A Standard Cell Based Synchronous Dual-Bit Adder with Embedded Carry Look-Ahead

PADMANABHAN BALASUBRAMANIAN*, KRISHNAMACHAR PRASAD[¶] and NIKOS E. MASTORAKIS[§]

* School of Computer Science, The University of Manchester, Oxford Road, Manchester M13 9PL, UNITED KINGDOM.

padmanab@cs.man.ac.uk

[¶]Department of Electrical and Electronic Engineering, Auckland University of Technology, Private Bag 92006, Auckland 1142, NEW ZEALAND. <u>krishnamachar.prasad@aut.ac.nz</u> [§] Department of Computer Science, Military Institutions of University Education, Hellenic Naval Academy, Piraeus 18539, GREECE. mastor@hna.gr

Abstract: - A novel synchronous dual-bit adder design, realized using the elements of commercial standard cell libraries is presented in this article. The adder embeds two-bit carry look-ahead generator functionality and is realized using simple and compound gates of the standard cell library. The performance of the proposed dualbit adder design is evaluated and compared vis-à-vis the conventional full adder (implemented using two half adder blocks) and the library's full adder element, when performing 32-bit addition on the basis of the fundamental carry propagate adder topology. Based on experimentations targeting the best case process corner of the high-speed 130nm UMC CMOS cell library and the highest speed corner of the inherently power optimized 65nm STMicroelectronics CMOS standard cell library, it has been found that the proposed adder module is effective in achieving significant performance gains even in comparison with the commercial library based adder whilst facilitating reduced energy-delay product.

Key-Words: - Adder, High-speed, Low power, PDP, EDP, Standard cells, Semi-custom design style.

1 Introduction

Integer addition forms the basis of computer systems. Addition was found to be the most frequently encountered operation amongst a set of real-time digital signal processing benchmarks in [1]. About 72% of the instructions of a prototype RISC machine. DLX. resulted in addition/subtraction operations [2]. A study of the operations performed by an ARM processor's ALU revealed that additions constituted nearly 80% [3]. Arithmetic circuit realizations have also evoked interest in the optical, quantum computing and evolutionary programming regimes [4] - [7].

Addition is realized in hardware through the use of single-bit adder blocks (usually full adder modules). Hence, the design of an adder module is of considerable interest to any computer design architect as it is one of the most critical components found in a processor's data path that eventually determines its throughput. It is present in the ALU, the floating-point unit and also responsible for address generation in case of cache or memory accesses. Both the half adder and the full adder blocks are made available as gates/elements in any commercial standard cell library, optimized for speed/power/area. Many transistor level full adder designs have been put forward in the literature [8] – [17], targeting optimization of any or a combination of the design metrics viz. speed, power and area. Hence, it may be that any of these designs might have been employed for realizing the full adder, for inclusion as an element in a commercial standard cell library such as [18] or [19].

In this article, our primary focus is on realizing the adder functionality using readily available off-theshelf components of a cell library and figure out whether they might enable higher performance over those existent in a typical standard cell library. To this end, we propose the design of a dual-bit adder module using standard cells, including technologydependent logic optimization to analyze its impact on delay improvement for the ripple carry adder (RCA) topology. The ripple carry structure serves as a good platform for validating the performance potential of any individual adder block [20]. Our approach is distinguished in that it is geared towards achieving a delay optimal and energy-efficient solution based on a semi-custom design approach rather than adopting a full-custom style. This additionally provides a framework for analysis using any generic standard cell library.

The remaining part of this paper is organized as follows. Section 2 provides a brief background about the conventional full adder realization. In section 3, we present the design of a dual-bit adder module and in the next section, make mention of the hybrid ripple carry adder architecture that could facilitate a marginal speed improvement over a dual-bit adder based ripple carry adder tree. In section 5, we describe the simulation mechanism and report the results obtained, followed by a discussion of the same. Finally, we make the concluding remarks in the last section.

2 The Binary Full Adder and Its Classical Realization

A single-bit full adder¹ (SBFA) has 3 inputs, namely an augend (say, a), an addend (say, b) and an input carry (say, cin) and produces 2 outputs, namely the sum (*Sum*) and output carry (*Cout*). Table 1 depicts the truth table of the binary full adder and the fundamental equations governing its outputs are given by (1) and (2) respectively.

$$Sum = a \oplus b \oplus cin \tag{1}$$

$$Cout = (a \oplus b)cin + ab$$
(2)

Table 1. Truth table of the 1-bit full adder



Fig. 1. A conventional SBFA constructed using two half-adder blocks

3 Proposal and Design of the Dual-Bit Adder Module

RCAs were found to occupy the least area and consume less energy per addition, next only to the Manchester carry chain adder, in relative comparison with many high-speed adder architectures [21]. Given this, we consider the effect of performing simultaneous addition of two binary bits rather than only one bit at a time based on the typical RCA topology, to ascertain whether this scheme will be able to improve the speed of addition for the worst-case topology. It is the intention of this article to primarily investigate this issue and then comprehensively comment on the resulting analysis. Moreover, it is expected that the article would appeal to the target audience in terms of its contribution to conventional pedagogical knowledge of digital logic and computer design (especially, arithmetic circuits).

At the outset, our proposal seems to enthuse an interest for the reason that the number of stages in the ripple carry cascade would simply get halved, i.e. in case of an *n*-bit RCA, as portrayed by figure 2, the longest data path delay would be the summation of the delay of all the *n* individual

¹ Here, a binary full adder is explicitly referred to as the *single-bit full adder* (SBFA) for the sake of clarity. The following sections will introduce an adder module that is used to perform simultaneous addition of two bits, named as the *dual-bit full adder* (DBFA).

single-bit full adder stages; in general, the computation complexity is O(n).

If addition is performed considering two bits at a time, as shown in figure 3, the maximum path delay would encounter approximately half the time complexity, i.e. $O(\frac{n}{2})$. We shall refer to the adder module, which adds two augend bits and two addend bits simultaneously taking into account the input carry as the dual-bit full adder (DBFA).

from the initial two-level reduced algebraic expressions viz. minimum sum-of-products, obtained using a standard logic minimizer: Espresso [22]. The physical logic synthesizing the DBFA has been implemented, taking cognizance of the appropriate complex library gates as shown in figure 4. In fact, the adder can be realized using only twelve cells, where a 3-input XOR gate can be used to derive the least significant sum output, but this has been avoided to reduce the loading on cin.



Fig. 2. *n*-bit basic carry propagate adder topology employing SBFA blocks



Fig. 3. *n*-bit RCA architecture utilizing DBFA modules

$$Cout = cin(a1+b1)(a0+b0) + a0b0(a1+b1) + a1b1$$
(3)

$$Sum1 = (a1 \oplus b1) \{ \overline{(cin + a0b0)(a0 + b0)} \} + (\overline{a1 \oplus b1}) \{ (cin(a0 + b0)) + a0b0 \}$$
(4)

 $Sum0 = cin \oplus a0 \oplus b0$

The DBFA block basically consists of five inputs (two augend inputs – a1, a0; two addend inputs – b1, b0 and the carry input – cin) and produces three outputs (most significant and least significant sum outputs – *Sum*1, *Sum*0 and a carry output – *Cout*). The DBFA's truth table is given in the Appendix.

The optimized Boolean equations of the DBFA are specified by (3) - (5). They are derived starting

4 Hybrid Ripple Carry Adder Architecture

From the longest signal path depicted in the adder modules corresponding to figures 1 and 4, it can be observed that the data path is longer in case of the dual-bit adder module compared to the single-bit adder block with respect to the least significant stage. Therefore, the delay associated with the dual-

(5)

bit adder module present in the least significant stage of the RCA shown in figure 3 would be higher than the delay associated with the single-bit adder module in a similar stage of the RCA structure of figure 2. However, in the subsequent stages of the adder cascade, the critical path for the DBFA would only be traversed through a single complex gate. Hence, from the above discussion, it can be safely generalized that a SBFA module is ideally suited for the least significant adder stage(s) as it exhibits reduced latency to produce the carry output. Henceforth, our discussion would deal with only the sample case of a 32-bit RCA. The reason for this specific discussion is because of our observation that the positioning of the SBFAs would ideally depend on the adder operand width, and the delay optimal placement needs to be confirmed through static timing analysis in every scenario.

With reference to our preceding arguments, it can be observed that two SBFA blocks could replace the DBFA module of the least significant RCA stage of figure 3. However, one of the SBFA blocks [23] [24] could be positioned in the most significant stage to effect good improvement in speed and/or power - however, this hypothesis is to be confirmed through static timing analysis on a case-by-case basis. Herein, we notice that the RCA now incorporates a hybrid combination of single-bit and dual-bit full adder modules. Therefore, such a topology is labelled as the hybrid RCA (HRCA) architecture, represented diagrammatically in figure 5. The HRCA structure is anticipated to result in only a marginal reduction in delay, area occupancy and/or power consumption, and so this topology exhibits only a minor optimization potential.



Fig. 4. Proposed gate level realization of the DBFA unit



Fig. 5. HRCA topology featuring SBFAs and DBFAs in the linear cascade

5 Simulation Mechanism, Results and Inferences

Three RCAs, all of size 32 bits, were implemented using standard cells, with the individual SBFA and DBFA blocks described in a semi-custom style. Thus, the physical implementation is in exact conformity with the logic description. The single-bit and dual-bit full adder modules were instantiated to realize a 32-bit RCA, consisting of a series cascade of adder stages (32 stages with SBFAs and 16 stages for usage of DBFAs). The gate level simulations were performed targeting the best case PVT corner (1.32V, -40°C) of the high-speed 130nm Faraday (UMC) CMOS process [18] and the fastest of the best speed corners (1.35V, -40°C) of the high-speed inherently power optimized 65nm and STMicroelectronics bulk CMOS process [19].

The simulation results purely reflect the performance, power and area metrics of the combinatorial adder logic with no consideration of any sequential components. This sets the tone for a legitimate comparison of various adder realizations. Cadence NC-Verilog had been used for functional simulation and also to generate the switching activity files corresponding to the gate level simulations. Timing, power and area estimation were done within the Synopsys PrimeTime environment [25]. PrimeTime was used to perform static timing analysis and PrimeTime PX was used to estimate average power dissipation with a small set of input vectors, corresponding to the input profile of a simple combinational benchmark, newcwp. The power dissipation figures correspond to input sequences fed at a nominal rate of every 6ns for the 130nm technology node and 4ns for the 65nm process technology. The adder inputs were configured with the driving strength of the minimum sized inverter in the library, while their outputs possess fanout-of-4 drive strength. Also, minimum sized library elements were preferred for all the simulations. Automatic selection of appropriate wire loads was performed during successive simulations; thereby estimated net parasitics were taken into account during the process of experimentation.

We first present the simulation results corresponding to different adders that pertain to the 130nm UMC bulk CMOS process. Critical path delay and area parameters are listed in Table 2, while the power figures are mentioned in Table 3. The total power metric is the gross of dynamic and static power components, while dynamic power is in turn the sum of switching and internal power components. From Table 2, we can infer that the proposed DBFA based 32-bit RCA features the least delay metric, reporting an impressive reduction in latency by 52.4%, in comparison with the 32-bit RCA implemented using the commercial library's full adder cell. The values mentioned within brackets in the 2nd column of Table 2 signify the increase in latency for the different 32-bit adders in comparison with that featuring the least value, while the values given within brackets in the 3rd column specify the relative area occupancy for the half adders based SBFA or the DBFA block compared to the commercial library's SBFA.

Table 2. Maximum data path delay and area metrics of 32-bit RCAs (130nm process)

Adder realization style	Critical path delay (ns)	Cells area (µm ²)	
Half adders based	3.52	1184	
SBFA	(117.3%)	(1.68×)	
Commercial	3.40	704	
library's SBFA	(109.9%)		
Proposed DBFA	1.62	1504	
		(2.14×)	

Table 3. Power dissipation parameters of 32-bitcarry propagate adders (130nm process)

Adder realization style	Total power (µW)	Static power (nW)
Half adders based	86.31	230.91
SBFA		
Commercial library's	74.48	120.78
SBFA		
Proposed DBFA	153.17	279.35

From the power figures listed in the above tabular column, it is evident that the 32-bit RCA constructed using the commercial library's SBFA reports the least power dissipation. This is mainly attributable to the finer optimization achieved at the transistor level. In comparison with it, the proposed DBFA based RCA is found to dissipate nearly twice more power. Also, it can be noticed that while the half adders based SBFA features approximately twice the leakage in comparison with the commercial library's SBFA; in terms of average power consumption, the former is only 1.2× expensive compared to the latter. For analysis with the high-speed 65nm CMOS process, the fastest speed corner among all the 26 best-case PVT corners has to be determined. Based on a perusal of the corner cases listed in Table 4, the fastest speed corner can be identified as that pertaining to (1.35V, -40°C); however, this has been confirmed through experimentation and the delay values corresponding to a 32-bit RCA employing DBFAs corroborate the manual observation.

Table 4.Determination of the fastest speed corneramongst the best case PVT specifications of the65nm STMicroelectronics CMOS technology

Best-case	Critical path delay		
electrical	of the 32-bit RCA		
specification	(ns)		
(1.05V, 105°C)	1.48		
(1.05V, 125°C)	1.48		
(1.05V, -40°C)	1.32		
(1.05V, -40°C, 10y)	1.34		
(1.10V, 105°C)	1.38		
(1.10V, 125°C)	1.38		
(1.10V, 150°C)	1.38		
(1.10V, -40°C)	1.21		
(1.10V, -40°C, 10y)	1.23		
(1.15V, 105°C)	1.29		
(1.15V, 125°C)	1.29		
(1.15V, -40°C)	1.12		
(1.15V, -40°C, 10y)	1.14		
(1.25V, 105°C)	1.17		
(1.25V, 125°C)	1.17		
(1.25V, -40°C)	0.99		
(1.25V, -40°C, 10y)	1.00		
(1.30V, 105°C)	1.11		
(1.30V, 125°C)	1.11		
(1.30V, 150°C)	1.11		
(1.30V, -40°C)	0.94		
(1.30V, -40°C, 10y)	0.95		
(1.35V, 105°C)	1.07		
(1.35V, 125°C)	1.07		
(1.35V, -40°C)	0.90		
$(1.35V, -40^{\circ}C, 10y)$	0.91		

The simulation results of the 32-bit RCAs pertaining to the fastest speed corner of the 65nm bulk CMOS process are listed in Tables 5 and 6. It can be observed that the proposed DBFA based RCA reports the least maximum path delay, featuring a reduction in longest path delay by 46.7% over the commercial library's SBFA based RCA. Nevertheless, this significant reduction in latency comes at a relative increase in area expenditure by

 $1.33 \times$ and consequently higher static power consumption (1.8×) and more average power dissipation by 1.15×.

Table 5. Longest data path delay and area metricsof 32-bit carry propagate adders (65nm process)

Adder	Critical path	Cells	
realization	delay	area	
style	(ns)	(μm^2)	
Half adders based	1.96	532.48	
SBFA	(117.8%)	(1.78×)	
Commercial	1.69	299.52	
library's SBFA	(87.8%)		
Proposed DBFA	0.90	698.88	
		(2.33×)	

Table 6.	Power consumption parameters of 32-bit
car	ry propagate adders (65nm process)

Adder realization style	Total power (µW)	Static power (nW)	
Half adders based	73.33	434.58	
SBFA			
Commercial library's	55.88	189.94	
SBFA			
Proposed DBFA	120.14	533.15	

Considering the simulation results corresponding to both the CMOS process technologies, it can be inferred that a substantial decrease in worst case data path delay (49.6%, on average) has been effected for the proposed DBFA based RCA in comparison with the RCA constructed using the full adder of commercial deep sub-micron standard cell libraries. However, the speed advantage is accompanied by more real estate (1.24×, on average) and consequently more power dissipation by 1.11×.

Though power-delay product (PDP) has been traditionally resorted to as a typical design parameter for evaluating digital system designs with respect to low power, the problem with this while addressing parameter lower energy consumption was illustrated in [26] and [27], based on a research carried out at Stanford University. Therefore, a more reliable parameter by name energy-delay product (EDP) was proposed and used for digital system (microprocessor) comparisons, wherein a smaller EDP value implied lower energy consumption for the same level of performance or a more energy-efficient design.

We now pictorially represent the PDP and EDP values of different 32-bit RCAs for an effectual comparison. Figure 6 highlights a comparison between various 32-bit RCAs in terms of the PDP for both the CMOS processes. It can be seen that the proposed DBFA based RCA reports the optimal PDP figure for the 130nm process, while the commercial library's SBFA based RCA is found to be the best with respect to the 65nm process. Overall, the PDP of the 32-bit RCA incorporating the commercial libraries full adder cell is found to be only slightly better than the DBFA based RCA, enabling a marginal reduction by 2.4%. On the other hand, the 32-bit RCA featuring the conventional half adders based SBFA has a poor PDP figure in comparison with the commercial libraries SBFA based RCA, reporting an increase of 28.7%.

The EDP metric corresponding to the different adder realizations for 130nm and 65nm processes is portrayed in figure 7.

With respect to both the bulk CMOS process technologies, the DBFA based RCA is found to yield the energy optimal solution among all the RCA implementations. This may be understandable as EDP is proportional to the square of the delay parameter and therefore the proposed DBFA module enjoys the advantage of being inherently high speed. Compared to the 32-bit RCA utilizing the full adder element of the respective cell library, the 32-bit RCA employing the DBFA module facilitates reduction in EDP by 53.3% and 39% for the 130nm and 65nm processes respectively, thus clearly demonstrating superior energy efficiency over its counterparts.



Fig. 6. PDP (10⁻¹⁵ J) values of various 32-bit RCAs corresponding to 130nm and 65nm CMOS processes



Fig. 7. EDP (10⁻²⁴ Js) figures of various 32-bit RCAs pertaining to 130nm and 65nm CMOS processes

In fact, the DBFA block that is implemented using 12 cells instead of 13 cells, as discussed earlier, by employing a 3-input XOR gate for realizing the least significant sum output results in a critical path delay 1.66ns and 0.95ns for 32-bit addition, of corresponding to the 130nm and 65nm processes respectively. Hence, the dual-bit adder unit designed using 13 cells is optimal in terms of speed compared to its 12 cells counterpart. With respect to area requirement, the latter occupies less area than the former by only 1.2% for the 65nm process and similar reduced area occupancy with respect to the 130nm process. Nevertheless, in terms of average power dissipation, the 12 cells DBFA is found to be more expensive than the 13 cells DBFA by 2%, on average, which is attributable to the greater size of the XOR3 cell compared to the XOR2 cell in case of the former adder. With respect to PDP and EDP parameters, the DBFA block realized using 12 cells is found to be more expensive than the DBFA block synthesized using 13 standard cells by 6% and 9% respectively.

In essence, the proposed DBFA is found to considerably speed-up the worst-case addition process in comparison with the RCA utilizing the full adder element of the commercial library by a whopping 95%. Further, it is found to be energyefficient as well, featuring the least EDP among all the other RCA realizations. The DBFA based RCA leads to average reduction in EDP to the tune of 51%, compared to the RCA constructed using the full adder block present in the cell libraries, thus asserting its supremacy. Possible applications of the conceived DBFA module, with a view to realize fast arithmetic circuits based on other adder topologies would be investigated as a future work.

As part of our simulations, we also considered sandwiching the DBFAs in between the most significant and least significant SBFAs [23] of a 32bit RCA to evaluate the potential of the HRCA configuration. Based on analysis targeting the 130nm and 65nm CMOS process technologies, it was found that the hybrid scheme resulted in only a minor reduction in critical path delay and area by 3% and 1.2% respectively over the pure DBFA based RCA, while both feature approximately similar power dissipation characteristics.

6 Conclusions

Two novel thoughts have been propounded in this paper, following a semi-custom design approach.

- Addition involving two bits at a time based on the simple RCA structure, rather than the conventional manner of adding only one bit, which led to the original design of a new DBFA module utilizing standard cells
- Sandwiching/padding the DBFA modules in between/with the SBFA blocks, paving the way for a HRCA architecture, which however constitutes only a peephole logic optimization

It may be thought of whether one could decrease the worst-case addition time complexity further, within the ambit of the RCA topology, by performing addition of 3-bits at a time, rather than adding 2-bits at a time (as described in this work). This, however, might not be beneficial at the hardware level. This is because of the exponential increase of the input space. In general, an n-input Boolean function would demand an input space requirement of $O(2^n)$. Although 'n' may increase linearly, the associated input state space would expand exponentially. With respect to the carry output signal of an *n*-bit adder block, the number of essential cubes comprising the disjunctive normal form can be deduced using the principle of mathematical induction, and is found to be of $O(2^{n+1}-1)$. As a result, though the theoretical conception of a triple-bit adder might be interesting, the physical realization might degrade the delay metric, as more logic gates would be required resulting in a corresponding increase in the number of logic levels and also, it is likely that the loading on the primary inputs would increase substantially. Our preliminary study showed that the triple-bit adder unit tends to degrade the speed when implementing a RCA in comparison with those utilizing single-bit/dual-bit adder modules.

References:

- [1] D.C. Chen, L.M. Guerra, E.H. Ng, M. Potkonjak, D.P. Schultz and J.M. Rabaey, "An Integrated System for Rapid Prototyping of High Performance Algorithm Specific Data Paths," *Proc. IEEE Conference on Application Specific Array Processors*, pp. 134-148, 1992.
- [2] J.L. Hennessy and D.A. Patterson, *Computer Architecture: A Quantitative Approach*, Morgan Kaufmann Publishers, 1990.
- [3] J.D. Garside, "A CMOS VLSI Implementation of an Asynchronous ALU," *Proc. IFIP Working*

Conference on Asynchronous Design Methodologies, pp. 181-207, 1993.

- [4] A.J. Poustie, K.J. Blow, A.E. Kelly and R.J. Manning, "All-Optical Full Adder with Bit-Differential Delay," *Optics Communications*, vol. 168, no. 1-4, pp. 89-93, September 1999.
- [5] T.A. Rahman, M.K. Ahmed and E.M. Saad, "All-Optical Arithmetic Unit Based on the Hardlimiters," Proc. 6th WSEAS International Conf. on Electronics, Hardware, Wireless and Optical Communications, pp. 64-67, 2007.
- [6] P.K. Lala, J.P. Parkerson and P. Chakraborty, "Adder Designs Using Reversible Logic Gates," WSEAS Trans. on Circuits and Systems, vol. 9, no. 6, pp. 369-378, June 2010.
- [7] C. Reis, J.A.T. Machado and J.B. Cunha, "Evolutionary Techniques in Circuit Design and Optimization," Proc. 6th WSEAS International Conf. on Simulation, Modelling and Optimization, pp. 307-312, 2006.
- [8] N. Zhuang and H. Wu, "A New Design of the CMOS Full Adder," *IEEE Journal of Solid-State Circuits*, vol. 27, no. 5, pp. 840-844, 1992.
- [9] N.H.E. Weste and K. Eshraghian, Principles of CMOS VLSI Design – A Systems Perspective, 2nd Edition, Addison-Wesley Publishing: Massachusetts, 1993.
- [10] R. Shalem, E. John and L.K. John, "A Novel Low-Power Energy Recovery Full Adder Cell," *Proc. ACM Great Lakes Symposium on VLSI*, pp. 380-383, 1999.
- [11] M. Margala, "Low-Voltage Adders for Power-Efficient Arithmetic Circuits," *Microelectronics Journal*, vol. 30, no. 12, pp. 1241-1247, 1999.
- [12] A.M. Shams, T.K. Darwish and M.A. Bayoumi, "Performance Analysis of Low-Power 1-Bit CMOS Full Adder Cells," *IEEE Transactions on VLSI Systems*, vol. 10, no. 1, pp. 20-29, 2002.
- [13] M. Zhang, J. Gu and C.H. Chang, "A Novel Hybrid Pass Logic with Static CMOS Output Drive Full Adder Cell," *Proc. IEEE International Symposium on Circuits and Systems*, pp. 317-320, 2003.
- [14] Y. Jiang, A. Al-Sheraidah, Y. Wang, E. Sha and J.G. Chung, "A Novel Multiplexer-Based Low-Power Full Adder," *IEEE Transactions on Circuits and Systems – II: Express Briefs*, vol. 51, no. 7, pp. 345-348, 2004.
- [15] S. Goel, S. Gollamudi, A. Kumar and M. Bayoumi, "On the Design of Low-Energy Hybrid CMOS 1-Bit Full Adder Cells," Proc. 47th IEEE International Midwest Symposium on Circuits and Systems, vol. II, pp. 209-212, 2004.

- [16] S. Goel, A. Kumar and M.A. Bayoumi, "Design of Robust, Energy-Efficient Full Adders for Deep Submicrometer Design using Hybrid-CMOS Logic Style," *IEEE Transactions* on VLSI Systems, vol. 14, no. 12, pp. 1309-1321, 2006.
- [17] C. Senthilpari, A.K. Singh and K. Diwakar,
 "Design of a Low-Power, High-Performance,
 8×8 Bit Multiplier using a Shannon-Based
 Adder Cell," *Microelectronics Journal*, vol. 39, no. 5, pp. 812-821, 2008.
- [18] Faraday Technology Corporation, Faraday Cell Library FSCOH_D 0.13µm Standard Cell, 2004.
- [19] STMicroelectronics CORE65LPLVT_1.10V Version 4.1 Standard Cell Library, User Manual and Databook, July 2006.
- [20] C.H. Chang, J. Gu and M. Zhang, "A Review of 0.18µm Full Adder Performances for Tree Structure Arithmetic Circuits," *IEEE Trans. on VLSI Systems*, vol. 13, no. 6, pp. 686-695, June 2005.
- [21] C. Nagendra, M.J. Irwin and R.M. Owens, "Area-Time-Power Tradeoffs in Parallel Adders," *IEEE Trans. on Circuits and Systems – II: Analog and Digital Signal Processing*, vol. 43, no. 10, pp. 689-702, October 1996.
- [22] R.K. Brayton, G.D. Hachtel, C.T. McMullen and A.L. Sangiovanni-Vincentelli, *Logic Minimization Algorithms for VLSI Synthesis*, Kluwer Academic Publishers, 1984.
- [23] P. Balasubramanian and N.E. Mastorakis, "A Delay Improved Gate Level Full Adder Design," in the Book, "COMPUTING AND COMPUTATIONAL INTELLIGENCE," Included in ISI/SCI Web of Science and Web of Knowledge, pp. 97-102, 2009.
- [24] P. Balasubramanian and N.E. Mastorakis, "A Low Power Gate Level Full Adder Module," in the Book, "SELECTED TOPICS ON APPLIED MATHEMATICS, CIRCUITS, SYSTEMS AND SIGNALS," Included in ISI/SCI Web of Science and Web of Knowledge, *Invited Paper*, pp. 246-248, 2009.
- [25] Synopsys Inc. <u>www.synopsys.com</u>
- [26] M. Horowitz, T. Indermaur and R. Gonzalez, "Low-Power Digital Design," Proc. IEEE Symposium on Low Power Electronics, pp. 8-11, 1994.
- [27] R. Gonzalez and M. Horowitz, "Energy Dissipation in General Purpose Microprocessors," *IEEE Journal of Solid-State Circuits*, vol. 31, no. 9, pp. 1277-1284, 1996.

Appendix:

Inputs			Outputs				
<i>a</i> 1	<i>a</i> 0	<i>b</i> 1	<i>b</i> 0	cin	Cout	Sum1	Sum0
0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1
0	0	0	1	0	0	0	1
0	0	0	1	1	0	1	0
0	0	1	0	0	0	1	0
0	0	1	0	1	0	1	1
0	0	1	1	0	0	1	1
0	0	1	1	1	1	0	0
0	1	0	0	0	0	0	1
0	1	0	0	1	0	1	0
0	1	0	1	0	0	1	0
0	1	0	1	1	0	1	1
0	1	1	0	0	0	1	1
0	1	1	0	1	1	0	0
0	1	1	1	0	1	0	0
0	1	1	1	1	1	0	1
1	0	0	0	0	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	0	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	0	1	0	0
1	0	1	0	1	1	0	1
1	0	1	1	0	1	0	1
1	0	1	1	1	1	1	0
1	1	0	0	0	0	1	1
1	1	0	0	1	1	0	0
1	1	0	1	0	1	0	0
1	1	0	1	1	1	0	1
1	1	1	0	0	1	0	1
1	1	1	0	1	1	1	0
1	1	1	1	0	1	1	0
1	1	1	1	1	1	1	1

Truth table of the dual-bit full adder