Energy-Efficient GHz-Class Charge-Recovery Logic

Visvesh S. Sathe, *Member, IEEE*, Juang-Ying Chueh, *Member, IEEE*, and Marios C. Papaefthymiou, *Senior Member, IEEE*

***Abstract*—In this paper, we present Boost Logic, a charge-recovery circuit family that can operate efficiently at clock frequencies in excess of 1 GHz. To achieve high energy efficiency, Boost Logic relies on a combination of aggressive voltage scaling, gate overdrive, and charge-recovery techniques. In post-layout simulations of 16-bit multipliers with a 0.13-μm CMOS process at 1 GHz, a Boost Logic implementation achieves 5 times higher energy efficiency than its minimum-energy pipelined, voltage-scaled, static CMOS counterpart at the expense of 3 times longer latency. In a fully integrated test chip implemented using a 0.13-μm bulk silicon process and on-chip inductors, chains of Boost Logic gates operate at clock frequencies up to 1.3 GHz with a 1.5-V supply. When resonating at 850 MHz with a 1.2-V supply, the Boost Logic test chip achieves 60% charge-recovery.**

***Index Terms*—Adiabatic, charge-recovery, energy recovery, resonant systems.**

I. Introduction

POWER minimization has become a primary concern in VLSI design. Several conventional techniques are utilized to curb dynamic and leakage power in conventional CMOS circuits. One of the most effective methods is pipelining and subsequent voltage scaling to minimize energy dissipation at a given operating frequency. At high operating frequencies, however, the energy and delay overhead of pipeline registers becomes significant and degrades overall system efficiency.

In systems with significant switching activity, charge-recovery circuits have the potential to dissipate less energy than their pipelined, voltage-scaled CMOS counterparts. Several charge-recovery logic styles have been proposed [1]–[5]. Over a range of relatively low operating frequencies (a few hundred megahertz), these charge-recovery techniques have been shown to achieve lower energy dissipation when compared to voltage-scaled CMOS. Achieving energy savings over CMOS at higher operating frequencies has remained elusive, however.

Although performance limits of charge-recovery circuits are fundamentally determined by the need for gradually transitioning power-clocks, prevalent operating frequencies in charge-recovery circuits are more a consequence of design than any such fundamental constraint. Some of the main factors that lead to lower speeds in charge-recovery circuits are the use of diode-connected transistors [6], [7], the use of pMOS devices in evaluation trees [8], [9], and the excessive time required to resolve the complementary outputs of the dual-rail gates during evaluation [2], [4], [10].

In this paper, we present a novel dynamic charge-recovery logic family called Boost Logic [11]. Boost Logic achieves significant energy savings over voltage-scaled static CMOS across a range of frequencies much higher than currently demonstrated in charge-recovery literature. A unique feature of Boost Logic gates that enables energy-efficient and high-throughput operation is an aggressively scaled, conventionally switching "Logic" stage that operates in tandem with a charge-recovery "Boost" stage. Logic performs the logical evaluation of a Boost Logic gate operating at an ultra-low DC supply voltage of approximately one threshold voltage, V_{th}. After Logic pre-resolves the differential outputs of a Boost Logic gate to the level of about one threshold voltage, Boost amplifies the difference between the outputs nodes to the full rail in an energy-efficient charge-recovery manner, providing a large gate overdrive to fanout gates and thereby reducing delay in their Logic stages. Thus, Boost Logic achieves lower energy dissipation without incurring the performance degradation typical of conventional voltage-scaled designs.

Fig. 1(a) illustrates the concept behind Boost Logic. Each Boost Logic gate consists of two parts operating in tandem over nonoverlapping time intervals: A conventionally switching logical evaluation stage (Logic) and a charge-recovering stage (Boost). Fig. 1(b) shows simplified voltage waveforms of a Boost Logic gate output. In the first phase of its operation, Logic resolves the output nodes to supply rails V'_{dd} and V'_{ss}. In the second phase of its operation, Boost amplifies this voltage difference between the outputs by making them track complementary resonating clock signals ϕ and $\overline{\phi}$, oscillating with peak voltage V_{dd}. These clocks will henceforth be referred to as *power-clocks*. This full-rail swing provides fanout Logic stages with a gate overdrive of $V_{gs} - V_{th} \approx (V_{dd} - V_{th})/2$, allowing them to perform evaluation at frequencies much higher than expected of such aggressively voltage-scaled logic. Although Boost enables aggressive voltage scaling in Logic, significantly reducing energy dissipation, it is vital that the power dissipation of Boost itself does not nullify these advantages. To that end, an initial voltage difference is provided to Boost by Logic, greatly aiding its sense-amplifying action, and resulting in efficient charge-recovery.

Although previously proposed logic families have used the idea of increased gate overdrive through the use of bootstrapping techniques [3], [12], these methods lack the robustness offered by the use of a Boost stage. These methods are also limited in the amount of achievable gate overdrive. More recently, LVS logic has been proposed [13], where sense amplifiers are used to amplify low-swing gate outputs.

Manuscript received April 21, 2006; revised August 10, 2006. This work was supported in part by the U.S. Army Research Office under Grant DAADA19-03-1-0122.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2121 USA (e-mail: vssathe@eecs.umich.edu).

Digital Object Identifier 10.1109/JSSC.2006.885053

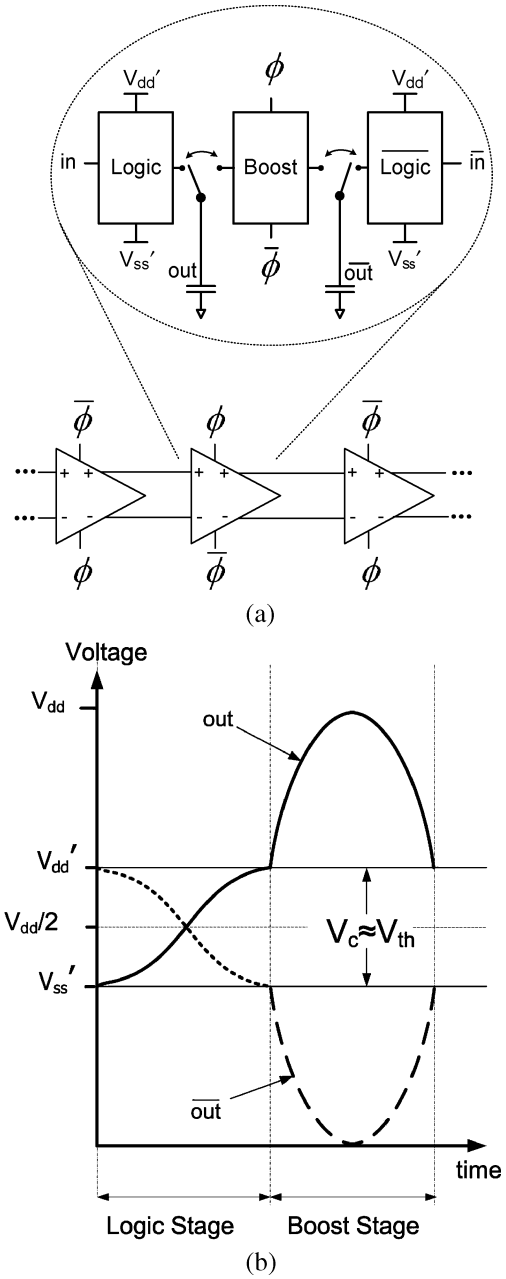


Fig. 1. Boost Logic. (a) Cascade. (b) Simplified waveform.

We have characterized the performance and energy-efficiency of Boost Logic through extensive simulations. Specifically, we have explored the robustness of Boost Logic gates to clock skew. We have also investigated the effect of power-supply variation on the energy and performance of Boost Logic. To comparatively assess the energy efficiency of Boost Logic, we have implemented 16-bit carry-save multipliers in both Boost Logic and pipelined, voltage-scaled static CMOS. Both multipliers have been designed for operation at 1 GHz. In post-layout simulations with extracted parasitics, Boost Logic achieves approximately 5 times higher energy efficiency than its minimum-energy static counterpart, although at the cost of 3 times longer latency.

Boost Logic derives significant performance improvements from the use of low V_{th} devices in its Logic stage. Post-layout simulations of a 16-bit multiplier with low V_{th} devices result in a 33% reduction in latency and a 32.6% reduction in energy dissipation as compared to Boost Logic with regular V_{th} devices.

Beyond simulations, in this paper we provide measurements from a fully-integrated test chip that demonstrates the operation of Boost Logic gate cascades at operating frequencies exceeding 1 GHz. We present chip measurements obtained over a range of frequencies in the neighborhood of its natural frequency. Moreover, we present measurements exploring the trade-offs involved in on-chip clock generation and the trade-offs between the DC supply and power-clock voltages with regard to overall chip efficiency. When operating at its resonant frequency of 850 MHz, the Boost Logic chip achieves 60% recovery in its resonant portion. Driven 17% off resonance at 1 GHz, it still recovers 40% of its resonating energy.

The remainder of the paper is organized as follows. In Section II, we present Boost Logic and discuss its structure and operation. We explain how the Boost stage achieves significant charge-recovery at high frequencies. The results of extensive simulations investigating the sensitivity of Boost gates to power-supply fluctuation and clock skew are discussed in Section III. In Section IV, we compare the energy and throughput of a 16-bit Boost Logic carry-save multipliers with its voltage-scaled pipelined CMOS counterpart. We also demonstrate the performance benefits obtained from the use of low V_{th} devices through simulation results obtained on a 16-bit Boot Logic multiplier implemented with low V_{th} devices. In Section V, we present the Boost Logic test chip along with measurements obtained. Conclusions are given in Section VI.

II. Boost Logic Basics

In this section, we first analyze the structure and operation of Boost Logic. We subsequently consider the energy and delay equations that govern the operation of Boost Logic and show how Boost Logic achieves high throughput with significant energy savings.

A. Structure

Fig. 2 shows the structure of a Boost Logic gate. Boost Logic is a two-phase, dual-rail, partially charge-recovering logic. The structure of a Boost gate can be divided into two parts—logical evaluation ("Logic") and charge-recovering amplification ("Boost"). Logic can be implemented in any transistor topology as long as it supports the use of the clocked transistors M5–M8. These clocked transistors de-couple Logic from the output nodes when Boost drives them. Depending on the gate implementation and the operating frequency, we have found the use of the clocked CMOS or pseudo-nMOS logic styles (shown in Fig. 2) to be particularly effective. The implementation of the clocked pseudo-nMOS logic evaluation trades off the voltage difference in the pre-resolved output nodes (pseudo nMOS gates do not swing to the full rail) for lower gate loading to achieve better performance at higher frequencies. In this implementation, transistors M6 and M8 shown in Fig. 2 also perform the role of pull-up load devices for the logic gate when conducting. At lower operating frequencies, the use of a dual-rail CMOS topology in Logic offers the advantages of full-rail evaluation, the absence of crowbar current

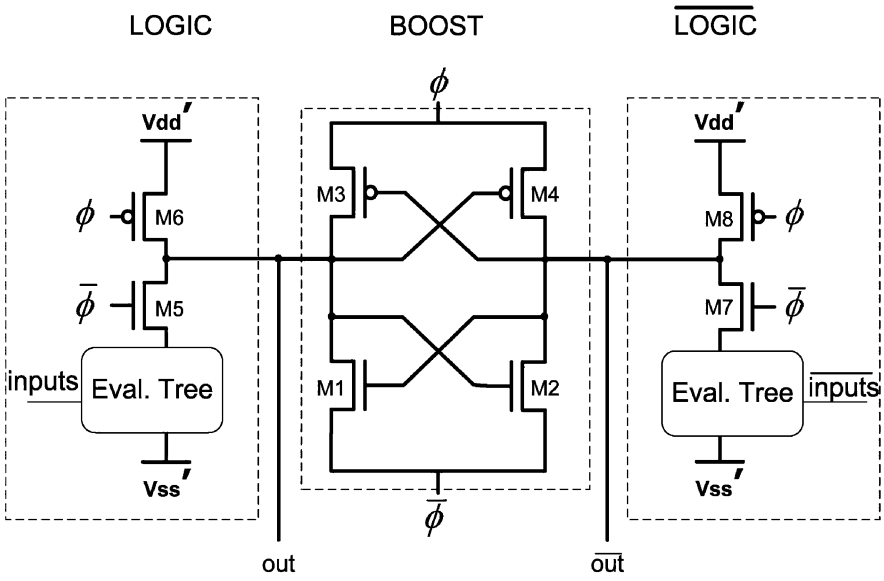


Fig. 2. Boost logic gate structure.

in Logic, and reduced susceptibility to process variation. The DC power-supply rails are at voltages

$$V'_{dd} = \frac{1}{2} \cdot (V_{dd} + V_{th}) \quad (1)$$

$$V'_{ss} = \frac{1}{2} \cdot (V_{dd} - V_{th}). \quad (2)$$

This choice of voltage values is motivated by the operation of Boost, which will be discussed in greater detail in Section II-B. The potential difference between the voltage supply rails in Logic is therefore $V_c \approx V_{th}$. As seen from Fig. 2, Boost resembles back-to-back CMOS inverters. The only difference is that the V_{dd} and Gnd rails are replaced by ϕ and $\overline{\phi}$.

B. Operation

This section describes Boost Logic operation. We show how the operation of Boost and Logic stages in tandem yields efficient charge-recovery at operating frequencies significantly higher than demonstrated in the literature.

Fig. 3(a) shows Boost Logic inverters in a ring configuration, while Fig. 3(b) illustrates the voltage waveforms at node out. By design, the Logic and Boost stages evaluate at mutually exclusive intervals. As such, when Logic evaluates, Boost does not drive the outputs and vice versa. Consider the operation of the gate whose waveforms are shown in Fig. 3(b). When Logic evaluates (ϕ falls and $\overline{\phi}$ rises), the header transistors M_6 and M_8 and footer transistors M_5 and M_7 turn on (see Fig. 2). As out evaluates high, the header transistor M_6 pulls the output node to V'_{dd}. The complementary output discharges through the evaluation tree to nearly V'_{ss}. At this time, the charge-recovering sense amplifier is in pre-charge with $\phi = 0$ and $\overline{\phi} = V_{dd}$. In this state, it is easily verified that as long as the outputs stay within the DC supply rails, all Boost transistors are off, and no current flows in any Boost stage devices. As ϕ begins to rise past V'_{ss}, Logic is deactivated and disconnected from V'_{dd} and V'_{ss}. As ϕ continues to rise past V'_{dd}, Boost turns on. Since out is at V'_{dd} and $\overline{out}$ at nearly V'_{ss}, transistors M_2 and M_3 turn on as $\overline{\phi}(\phi)$ goes

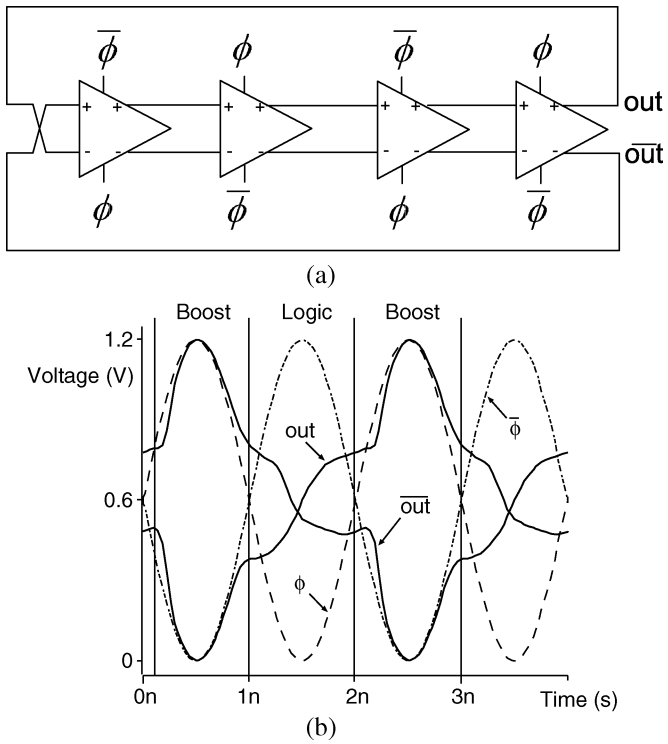


Fig. 3. Boost Logic inverter. (a) Cascade and (b) voltage waveform at nodes out and $\overline{out}$.

past V'_{ss} (V'_{dd}), causing $\overline{out}$ (out) to subsequently follow $\overline{\phi}$ (ϕ). As the voltage difference between out and $\overline{out}$ increases, transistors M_2 and M_3 turn more strongly on, further reducing the voltage difference across these current-carrying transistors. Finally, out ($\overline{out}$) reach the rails V_{dd} (Gnd). These outputs drive fanout Logic stages.

As ϕ and $\overline{\phi}$ transition once again, entering the next logic phase, the outputs track the corresponding complementary clocks through the same transistors M_2 and M_3. As the voltage difference between out and $\overline{out}$ approaches V_{th}, all four Boost

transistors are in cutoff, isolating Boost from the outputs, and Logic once again begins to evaluate.

Boost Logic achieves charge-recovery at high frequencies due to several design features. First, Boost does not require diodes to perform charge-recovery and can therefore operate efficiently at relatively higher frequencies. Although an option, Boost Logic does not require the use of pMOS evaluation trees in the Logic stage, greatly reducing capacitive loading of gate inputs and improving performance. By not having to follow the power-clock when it transitions at its fastest rate ($V_{dd}/2$ for sinusoidal clocks), energy dissipation in Boost is reduced. This form of pre-charge also provides more time for the Logic to evaluate as compared to charge-recovery designs that pre-charge to nearly V_{dd} or Gnd.

The most significant feature in Boost Logic which enables efficient charge-recovery is the (scaled) voltage difference between the nodes out and $\overline{out}$. This voltage difference greatly improves the efficiency of the sense-amplifying action of Boost Logic. The presence of a nearly uniform voltage difference between the output nodes regardless of data inputs at the onset of the Boost stage results in a data-independent capacitance seen by the clock generator, minimizing data-induced jitter. Previous charge-recovery logic designs did not use pre-resolved output nodes, resulting in a degradation of both recovery efficiency and data-dependent clock jitter performance. Note that even if $V_c < V_{th}$, the Logic stages are not operating in the subthreshold region. The gate overdrive provided by Boost keeps Logic devices in the super-threshold linear region.

As shown in Fig. 3(a), cascading Boost Logic gates is straightforward. Since the boost conversion of a gate is required to occur concurrently with the logic evaluation stage in its fanout gates, Boost Logic gates are cascaded by driving the Boost stages with alternating clock phases ϕ and $\overline{\phi}$. Notice that from a timing (and to a large extent, functional) perspective, Boost Logic consists of a logic gate driving a level-converting latch.

C. Energy and Delay

In this section we consider the equations that govern the energy dissipation and the Logic delay of a Boost Logic gate. We also highlight the low delay variation of Boost Logic upon scaling V_c.

Given that the transistors in the evaluation tree conduct in the linear mode, the delay δ in the Logic stage of a Boost Logic gate can be approximated as

$$\delta \propto \frac{C \cdot V_c}{\left[\left(\frac{V_{dd}}{2} + \frac{V_C}{2} - V_{th}\right) V_C - \frac{1}{2} V_c^2\right]} \tag{3}$$

where V_c is the voltage swing of the gate, and V_{dd} is the amplitude of the power-clock. This equality can then be rewritten as

$$\delta \propto \frac{C}{\frac{V_{dd}}{2} - V_{th}}. \tag{4}$$

Considering first-order transistor effects, this result implies that Logic delay is independent of supply voltage V_c. Indeed, the extent to which this energy-delay correlation can be exploited to reduce supply voltage is determined by noise considerations and Boost efficiency.

To analyze the energy dissipation of Boost, we make simplifying assumptions. First, it is assumed that the output waveforms are sinusoidal as they track the power-clocks over a voltage swing of $(V_{dd} - Vc)/2$. The resistance offered by the conducting devices (operating in the linear region) during charge transfer in each output node depends on the voltage of the complementary output node. However, it is assumed that for a given clock voltage waveform, the conducting device can be replaced by an appropriate equivalent time-invariant resistance R so that the I^2R dissipation of the gate remains unchanged. Furthermore, it is assumed that the gate is designed so that the on-resistance of the nMOS devices equals the on-resistance of the pMOS devices in the Boost stage.

Under the above assumptions, considering two identical gates operating at opposite clock phases, the current flowing through any clock phase can be estimated to be sinusoidal with an amplitude of $j\omega C(V_{dd} - Vc)/2$. The energy dissipation of the two gates over one cycle is then equal to the I^2R dissipation incurred in both clock phases. The dissipation of a single Boost stage can then be estimated to be

$$E_{\text{boost}} \approx \frac{\pi^2 \tau}{2T} C_{\text{boost}} (V_{dd} - V_C)^2 \tag{5}$$

where V_{dd} is the amplitude of the power-clock, T is the clock period, and $\tau = RC_{\text{boost}}$ is the product of the resistance of Boost looking into a power-clock terminal, and the total capacitance driven by the gate. Since $V_c \approx V_{th}$ by design, for the pseudo-nMOS Logic stage implementation of Boost Logic, the total energy dissipation of the gate per cycle can be rewritten as

$$E \approx k \cdot C V_{th}^2 + \frac{\pi^2 \tau}{2T} C_{\text{boost}} (V_{dd} - V_{th})^2 \tag{6}$$

where C is the total capacitive load driven by the gate. The coefficient k, for the energy dissipation in Logic, is greater than 1/2 due to the crowbar current that flows in the output rail that evaluates to V'_{ss}. The difference in the power dissipation of the pseudo-nMOS Logic when the output toggles and when it remains unchanged is a function of the relative sizing of the pull-up and pull-down devices. Simulations of implemented pseudo-nMOS logic gates indicate a difference of less than 20% between the energy dissipation of switching and non-switching gates. As such, (6) does not account for the switching activity of the gate. If a CMOS topology were used for Logic, then $k = s/2$, where s is the switching activity of the gate. The energy dissipation of Boost is independent of the switching activity since amplification is performed every cycle. Equation (5) is a good approximation of the actual energy dissipation, because the gate outputs follow the power-clock closely and do not contain any additional energy dissipation terms due to diode drops in the gate.

III. Robustness to Variation

In this section, we present simulations performed to explore the robustness of Boost Logic to variation. More specifically, in Section III-A we investigate the robustness of Boost Logic to

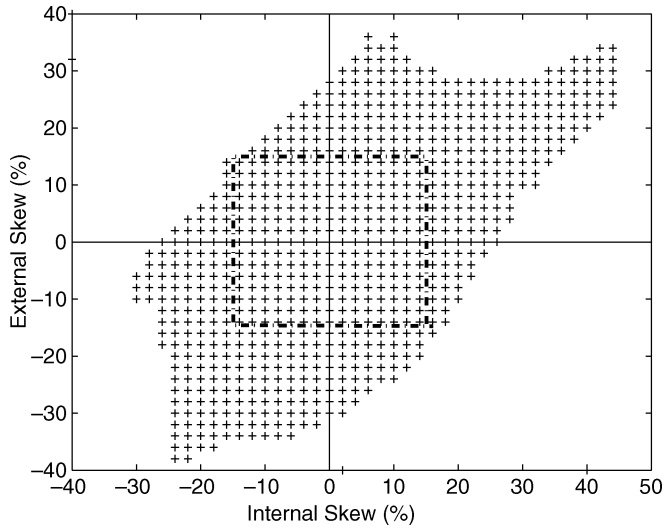


Fig. 4. Shmoo plot for functional correctness over a range of internal and external skew values.

clock skew. In Section III-B, we consider the delay variation in Boost Logic as a result of power-supply variation.

A. Clock Skew

Boost gates depend on the power-clock for driving Boost as well as providing timing information for the correct operation of the gate. Robustness to clock skew is therefore a strict requirement for fine-grained charge-recovery logic.

In a cascade of gates, the phase difference between the power-clock of a given gate and that of its fanout can affect the energy efficiency and functionality of the charge-recovery gate. We refer to this kind of clock skew as *external* clock skew. Since Boost Logic requires two complementary clock phases to perform any computation, correct operation can be compromised by a phase difference between ϕ and $\overline{\phi}$ for any given gate. We refer to such a phase difference between ϕ and $\overline{\phi}$ as *internal* skew.

To determine the robustness of Boost gates to both kinds of skew, we evaluated a parallel arrangement of AND, OR, XOR, and INV gates, providing them with random inputs and verifying functional correctness in each gate over varying amounts of both types of clock skew. The clock signals used in the experiments were forced signals and a FO4 load was applied to each gate. Simulations were carried out over the range of different internal skew and external skew values from -45% to $+45\%$ of the clock period.

Fig. 4 shows the shmoo plot obtained. The points marked "+" indicate that all gates operated correctly at the corresponding values of internal and external skew. The skew values are given as a percentage of the cycle time. It can be inferred from Fig. 4 that Boost Logic operates correctly over a large range of possible conditions of internal and external skew. In particular, all Boost Logic gates simulated correctly under simultaneous internal and external skew, each amounting to 15% of the clock cycle.

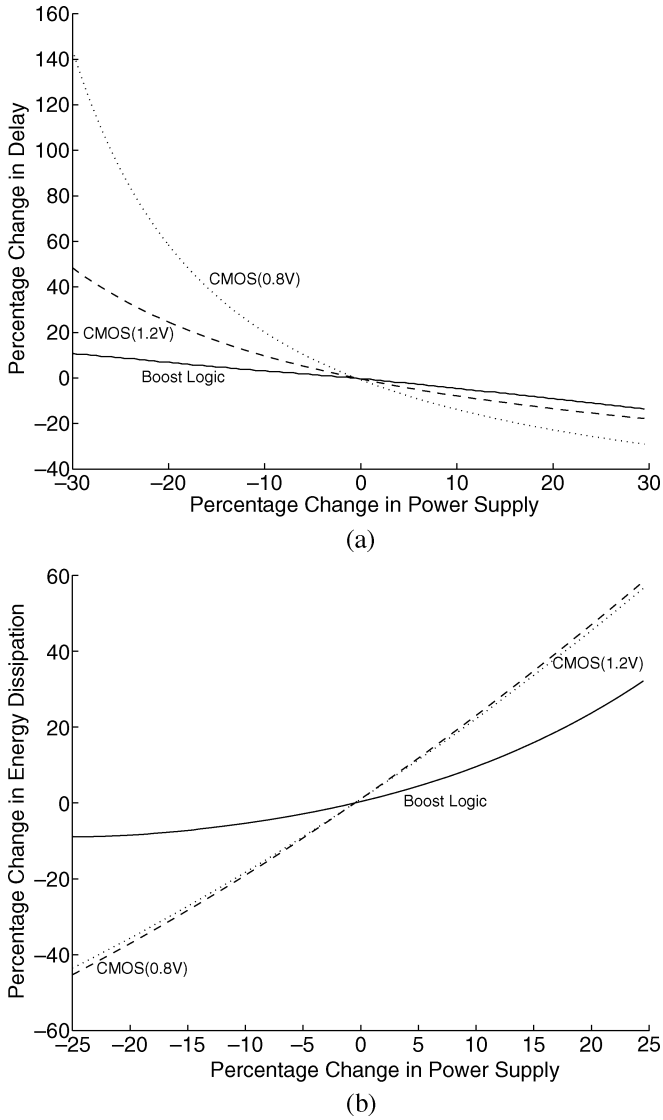


Fig. 5. Effect of power-supply variation from nominal values on (a) delay (b) energy dissipation in boost and CMOS NAND gates.

B. Power-Supply Variation

The sensitivity of Boost Logic to power-supply variation is an important property from an operational standpoint. Boost Logic is powered by two supplies: the two-phase power-clock and the ultra-low DC supply voltage. Voltage fluctuation in either power supply affects the performance of Boost Logic. From (4), it can be inferred that the Logic delay is independent of V_c and inversely proportional to V_{dd}. While the sensitivity of Logic delay to V_{dd} variation is to be expected, its predicted insensitivity to V_c needs further verification. To that end, we have evaluated the delay of a Boost Logic NAND gate when driving another identical gate in the presence of variation in V_c.

Fig. 5 illustrates the effect of power-supply variation on the delay and power dissipation of Boost Logic and CMOS at supply voltages of 1.2 V and 0.8 V. This experiment measures the percentage change in delay as the DC supply of the CMOS and Boost NAND gates is varied over a range of $\pm 30\%$. Results show that the corresponding Boost Logic delay and energy dissipation vary over the range $[-13\%, +12\%]$ and

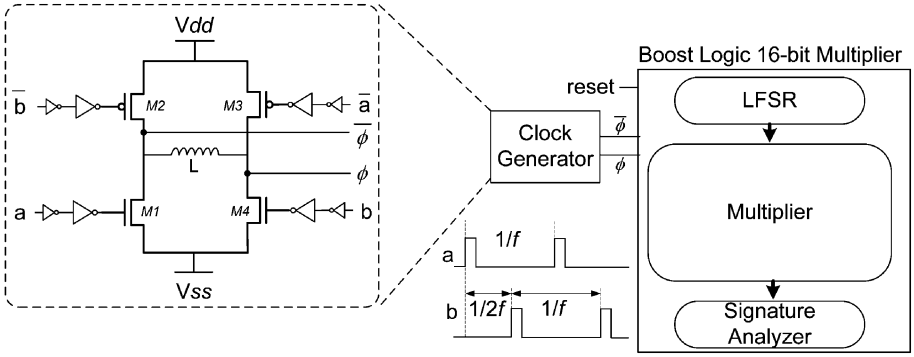


Fig. 6. Simulation setup for 16-bit carry-save multiplier.

[−10%,+30%], respectively. This variation in delay and energy dissipation is significantly lower than that observed in CMOS, in spite of the near-threshold DC supply in Boost Logic.

IV. Simulation Results

In this section, we present and compare results obtained from post-layout simulation of regular and low V_{th} 16-bit Boost Logic multipliers and their regular V_{th} static CMOS counterpart. The multipliers were designed in a 0.13-μm CMOS process with $V_{th} = 340$ mV.

An equitable comparison between Boost Logic and voltage-scaled CMOS cannot be performed on a gate-by-gate basis due to the vastly different implementations of the two design methodologies. While Boost Logic performs two logic evaluations per cycle, several conventional static CMOS gates fit in a single pipeline stage. Also, whereas Boost Logic implementations ideally consist of large complex gates, conventional CMOS gates rarely employ such gates in high-frequency designs. Furthermore, the very structure of Boost Logic gates (and charge-recovery gates in general) obviates the use of timing elements such as flip-flops. On the other hand, conventional CMOS designs are heavily pipelined in order to achieve higher operating frequencies and aggressive voltage-scaling. These timing elements present a timing and energy overhead that would not be accounted for in a gate-to-gate comparison.

An objective comparison between the two design methodologies can be made by considering design implementations. To that end, we designed 16-bit carry-save multipliers and their accompanying built-in self-test (BIST) logic in Boost Logic and static CMOS. As shown in Fig. 6, a linear feedback shift register (LFSR) was used to provide pseudo-random input vectors to the multiplier. The average switching activity at the output nodes of the LFSR was 50%. Multiplier outputs were processed by a signature analyzer. In the Boost multiplier, power-clock signals ϕ and $\overline{\phi}$ were derived using an H-Bridge clock generator. Pulses a and b were used to control switches that replenish the energy in the clock generator. Being a periodically driven oscillator, no separate start-up circuitry was required, and stable oscillations in the multiplier were observed within two cycles. The total capacitance driven by the clock generator (including the parasitic capacitance of the interconnect, inductor and switches) was approximately 18 pF per phase. The inductor parameters used in the experiment were $L = 2.55$ nH and $R_L = 400$ mΩ. The network impedance per phase was estimated at 600 mΩ per phase. The multiplier itself was implemented in a carry-save architecture. The final addition was also carried out using a carry-save adder.

To compare the energy efficiency of Boost Logic with respect to CMOS, an industrial tool was used to synthesize a pipelined multiplier. The synthesis tool was allowed to size the gates in the design, but the multiplier was constrained to the same carry-save topology as the Boost design. The tool was free to design its own final adder stage for the multiplier. The minimum-energy pipeline depth of the multiplier for 1-GHz operation was found to be eight stages. Fewer pipeline stages resulted in higher operating voltages and higher dissipation. On the other hand, the operating voltage reduction obtainable from deeper pipelining was unable to offset the increased switching capacitance in the design. Physical design was performed using a commercial place-and-route tool at a 75% row utilization.

The use of low V_{th} devices in Logic yields a higher gate overdrive, thus enabling quicker logic evaluation, and increases the amount of time available for logic evaluation. Thus, V_{th} devices enable substantially deeper Logic stacks to be implemented, reducing the latency and energy dissipation of the design. To substantiate this claim, we implemented 16-bit carry-save multiplier using devices with $V_{th} = 50$ mV. Leveraging the higher gate overdrive obtained from low V_{th} devices, complex gates were implemented so that two carry-save stages fit in a single cycle. The corresponding low V_{th} CMOS multiplier design was not implemented due to the absence of timing libraries for synthesis.

Table I presents results obtained from post-layout simulation of the three multipliers: the pipelined, voltage-scaled static CMOS (CMOS), regular V_{th} Boost Logic (Boost(rvt)) and the low V_{th} Boost Logic (Boost(lvt)). All Boost simulations include the energy dissipation in the multipliers as well as energy dissipated in clock generation and distribution. An on-chip inductor was designed to yield an operating frequency of 1 GHz, and an extracted 13-element RLC model was used for simulating the inductor along with the clock generator and the Boost circuitry. Similarly, the energy reported in the conventional multiplier includes energy dissipation of the logic, the flip-flops as well as the clock tree. For both Boost designs, the reported energy dissipation in the switches accounts for the energy dissipation incurred in driving the clock generator switches as well as the losses incurred in the resistance of the clock-generator switches during current injection. The energy dissipation in driving the clock

TABLE I
SIMULATION RESULTS OF 16-BIT CARRY-SAVE MULTIPLIER

	Static CMOS	Boost(rvt)	Boost(lvt)
Frequency (GHz)	1.0	1.0	1.0
V_{dd} (V)	1.0	1.04	1.02
Pipeline depth	8	24	16
Area(mm^2)	0.144	0.095	0.058
Energy per cycle (pJ)			
Logic	27	2.64	2.5
Boost/Flip-Flops	49.9	5.62	3.98
Clock Distribution	3.2	1.91	1.01
Clock Generator	N/A	5.62	3.15
Total	80.1	15.79	10.64

generator switches was estimated by considering the dissipation of a buffer chain with a tapering factor of 3, including parasitics.

Table I shows that at 1 GHz, the pipelined, voltage-scaled design requires a 1-V supply for correct operation, with clock buffers and flip-flops dominating overall power dissipation. The computational latency of the design is eight cycles. In Boost(rvt), the two most significant sources of dissipation are the dissipation in Boost stages and the losses incurred in the clock generator. The dissipation of the voltage-scaled Logic is less than half that of the Boost stages. Although the computational latency of Boost(rvt) is 24 cycles (3 times longer than CMOS), the overall energy dissipation of Boost(rvt) is 5 times lower (80.1 pJ versus 15.8 pJ). For Boost(lvt), the major sources of dissipation are the Boost stages and the clock generator. Boost(lvt) achieves additional energy savings over Boost(rvt) through the reduction of switching capacitance. The reduced area of Boost(lvt) also contributes to a reduction in clock network dissipation. Notice the low clock network dissipation in the Boost designs, despite a significantly higher current flowing in these designs as compared to the conventional design. This reduced dissipation is a consequence of the resonating power-clocks.

V. BOOST LOGIC TEST CHIP

In this section, we discuss a Boost Logic test chip and present measurement results obtained from the test chip. We also present results pertaining to the efficient operation of the H-Bridge clock generator for charge-recovery logic.

A. Test Chip Modules

The Boost Logic test chip is the first demonstration of a fully integrated charge-recovery logic. Implemented in a 0.13-μm CMOS logic process, the test chip operates correctly at frequencies of up to 1.3 GHz. In addition to the required power supplies, the test chip also features independent body bias supplies for the well and substrate contacts to provide more flexibility in experimentation. An on-chip clock generator is used in an H-Bridge topology to generate the complementary sinusoidal power-clocks. The system is resonated using an on-chip 2.4-nH spiral inductor.

Fig. 7(a) is a block diagram of the Boost Logic test chip. The test structures in the chip were eight gate chains consisting of AND, OR, XOR, and INV gates with 200 gates per chain. The outputs of the gate chains were taken to output pads through digitally programmable Schmitt triggers.

A significant source of energy dissipation in charge-recovery systems is the clock generator. The clock generator periodically replenishes the energy dissipated in the resonant system, thereby sustaining a periodic clock with a constant amplitude. The frequency of the resonant clock is set by the rate at which energy is provided by the clock generator. The amount of energy replenished depends on the time for which clock generator switches M1–M4 conduct every cycle. This duration is determined by the pulsewidth of the signals that drive the switches. To enable the sweeping of operating frequency and pulsewidth, the design includes a pulse generator with programmable frequency and duty cycle. Furthermore, to allow for the exploration of alternative configurations for efficient clock generation, the width of the clock generator switch is programmable.

The H-Bridge clock generator consists of four clock generator switches which are driven by signals a and b. Once every cycle, for the duration of the pulse $a(b)$, switches M1 (M2) and M3 (M4) turn on. As the diagonally-situated switches turn on, current builds up in the integrated inductor L. To sustain oscillations, the current I thus injected in the inductor must satisfy the equation

$$\frac{1}{2}LI^2 = E_{\text{diss}}(T/2) \tag{7}$$

at the end of the pulse, so that the energy stored in the magnetic field of the inductor replenishes the energy $E_{\text{diss}}(T/2)$ dissipated in the half-cycle preceding it. L is chosen so that the natural frequency of the design is set close to its operating frequency as per the expression

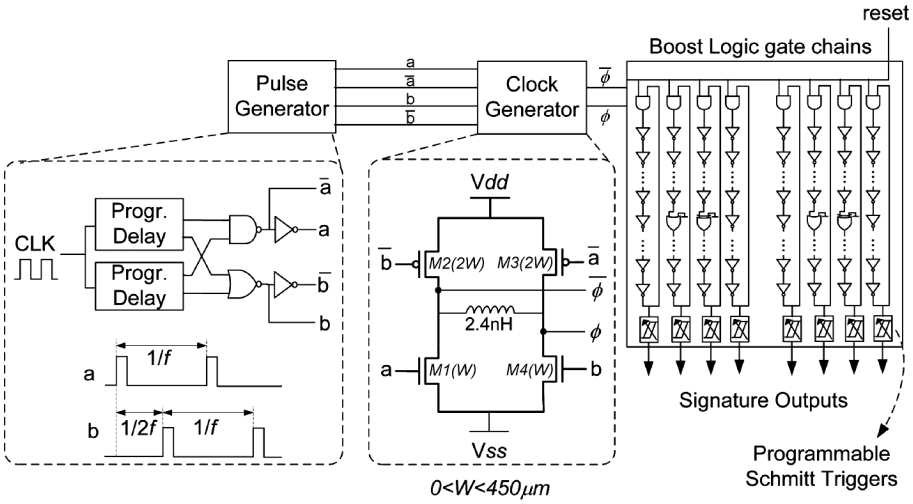
$$f_n = \frac{1}{2\pi}\sqrt{\frac{1}{LC} - \zeta^2} \tag{8}$$

where C is the capacitance of the oscillating system and ζ is its damping factor. Since L is fixed, the energy stored in the inductor is controlled through the injected current I, by varying either the clock generator switches or the pulse widths. Both measures increase either the CV^2 or the I^2R dissipation in the clock generator switches. While the pulse generator provides programmable duty cycle pulses, the clock generator was designed with programmable switch widths. The nMOS switches were variable from 0 to 405 μm and the pMOS switches programmable from 0 to 810 μm.

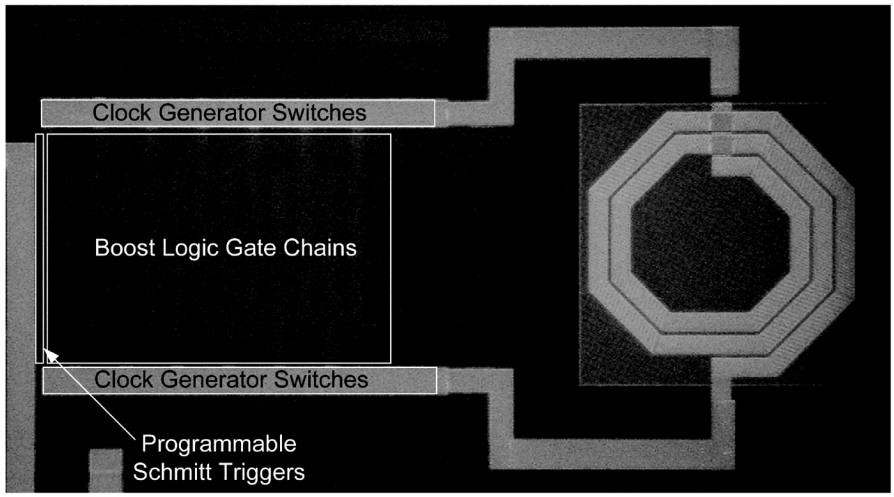
Fig. 7(b) shows a die-shot of the Boost Logic test chip measuring 315×320 μm^2. Including the moat around it, the 2.4-nH inductor occupied about 0.078 mm^2. The power-clocks were distributed on two main trunks along the sides of the gate chains, and shielded spines with a 16-μm pitch were used to route over the logic. The replenishing switches were placed below the main trunks on either side of the logic.

B. Test Chip Measurements

Fig. 8 shows the measured current and inferred energy dissipation in the clock generator supply V_{dd} of the test chip over a range of operating frequencies from 700 MHz to 1.1 GHz. By scaling V_{dd} to 1.5 V, correct operation was verified up to 1.3 GHz. The clock generator supply V_{dd}, together with the DC supply voltage V_C, accounts for the total energy dissipation in the test chip. Dissipation in the V_{dd} supply accounts for losses in



(a)



(b)

Fig. 7. Boost Logic test chip. (a) Block diagram. (b) Dieshot.

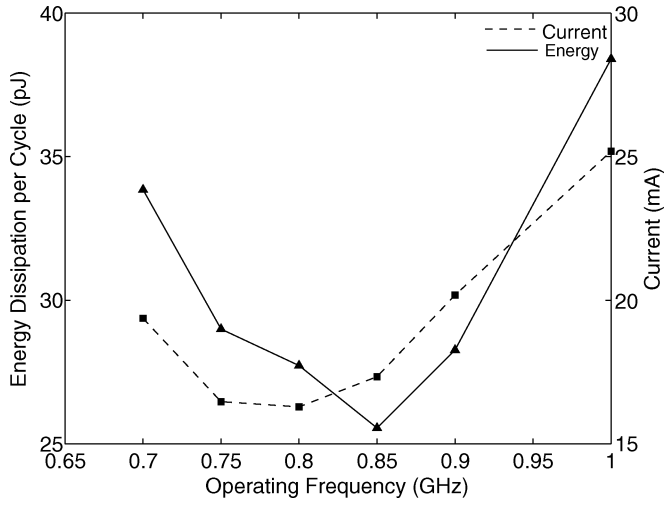


Fig. 8. Boost logic test chip: measured energy and current versus frequency.

the Boost stages, the clock distribution network, and the clock generator replenishing switches, as well as the buffers needed to drive them. Not shown in the figure is the measured energy dissipation in the voltage-scaled power supply for Logic, which remains nearly constant at 5 pJ per computation. Each point in the plot corresponds to the minimum energy dissipation obtained over the entire range of possible values for V_{dd}, V_C, D (the duty cycle of the clock generator), and W (the switch width of the clock generator). The natural frequency of the chip was measured to be approximately 850 MHz. At this frequency, the minimum energy per cycle in the test chip was measured at 26 pJ with $V_{dd} = 1.4$ V, $V_c = 0.45$ V, $D = 22\%$, and $W = 225$ μm. Since the CV^2 dissipation in switching the same load capacitance with an identical amplitude would be 60 pJ, it can be inferred that approximately 60% of the energy provided to the system is recovered. As the operating frequency shifts from the natural frequency, we observe an increase in the energy dissipation of the system as shown in Fig. 8.

Fig. 9 shows a textured shmoo plot of the energy dissipation of the test chip at $V_{dd} = 1.4$ V and $V_C = 0.4$ V as the duty cycle of the pulse generator D and the clock generator switch width W are varied. In this figure, points with the same symbol fall in the same energy dissipation band of 4 pJ. Higher energy dissipation occurs at larger values of D and W, which reflect the higher switching losses as well as dissipation in driving the larger clock switches. The lower iso-energy band reflects a trade-off that

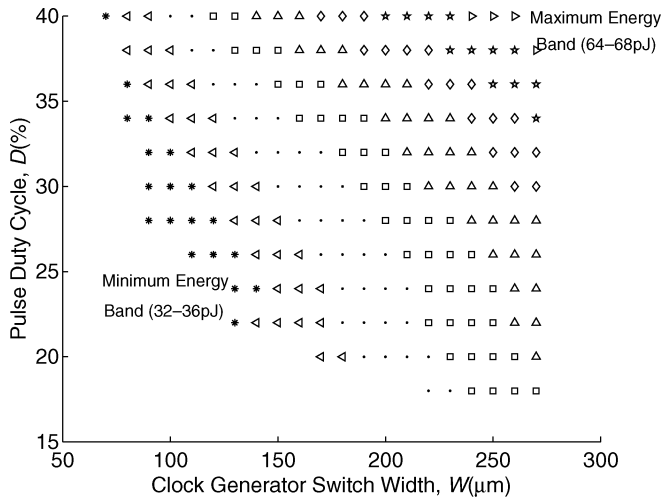


Fig. 9. Shmoo plot obtained by varying D and W.

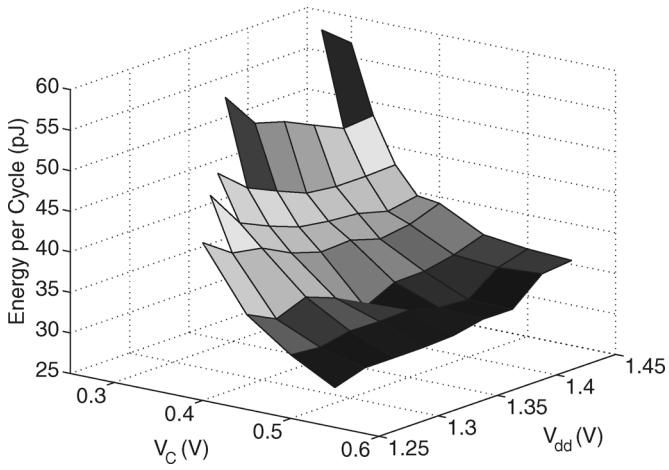


Fig. 10. Measured energy dissipation as a function of V_{dd} and V_c.

can be made between D and W. From our measurements, the region of efficient operation of the clock generator is for pulse duty cycles between 20% and 30% and for the minimum switch width that allows correct operation.

Fig. 10 illustrates the total energy dissipation in the test chip as a function of V_{dd} and V_C. Each point on the surface corresponds to the minimum energy achievable across all possible values of D and W. The figure shows a pronounced increase in energy dissipation for very low values of V_C. This observation demonstrates the importance of pre-resolved output nodes before charge-recovery for efficient operation at higher frequencies. As V_C increases beyond a certain point, however, the increased CV_C^2 dissipation as well as the leakage from Boost into Logic nullifies the power savings of improved charge-recovery due to better output node resolution. The optimal value of V_C depends on the clock amplitude and operating frequency of the design.

VI. Conclusion

In this paper, we have presented Boost Logic, a charge-recovery logic family which is capable of efficient operation at GHz-class clock frequencies. This efficient operation is achieved through the combined use of aggressive voltage-scaling, gate-overdrive, and charge-recovery techniques.

In post-layout simulations of 16-bit carry-save multipliers at 1 GHz, the Boost Logic implementation achieves energy savings in excess of 80% compared to its minimum-energy voltage-scaled static CMOS counterpart at the expense of a threefold increase in computational latency.

Considerable performance benefits can be achieved from the use of low V_{th} devices in the evaluation tree of Boost Logic gates. The use of low threshold devices enables the logic depth of a gate to be approximately doubled, resulting in designs with lower latency and energy dissipation than regular threshold devices as confirmed through the implementation of the low V_{th} 16-bit Boost multiplier.

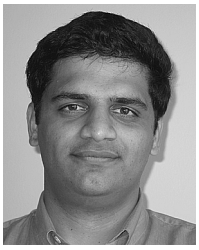
We have demonstrated the correct operation of a Boost Logic prototype chip with on-chip inductors in a 0.13-μm bulk silicon process. Our measurements show that approximately 60% charge-recovery was achieved at the resonant frequency of 850 MHz.

Acknowledgment

The authors would like to thank S. Pant and D. Roberts for their valuable assistance in design and test. They are grateful to MOSIS for enabling the fabrication of the test chip, and to the anonymous reviewers for constructive comments on the manuscript.

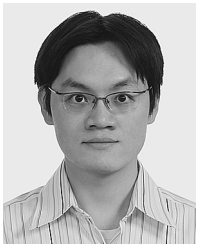
References

[1] S. G. Younis and T. Knight, "Practical implementation of charge recovering asymptotically zero power CMOS," in *Proc. Symp. Integrated Systems*, 1993, pp. 234–250.
[2] Y. Moon and D. Jeong, "An efficient charge-recovery logic circuit," *J. Solid-State Circuits*, vol. 31, no. 4, pp. 514–522, Apr. 1996.
[3] W. Athas, N. Tzartzanis, L. Svensson, and L. Peterson, "A low-power microprocessor based on resonant energy," *J. Solid-State Circuits*, vol. 32, no. 11, pp. 1693–1701, Nov. 1997.
[4] D. Maksimovic, V. Oklobdzija, B. Nikolic, and K. Current, "Clocked CMOS adiabatic logic with integrated single-phase power-clock supply: Experimental results," in *Proc. ISLPED*, 1997, pp. 323–327.
[5] C. Ziesler, J. Kim, V. Sathe, and M. Papaefthymiou, "A 225 MHz resonant clocked ASIC chip," in *Proc. ISLPED*, 2003, pp. 48–53.
[6] V. De and J. D. Meindl, "Complementary adiabatic and fully adiabatic MOS logic families for gigascale integration," in *IEEE ISSCC Dig. Tech. Papers*, 1996, pp. 298–299, 461.
[7] A. G. Dickinson and J. S. Denker, "Adiabatic dynamic logic," *J. Solid-State Circuits*, vol. 30, no. 3, pp. 311–315, Mar. 1995.
[8] S. Kim, C. H. Ziesler, and M. C. Papaefthymiou, "A true single-phase 8-bit adiabatic multiplier," in *Proc. Design Automation Conf.*, 2001, pp. 758–763.
[9] Y. Yibin and K. Roy, "QSERL: Quasi-static energy recovery logic," *J. Solid-State Circuits*, vol. 36, no. 2, pp. 239–248, Feb. 2001.
[10] A. Vetuli, S. D. Pascoli, and L. M. Reyneri, "Positive feedback in adiabatic logic," *Electron. Lett.*, vol. 32, no. 20, pp. 1867–1869, Sep. 26, 1996.
[11] V. S. Sathe, J. Chueh, and M. C. Papaefthymiou, "A 1.1GHz charge-recovery logic," in *IEEE ISSCC Dig. Tech. Papers*, 2006, pp. 1540–1549.
[12] C. S. *et al.*, "Hot-clock nMOS," in *Proc. Chapel Hill Conf. VLSI*, 1985, pp. 1–17.
[13] D. Deleganes, M. Barany, G. Geannopoulos, K. Kreitzer, M. Morrise, D. Milliron, A. Singh, and S. Wijeratne, "Low-voltage-swing logic circuits for a 7GHz x86 integer core," in *IEEE ISSCC Dig. Tech. Papers*, 2004, pp. 154–155.
[14] V. Svilan, M. Matsui, and J. Burr, "Energy-efficient 32×32-bit multiplier in tunable near-zero threshold CMOS," in *Proc. ISLPED*, 2000, pp. 268–272.

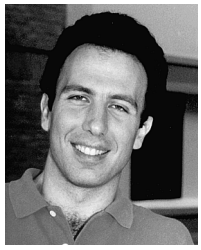


Visvesh S. Sathe (S'02) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Bombay, India, in 2001, and the M.S. degree in electrical engineering and computer science in 2004 from the University of Michigan, Ann Arbor, where he is currently working toward the Ph.D. degree.

In the summers of 2003 and 2004, he was with the Low Power Circuits and Technology group, IBM T. J. Watson Research Center, Yorktown Heights, NY, where he was a graduate intern. His research interests include VLSI design with particular emphasis on low-power design techniques.



Juang-Ying Chueh (S'02) was born in Taipei, Taiwan. He received the B.S. and M.S. degrees from National Taiwan University in 1999 and 2001, respectively. In Fall 2001, he joined the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, where he is working toward the Ph.D. degree on resonant clock generation and distribution, and low-power circuit design.



Marios C. Papaefthymiou (M'93–SM'02) received the B.S. degree in electrical engineering from the California Institute of Technology, Pasadena, in 1988, and the S.M. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, in 1990 and 1993, respectively.

After a three-year term as Assistant Professor at Yale University, he joined the University of Michigan, Ann Arbor, where he is currently Professor of electrical engineering and computer science and Director of the Advanced Computer Architecture Laboratory. His research interests encompass algorithms, architectures, and circuits for energy-efficient high-performance VLSI systems. He is also active in the field of parallel and distributed computing.

Among other distinctions, Dr. Papaefthymiou has received an ARO Young Investigator Award, an NSF CAREER Award, and a number of IBM Partnership Awards. Together with his students, he received a Best Paper Award in the 32nd ACM/IEEE Design Automation Conference and the First Prize (Operational Category) in the VLSI Design Contest of the 38th ACM/IEEE Design Automation Conference. He has served multiple terms as Associate Editor for the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS, the IEEE TRANSACTIONS ON COMPUTERS, and the IEEE TRANSACTIONS ON VLSI SYSTEMS. He has served as the General Chair and Technical Program Chair for the ACM/IEEE International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems. He has also participated several times in the Technical Program Committee of the IEEE/ACM International Conference on Computer-Aided Design.