

A Low Energy and High Performance DM² Adder

Itamar Levi, Amir Albeck, Alexander Fish, *Member, IEEE*, and Shmuel Wimer, *Member, IEEE*

Abstract—A novel Dual Mode Square (DM²) adder is proposed. The DM² adder achieves low energy, high performance and small area by combining two independent techniques recently proposed by the authors: dual-mode logic (DML) and dual-mode addition (DMADD). DML is a special gate topology that allows on-the-fly adaptation of the gates to real time system requirements, and also shows a wide energy-performance tradeoff. DMADD is probability based circuit architecture with a wide energy-performance tradeoff; however its utilization in a pipelined processor requires multi-cycle operation in some cases. We show how DML circuits avoid this requirement, and thus make it possible to transparently plug-in the DM² adder and derive full benefits from the DMADD. Previous work showed that the DMADD can lead to energy savings of up to 50% at the same clock cycle, compared to conventional CMOS solutions. Simulation results in a 40 nm standard process shows that the proposed DM² approach achieves additional energy savings of 27% to 36% for 64-bit and 32-bit adders, respectively, compared to DMADD.

Index Terms—Adders, DML, low-power design.

I. INTRODUCTION

OBTAINING energy efficiency and low peak power while maintaining computational performance is one of the primary goals in contemporary processor design. Energy reduction and performance improvement have been studied extensively from the very high level of application algorithms, through system [1], architecture [2], [19] and logic levels, to the gate [3]–[7], [19], circuit, device and interconnect levels [8], [9]. Energy reduction in the context of pipelined digital systems has also been studied in [19] and [20]. For example, approaches such as circuit sizing and supply voltage scaling have been utilized and analyzed [19].

This work combines recently proposed gate and architecture levels approaches. It shows how the combination of two independent methods yields considerable performance enhancement and energy efficiency.

The first method is *dual-mode addition* (DMADD) [10]. It takes advantage of the carry probability to perform low-power addition and leading to a considerable energy reduction of up to 50% compared to conventional designs. However, it requires some pipeline modifications to support multi-cycle addition. The second method is a logic gate topology called *dual-mode*

logic (DML) comprising static and dynamic operation modes within the same gate [11]–[14].

In this paper we propose Squared Dual Mode (DM²) approach combining DMADD and DML. DM² main objective is to eliminate the DMADD need for multi-cycle addition by replacing its ordinary CMOS logic with DML, thus avoiding the architectural overheads. Furthermore, DM² enables considerable energy savings due to the inherent properties of the DML gates.

Two adders were implemented using the DM² method in a standard 40 nm process. Theoretical analysis and post-layout simulations prove the efficiency of DM² exhibiting energy saving of up to 36%, as compared to the DMADD.

The rest of this paper is organized as follows: Section II briefly presents DMADD and DML techniques. Section III describes the DMADD and DML integration into the DM², including a theoretical analysis and circuit design optimization. Simulations of 40 nm DM² adders and their comparison to standard CMOS based DMADD, Brent-Kung and Ripple adders are presented in Section IV. Section V concludes the paper.

II. DML AND DMADD OVERVIEW

A. DMADD

DMADD comprises two addition modes [10]. The energy efficient one-cycle mode, called *normal*, is used most of the time to properly compute addition. It takes advantage of the average (expected) longest carry in addition which is $O(\log_2 n)$, and is much shorter than the adder size n . The probability of $O(n)$ -bit carry propagation is nearly zero [17]. The second mode, called *extended*, occurs very infrequently and requires several clock cycles to properly add. The decision of which mode should take place requires an appropriate control circuit. When this control is used in a pipelined processor it selects the proper mode at the instruction decode (ID) stage, prior to the ALU stage.

The probability q of a carry to propagate through a bit is $(1/2)$ and the propagation probability through successive k bits is therefore 2^{-k} . The probability q_k that it takes exactly k bits for a carry to either be generated or killed is:

$$q_k = \Pr \left(\prod_{i=1}^{k-1} p_i = 1 \right) \times \Pr(p_k = 0) = 2^{-k} \quad (1)$$

where p_i is the propagate signal of bit i . It was shown in [10] that adders designed for $2 \log_2 n$ -bit carry propagation yields considerable energy efficiency compared to ordinary n -bit carry propagation designs.

An n -bit DMADD comprises m groups of k bits each, where, $n = mk$, such that the carry propagation delay of two k -bit adders meets the clock cycle. It enables a few design alternatives to reduce energy. A design for a $(2k - 1)$ -bit delay rather

Manuscript received December 24, 2013; revised May 12, 2014 and June 10, 2014; accepted June 18, 2014. Date of publication July 17, 2014; date of current version October 24, 2014. This work was supported by the Israel Science Foundation (ISF Grant) under Grant Number 1678/13. Dual Mode Logic methodology was developed in the frame of the Kamin Grant of the Office of the Chief Scientist (OCS) in the Ministry of Economy. This paper was recommended by Associate Editor M. Seok.

The authors are with the Bar-Ilan University, Ramat Gan 52900, Israel (e-mail: itamarlevi@gmail.com; amiralbeck@gmail.com; shmuel.wimer@biu.ac.il; alexander.fish@gmail.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSI.2014.2334793

than n -bit enables transistor downsizing, high threshold voltage usage, or voltage scaling [1]. To compensate for cases where the carry propagates through more than $(2k - 1)$ -bits, m clock cycles are used to compute. The normal operation mode of the DMADD requires each of the m groups to kill or generate a carry, for which the probability q_{norm} is

$$q_{\text{norm}}(k, m) = (1 - 2^{-k})^m = 1 - m2^{-k} + O(2^{-2k}) > 1 - m2^{-k} \quad (2)$$

whereas the probability q_{ext} of the extended mode is

$$q_{\text{ext}}(k, m) < m2^{-k}. \quad (3)$$

The deployment of DMADD in an in-order pipelined processor requires stalling the pipe for m cycles in case of extended-mode addition. This imposes some design overhead and performance degradation. More severely, DMADD in out-of-order [15] architectures may be extremely difficult. Here we show how DML avoids the extended, multi-cycle mode. It ensures that regardless of the carry propagation, the DMADD will always properly compute within a single cycle.

B. DML

DML enables on-the-fly switching in clock cycle resolution between the high performance dynamic and energy efficient static operation modes. This instantaneous switching is obtained by a unique circuit topology supplemented by appropriate transistor sizing [11]–[14],[21]. The DML basic gate structures are shown in Fig. 1. Although the topology of a DML gate is similar to a static logic family gate (e.g., a conventional CMOS gate), it comprises an additional transistor. DML gates have a very intuitive structure; however they require an unconventional sizing scheme to achieve the target behavior [11]–[14]. In the static DML operation mode, the M1 transistor is cut off by applying the high Clk signal for “Type A” and the low Clk_{bar} for “Type B” topology. Therefore, the gates operate similarly to static CMOS logic. For the dynamic operation mode, the Clk toggles, providing two separate pre-charge and evaluation phases. During the pre-charge phase, the output is charged to V_{DD} in the “Type A” gate (and discharged to GND in “Type B”). During the evaluation, the output is evaluated according to the values of the gate inputs. Transistor upsizing enables evaluation through a low resistive network, which results in faster operation in the dynamic mode. The complementary (non evaluate) network transistors are all minimally sized (low capacitance), yielding a rather slow static mode, with, however, very low energy consumption [11]–[14]. Switching between DML modes is possible at any circuit and system level: single gate, logic paths, complete block and whole system. DML logic can naturally be mixed with ordinary CMOS logic. It works robustly at any supply voltage, down to the sub-threshold region. The dual modality enables both lower energy and higher performance in most commonly used designs.

DMADD can benefit from DML by instantaneous switching between DML static and dynamic modes, depending on whether the DMADD needs to operate in the normal or extended mode. In both cases a single clock cycle suffices, making the dual-mode useful for out-of-order architecture without any penalty in clock cycles of pipeline stall. As mentioned, DML utilization also yields considerable energy reduction. This follows from the lower energy of its static mode compared to CMOS, operated

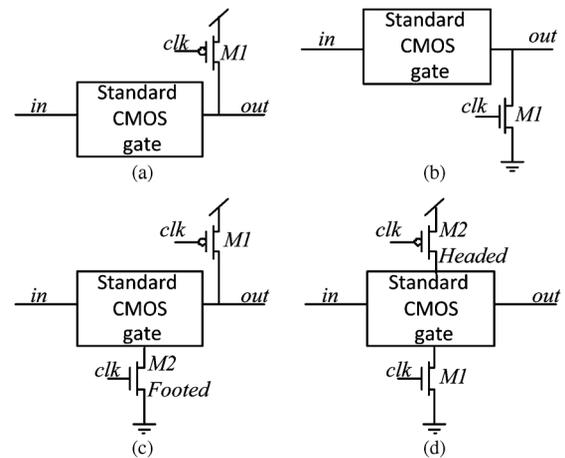


Fig. 1. (a) *Type_A* DML topology (b) *Type_B* DML topology (c) Footed *Type_A* DML gate (d) Headed *Type_B* DML gate.

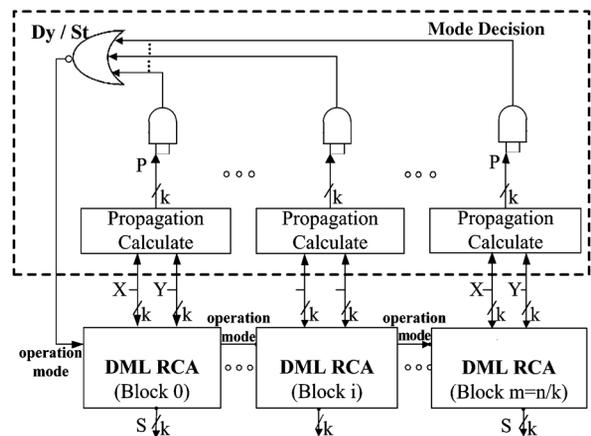


Fig. 2. DMADD adder topology and control circuit.

with very high probability. Though its dynamic mode energy is higher than CMOS, this mode only rarely occurs.

III. SYSTEM ARCHITECTURE AND TRANSISTOR SIZING

A. DM^2 Architecture

The DM^2 adder is an n -bit Ripple Carry Adder (RCA) divided into $m = n/k$ groups of k bits each, as illustrated in Fig. 2. As discussed in Section II, the probability of the normal addition mode is approximately $1 - m/2^k$. The longest carry path in this mode does not exceed $(2k - 1)$ bits, far shorter than the n bit worst-case. The underlying DML gates can therefore be operated in their static, energy efficient mode. In its extended mode, where the carry propagates through more than $2k - 1$ bits with probability $m/2^k$, the DML logic will turn into its fast dynamic mode. In this mode the worst-case n -bit carry path must be completed within the given clock cycle, which is done by transistor sizing. The tradeoff is clear: while most of the time the DMADD consumes very low power, high power consumption occurs very infrequently. Obviously, k must be determined such that the propagation delay of a $(2k - 1)$ -bit carry path, where the logic is static, will not exceed the clock cycle. To minimize the dynamic mode probability, k is maximized subject to that delay constraint.

Nowadays processors are pipelined. To illustrate the advantages of DM^2 , we used a simple, yet realistic in-order pipelined processor [15]. We take advantage of the fact that the ID stage

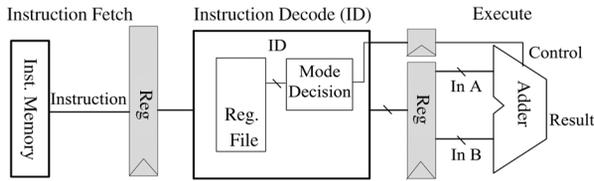


Fig. 3. Incorporation of mode decision logic.

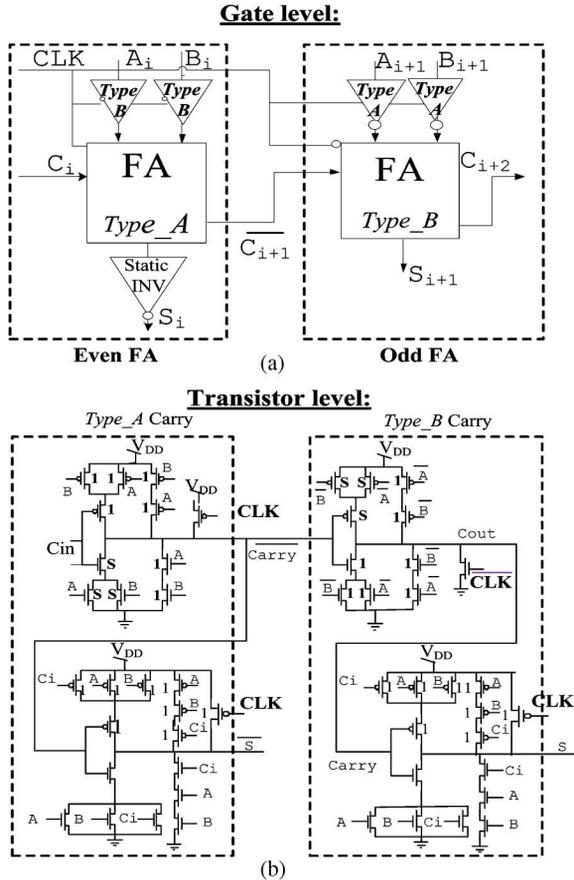


Fig. 4. Even and Odd FA, levels: (a) Gate (b) transistor.

occurs one cycle prior to execution, thus making the ALU arguments available one cycle ahead of their usage. This enables to determine the operation mode of the DMADD by using a *mode decision* block, as illustrated in Fig. 3.

The *mode decision* block architecture is shown in Fig. 2, where St and Dy denote the static-normal and dynamic-extended modes, respectively. The RCA is standard, comprising alternating polarity full-adders (FAs) [16]. The alternating polarity of the RCA bits and the inherent DML alternating pre-charge polarity [11], [12],[21] dictate the different internal designs of the polarity alternating FAs. Fig. 4 depicts the internal circuits of the two bit types.

To speed up the critical path it uses un-footed gates in even and odd bits. Design considerations of how to optimally use the DML gates within the even and odd bits can be found in [11]–[13].

It is important to note that the pre-charge of all the bits occurs simultaneously, and hence does not affect the critical path delay. To ensure proper pre-charge, the gates connected to the RCA inputs are of footed *Type_A* in the even bits and footed *Type_B* in the odd ones. The transistor-level schematics and the sizes of

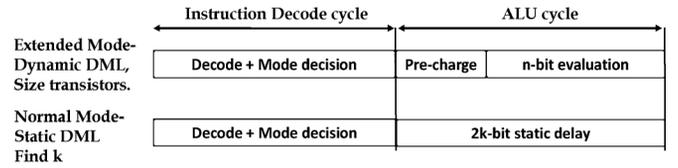


Fig. 5. System timing diagram.

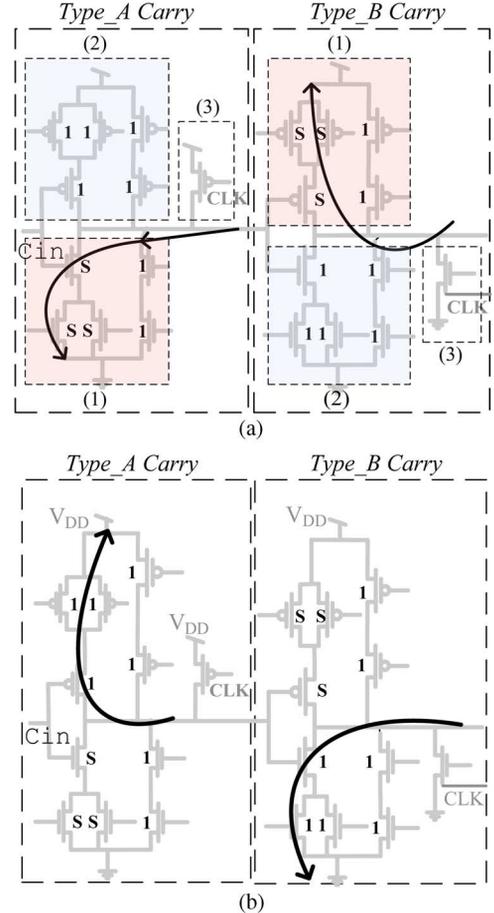


Fig. 6. DML worst-case delay path: (a) DML dynamic mode (b) DML static mode, where the blocks represent: (1) the evaluation path, (2) the complementary networks and (3) the pre-charge transistors.

the alternating bits are shown in Fig. 4(b). These are based on CCMOS (Mirror) FA [16].

B. DM^2 Transistor Sizing

Let T be the system's clock cycle, and T_{pre} the pre-charge delay of a full-adder. The size of the DML gates is determined in such a way that the carry evaluation through all the n bits will meet $T - T_{pre} = nT_{eval}$, where T_{eval} is the carry evaluation delay of a FA. Notice that the pre-charge takes place simultaneously for all bits prior to evaluation, as illustrated in Fig. 5.

It is also important to note that in DML only the transistors involved in the evaluation network are considered for up sizing. Whereas the other half of the pre-charge transistors stay minimal [11]–[14],[21]. Furthermore, not all the evaluation transistors require upsizing; i.e., only those designated by S , as shown in Fig. 6 for the carry logic of two successive bits. The DML design methodology requires those to be of opposite types.

Shown in Fig. 6(a), the critical carry path in the *Type_A* gate is passing through the lower left branch, whereas in the *Type_B* gate it is passing through the upper left branch. Consequently,

the remaining evaluation transistors can stay minimal, and are designated by 1. The smallest sizing factor S of the evaluation transistors that meet the timing constraints is obtained by simulation.

Once the transistor sizes have been determined by the DML dynamic mode, the maximal group size meeting the timing constraints in the static mode can be set. Recall that in the DMADD normal mode, the static DML mode is operational, where the carry propagates through $2k - 1$ bits at most.

The worst-case delay path for the static DM² mode differs from the dynamic one. Fig. 6(b) illustrates the critical path in the static mode that passes through the highly resistive minimal size transistors.

Let T_{stat} denote the carry delay in a FA operated in a DML static mode. The group size k is determined to satisfy $T = (2k - 1)T_{\text{stat}}$, yielding

$$k = \frac{1}{2} \left(\frac{T}{T_{\text{stat}}} + 1 \right) = \frac{1}{2} \left(\frac{T_{\text{pre}} + nT_{\text{eval}}}{T_{\text{stat}}} + 1 \right) \quad (4)$$

Usually, n is a power of two, and due to practical design considerations, k is set to the nearest power of two [10]. Since the size of the devices was chosen to be as small as possible, k must always be rounded down, since rounding up may cause timing violations.

C. Energy Savings

Although the primary motivation for DM² is to avoid the DMADD architecture overheads described above, it also enables considerable energy savings. To assess the DM² energy savings compared to DMADD, the latter was optimally designed in ordinary CMOS logic to meet the worst $2k - 1$ bit delay occurring in group size k . Once the size of the gates was determined, the switching and leakage energies, $E_{\text{switch}}^{\text{DMADD}}$ and $E_{\text{leakage}}^{\text{DMADD}}$, respectively, were measured by simulation.

Consider DM² adder energy consumption per addition, and let $E_{\text{stat}}^{\text{DM}^2}$ and $E_{\text{dyn}}^{\text{DM}^2}$ be worst-case static normal mode (most often) and dynamic extended mode (less often), respectively. Recalling that the normal mode probability is $1 - m/2^k$, we obtain

$$\frac{E^{\text{DM}^2}}{E^{\text{DMADD}}} = \frac{(1 - m/2^k) E_{\text{stat}}^{\text{DM}^2} + m/2^k E_{\text{dyn}}^{\text{DM}^2}}{E_{\text{switch}}^{\text{DMADD}} + E_{\text{leakage}}^{\text{DMADD}} + (m - 1)m/2^k E_{\text{leakage}}^{\text{DMADD}}} \quad (5)$$

The term $(m - 1)m/2^k E_{\text{leakage}}^{\text{DMADD}}$ in (5) follows from the extra $m - 1$ cycles required by the DMADD extended mode. Equation (5) can be simplified by noting that $m/2^k \ll 1$, $E_{\text{dyn}}^{\text{DM}^2} \simeq 4E_{\text{stat}}^{\text{DM}^2}$ for a DML FA (obtained by simulation), and that $E_{\text{switch}}^{\text{DMADD}} + E_{\text{leakage}}^{\text{DMADD}} \gg (m - 1)m/2^k E_{\text{leakage}}^{\text{DMADD}}$. All in all we obtain the following approximation

$$\frac{E^{\text{DM}^2}}{E^{\text{DMADD}}} \simeq \frac{E_{\text{stat}}^{\text{DM}^2}}{E_{\text{switch}}^{\text{DMADD}} + E_{\text{leakage}}^{\text{DMADD}}} \quad (6)$$

Note that we did not include the energy consumed by the adder's controller as it is similar to DMADD and DM² adders.

IV. SIMULATION RESULTS

We compare 32 and 64-bit DMADD with DM² adder and DM² with ripple carry and Brent-Kung adders designed in a 40 nm process technology, targeting 1 GHz clock frequency. Energy, area, extended mode probability and reliability are compared. As described in Section III, the first DM² design step is

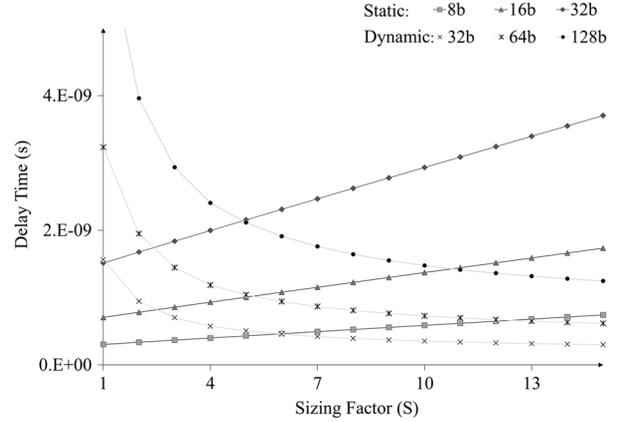


Fig. 7. Delays of 32, 64 and 128 DM² bit adders operated in both the dynamic and static modes.

to set the device sizes S of the FAs to meet the clock cycle in the DML dynamic mode, which defines the optimal group size k .

Note that the DM² adder requires extra circuitry for precharge, which should be carefully designed. As for other components of the design, the precharge circuits were carefully designed under PVT corners and mismatch, and the necessary margins were taken. The energy and delay overheads, which are negligible, are represented by the final results, as shown in this Section.

A. Transistor Sizing and Setting the Group Size

Fig. 7 presents the delays of 32, 64 and 128 bit adders operating in both dynamic and static modes. The delay of the dynamic mode decreases with S increase and is given by:

$$T_{\text{eval}} = (n - 1) \frac{R}{S} (\alpha C + \beta SC) + \frac{R}{S} (\delta C + \gamma SC) \quad (7)$$

where R is the resistance and C is the capacitance of a minimal size transistor, and α , β , γ and δ are process dependent parameters. The $(n - 1)$ factor represents the first $(n - 1)$ gate delay in the chain that charges similar capacitors, where the second term represents the last gate which charges the output register. Though not intuitive, the delay in the static DML mode increases with the increase in transistor size. This follows from the inherent structures of the DML logic. Recall that in *Type_A* the size of the pull-up transistors through which capacitive load is pre-charged are minimal. Similar arguments apply for *Type_B* pull-down transistors (Fig. 6(b)). All in all, the static delay is given by:

$$T_{\text{stat}} = (2k - 1)R(\alpha C + \beta SC) + R(\delta C + \gamma SC) \quad (8)$$

Consider the design of a 32-bit adder targeting 1 GHz clock frequency. Ignoring setup time, the intersection point (a) of the dynamic curve with the 1 GHz horizontal line in Fig. 8 dictates the smallest sizing factor S that meets the timing constraints. Ideally DM² should pursue the largest possible group size k , which results in the smallest probability of dynamic (high energy) mode. This could theoretically be achieved by the static curve passing through (a). Practically, since k is a power of two, it is obtained by the nearest k below point (a).

To follow the common design methodology where sizing factors are integers and the DMADD group size is a power of two, the first practical design points (a') corresponding to $S = 2$

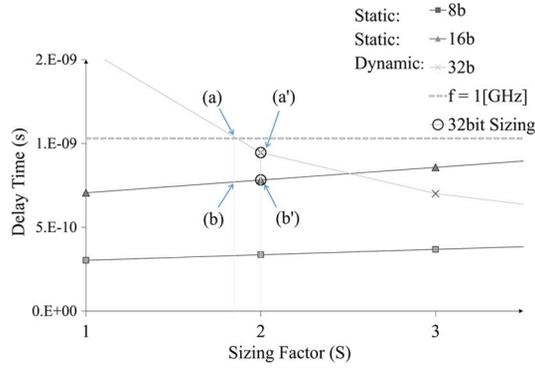


Fig. 8. 32 bit adder design operation point.

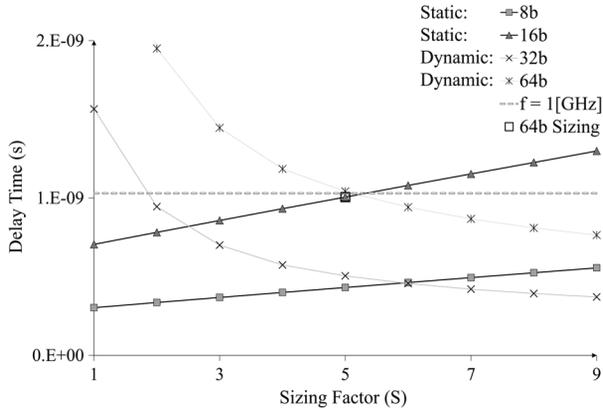


Fig. 9. 64 bit adder design operation point.

was chosen. The largest practical group size that meets the delay constraints for $S = 2$ is (b'), yielding a 32-bit adder for which $2k = 16$.

Another adder of 64-bit was designed with similar considerations, yielding $2k = 16$ and $S = 5$ as illustrated in Fig. 9 (The intersection of the dynamic, static and clock-cycle curves in a single point is merely a coincidence.)

The procedure of finding the minimal device sizing factor automatically determines the maximal group size, which in turn minimizes the dynamic DML operation mode probability. To summarize, the determination of the device sizes on one hand and the group size on the other, is optimal by all means, and there is no other design point meeting the clock-cycle and yielding lower energy.

The dynamic operation mode probability is derived by substituting n and k in (3) which for a 32-bit adder yields 1.56%, and 3.12% for a 64-bit adder.

B. Energy Saving Measurements and Bounds

To account for the energy, the adder's inputs were set to result in the worst-case of maximum energy consumption. Fig. 10 illustrates two successive bits of alternating types as dictated by the DML design methodology. To trigger the worst case, the gates of all the evaluating devices in *Type_A* cell should be at 1 logic level. This is obtained by providing two 1 input bits and enforcing 1 carry in, obtained by providing two 0 logic levels to the inputs of a *Type_B* cell. The symmetric argument of ensuring that all the evaluating devices in *Type_B* cell are conducting holds similarly. The gates of all the evaluating devices in *Type_B* cell should be at 0 logic level. This is obtained by

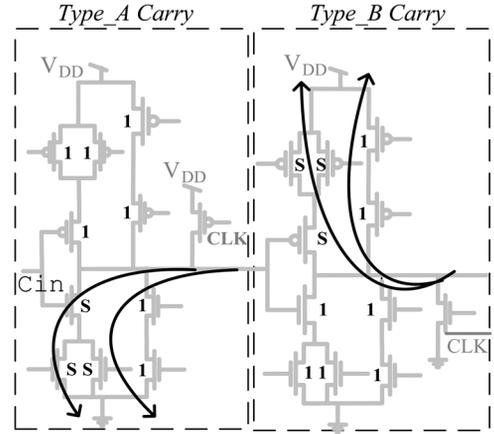


Fig. 10. Worst-case energy triggered paths in both the static-normal and dynamic-extended modes.

providing two 0 input bits and enforcing 0 carry in, obtained by providing two 1 logic levels to the inputs of a *Type_A* cell.

Consequently, according to Fig. 4 and the alternating polarity of successive FA bits, the worst input of the adder is $A[0 : n - 1] 111111 \dots 111111 B[0 : n - 1] 111111 \dots 1111$. This worst-case input applies to both the static and dynamic DML modes. However, there is a considerable difference between these cases. Whereas the energy of the static mode is consumed by the evaluation devices alone, the dynamic mode consumes additional pre-charge energy, which was measured in the experiments. It is interesting to note that the propagate signals of all the bits are 0, and therefore, since it is embedded in the pipeline, the DM² adder controller will turn it into a normal static mode. Note that although we used this input to calculate the worst dynamic mode energy consumption, in the actual pipeline this scenario will operate in a static DML mode.

To compare the energy consumption of the ordinary DMADD CMOS adder with the DM², the worst-case stimuli were used for both. For DMADD normal mode, the inputs led to the longest carry propagation through $2k - 1$ bits, whereas in the extended mode it propagated through the entire n bits. The worst stimulus for DM² is the one described previously.

Both DMADD and DM² adders were implemented in 40 nm process technology. The layout of the 64-bit DM² adder is shown in Fig. 13. It was designed with Cadence's Virtuoso tool, and extracted and simulated with SPICE. The energy measure for each mode was weighted by its corresponding probability. The results are summarized in Fig. 11, showing 36% energy reduction for the 32-bit DM² adder and 27% reduction for the 64-bit one, compared to the corresponding DMADD adders.

Recall that the motivation for DM² design was to simplify the pipeline and avoid the multi-cycle mode required by the DMADD design. In addition, there is no tradeoff in achieving the primary objective, furthermore considerable energy savings are achieved.

C. Comparison of DM² to Brent-Kung and Simple Ripple Carry Adders

Extensive experiments were carried out for both 32 bit and 64 bit adders to compare the average power per cycle (henceforth power) efficiency of DM² to a variety of adder architectures. The experiments covered three architectures at the same target frequency: High performance Brent-Kung [18],

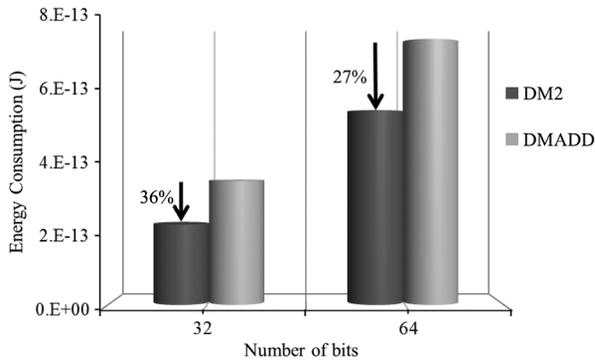


Fig. 11. Energy consumption for the 32-bit and the 64-bit adders.

Low performances Ripple carry adder and DMADD, which is the base addition architecture [10] used for DM² (results reported above). As can be seen from Table II, DM² achieved a power reduction of 1.9X–5X. All the adders were designed to meet 1 GHz performance, and their power consumption was minimized. Adders were Verilog designed and synthesized with the RTL compiler synthesis tool with the given 40 nm technology library and cadence encounter's Place&Route capabilities. Then, all designs were imported to Cadence virtuoso for spectre (SPICE) analog simulations. All adders were simulated with their worst case input transitions for power measurements and their slowest critical path frequency, as done in the previous section for DM² and DMADD.

In the first experiment, Brent-Kung architecture was used to meet 1 GHz clock frequency; all the related attributes for the synthesis tool were set to minimize the power consumption. This resulted in a power consumption of ~2.6X (593/230) and ~2.1X (1082/520) for the 32 and 64 bit designs, respectively, compared to the DM² adder. Note, however, that the Brent-Kung architecture is the only alternative if very high performance is required. For example, the maximum frequency of 2.5 GHz can be achieved by this architecture, but at a very significant power consumption cost. In this case the power consumption increased to ~12.2X and ~9.2X compared to 1 GHz DM² for 32 and 64 bit designs, respectively.

The goal of the second experiment was to compare the DM² adder to the Ripple carry adder (RCA) at 1 GHz. Unfortunately, the ripple carry adder could not achieve this design goal with the 40 nm technology (given the STD Cell library sizing factors). The maximum achievable frequencies were 370 MHz and 195 MHz for the 32 and 64 bit designs, respectively.

Nevertheless, to show that DM² is more power efficient than the ripple carry adder, it was optimally designed to meet the 370 MHz and 195 MHz frequencies (32 and 64 bit). In this case DM² design achieved a power reduction of ~4X (410/102 and ~5.8X (435/75), compared to the ripple carry adders. Note, in this case the power improvement difference between 32 and 64 bit designs (4 and 5.8) was not large since some of the DM² gates were already minimum sized given the relaxed performance specifications.

D. Mode Decision Overhead

As previously mentioned, mode decision operates in the ID Stage. In order to fully grasp the system tradeoffs, we extracted both the 32 and 64 bit control circuitry (mode decision) average power consumption and performance, which are listed

in Table III. As clearly shown in the table, the mode decision logic delay is much smaller than the clock cycles. Shown in Fig. 3, the logic use the output of the register file, which is usually not a critical path, consuming less than half clock cycle (1 GHz). The ID stage could therefore tolerate the incorporation of the decision logic, with no timing problems. Simulation results showed that the average power of the mode decision circuitry was ~20–25% of the DM² adder. Up to now the mode decision average power has not been taken into account (Table II). Although the mode decision clearly introduces power overhead, its contribution was negligible and did not disconfirm the advantages of the DM², compared to other alternatives. Table IV presents the average power dissipation of the DM² adder including the mode decision unit. As can be seen ~2X (593/291) and ~1.7X (1082/639) power reduction was achieved compared to Brent-Kung operating at 1 GHz for the 32 and 64 bit designs, respectively. Compared to RCA operating at its maximum frequency, ~2.51X (410/163) and ~2.24X (435/194) power reduction was achieved for the 32 and 64 bit designs, respectively.

E. Design Analysis Accuracy

To grasp the accuracy of the optimal DM² design analysis, the calculated energy reduction was compared to the SPICE simulation results for 32-bit and 64-bit adders. For the 32-bit adder the following parameters were measured. Note that these parameters are the delays and energy measurements per bit.

$$\begin{aligned} T_{\text{pre}} &= 10^{-10} [\text{Sec}], & E_{\text{dyn}}^{\text{DM}^2} &= 4.76 \cdot 10^{-14} [J], \\ T_{\text{eval}} &= 3.13 \cdot 10^{-11} [\text{Sec}], & E_{\text{stat}}^{\text{DM}^2} &= 6.42 \cdot 10^{-15} [J], \\ T_{\text{stat}} &= 4.68 \cdot 10^{-11} [\text{Sec}], & E_{\text{stat}}^{\text{DM}^2} + E_{\text{leakage}}^{\text{DMO}} &= 1.08 \cdot 10^{-14} [J]. \end{aligned}$$

For 64-bit adder the following parameters were measured.

$$\begin{aligned} T_{\text{pre}} &= 10^{-10} [\text{Sec}], & E_{\text{dyn}}^{\text{DM}^2} &= 4.96 \cdot 10^{-14} [J], \\ T_{\text{eval}} &= 1.56 \cdot 10^{-11} [\text{Sec}], & E_{\text{stat}}^{\text{DM}^2} &= 7.24 \cdot 10^{-15} [J], \\ T_{\text{stat}} &= 6.25 \cdot 10^{-11} [\text{Sec}], & E_{\text{stat}}^{\text{DM}^2} + E_{\text{leakage}}^{\text{DMO}} &= 1.11 \cdot 10^{-14} [J]. \end{aligned}$$

Table I shows the measured k derived from Fig. 8 and Fig. 9 for 32-bit and 64-bit adders, respectively, and the corresponding practical rounded k . Energy reductions of 36% and 27%, respectively, were achieved. Note that the computed energies are lower bounds since the practical group size k may be smaller than the computed k due to rounding.

The table shows that the energies measured by simulations fall close to those computed by (5), yielding small inaccuracies of 7.8 and 5.5 percent, respectively, for 32-bit and 64-bit adder designs.

F. Reliability

Dynamic Voltage and Frequency Scaling (DVFS) has become a popular energy reduction technique. Sensitivity to process variations has also become a major design concern. It is therefore important to verify the voltage scalability of DM² design and its sensitivity to process variations. Ideally we desire the minimum energy design point at which the group size k was determined to be invariant to the operation voltage.

Recall that the value of k is set such that the n -bit DML dynamic mode propagation delay is equated to $(2k - 1)$ -bit static mode propagation delay. We therefore desire that the delay ratio

TABLE I
COMPUTED VERSUS MEASURED COMPARISON

#Bits		K	$\frac{E^{DM^2}}{E^{DMADD}}$ Eq. (5)	$\frac{E^{DM^2}}{E^{DMADD}}$ Eq. (6)
32-bit adder	Computed	12.2	0.59	0.59
	Measured	8	0.64	
	Inaccuracy [%]		7.8	7.8
64-bit adder	Computed	9.3	0.69	0.65
	Measured	8	0.73	
	Inaccuracy [%]		5.5	11

TABLE II
PERFORMANCE, POWER, AREA AND NUMBER OF CELLS COMPARISON OF BRENT-KUNG AND RIPPLE ADDERS COMPARED TO DM²

#Bit		Brent-Kung	Brent-Kung	DM ²	DM ²	RCA
32	Power [uW]	2840	593	230	102	410
	Frequency [GHz]	2.503	1.09	1	0.3728	0.38
	Area [um ²]	824	824	400	290	488
	#Cells	435	435			130
64	Power [uW]	4800	1082	520	75	435
	Frequency [GHz]	2.142	1.06	1	0.194	0.199
	Area[um ²]	1594	1200	800	350	967
	#Cells	926	926			260

TABLE III
32 AND 64 BIT CONTROL CIRCUITRY (MODE DECISION) AVERAGE POWER CONSUMPTION AND PERFORMANCE

Clk Cycle [ps]	Delay Time [ps]	Av Power consumption [uW]	Control Circuit #bits
1000	320	61	32
1000	380	119	64

TABLE IV
AVERAGE POWER DISSIPATION OF THE DM² ADDER INCLUDING MODE DECISION UNIT

	Brent-Kung	DM ²	DM ²	RCA
32b-Power [uW]	593	291	163	410
32b-Frequency [GHz]	1.09	1	0.3728	0.38
64b-Power [uW]	1082	639	194	435
64b-Frequency [GHz]	1.06	1	0.194	0.199

should be independent of the operation voltage. The following expression shows an approximate delay ratio:

$$\begin{aligned}
 \frac{T_{eval(n)}}{T_{stat}(2k-1)} &= \frac{nC_{dyn} \int_0^{V_{DD}/2} \frac{dV}{I_{dyn}}}{(2k-1)C_{stat} \int_0^{V_{DD}/2} \frac{dV}{I_{stat}}} \\
 &= \frac{\frac{nC_{dyn} \delta_{dyn} \int_0^{V_{DD}/2} \frac{dV}{f(V_{DD})}}{\delta_{dyn}}}{\frac{(2k-1)C_{stat} \int_0^{V_{DD}/2} \frac{dV}{f(V_{DD})}}{\delta_{stat}}} \\
 &\approx \frac{nC_{dyn} \delta_{stat}}{(2k-1)C_{stat} \delta_{dyn}} \quad (9)
 \end{aligned}$$

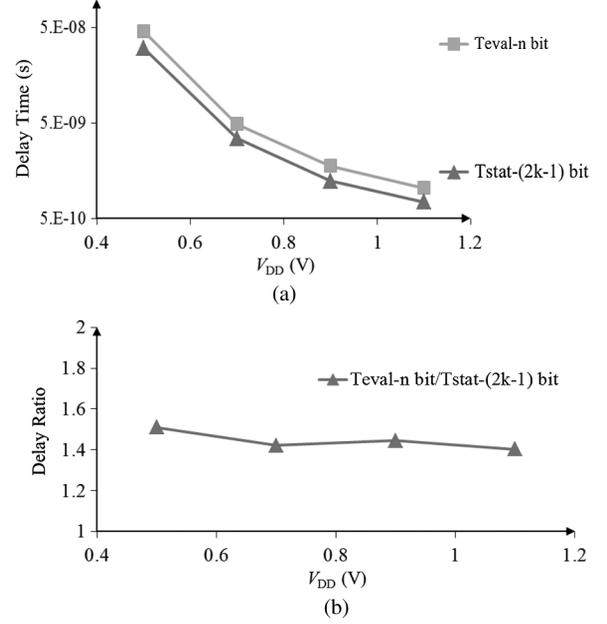


Fig. 12. (a) $n = 64$ -bit dynamic and $(2k - 1) = 16$ -bit static delays and their ratio in (b).

The constant ratio in (9) stems from the current equation $I_{dyn} = \delta_{dyn} f(V_{DD})$ and $I_{stat} = \delta_{stat} f(V_{DD})$. The factors δ_{dyn} and δ_{stat} are the current driving strength of the respective topologies, dependent solely on the device sizes and process parameters. $f(V_{DD})$ depicts the current dependency on the supply voltage, where the device is operated in one of the possible operation modes, e.g., strong inversion, near-threshold and sub-threshold.

Fig. 12(a) illustrates the n -bit dynamic and $(2k - 1)$ -bit static delays in a logarithmic scale. The two curves should theoretically coincide. They may practically be separated slightly as a result of k rounding (see Fig. 8). Fig. 12(b) depicts the ratio of these delays, and shows that it is almost constant across a wide voltage range.

To study its sensitivity to process variations, the DM² and ordinary CMOS DMADD adders were tested by running 2000 Monte-Carlo simulations for its static and dynamic modes. The results are summarized in Table V, showing a very small change in the sensitivity of the DM² adder compared to the DMADD. This is not surprising, as DML was previously shown to be robust [11], [13].

G. Area Utilization

DM² and DMADD adders were designed to compare their areas. Fig. 13 shows the layout of the DM² which were custom

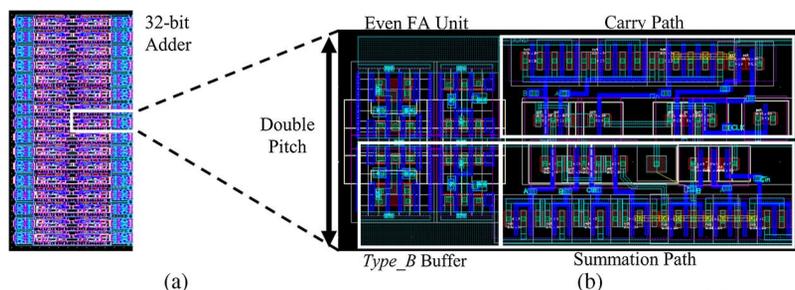


Fig. 13. Layouts of a complete DM^2 adder occupying $400 \mu m^2$, and a single FA cell in (b).

TABLE V
2000 RUNS MONTE-CARLO DELAY RESULTS

Delay	Dynamic 64 bit	Static 16 bit	DMADD CMOS 64 bit
Variance σ	131 [ps]	58 [ps]	123 [ps]

designed. The DMADD was synthesized with a Cadence Encounter RTL Compiler synthesizer. DM^2 was 32% smaller than DMADD. This follows from the smaller cell sizes of the DML family compared to the CMOS (in DML either the pull-up or the pull-down transistor network is always of minimum size).

V. CONCLUSION

A novel, low-energy and high-performance DM^2 adder combining DML logic and dual-mode addition was described. It simplifies the usage of dual-mode addition in a pipelined processor, while further reducing the computation energy by 36% to 27% for 32-bit and 64-bit adders, respectively, compared to DMADD implementation. The proposed adder achieved 32% less area and its robustness for process variations is proven. The combination of novel circuit topologies and probability-based computational circuit architecture has the potential to achieve considerably higher efficiency than traditional designs. Future work includes investigation of whether DM^2 can be employed with multipliers, which will first require determining whether multipliers have small carry probabilities.

ACKNOWLEDGMENT

The authors thank T. Paz and A. Garay for their valuable contribution with simulations and discussions.

REFERENCES

- [1] W. Kim, M. S. Gupta, G. Wei, and D. Brooks, "System level analysis of fast, per-core DVFS using on-chip switching regulators," in *Proc. HPCA*, 2008, pp. 123–134.
- [2] B. R. Zeydel, D. Baran, and V. G. Oklobdzija, "Energy-efficient design methodologies: High-performance vlsi adders," *IEEE J. Solid-State Circuits*, vol. 45, no. 6, pp. 1220–1233, Jul. 2010.
- [3] W. Shen, Y. Cai, X. Hong, and J. Hu, "An effective gated clock tree design based on activity and register aware placement," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 12, pp. 1639–1648, Dec. 2010.
- [4] J. Shinde and S. S. Salankar, "Clock gating—A power optimizing technique for VLSI circuits," in *Proc. INDICON*, 2011, pp. 1–4.

- [5] K. Roy and S. C. Prasad, *Low-Power CMOS VLSI Circuit Design*. New York, NY, USA: Wiley, 2009.
- [6] M. Alioto, "Ultra-low power VLSI circuit design demystified and explained: A tutorial," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 1, pp. 3–29, Jan. 2012.
- [7] D. Bol, "Robust and energy-efficient ultra-low-voltage circuit design under timing constraints in 65/45 nm CMOS," *J. Low Power Electron. Appl.*, vol. 1, pp. 1–19, 2011.
- [8] H. Zhang and J. Rabaey, "Low-swing interconnect interface circuits," in *Proc. Int. Symp. Low Power Electron. Design*, 1998, pp. 161–166.
- [9] J. Seo, D. Sylvester, D. Blaauw, H. Kaul, and R. Krishnamurthy, "A robust edge encoding technique for energy-efficient multi-cycle interconnect," in *Proc. Int. Symp. Low Power Electron. Design*, 2007, pp. 68–73.
- [10] S. Wimer, A. Albeck, and I. Koren, "A low energy dual-mode adder," *Comput. Electrical Eng.*, vol. 61, no. 5, pp. 1524–1537, Jul. 2014.
- [11] A. Kaizerman, S. Fisher, and A. Fish, "Subthreshold dual mode logic," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 5, pp. 979–983, May 2013.
- [12] I. Levi, A. Belenky, and A. Fish, "Logical effort for CMOS-based dual mode logic gates," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 5, pp. 1042–1053, May 2013.
- [13] I. Levi and A. Fish, "Dual mode logic—Design for energy efficiency and high performance," *IEEE Access*, vol. 1, pp. 258–265, 2013.
- [14] I. Levi, O. Bass, A. Kaizerman, A. Belenky, and A. Fish, "High speed dual mode logic carry look ahead adder," in *Proc. ISCAS*, 2012, pp. 3037–3040.
- [15] K. C. Yeager, "The MIPS R10000 superscalar microprocessor," *IEEE Micro*, vol. 16, pp. 28–41, 1996.
- [16] N. H. Weste and D. Money, *CMOS Vlsi Design*. New York, NY, USA: Pearson/Addison Wesley, 2005.
- [17] B. Pahami, *Computer Arithmetic: Algorithms and Hardware Designs*. Oxford, U.K.: Oxford Univ. Press, 2009.
- [18] R. Brent and H. Kung, "A regular layout for parallel adders," *IEEE Trans. Comput.*, vol. C-31, no. 3, pp. 260–264, Mar. 1982.
- [19] H. Q. Dao, B. R. Zeydel, and V. G. Oklobdzija, "Energy optimization of pipelined digital systems using circuit sizing and supply scaling," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 14, no. 2, pp. 122–134, 2006.
- [20] S. J. Wilton, S. S. Ang, and W. Luk, "The impact of pipelining on energy per operation in field-programmable gate arrays," in *In Field Programmable Logic and Application*. New York, NY, USA: Springer Berlin Heidelberg, 2004, pp. 719–728.
- [21] I. Levi, A. Kaizerman, and A. Fish, "Low voltage dual mode logic: Model analysis and parameter extraction," *Microelectronics J.*, vol. 44, no. 6, pp. 553–560, 2013.



Itamar Levi received the B.Sc. and M.Sc. degree in electrical and computer engineering from Ben-Gurion University, Beer-Sheva, Israel, in 2012 and 2013, respectively. Currently, he is pursuing the Ph.D. degree in electrical engineering from Bar-Ilan University, Ramat Gan, Israel.

From 2011, he has been working at the VLSI Systems Center, Ben-Gurion, where he was responsible for various aspects of VLSI systems design: low-energy design, dual mode logic family and digital systems optimization. His research interests are hardware security and cryptography.



Amir Albeck received the B.S. degree from Bar-Ilan University, Ramat Gan, Israel, where he is currently pursuing the M.S. degree in computer engineering.

His research interests are in low power arithmetic calculation techniques.



Alexander Fish received the B.Sc. degree in electrical engineering from the Technion, Israel Institute of Technology, Haifa, Israel, in 1999. He received the M.Sc. in 2002 and the Ph.D. (*summa cum laude*) degree in 2006, respectively, at Ben-Gurion University, Beer-Sheva, Israel.

He was a postdoctoral fellow in the ATIPS laboratory at the University of Calgary, Canada, from 2006—2008. In 2008 he joined the Ben-Gurion University in Israel, as a faculty member in the Electrical and Computer Engineering Department. There

he founded the Low Power Circuits and Systems (LPC&S) laboratory, specializing in low power circuits and systems. In July 2011 he was appointed as a head of the VLSI Systems Center at BGU. In October 2012 Prof. Fish joined the Bar-Ilan University, Faculty of Engineering as an Associate Professor and the head of the nanoelectronics track. He also leads new Emerging Nanoscaled Integrated Circuits and Systems (ENICS) Labs. His research interests include development of secured hardware, ultra low power SRAM, DRAM and Flash memory arrays, CMOS image sensors and energy efficient design techniques

for low voltage digital and analog VLSI chips. He has authored over 70 scientific papers in journals and conferences, including IEEE Journal of Solid State Circuits, IEEE TRANSACTIONS ON ELECTRON DEVICES, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS and many others. He also submitted 21 patent applications. He has published two book chapters.

Prof. Fish was a co-author of papers that won the Best Paper Finalist awards at IEEE ISCAS and ICECS conferences. Prof. Fish serves as an Editor in Chief for the MDPI Journal of Low Power Electronics and Applications (JLPEA) and as an Associate Editor for the IEEE SENSORS, IEEE ACCESS, Elsevier *Microelectronics and Integration*, the VLSI Journals. He also served as a chair of different tracks of various IEEE conferences. He was a co-organizer of many special sessions at IEEE conferences, including IEEE ISCAS, IEEE SENSORS and IEEE conferences. Prof. Fish is a member of Sensory, VLSI Systems and Applications and Bio-medical Systems Technical Committees of IEEE Circuits and Systems Society.



Shmuel Wimer received the B.Sc. and M.Sc. degrees in mathematics from Tel-Aviv University, Tel-Aviv, Israel, and the D.Sc. degree in electrical engineering from the Technion-Israel Institute of Technology, Haifa, Israel, in 1978, 1981 and 1988, respectively.

He worked for 32 years at industry in R&D, engineering and managerial positions, for Intel from 1999 to 2009, and prior to that for IBM, National Semiconductor and Israeli Aerospace Industry (IAI). He is presently an Associate Professor with the Engi-

neering Faculty of Bar-Ilan University, and an Associate Visiting Professor with the Electrical Engineering Faculty, Technion. He is interested in VLSI circuits and systems design optimization and combinatorial optimization.