VLSI design methods and architectures, low-power circuits, and systems, digital signal processing architectures, parallel algorithm design, computer arithmetic, image and video signal processing neural networks, and wideband network architectures.

Dr. Bayoumi was Vice President for the technical activities of the IEEE Circuits and Systems Society. Currently, he is Chairman of the Technical Committee (TC) on Circuits and Systems for Communication and the TC on Signal Processing Design and Implementation. He was a founding member and Chairman of the VLSI Systems and Applications Technical Committee. He is also a member of the Neural Network and the Multimedia Technology Technical Committees. He has been on the technical program committee for ISCAS for several years, and he was the publication chair for ISCAS'99. He was the General Chairman of the 1994 MWSCAS and is a member of the Steering Committee of this symposium. He was an Associate Editor of the IEEE CIRCUITS AND DEVICES MAGAZINE, IEEE TRANSACTION ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, IEEE TRANSACTIONS ON NEURAL NETWORKS, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II. He was the cochairman of the Workshop on Computer Architecture for Machine Perception in 1993 and is a member of the Steering Committee of this workshop. He was general chairman for the 8th Great Lake Symposium on VLSI in 1998, and general chairman of the 2000 Workshop on Signal Processing Design and Implementation. He is an Associate Editor of the VLSI Journal *INTEGRATION*, and the *Journal of VLSI Signal Processing Systems*. He is a regional editor for the *VLSI Design Journal* and on the Advisory Board of the *Journal on Microelectronics Systems Integration*. He served on the Distinguished Visitors Program for the IEEE Computer Society from 1991 to 1994 and is currently on the Distinguished Lecture program of the Circuits and Systems Society. He won the UL Lafayette 1988 Researcher of the Year Award and the 1993 Distinguished Professor Award at UL Lafayette.