Resonant-Clock Design for a Power-Efficient, High-Volume x86-64 Microprocessor

Visvesh S. Sathe, *Member, IEEE*, Srikanth Arekapudi, *Member, IEEE*, Alexander Ishii, *Member, IEEE*, Charles Ouyang, *Member, IEEE*, Marios C. Papaefthymiou, *Senior Member, IEEE*, and Samuel Naffziger, *Senior Member, IEEE*

***Abstract*—AMD's 32-nm x86-64 core code-named "Piledriver" features a resonant global clock distribution to reduce clock distribution power while maintaining a low clock skew. To support a wide range of operating frequencies expected of the core, the global clock system operates in two modes: a resonant-clock (rclk) mode for energy-efficient operation over a desired frequency range and a conventional, direct-drive mode (cclk) to support low-frequency operation. This dual-mode feature was implemented with minimal area impact to achieve both reduced average power dissipation and improved power-constrained performance. In Piledriver, resonant clocking achieves a peak 25% global clock power reduction at 75 °C, which translates to a 4.5% reduction in average application core power.**

***Index Terms*—Clocks, high-performance computing, low-power electronics, microprocessors.**

I. Introduction

LARGE high-performance microprocessors continue to dissipate a significant amount of power in their clock distribution networks. With energy efficiency increasingly determining cost and performance, efficient clocking strategies have gained importance. To this end, the 32-nm AMD core code-named "Piledriver" employs resonant clocking [1]–[5] on the global clock distribution, using distributed integrated inductors to achieve *LC* resonance with parasitic clock capacitance.

During the past decade, several test chips successfully demonstrated a variety of resonant clocking implementations [1]–[9]. Early instances of resonant clocks can be found in so-called *adiabatic* circuits, where resonant "power-clocks" recover charge stored in the parasitic capacitance of internal dynamic logic-gate nodes into discrete [8]–[10] or integrated inductors [2].

To attain improved energy efficiency, researchers have explored resonant clocking by confining it to the clock distribution network and using the resonant clock waveform to drive timing elements (e.g., clock gaters or flip-flops) [5]–[7], [11]–[13]. In these works, all of the sink nodes of the resonant clock network oscillate essentially in phase. Resonant clock designs in which clock waveforms are either traveling waves [4] or standing waves [3] have also been explored. In these methodologies, however, the clock phase or amplitude varies considerably across the distribution at a given time.

Recently, the feasibility of resonant clocking in a high-performance microprocessor was explored on the Cell Broadband Engine [1], providing the first gigahertz-speed evaluation of the technology in a commercial context and in the presence of several real-world constraints. While this evaluation yielded promising results, it also underscored critical challenges that need to be addressed towards a production-ready resonant-clocked microprocessor. A thick copper metal layer was used to implement the inductor and the required capacitance structures. While the additional metal layer leads to substantial resonant-clocking efficiencies, it increases fabrication cost and is an unsuitable option for a high-volume processor. Another significant limitation of the implementation was its inability to operate outside a limited range around the natural frequency, which precludes effective core power management through dynamic voltage-frequency scaling (DVFS) and raises adverse test implications.

Achieving efficient *LC* resonance for clocking in high-performance digital circuits poses several additional challenges. Interactions between the inductor and nearby signal wires present potential noise implications. Efficient operation relies on achieving a good system quality factor (Q), which depends on the inductor winding resistance, as well as the extent of inductive coupling to nearby conductors. This detrimental coupling has the highest impact in low-impedance power grid loops, resulting in the formation of Q-degrading eddy currents. Prior test chips with integrated inductors avoided this Q-degradation by enforcing keep-out regions [5], [6], [12] around the inductors, avoiding the presence of any nearby conductors. Physical constraints due to the C4 bump pitch, and the increased area overhead of such a technique, however, limit its applicability in high-volume microprocessors. This work addresses these and other challenges encountered in a production-ready implementation of resonant clocking for a high-volume commercial x86-64 microprocessor core, Piledriver, capable of operating frequencies over 4 GHz.

The remainder of this paper is organized as follows. Section II provides an overview of the resonant clock implementation preliminaries. Section III outlines the Piledriver core global clock architecture. Clock driver and inductor design, central to efficient resonant clocking, are discussed in Sections IV and V, respectively. The design of other key circuit components that enable the implementation are discussed in Section VI.

Manuscript received April 18, 2012; revised July 01, 2012; accepted August 08, 2012. Date of publication October 18, 2012; date of current version December 31, 2012. This paper was approved by Guest Editor Wim Dehaene.

V. S. Sathe and S. Naffziger are with Advanced Micro Devices, Fort Collins, CO 80525 USA.

S. Arekapudi is with Advanced Micro Devices, Sunnyvale, CA 94086 USA.

A. Ishii and M. C. Papaefthymiou are with Cyclos Semiconductor Inc., Berkeley, CA 94709 USA.

C. Ouyang is with Advanced Micro Devices, Sunnyvale, CA 94086 USA.

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/JSSC.2012.2218068

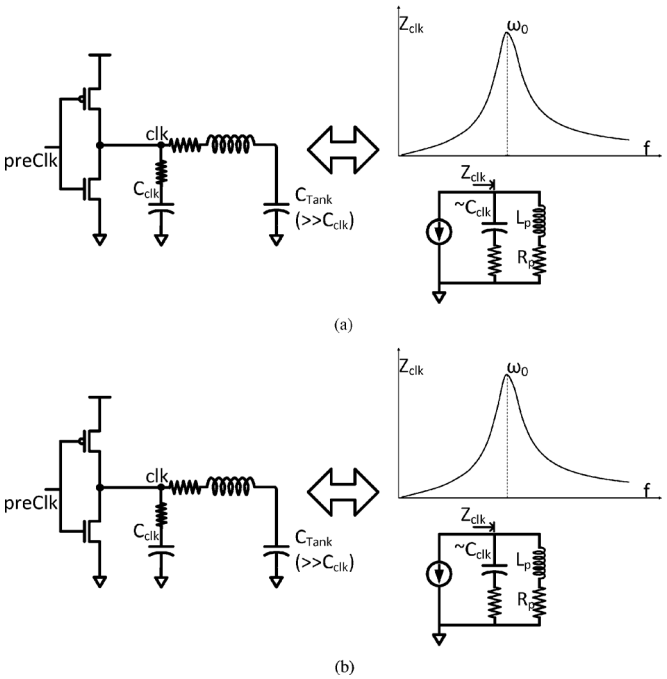


Fig. 1. Simplified lumped model of a resonant clock system. (a) Basic resonant clocking model. (b) Basic resonant clocking model with inductive coupling to signal and power grid wires.

Global clock optimization, crucial for the design of an efficient, skew-compliant clock network is discussed in Section VII. Measurement results are presented in Section VIII.

II. Resonant Clocking Preliminaries

Here, we highlight the salient aspects of resonant clocking. The importance of system Q to energy efficiency, and the main contributors to system Q are discussed. The impact of inductive coupling to nearby signal and power wires is also explored.

A. Energy Efficiency and System Quality Factor

The resonant clocking approach in Piledriver involves achieving efficient *LC* resonance between the parasitic capacitance in the global clock grid and distributed integrated inductors connected to this grid. Fig. 1(a) illustrates this concept. For simplicity, parallel inductors and the distributed clock drivers and clock load are lumped into single elements. From an equivalent ac circuit perspective, clock drivers serve as current sources driving a parallel *LC* tank circuit. The capacitance $C_{\rm Tank}$, employed to prevent dc current flow through the inductor is large enough to serve as an ac ground at frequencies around the natural frequency of the network, ω_0. At frequencies close to ω_0, the tank circuit impedance increases, requiring a smaller current to sustain a given clock amplitude. Unlike a conventional clock system, in which the clock drivers serve to completely charge and discharge the clock grid, the resonant clock drivers serve to primarily replenish the I^2R losses in the tank circuit. $C_{\rm Tank}$ is implemented as a stacked capacitor to allow for low-impedance ac current-return paths through the clock load capacitance, coupled to both power and ground. A stacked capacitor also provides the additional benefit of contributing to the core decoupling capacitance.

The efficiency achieved by resonant clocking is a function of the Q of the oscillating system

$$Q = \frac{\omega \cdot E_{\rm peak}}{P_{\rm diss}} \tag{1}$$

where $E_{\rm peak}$ is the peak energy stored, ω is the angular frequency of oscillation, and $P_{\rm diss}$ is the per-cycle average power dissipation in the system. In Piledriver, the clock voltage transitions between $V_{\rm ss}$ and $V_{\rm dd}$, centered at $V_{\rm dd}/2$. Applying (1) to our simplified tank circuit representation [Fig. 1(a)], the power dissipation of the resonant clock system $P_{\rm diss}$ can be shown to be [14]

$$P_{\rm diss} = \frac{\pi}{4Q} C V_{dd}^2 f \tag{2}$$

where C is the clock capacitance and f is the operating frequency. Equation (4) illustrates the impact of system Q on the efficiency of the network. The system Q is a "parallel combination" of the Q of the inductor Q_L and that of the clock distribution network Q_C [5] as

$$\frac{1}{Q} = \frac{1}{Q_L} + \frac{1}{Q_C}. \tag{3}$$

B. Impact of Adjacent Power and Signal Lines

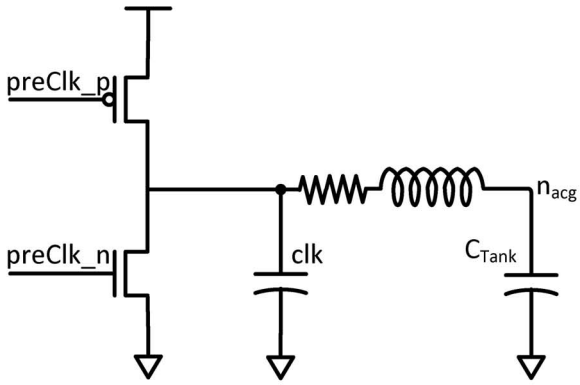
Fig. 1(b) shows a simplified view of a resonant clock system that experiences significant mutual inductance interaction between the implemented inductor and neighboring wires. In an actual implementation, there are a large number of conductors placed at various distances and oriented at various angles with respect to the inductor winding, resulting in different loop impedances, and coupling coefficients. For the purposes of discussion. however, the effect of these interactions has been modeled by a single "secondary" RL loop, which is inductively coupled to the implemented inductor with an effective coupling factor k to the inductor winding.

The most important effect of this inductive coupling is a reduction in Q_L (and therefore Q). To understand why, we first consider the effect of mutual inductance interactions shown in Fig. 1(b) on Z_L, the imaginary component of $Z_{\rm clk}$ as

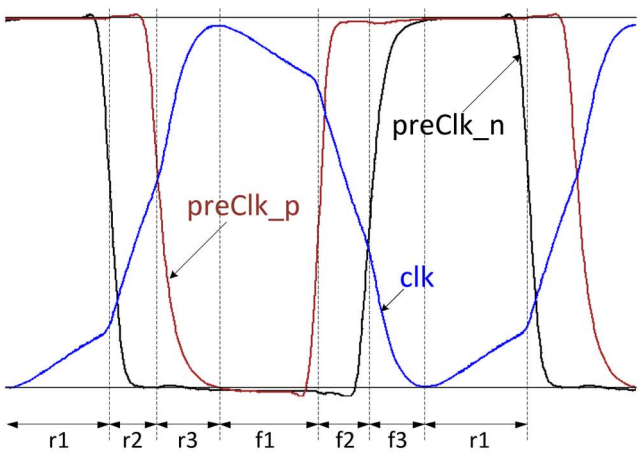
$$Z_L = R_p + j\omega L_p + \frac{k^2 L_p L_s \omega^2}{R_s + j\omega L_s}. \tag{4}$$

Mutual inductance increases the resistive component of Z_L while reducing its inductive component. The impact of this interaction on Z_L depends on the coupling coefficient k, and the secondary loop impedance—a low k (obtained using a keep-out region) or a large R_s (due to an absence of low-impedance power loops) has a smaller impact on Z_L. Conversely, the presence of low-impedance conductor loops close to the inductor winding (such as in a power grid) results in the formation of eddy currents, reducing Q_L and degrading in overall efficiency. This undesirable change in Z_L and ω_0 is illustrated in Fig. 1(b).

Another important aspect of resonant clocking is that, while energy efficiency is achieved at frequencies in the neighborhood of the resonant frequency, driving a resonant clock network at frequencies substantially away from the natural frequency results in increased energy consumption. In particular, driving the



(a)



(b)

Phase	Pulldown (NMOS)	Pullup (PMOS)	Description
r1	ON	OFF	RL current build-up in inductor through NMOS
r2	OFF	OFF	Inductor current charges clock capacitance
r3	OFF	ON	Pmos assists inductor current to fully charge clock capacitance
f1	OFF	ON	RL current buildup in inductor through PMOS
f2	OFF	OFF	Inductor current discharges clock capacitance
f3	ON	OFF	Nmos assists inductor current to fully discharge clock capacitance

(c)

Fig. 2. Steady-state analysis of a single resonant clock cycle (a) Simplified resonant clock system for steady-state analysis. (b) Constituent phases of a single resonant clock cycle. (c) Description of circuit activity during the resonant clock cycle.

network at low frequencies also results in a warped clock waveform, compromising functional operation of the design [1].

C. Resonant Clock Waveforms

An ac analysis of the simplified resonant clock network of Fig. 1(a) provides a simple and consistent framework for understanding the major aspects of resonant clocking. However, an understanding of the sources of power dissipation in the system, and the impact of resonant clocking on core performance is better aided by steady-state clock analysis.

Fig. 2(a) shows the simplified resonant clock system used for the analysis. A split buffer topology is adopted to allow for independent pull-up and pull-down control allowing for the insertion of a "dead time" during which both devices are nonconducting.

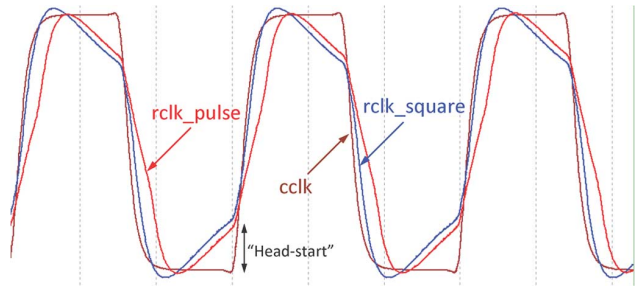


Fig. 3. Clock waveforms corresponding to conventional (cclk) rclk_square (no driver dead-time) and rclk_pulse (with driver dead time).

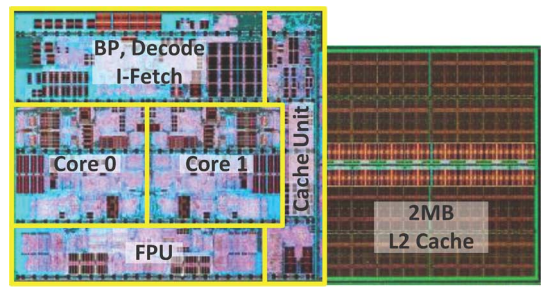


Fig. 4. Die microphotograph of the Piledriver core.

Because the analysis is being performed in steady state, due to the large $C_{\rm Tank}$, the voltage at node $n_{\rm acg}$ (intended ac ground) can be considered to be nearly steady at $D \cdot V_{\rm dd}$, where D is the duty cycle of the output clock waveform. The clock waveform is divided into six phases as shown in Fig. 2(b), and the corresponding behavior of the circuit is summarized in Fig. 2(c).

Fig. 3 compares simulation waveforms in various modes at a clock receiver in the Piledriver core. The waveforms correspond to a conventional clock waveform (cclk) and two resonant clock waveforms (rclk_square and rclk_pulse). The rclk_pulse mode is an energy-efficient rclk mode in which a dead time is deliberately introduced in the clock driver, effectively trading off switching and conduction losses in the clock driver. Driving a resonant clock with a driver dead-time is henceforth referred to as pulse-mode resonant clocking.

A relative insertion delay (phase-shift) observed between the $cclk$ and $rclk$ waveforms is determined by two opposing effects. Achieving the full benefits of resonant clocking requires using a lower clock driver strength. In Piledriver, rclk slew rates are 50%–70% of those of cclk, and the resulting slew degradation causes rclk waveforms to have a higher insertion delay. A countervailing effect is the "head start" that rclk waveforms experience due to the increasing IR drop across the conducting device in phases $r1$ and $f1$. The slew impact is found to be more significant in Piledriver, resulting in a phase push-out for rclk_square in comparison with $cclk$. The rclk_pulse waveform sees a further phase offset with respect to $cclk$ due to the dead time introduced in the driver, which delays the onset of the asserting edge.

III. Piledriver Resonant Clocking

Here, we provide an overview of the Piledriver core and motivate the implementation of resonant clocking for the global clock distribution grid.

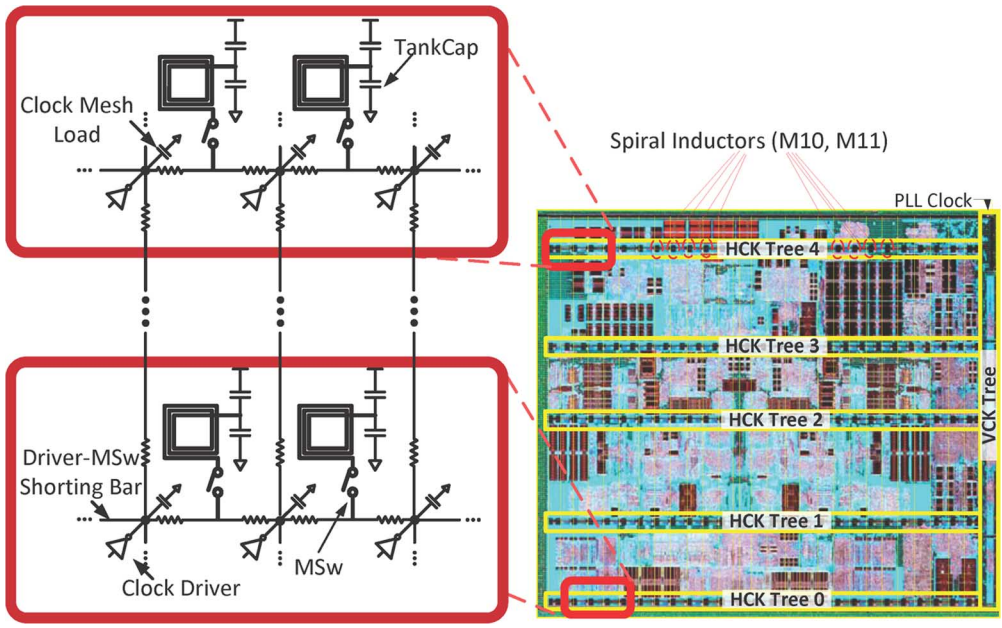


Fig. 5. Physical view of Piledriver resonant clock implementation.

A. Piledriver Core

Piledriver is AMD's two-core x86-64 processor based on the company's Bulldozer module [15] to meet the demanding compute needs of both client and server workloads. Fig. 4 shows a chip microphotograph of the Piledriver core-pair with a shared L2 cache. The 30.9-mm^2 design is built in 11-level metal HKMG 32-nm SOI CMOS and achieves an operating frequency improvement of more than 20% compared with AMD's previous x86-64 processor built in the same process node [16]. The two-core Piledriver module contains 216 million transistors and is designed to operate in the (0.8 V, 1.3 V) range.

Fig. 5 illustrates the global clock distribution architecture of the Piledriver core. The PLL clock is distributed along the right edge of the core using a folded vertical clock tree macro (VCK tree). The VCK tree in turn drives five horizontal clock tree macros (HCK trees). The clock drivers are placed inside the HCK trees and drive a global clock mesh. The distribution of a low-skew clock across a large high-performance microprocessor necessitates the use of a clock grid to address process, voltage and temperature gradients across a large die-area, exacerbated by long-latency pre-clock distribution networks. Robust top-level clock routing resources are employed to meet an aggressive 7 ps within-grid skew target. A hold time driven methodology constraint also tightly controls the latency of the clock from the grid to any downstream timing elements, limiting the use of the staging buffers or multilevel clock gating to reduce the load on the grid. These factors contribute to a substantial loading on the clock grid. With about 24% of the average application power in Piledriver dissipated in the global clock, efficient global clock distribution is crucial to achieve efficient processor design.

B. Resonant Clocking Architecture

The Piledriver core operates across a wide operating frequency range from 500 MHz to over 4 GHz. The power-up sequence and certain test modes require support for even lower frequencies. Consequently, a dual-mode clock design is implemented, employing a mode switch (MSw) to support both resonant and conventional clocking.

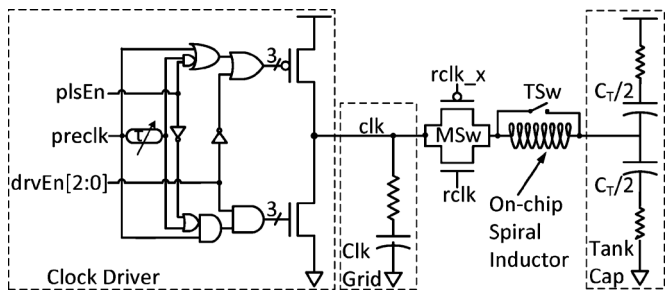


Fig. 6. Simplified schematic representation of Piledriver resonant clock implementation.

Fig. 6 shows a simplified view of the Piledriver resonant clock system. The MSw is introduced between the inductor and the clk node. During rclk(cclk), the MSw is closed (open), connecting (disconnecting) the inductor and tank capacitor C_{Tank} to the clock network. Robust clock operation over a wide range of frequencies is supported thus.

Clock drivers play an important role in supporting a dual-mode clock system and were designed with a split-buffer topology for crossover current reduction and pulse-mode support. Furthermore, clock driver strengths are programmed at runtime depending on the operating voltage-frequency and the clock mode, thereby improving energy efficiency.

C_{Tank} needs to be sufficiently large to serve as an ac ground in the range of rclk target frequencies. In the Piledriver implementation, C_{Tank} is approximately six times the clock network loading with low ESR, allowing the capacitor to serve as an adequate ac ground. Connecting the inductor in or out of the clock grid (depending on the operating clock mode) causes transient voltage overshoots, which raise electrical reliability concerns.

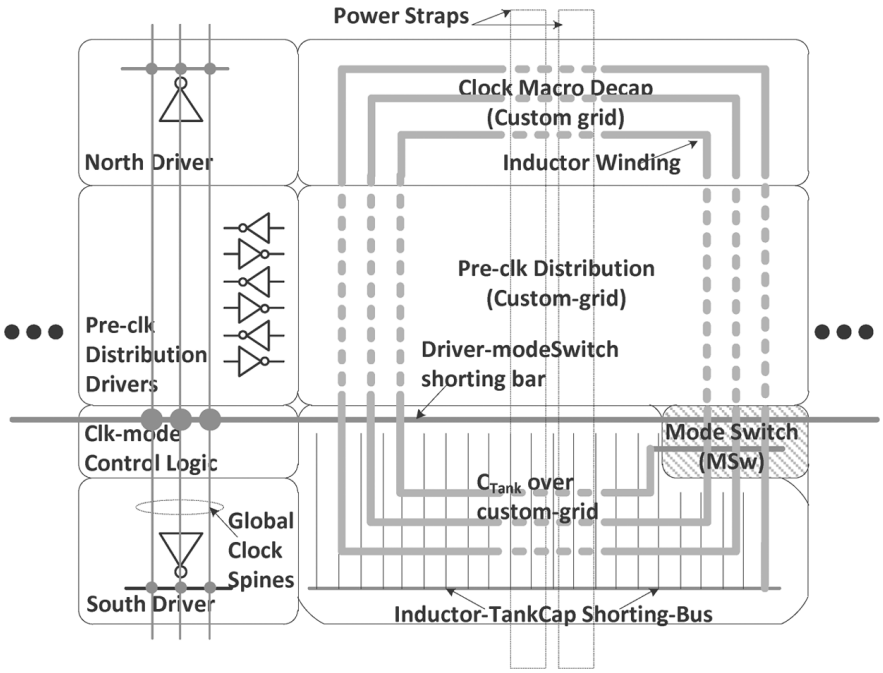


Fig. 7. Repeated-section view of the HCK tree macro illustrating the organization of the inductors, final drivers, programming logic, and other rclk-related components.

These concerns were addressed with a throttle-switch (TSw) to dampen the transient voltage excursions.

Fig. 5 illustrates the physical representation of the dual-mode clock system. To minimize I^2R losses, 92 spiral inductors were distributed across the core. These inductors "resonate" with the distributed clock grid cap, forming strongly connected oscillating clock domains. All of the inductors and associated resonant clocking circuitry were contained entirely inside the HCK tree, enabling feature implementation with minimal impact on the rest of the core design process.

The Piledriver core features timing arcs to and from the L2 cache and the north-bridge (NB) interface. In this implementation, however, only the core clock is implemented as a dual-mode clock, with L2 and NB clocks phase-aligned to the core clock in cclk. Consequently, the phase offset between rclk and cclk modes has performance implications on the core, as discussed in Section VIII.

Fig. 7 shows a repeated section of the horizontal clock macro and the arrangement of the various clock-related structures contained inside the HCK tree macro. A shorting-bar runs across the width of the HCK tree, allowing for tighter skew control at the final clock driver outputs. The shorting-bar connects the inductor to the vertical clock spines (which make up the global clock mesh) through the MSw. The horizontal clock tree distribution is situated in the center of the HCK tree and runs through the inductor. The programming logic required to support the dual-mode clock system and runtime programmable drive strength, is also distributed across the HCK Tree macro underneath the inductor.

C. Clock Configuration Programming

Operating the core in a given clock mode requires that every driver and inductor be programmed with the correct configuration. Because clock modes and their associated configuration programming are driven by operating frequency, programming of the clock macros is performed during performance state (PState) transitions, during which the core transitions to a new target frequency. Fig. 8 illustrates the system-level organization of the configuration programming interface. On receiving notification from the NB of a PState transition, the core implements a PState entry sequence which transitions the core into a clock-gated state. A program sequencer then accesses a frequency-indexed fuse table to obtain the clock macro configuration programming based on the target PState frequency, and coordinates the broadcast of this configuration programming to the drivers and rclk components in the core. Some of the more important configuration bits that are broadcast by the program sequencer are the driver strength, pulse-mode, and pulse duty-cycle settings. The transfer of configuration programming bits to each clock driver location in the core, is performed through a source-synchronous (SS) interface. During the PState transition, the core is designed to operate in cclk, regardless of the initial and final clock modes of the core. Because the PLL frequency output cannot be guaranteed during the PState transition, this measure ensures that the core does not operate in rclk at an unsuitable transient frequency.

IV. Driver Design

The details of driver design including runtime-programmable drive strength modulation, and pulse-mode drive support are presented here.

Fig. 6 shows a simplified representation of the clock driver. The driver cell consists of the final four stages of the clock distribution. The implemented split-buffer topology supports pulse-drive, and the efficient implementation of run-time drive-strength modulation. $preclk$ is a free running clock from the clock tree, $drvEn[2:0]$ are the drive-strength configuration bits, and $plsEn$ is the pulse-mode enable signal. Each

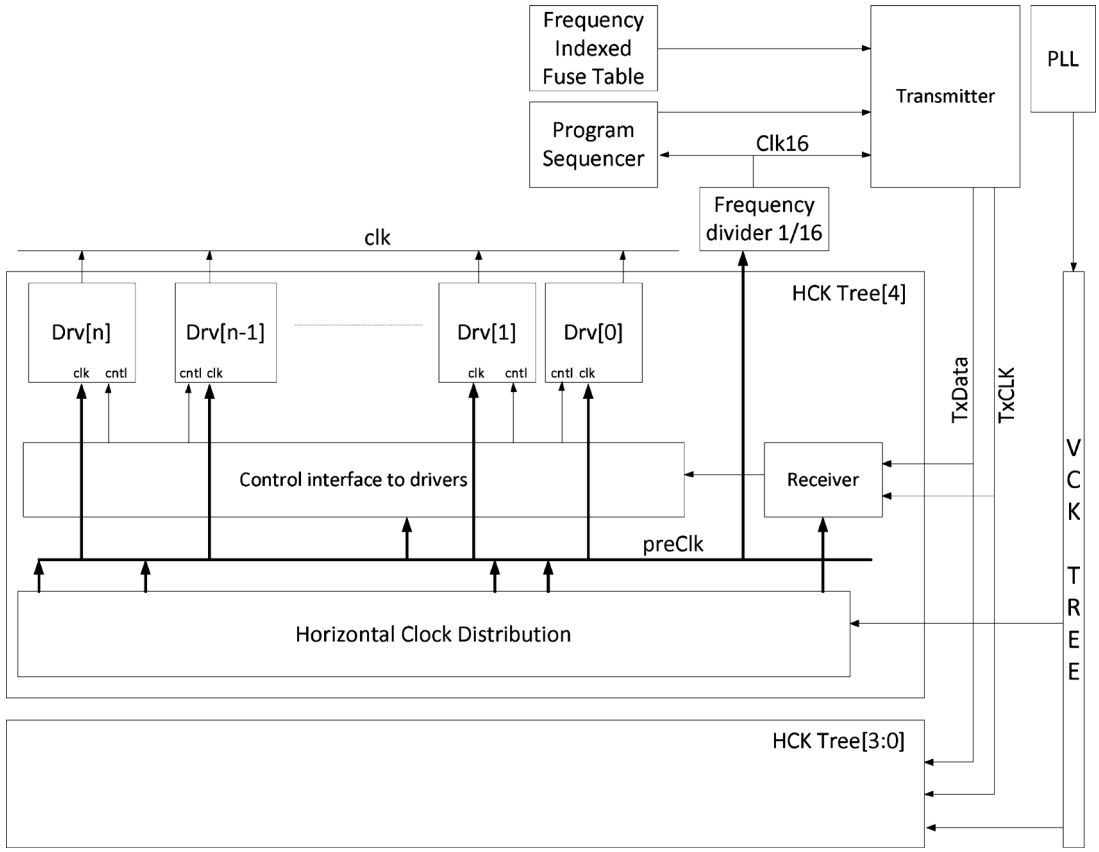


Fig. 8. Clock spine programming architecture.

driver in the HCK tree is selected to drive the grid load in its vicinity. Skew-optimal driver allocation is done through a linear programming formulation (discussed in greater detail in Section VII).

Pulse-mode operation in Piledriver is supported through the implementation of a subtractive pulse-mode scheme in the driver. Accordingly, each final-stage clock driver contains a small delay chain, used to adjust the pulse width at the input of the pull-up and pull-down devices in the split driver. The desired pulse width is achieved by delaying the arrival of the rising (falling) edge of the split NMOS (PMOS) device in the clock driver, without introducing any delay in the corresponding de-asserting edges. Shown in Fig. 6, such edge-selective behavior is implemented with logic gates at the input of the driver.

In contrast to traditional pulse-generation mechanisms, where the delay chain sets the duration of the pulse, the pulse width in the proposed pulse-generation scheme subtracts the delay of the delay chain from the on-time in each leg of the driver. This method of pulse generation is both necessary and advantageous for several reasons. The proposed pulse generation supports robust operation of multi-core systems with a common power plane, in which some cores operate at a voltage higher than required for their operating frequency. The use of a traditional pulse-generation scheme results in duty-cycle shrinking, adversely impacting clock amplitude. In contrast to a conventional pulse-generation scheme, the resonant clock duty cycle can be modulated by the PLL clock, thereby allowing PLL duty-cycle tuning for phase-path timing optimization. Finally, subtractive pulse-generation greatly reduces the impact of variation. Typical rclk_pulse widths are in the range of 30%–40%. Using a subtractive scheme allows for a smaller delay chain which is more energy efficient and less susceptible to process variation. As discussed in Section II, the clock edge transition governed by the de-asserting edge. By allowing this de-asserting edge to propagate through the logic gates without delay, the sensitivity of the global clock skew to variation in the delay chains is reduced.

V. Inductor Design

The design of Piledriver spiral inductors used for the rclk implementation is discussed here. In particular, we discuss the challenges posed to Piledriver inductor design, and how they were addressed.

Piledriver inductor design is heavily constrained both physically and electrically. The Piledriver inductors are constrained to an outer winding dimension of less than 100 μm by the HCK tree height, and the C4 bump pitch heavily influences inductor placement. Additionally, the presence of the power supply grid, and other signal wires under and around the inductors pose a challenge to achieving useful inductor quality factors (Q_L), as discussed in Section II. Unlike conventional spiral inductor applications, the area-overhead and routing constraints imposed by keep-outs make them infeasible for Piledriver.

The process provides two thick metals (M10 and M11), which are crucial to build a high-Q inductor, but these primarily as power redistribution (RDL) and global clock mesh layers. The pre-clock distribution accounts for most of the M10 routing

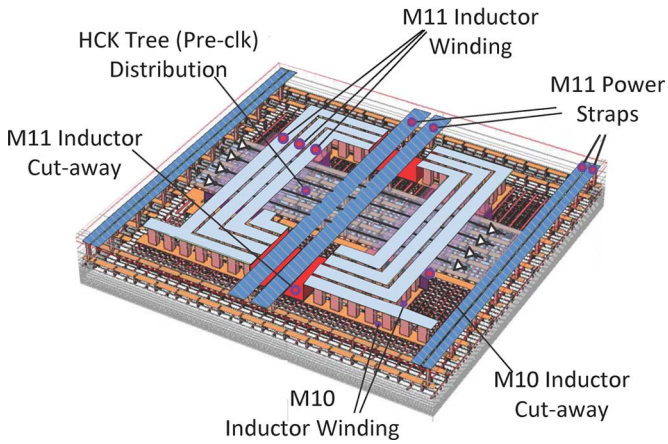


Fig. 9. 3-D representation of Piledriver inductor. Cut-aways are created in the stacked winding to make way for RDL power stripes and pre-clock distribution metal.

within the HCK tree. Therefore, inductor design using these top two layers (M10 and M11) needed to be performed in the presence of the power and signal nets in these metal layers.

Fig. 9 shows a 3-D representation of the Piledriver inductors. At the frequencies in question, Q_L (and therefore Q) is dominated by series losses in the winding. As such, "maximally thick" inductor windings were built by stacking the top two levels of metal. Cut-aways were built in specific regions to allow the power supply RDL metal on M11, or the horizontal pre-clock distribution on M10 to be routed through the inductor. The inductive coupling interaction between the winding, and the power and pre-clock distribution wires was minimized by positioning the inductor so the power and signal wires run through the middle of the winding, allowing for maximal magnetic vector potential [17] cancellation along the inductor winding.

Eddy currents that form in the power grid underneath and around the inductor are a major source of Q degradation. To address the severe Q degradation that results from these eddy currents, a custom "loop-less" power grid was developed. This loop-less grid avoids the formation of eddy currents beneath the inductor, while meeting power grid robustness requirements for the circuits underneath. The region outside the inductor (and outside the HCK tree) could not be built with a custom grid due to methodology constraints. Therefore, eddy current flow around the inductor continues to contribute (though to a smaller extent) to Q degradation. At 3.4 GHz, the achievable inductor Q is 3.6–3.8 depending on the value of inductor chosen.

VI. Other Rclk Components

A. Mode Switch (MSw)

The MSw is essential for the implementation of a dual-mode clock, but its inclusion increases the effective resistance of the inductor section of the tank circuit, degrading Q_L. That the devices in the MSw do not conduct with a full gate overdrive for most of the cycle (and in particular during maximum current flow) further exacerbates the problem. Furthermore, the MSw capacitively loads the clock grid, presenting a source of overhead. The resulting tradeoff between series resistance and capacitive loading drove careful design of the MSw.

Before the MSw turns on at the onset of a cclk to rclk transition, the dc voltage levels at nodes on the inductor side of the MSw and those of the clock network ($DV_{\rm dd}$ for a clock duty cycle D) cannot be guaranteed to match. In this situation, turning on a relatively low-resistance MSw can lead to loading the clock grid with the large, partially charged $C_{\rm Tank}$, degrading slew and causing potential electrical reliability issues in the MSw. Consequently, transitions from the cclk to the rclk mode are performed by staging the turn-on of the banks that make up the MSw over several cycles (analogous to a power-gating wake-up sequence).

The MSw implementation for rclk support presents a 6.5% capacitive overhead to the clock network during cclk operation. A variety of techniques are currently under consideration to reduce this overhead.

B. Throttle Switch (TSw)

Transitioning from rclk to cclk mode potentially raises reliability concerns. If the MSw is opened at a time when the current flow in the inductor is at or near its peak, a voltage overshoot results, increasing the gate oxide stress on particularly the NMOS device in the MSw. In Piledriver, the overshoot has been addressed by the use of a TSw [Fig. 10(a)] connected in parallel with the inductor. Fig. 10(b) shows the voltage overshoot observed on node *n1* without the use of a TSw. When the MSw opens, the TSw, controlled by complementary signals, closes. The presence of a sufficiently low-impedance switch across the inductor damps the resulting RLC system made up of the inductor, the switch, and the parasitic capacitance of the MSw, thereby avoiding overshoot. Fig. 10(c) shows simulation waveforms illustrating the suppression of the voltage overshoot by the throttle switch.

VII. Global Clock Optimization

The global clock distribution was optimized in both cclk and rclk modes to efficiently meet the target grid skew. This effort involved optimization of the clock spines, the clock driver assignment, and the allocation of the inductors.

A clock tuning algorithm was developed to minimize global clock skew while controlling transmission-line effects and minimizing the capacitive loading on the clock grid. An iterative linear programming (LP) formulation was implemented to determine the optimal clock assignment for each of the 270 drivers to achieve the skew target. Charge-based measurements are used to obtain an initial solution for the driver assignment. The tuning algorithm then derives sensitivities of the clock arrival time at various points on the grid to perturbations in the clock driver strength, and uses the sensitivity matrix to solve the linear program to minimize skew. Because this approach involves linearizing an inherently nonlinear problem, multiple iterations of sensitivity computation and linear programming are required to arrive at the desired solution.

To address the significant spatial variation of the global clock load on the grid, we implemented a palette of five inductors in the 0.5–1.3-nH range. While having more inductors to choose from was clearly more desirable, resource constraints limited the palette to five. To minimize clock skew from the quantization error arising from a sparse inductor palette, an iterative linear programming technique similar to the one used for driver assignment was employed. Energy efficiency was also traded to

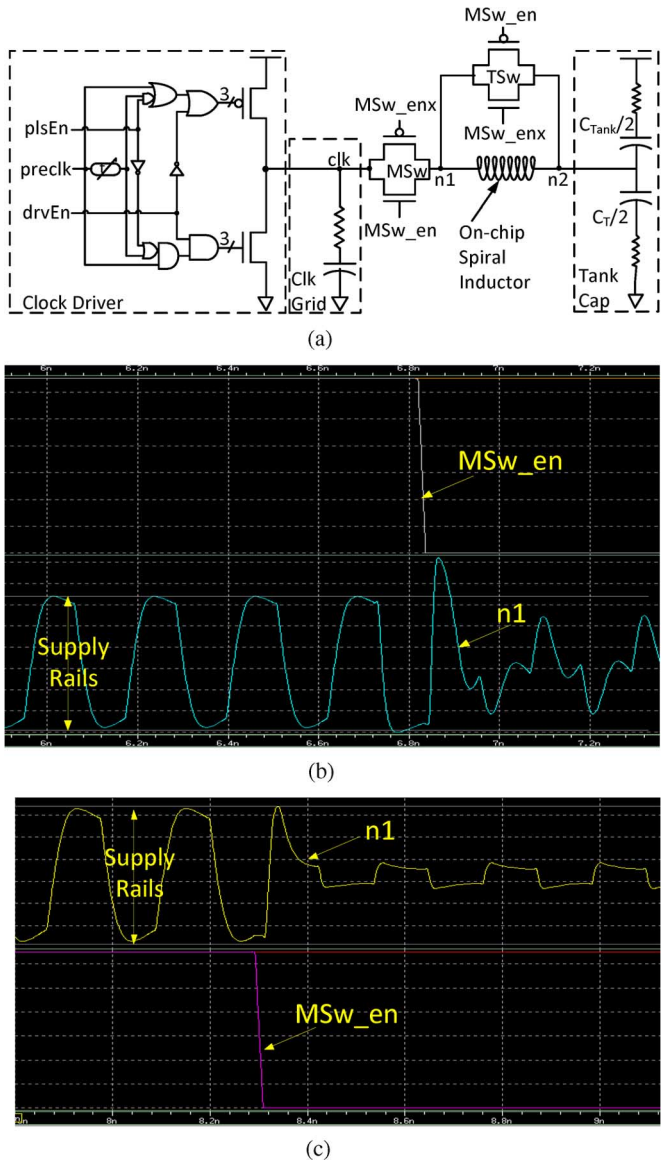


Fig. 10. Use of the throttle switch to limit voltage overshoot when transitioning from rclk to cclk. (a) Simplified rclk system with TSw. (b) Simulation waveforms showing overshoot in the absence of a TSw. (c) Simulation waveforms showing the damping of the overshoot with the TSw.

achieve skew control by interleaving inductors with drivers so each inductor is shared by two strongly connected clock "domains," and each domain is serviced by two inductors. As a result of the skew optimization efforts and the interleaved driver-inductor configuration, the skew associated with rclk (7.2 ps) was controlled to within 1 ps of cclk-skew(6.3 ps) in full-chip clock simulations.

VIII. Measurement Results

Here, we discuss measurement results pertaining to the energy efficiency, frequency overhead, and functional stability of the dual-mode clock in the Piledriver core. Piledriver parts successfully ran system stress test (SST), which stresses the stability of the rclk feature through several weeks of continuous targeted operation. Energy efficiency measurements obtained from automatic test equipment (ATE) on multiple parts are discussed. Also discussed are measurement results obtained from 32 parts

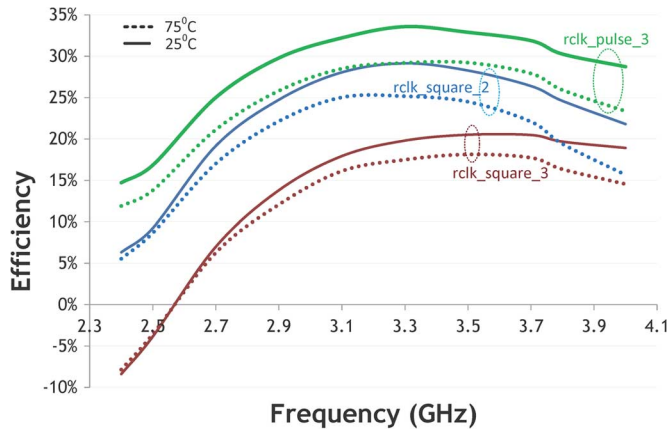


Fig. 11. Plot of energy efficiency versus. frequency at 25 °C and 75 °C.

running in the hybrid system test (HST) environment, which typically is used to determine the product shipping frequencies.

Fig. 11 compares energy efficiency and frequency for various clock modes at 25 °C and 75 °C. Suffixes in the clock mode names refer to a drive modulation, so rclk_sq_3 corresponds to a drive strength modulation of 3/7 in the driver. Efficiency is defined as the percentage savings in clock power obtained relative to the conventional clock-mode implementation. As expected pulse-mode is the most efficient of the three modes, with a peak efficiency of 35% at 25 °C and 29% at 75 °C while operating at 3.4 GHz. Improved efficiency at 25 °C is expected due to reduced resistance in the inductor winding and the clock distribution network. The rclk_square_2 mode achieves a peak efficiency of 29% at 25 °C and more than 25% at 75 °C. Each frequency point in Fig. 11 corresponds to a specific operating voltage, as defined by the voltage-frequency table for the part. The contribution of the voltage-dependent MSw resistance compounds the natural asymmetry in the efficiency–frequency relationship around the natural frequency.

Another notable trend in the efficiency curves is the more pronounced degradation of efficiency at frequencies beyond the natural frequency at higher temperatures. This is likely due to the increased significance of crossover current at higher operating temperatures, magnified by the reduced slew of the resonant clock waveform.

The efficiency of rclk displays only a slight variation across test patterns. Patterns with larger switching activity result from a smaller percentage of gaters being turned off. Such patterns cause a higher crossover current overhead in the switching clock gaters due to the reduced rclk slew, which can be 50%–70% of that in cclk. On the other hand, the increasing proportion of switching clock gaters increases the clock capacitance of the grid due to the Miller capacitance contribution to clock load, allowing for additional savings from resonant clocking. It has been observed, however, that patterns with higher switching activity are more efficient than low-activity patterns, indicating that the Miller capacitance effect more than compensates for the increased crossover current.

The reduced rclk slew rates in comparison with cclk have a potential timing impact on the core. Static timing analysis was performed with rclk slew rates at design time and any newly resulting critical paths were fixed. Fig. 12 shows maximum operating frequency ($F_{\max}$) data obtained from a sample of 32 parts.

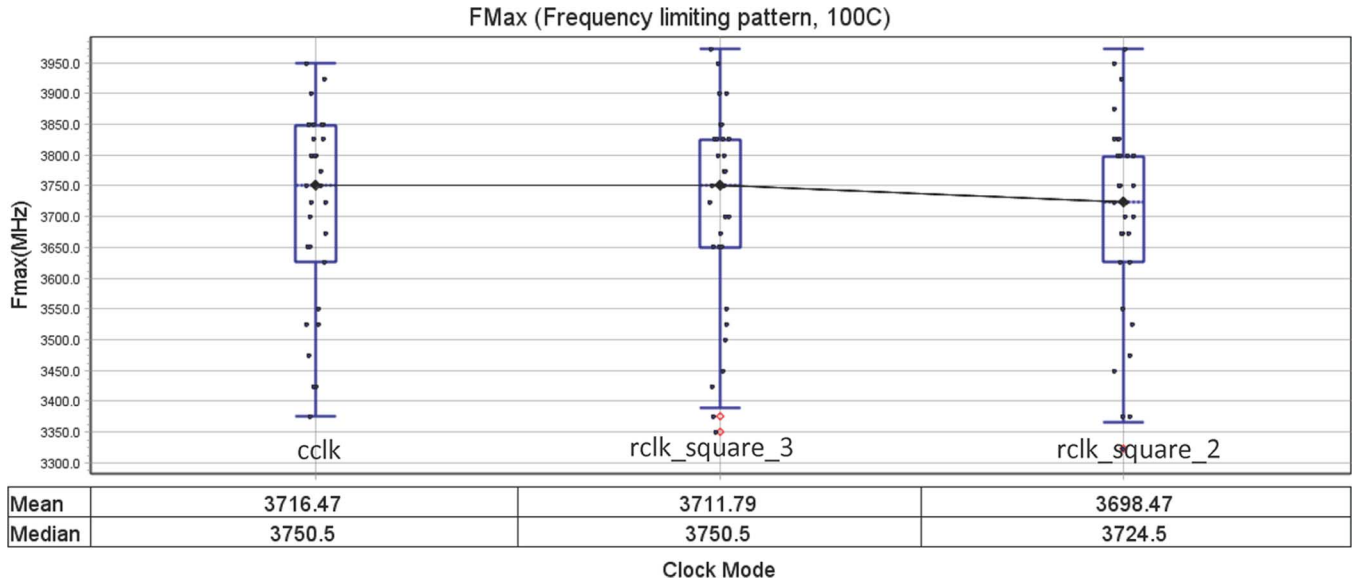


Fig. 12. $F_{\max}$ data obtained from 32 Piledriver cores in cclk, rclk_square_2, and rclk_square_3 modes.

The median frequency overhead with the rclk_square_3 mode is 0 MHz for this sample set, with a 5-MHz mean frequency impact. The mean frequency impact operating the rclk_square_2 mode is 18 MHz. As a result of the significant phase offset between the core clock in rclk_pulse mode and the conventional L2 and NB interface clocks (which are phase aligned with cclk), rclk_pulse-modes see a significant frequency impact in this implementation, and are infeasible for operation. This phase offset is being corrected in an upcoming implementation.

IX. Conclusion

We have presented the implementation of resonant clocking on a high-volume x86-64 AMD microprocessor codenamed Piledriver. To achieve energy-efficient clocking while supporting a wide range of operating frequencies, a dual-mode clock system was implemented. At frequencies sufficiently below the natural frequency of the resulting *LC* tank circuit, the core operates in conventional mode (cclk), while operating in resonant (rclk) mode at higher frequencies.

Various challenges pertaining to the implementation of integrated inductors without a keep-out region were addressed through judicious inductor and power grid design. Subtractive pulse-mode resonant clocking was also introduced to allow for increased efficiency while supporting key features such as PLL duty-cycle tuning and off-PState core operation. Offering a 6.5% clock capacitance overhead, the Piledriver resonant clocking implementation achieved a peak efficiency of 29% at 25 °C and 25% at 75 °C. A 25% reduction in clock power translates to a 4.5% reduction in average application core power and a 10% reduction in idle power.

The pulse mode achieves a peak energy efficiency of 35% but was not production-ready due to the timing impact arising from a conventional L2 and North Bridge interface clocks, and the significant phase offset introduced by the feature in comparison to cclk. This phase offset is being corrected in a current implementation. The frequency impact of the rclk_square modes was found to be marginal, with a 5–18 MHz measured mean frequency overhead over cclk, depending on the resonant clock mode chosen.

Acknowledgment

The authors would like to thank M. Bhoopathy, K. Viau, T. Meneghini, J. Kim, J. Kao, F. Brauchler, A. Arakawa, S. Obaidulla, K. Hurd, V. Palisetti, and D. Renfrow for their valuable contribution to this work.

References

[1] S. C. Chan, P. J. Restle, T. J. Bucelot, J. S. Liberty, S. Weitzel, J. M. Keaty, B. Flachs, R. Volant, P. Kapusta, and J. S. Zimmerman, "A resonant global clock distribution for the cell broadband engine processor," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 64–72, Jan. 2009.

[2] V. S. Sathe, J. Y. Chueh, and M. C. Papaefthymiou, "Energy-efficient GHz-class charge recovery logic," *IEEE J. Solid-State Circuits*, vol. 42, no. 1, pp. 38–47, Jan. 2007.

[3] F. O'Mahony, C. P. Yue, M. A. Horowitz, and S. S. Wong, "A 10-Ghz global clock distribution using coupled standing-wave oscillators," *IEEE J. Solid-State Circuits*, vol. 38, no. 11, pp. 1813–1820, Nov. 2003.

[4] J. Wood, T. Edwards, and S. Lipa, "Rotary traveling-wave oscillator arrays: A new clock technology," *IEEE J. Solid-State Circuits*, vol. 36, no. 11, pp. 1654–1665, Nov. 2001.

[5] A. J. Drake, K. J. Nowka, T. Y. Nguyen, J. L. Burns, and R. B. Brown, "Resonant clocking using distributed parasitic capacitance," *IEEE J. Solid-State Circuits*, vol. 39, no. 9, pp. 1520–1528, Sep. 2004.

[6] V. S. Sathe, J. C. Kao, and M. C. Papaefthymiou, "Resonant-clock latch-based design," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 864–873, Apr. 2008.

[7] C. Ziesler, J. Kim, V. Sathe, and M. Papaefthymiou, "A 225 MHz resonant clocked ASIC chip," in *Proc. Int. Symp. Low-Power Electron. Design*, Aug. 2003, pp. 48–53.

[8] W. Athas, N. Tzartzanis, W. Mao, R. Lai, K. Chong, L. Peterson, and M. Bolotski, "Clock-powered CMOS VLSI graphics processor for embedded display controller application," in *Proc. Int. Solid State Circuits Conf.*, Feb. 2000, pp. 296–297.

[9] S. Kim and M. C. Papaefthymiou, "Single-phase source-coupled adiabatic logic," in *Proc. Int. Symp. Low-Power Electron. Design*, Aug. 1999, pp. 97–99.

[10] W.-H. Ma, J. C. Kao, V. S. Sathe, and M. C. Papaefthymiou, "A 187 MHz subthreshold-supply robust FIR filter with charge-recovery logic," *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 793–80, Apr. 2010.

[11] J. Kao, W. H. Ma, S. Kim, and M. C. Papaefthymiou, "2.07 GHz floating-point unit with resonant-clock precharge logic," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Athens, Greece, Sep. 2009, pp. 160–163.

[12] M. Hansson, B. Mesgarzadeh, and A. Alvandpour, "1.56 GHz on-chip resonant clocking in 130 nm CMOS," in *Proc. Custom Integr. Circuits Conf.*, Sep. 2006, pp. 241–244.
[13] A. Ishii, J. Kao, V. Sathe, and M. C. Papaefthymiou, "A resonant-clock 200 MHz ARM926EJ-s™ microcontroller," in *Proc. Eur. Solid-State Circuits Conf.*, Sep. 2009, pp. 356–359.
[14] V. S. Sathe, "Hybrid resonant-clocked digital design," Ph.D. dissertation, Dept. Electr. Eng. Comput. Sci., Univ. of Michigan, Ann Arbor, May 2007.
[15] H. McIntyre, S. Arekapudi, E. Busta, T. Fischer, M. Golden, A. Horiuchi, T. Meneghini, S. Naffziger, and J. Vinh, "Design of the two-core x86-64 AMD bulldozer module in 32 nm SOI CMOS," *IEEE J. Solid-State Circuits*, vol. 47, no. 1, pp. 164–176, Jan. 2012.
[16] R. Jotwani, S. Sundaram, S. Kosonocky, A. Schaefer, V. F. Andrade, A. Novak, and S. Naffziger, "An x86-64 core in 32 nm SOI CMOS," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 162–172, Jan. 2011.
[17] S. Ramo, J. R. Whinnery, and T. V. Duzer, *Fields and Waves in Communication Electronics*, 3rd ed. Reading, MA: Addison-Wesley, 1993.

Visvesh S. Sathe (S02–M07) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Bombay, India, in 2001, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of Michigan, Ann Arbor, in 2004 and 2007, respectively.

While with the University of Michigan, his research focused on low-energy circuit design with a particular emphasis on circuits and architectures for resonant-clocked digital design. He has held internship positions with the IBM T. J. Watson Research Center and Cyclos Semiconductor. In 2007, he joined the Low Power Advanced Development Group, Advanced Micro Devices, Fort Collins, CO, where he is currently a Member of Technical Staff working on the exploration and implementation of low-power technologies for future microprocessors. He has authored or coauthored 20 publications and holds six patents.

Dr. Sathe presently serves on the technical program committees of CICC and the International Conference on VLSI Design.

Srikanth Arekapudi (M'12) received the B.Tech. degree in electronics and communication engineering from the National Institute of Technology, Warangal, India, in 1999, the M.S. degree in electrical and computer engineering from the University of Massachusetts at Amherst in 2001, and the D.Eng. degree in electrical engineering from Stanford University, Stanford, CA, in 2004.

He was with Agilent Technologies and Silicon Graphics Inc. before he joined Advanced Micro Devices (AMD), Sunnyvale, CA, in 2003. At AMD, he has contributed to several custom circuit and stdcell designs for K8 and Bulldozer-based projects. In the past, he served as a lead for Integer Execution Unit and High Speed Clock Design. He is currently a Principal Member of Technical Staff involved in the design of next-generation x86 processors.

Alexander T. Ishii (M'12) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge.

While with the Massachusetts Institute of Technology, he researched algorithms for the optimization of VLSI system timing. With over 20 years in a variety of research, commercial development, and management positions, he has pursued his interest in the realization of practical technologies for high-performance computing and communication systems and accumulated technical publications and patents in areas ranging from telecommunications network protocol stacks to ultra-low-power circuits. Since 2004, he has served as the Vice President of Engineering with Cyclos Semiconductor, Inc., Berkeley, CA, where he has led the development of resonant-clocking technologies which enable SoC devices to achieve new levels of energy efficiency.

Charles Ouyang (M'08) received the B.S. and M.S. degrees in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, in 1993 and 1998, respectively.

He has been with Advanced Micro Device, Sunnyvale, CA, since 2006, where he is currently a Member of Technical Staff on the DFx design team.

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 an Assistant Professor with Yale University, he joined the University of Michigan, Ann Arbor, where he is currently a Professor of electrical engineering and computer science and Chair of the Computer Science and Engineering Division. He is also cofounder of Cyclos Semiconductor Inc., Berkeley, CA, a startup company commercializing resonant clocking design technologies for energy-efficient high-performance semiconductor devices. 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.

Dr. Papaefthymiou was the recipient of an ARO Young Investigator Award, the National Science Foundation CAREER Award, and a number of IBM Partnership Awards. Furthermore, together with his students, he has received Best Paper awards at leading conferences, including the ACM/IEEE Design Automation Conference and the IEEE International Symposium on High-Performance Computer Architecture. He has served multiple terms as an associate editor for the IEEE TRANSACTIONS ON THE COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS, the IEEE TRANSACTIONS ON COMPUTERS, and the IEEE TRANSACTIONS ON VERY LARGE-SCALE INTEGRATION (VLSI) SYSTEMS. He has served as the General Chair and as the 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 on the Technical Program Committee of the IEEE/ACM International Conference on Computer-Aided Design.

Samuel Naffziger (SM'10) received the B.S.E.E. degree from the California Institute of Technology, Pasadena, in 1988, and the M.S.E.E. degree from Stanford University, Stanford, CA, in 1993.

He joined Hewlett-Packard in 1988 and spent eight years working on PA-RISC processor development including floating point, out-of-order execution and circuit methodologies. He then became part of the Itanium2 Joint Development Team with Intel Corporation, Fort Collins, CO, and led the design of both the first Itanium2 processor (McKinley) and Montecito. In 2006, he helped start the Mile High Design Center of Advanced Micro Devices (AMD), Fort Collins, CO, to work on next-generation processor designs. He holds 101 U.S. patents on processor circuits and architecture and has authored or coauthored over 30 IEEE publications and presentations.

Mr. Naffziger chaired the Digital subcommittee of the International Solid-State Circuits Conference for 5 years, and is a Corporate Fellow at AMD.