CACTI-IO: CACTI With OFF-chip Power-Area-Timing Models

Norman P. Jouppi, *Fellow, IEEE*, Andrew B. Kahng, *Fellow, IEEE*, Naveen Muralimanohar, *Member, IEEE*, and Vaishnav Srinivas, *Member, IEEE*

***Abstract*— In this paper, we describe CACTI-IO, an extension to CACTI that includes power, area, and timing models for the IO and PHY of the OFF-chip memory interface for various server and mobile configurations. CACTI-IO enables design space exploration of the OFF-chip IO along with the dynamic random access memory and cache parameters. We describe the models added and four case studies that use CACTI-IO to study the tradeoffs between memory capacity, bandwidth (BW), and power. The case studies show that CACTI-IO helps to: 1) provide IO power numbers that can be fed into a system simulator for accurate power calculations; 2) optimize OFF-chip configurations including the bus width, number of ranks, memory data width, and OFF-chip bus frequency, especially for novel buffer-based topologies; and 3) enable architects to quickly explore new interconnect technologies, including 3-D interconnect. We find that buffers on board and 3-D technologies offer an attractive design space involving power, BW, and capacity when appropriate interconnect parameters are deployed.**

***Index Terms*— CACTI, CACTI-IO, dynamic random access memory (DRAM), IO, memory interface, power and timing models.**

I. Introduction

THE interface to the dynamic random access memory (DRAM), including the PHY, I/O circuit (IO), and interconnect, is becoming increasingly important for the performance and power of the memory subsystem [18]–[20], [31], [38], [44]. As capacities scale faster than memory densities [8], there is an ever-increasing need to support a larger number of memory dies, especially for high-end server systems [36], often raising cooling costs. Mobile systems can afford to use multichip package or stacked-die point-to-point memory configurations; by contrast, servers have traditionally relied on a dual-inline memory module (DIMM) to support larger capacities. With modern server memory sizes exceeding 1 TB, the contribution of memory power can reach 30%–57% of total server power [44], with a sizable fraction (up to 50% in some systems) coming from the OFF-chip interconnect. The memory interface incurs performance bottlenecks due to challenges with interface bandwidth (BW) and latency. The BW of the interface is limited by: 1) the data rate, owing to the DRAM interface timing closure, signal integrity over the interconnect, and limitations of source-synchronous signaling [4], [48]; and 2) the width of the bus, which is often limited by size and the cost of package pins.

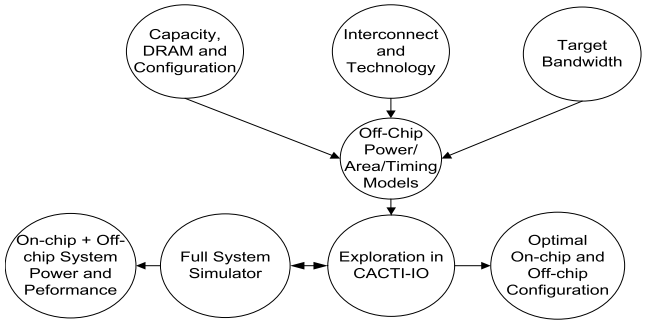


Fig. 1. CACTI-IO: OFF-chip modeling and exploration within CACTI.

CACTI [5] is an analytical memory modeling tool, which can calculate delay, power, area, and cycle time for various memory technologies. For a given set of input parameters, the tool performs a detailed design space exploration across different array organizations and on-chip interconnects, and outputs a design that meets the input constraints. CACTI-D [22] is an extension of CACTI with on-chip DRAM models.

In this paper, we describe CACTI-IO [1], an extension to CACTI, shown in Fig. 1. CACTI-IO allows the user to describe the configuration(s) of interest, including the capacity and organization of the memory dies, target BW, and interconnect parameters. CACTI-IO includes analytical models for the interface power, including suitable lookup tables for some of the analog components in the PHY. It also includes voltage and timing uncertainty models that help relate parameters that affect power and timing. Voltage and timing budgets are traditionally used by interface designers to begin building components of the interface [2], [4], [41], [49] and budget the eye diagram between the DRAM, interconnect, and the controller, as shown in Fig. 2. The eye mask represents the portion of the eye budgeted for the Rx (receiver). The setup/hold slacks and noise margins represent the budgets for the interconnect and the Tx (transmitter).

Final optimization of the IO circuit, OFF-chip configuration, and signaling parameters requires detailed design of circuits

Manuscript received September 18, 2013; revised April 4, 2014; accepted April 28, 2014. Date of publication August 22, 2014; date of current version June 23, 2015.

N. P. Jouppi is with Google, Mountain View, CA 94043 USA (e-mail: norm.jouppi@gmail.com).

A. B. Kahng is with the Department of Computer Science and Engineering and the Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA 92093 USA (e-mail: abk@ucsd.edu).

N. Muralimanohar is with Hewlett-Packard Laboratories, Palo Alto, CA 94304 USA (e-mail: naveen.muralimanohar@hp.com).

V. Srinivas is with the Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA 92093 USA (e-mail: vaishnav@ucsd.edu).

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/TVLSI.2014.2334635

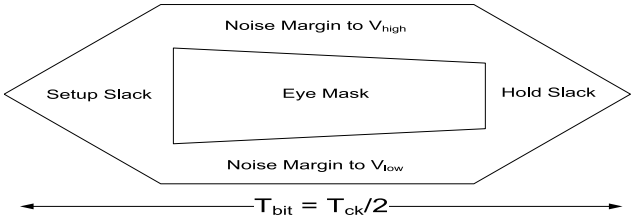


Fig. 2. Memory interface eye diagram for voltage and noise budgets.

along with SPICE analysis, including detailed signal integrity and power integrity analyses; this can take months for a new design [4]. CACTI-IO is not a substitute for detailed analyses, but rather serves as a quick estimate for the system architect to enable the right tradeoffs between the large number of nontrivial IO and OFF-chip parameters. Up-front identification of the OFF-chip design space at an architectural level is crucial for driving next-generation memory interface design.

The main objectives for the CACTI-IO tool are as follows.

a) Obtain IO power numbers for different topologies and modes of operation that can be fed into a full-system simulator: The tradeoffs between performance, power, and capacity in the memory subsystem are nontrivial [17], [22], but previous studies often do not explore alternatives to a standard Double Data Rate 3 (DDR3) configuration for the memory interface. Furthermore, most of the modeling tools, including McPAT [21] and DRAMSIM [35], do not model the interface power and timing, and have no visibility into the details of the PHY and IO. CACTI-IO provides IO power numbers for read, write, idle (only clock active), and sleep modes that can easily be integrated into a system simulator. This enables architects to compare on- and OFF-chip sources of power across modes.

b) Enable co-optimization of off- and on-chip power and performance, especially for new OFF-chip topologies: Historically, OFF-chip parameters, (i.e., signaling properties and circuit parameters) have been limited to standardized configurations including DIMMs, with operating voltage, frequency, data rates, and IO parameters strictly governed by standards. A major drawback and design limiter—especially when operating at high frequencies—in this simplistic design context is the number of DIMMs that can be connected to a channel. This often limits memory capacity, creating a memory wall. Recent large enterprise servers and multicore processors instead use one or more intermediate buffers to expand capacity and alleviate signal integrity issues. Such a design still adheres to DRAM standards but has more flexibility with respect to the interconnect architecture that connects memory and compute modules, including serial interfaces between the buffer and the CPU. While current and future memory system capacity and performance greatly depend on various IO choices, to date there is no systematic way to identify the optimal OFF-chip topology that meets a specific design goal, including capacity and BW. CACTI-IO provides a way for architects to systematically optimize IO choices in conjunction with the rest of the memory architecture. Below, we illustrate how CACTI-IO can help optimize a number of OFF-chip parameters—number of ranks (fanout on the data bus), memory data width, bus frequency, supply voltage, address bus fanout, and bus width—for given capacity and BW requirements. CACTI-IO can also be used to evaluate the number of buffers needed in complex, high-end memory configurations, along with their associated overheads.

c) Enable exploration of emerging memory technologies: With the advent of new interconnect and memory technologies, including 3-D through-silicon stacking (TSS)-based interconnect being proposed for DRAM [40] as well as new memory technologies such as magnetic RAM and phase-change RAM (PCRAM) [43], architects are exploring novel memory architectures involving special off-chip caches and write buffers to filter writes or reduce write overhead. Most of the emerging alternatives to DRAM suffer from high write energy or low write endurance. The use of additional buffers plays a critical role in such OFF-chip caches, and there is a need to explore the changing ON- and OFF-chip design space. When designing new OFF-chip configurations, many new tradeoffs arise based on the choice of OFF-chip interconnect, termination type, number of fanouts, operating frequency, and interface type (serial versus parallel). CACTI-IO provides flexible baseline IO models that can be easily tailored to new technologies and used to explore tradeoffs at a system level.

In summary, the key contributions of this paper are:

1) models for power, area, and timing of the IO, PHY, and interconnect for server and mobile configurations;
2) CACTI-IO, an extension to CACTI that includes these models;
3) four industry-driven case studies that use CACTI-IO to optimize parameters of the OFF-chip topology, including the number of ranks and memory data width.

In the remainder of this paper, Section II describes the interface models, including those for power, voltage margins, timing margins, and area. Section III describes how the models can be ported for a different technologies. Section IV shows comparisons of the model against SPICE. Section V presents CACTI-IO using four case studies, showing a summary of the power and timing as well as optimal OFF-chip configurations. Section VI summarizes our conclusion.

II. IO, PHY, and Interconnect Models

In this section, we give complete details of the IO, PHY, and interconnect models included in CACTI-IO. Power and timing models for interconnect and terminations have been well documented and validated over [2], [3], and [7]. Our goal here is to show the framework of the baseline models, which can then be adapted to any customized configuration needed, including new interconnect technologies.

A. Power Models

Power is calculated for four different modes: WRITE (peak activity during WRITE), READ (peak activity during READ), idle (no data activity, but clock is enabled and terminations are on), and sleep (clock and terminations are disabled, in addition to no data activity). The mode of the OFF-chip interconnect can be chosen by setting the iostat input parameter to W (WRITE), R (READ), I (IDLE), or S (SLEEP). CACTI-IO OFF-chip power models include the following.

1) Dynamic IO Power: The switching power at the load capacitances is described in (1), where N_{pins} is the number of signal pins; D_c is the duty cycle when the link is enabled; α is the activity factor for the signal switching (number of 0 to 1 transitions per clock period, i.e., $\alpha = 1$ for a clock signal); i denotes various nodes along the interconnect, with possibly different swings in a terminated or low-swing scheme; C_{Total_i} is the capacitance at node i; V_{sw_i} is the swing of the signal at node i; V_{dd} is the supply voltage; and f is the frequency of operation

$$P_{\text{dyn}} = N_{\text{pins}} D_c \alpha \left(\sum_i C_{\text{Total}_i} V_{sw_i}\right) V_{\text{dd}} f. \tag{1}$$

2) Interconnect Power: The power dissipated on the interconnect ($P_{\text{dyn\_interconnect}}$) is given by (2). Energy/bit $\left(E_{\text{bit}}^{\text{interconnect}}\right)$ is given by (3), where Z_0 is the characteristic impedance of the line, t_L is the flight time (time taken for the signal to traverse the line length), and t_b is the bit period. For high-end servers, generally $2t_L > t_b$ since the interconnect is long, while for mobile configurations, generally $2t_L < t_b$. For an FR-4-based interconnect used on printed circuit boards, t_L is approximately 180 ps/in. The interconnect is generally modeled as a transmission line when $t_L > t_r/3$ (t_r is the rise-time of the signal), unlike an on-die RC network [3]

$$P_{\text{dyn\_interconnect}} = N_{\text{pins}} D_c \alpha E_{\text{bit}}^{\text{interconnect}} f \tag{2}$$

$$E_{\text{bit}}^{\text{interconnect}} = \begin{cases} \frac{t_L V_{\text{sw}} V_{\text{dd}}}{Z_0} & \text{if } 2t_L \le t_b \\ \frac{t_b V_{\text{sw}} V_{\text{dd}}}{Z_0} & \text{if } 2t_L > t_b. \end{cases} \tag{3}$$

3) Termination Power: The IO termination power is provided for various termination options, including unterminated (as used in LPDDR2 and wide-IO), center-tap (as used in DDR3), IO Voltage Supply (VDDQ) (as in DDR4), and differential terminations (as used in Mobile X Data Rate). The voltage swing set by the terminations is fed into the dynamic power equation described in (1).

The termination power is then calculated for source and far-end terminations. $P_{\text{term\_ol}}$ is the termination power when the line is driven to 0 (V_{ol}), and $P_{\text{term\_oh}}$ is the termination power when the line is driven to 1 (V_{oh}). The average power is reported assuming that 0 and 1 are equiprobable during peak activity. V_{dd} is the supply voltage, V_{TT} is the termination voltage and R_{TT} is the termination resistance

$$P_{\text{term\_oh}} = (V_{\text{dd}} - V_{\text{TT}})(V_{\text{oh}} - V_{\text{TT}})/R_{\text{TT}} \tag{4}$$

$$P_{\text{term\_ol}} = V_{\text{TT}}(V_{\text{TT}} - V_{\text{ol}})/R_{\text{TT}} \tag{5}$$

$$P_{\text{avg}} = (P_{\text{term\_oh}} + P_{\text{term\_ol}})/2 \tag{6}$$

$$P_{\text{Totavg\_term}} = \sum P_{\text{avg}}. \tag{7}$$

Terminations are used to improve signal integrity and achieve higher speeds, and the values depend on the interconnect length as well as the frequency or timing requirements. Terminations on the DQ (data) bus typically use an on-die termination (ODT) scheme, while those on the command-address (CA) bus use a fly by termination scheme to the multiple loads. Figs. 3 and 4 show the DDR3 DQ and CA termination schemes along with the static current consumed by them as used in micrometer's power calculator [23].

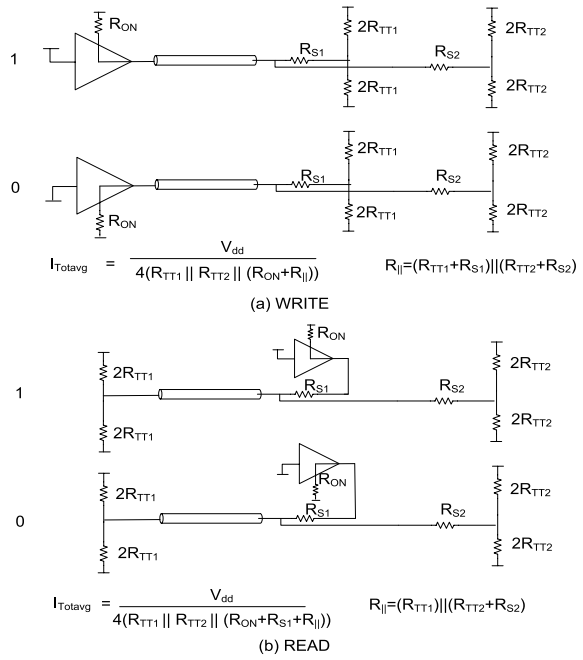


Fig. 3. DDR3 DQ dual-rank termination.

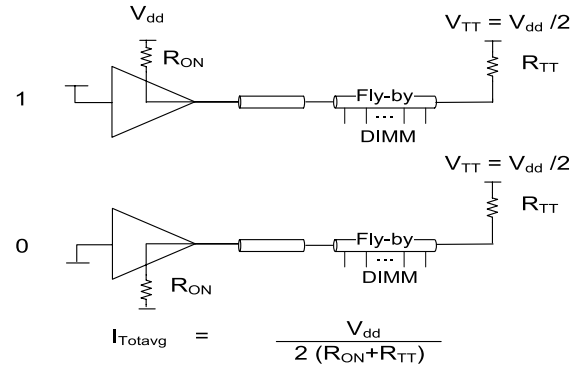


Fig. 4. DDR3 CA termination.

a) Unterminated: No termination power.

b) Center-tap termination, as in DDR3: The DQ WRITE, DQ READ, and CA powers are described in (8)–(10), respectively. R_{ON} is the driver impedance, R_{TT1} and R_{TT2} are the effective termination impedance of the used and unused ranks, respectively. $R_{||}$ is the effective impedance of both the ranks observed together. For the CA case, R_{TT} is the effective fly by termination. R_{S1} and R_{S2} are the series resistors used for better signal integrity

$$P_{\text{DQ\_Term}} = \frac{V_{\text{dd}}^2}{4} \cdot \left(\frac{1}{R_{\text{TT1}}} + \frac{1}{R_{\text{TT2}}} + \frac{1}{R_{\text{ON}} + R_{||}}\right) \tag{8}$$

$$P_{\text{DQ\_Term}} = \frac{V_{\text{dd}}^2}{4} \cdot \left(\frac{1}{R_{\text{TT1}}} + \frac{1}{R_{\text{TT2}}} + \frac{1}{R_{\text{ON}} + R_{S1} + R_{||}^{\text{read}}}\right) \tag{9}$$

$$P_{\text{CA\_Term}} = \frac{V_{\text{dd}}^2}{4} \cdot \left(\frac{1}{50 + R_{\text{TT}}}\right). \tag{10}$$

CACTI-IO calculates the voltage swing as follows. This calculation feeds into the dynamic power calculation of (1). The swing is calculated at the two loads and on the line as shown in Fig. 3 for both WRITE and READ modes.

WRITE

$$V_{\text{sw-line}} = \frac{V_{\text{dd}} \cdot R_{||}}{(R_{\text{ON}} + R_{||})} \tag{11}$$

$$V_{\text{sw-load1}} = \frac{V_{\text{dd}} \cdot R_{\text{TT1}}(R_{\text{S2}} + R_{\text{TT2}})}{(R_{\text{S1}} + R_{\text{TT1}} + R_{\text{S2}} + R_{\text{TT2}})(R_{\text{ON}} + R_{||})} \tag{12}$$

$$V_{\text{sw-load2}} = \frac{V_{\text{dd}} \cdot R_{\text{TT2}}(R_{\text{S1}} + R_{\text{TT1}})}{(R_{\text{S1}} + R_{\text{TT1}} + R_{\text{S2}} + R_{\text{TT2}})(R_{\text{ON}} + R_{||})} \tag{13}$$

$$\text{where } R_{||} = (R_{\text{TT1}} + R_{\text{S1}})||(R_{\text{TT2}} + R_{\text{S2}}). \tag{14}$$

READ

$$V_{\text{sw-line}} = \frac{V_{\text{dd}} \cdot R_{||}^{\text{read}}}{(R_{\text{ON}} + R_{\text{S1}} + R_{||}^{\text{read}})} \tag{15}$$

$$V_{\text{sw-load1}} = \frac{V_{\text{dd}} \cdot R_{\text{TT1}}(R_{\text{S2}} + R_{\text{TT2}})}{(R_{\text{TT1}} + R_{\text{S2}} + R_{\text{TT2}})(R_{\text{ON}} + R_{\text{S1}} + R_{||}^{\text{read}})} \tag{16}$$

$$V_{\text{sw-load2}} = \frac{V_{\text{dd}} \cdot R_{\text{TT2}} R_{\text{TT1}}}{(R_{\text{TT1}} + R_{\text{S2}} + R_{\text{TT2}})(R_{\text{ON}} + R_{\text{S1}} + R_{||}^{\text{read}})} \tag{17}$$

$$\text{where } R_{||}^{\text{read}} = (R_{\text{TT1}})||(R_{\text{TT2}} + R_{\text{S2}}). \tag{18}$$

c) Differential termination for low-swing differential interfaces: The power for a typical differential termination scheme is as follows:

$$P_{\text{diff\_term}} = 2 \cdot V_{\text{dd}} V_{\text{sw}} / R_{\text{TT}}. \tag{19}$$

In some cases, differential low-swing transmitter circuits could use a small voltage-regulated supply to generate a voltage-mode output [38]. In such a situation, the termination power would be one half of the value given in (19).

d) VDDQ and VSSQ terminations: We next present a power equation for a VDDQ-termination for DDR4 [26] and LPDDR3 [27]. The DDR4 and LPDDR3 specifications use a VDDQ termination scheme [28], i.e., a single termination resistor connected to the VDDQ supply. This is similar to other pseudo-open-drain schemes used by JEDEC [28]. The equations for the voltage swing for such a termination scheme are the same as for DDR3 above in (11)–(18). However, the signal is referenced to VDDQ rather than VDDQ/2, resulting in the power equation of (20), where $R_{||}$ is calculated for WRITE and READ modes similar to the DDR3 DQ case [(14) and (18)]. The power shown in (20) assumes 50% 0s and 50% 1s on the line. It must be noted that driving a 1 in this case results in no termination power.

TABLE I
PHY ACTIVE DYNAMIC POWER PER BIT FOR 3-D CONFIGURATIONS

Building Block	Dynamic Power (mW/Gbps)		
	500 Mbps	1 Gbps	2 Gbps
Datapath	0.1	0.2	0.5
Phase Rotator	N/A	0.1	0.2
Clock Tree	0.05	0.2	0.4
Duty Cycle Correction	N/A	N/A	0.05
Deskewing	N/A	N/A	0.1
PLL	N/A	N/A	0.1

TABLE II
PHY STATIC POWER FOR A ×128 3-D CONFIGURATION

Building Block	Static Power (mW)		
	500 Mbps	1 Gbps	2 Gbps
Phase Rotator	N/A	1	10
PLL	N/A	N/A	5

The CA termination would be similar to the DDR3 fly by scheme

$$P_{\text{DQ\_Term}} = 0.5 \cdot V_{\text{dd}}^2 \cdot \left(\frac{1}{R_{\text{ON}} + R_{||}}\right). \tag{20}$$

Termination schemes that are VDDQ or IO Ground (VSSQ) terminated can benefit from significant idle power reductions by idling the bus at the same polarity of the termination. LPDDR3 supports the unterminated, full-swing interface as well.

4) PHY Power: The PHY includes analog and digital components used to retime the IO signals on the interface. A wide range of implementations [18]–[20], [31]–[33] exist for the PHY that vary in power and are fine-tuned to specific design requirements. Currently, the user can change the inputs for the PHY power based on a specific implementation. Tables I and II, respectively, show the active dynamic power per bit and static power for the entire PHY of an example PHY implementation for a x128 3-D configuration. The building blocks are representative of typical PHY components [18]–[20], [31]–[33]. Table III shows the dynamic and static power, for example DDR3-1600 PHY. At lower data rates, certain components are not required, indicated by N/A in Tables I and II.

The building blocks listed include blocks that typically retime a source-synchronous interface using a forwarded clock scheme [2]. The datapath refers to the data transmit path until the input to the IO Tx and the data receive path after the IO Rx. The phase rotator is a delay element used to generate a T/4 delay to center-align the data-strobe (DQS) with respect to the data (DQ) pins. It could be a Delay Locked Loop or any other delay element that meets the requirements on the edge placement error (T_{error}) described in Section II. The clock tree is the local clock-tree within the PHY that distributes the clock to all the bit lanes. The Rx refers to the IO receiver, which typically consumes some static power for DDR3 stub-series terminated logic, owing to a pseudodifferential V_{ref}-based receiver first stage. Some PHY implementations have a duty cycle correction that corrects duty-cycle distortion, deskewing that reduces static skew offsets, write/read leveling that lines up the various data byte lanes with the fly by

TABLE III
PHY DYNAMIC POWER PER BIT AND STATIC POWER FOR A ×64 DDR3-1600

Building Block	Dynamic Power (mW/Gbps)	Static Power (mW)
Datapath	0.5	0
Phase Rotator	0.01	10
Clock Tree	0.05	0
Rx	0.5	10
Duty Cycle Correction	0.05	0
Deskewing	0.1	0
Write/Read Leveling	0.05	0
PLL	0.05	10

TABLE IV
PHY WAKEUP TIMES FROM SLEEP AND IDLE MODES

Building Block	Sleep to Active	Idle to Active
PLL	10 μs	0
Phase Rotator	5 μs	0
Rx	2 μs	2 ns
Bandgap	10 μs	0
Deskewing	3 ns	0
Vref Generator	0.5 μs	0

clock and a phase-locked loop dedicated for the memory interface. The static skew ($T_{\text{skew\_setup}}$ and $T_{\text{skew\_hold}}$) on the interface and the duty-cycle distortion (T_{DCD}) can be reduced if the PHY implements a deskewing scheme and a duty-cycle corrector.

Specific implementations could have other blocks not listed here, but the framework supports easy definition of dynamic and static active and idle power for each of the building blocks. Each building block in the PHY has an idle and sleep state, similar to the IO. CACTI-IO provides these PHY parameters for a few standard configurations included within it. If a new PHY architecture is being investigated, the architect will have to work with the PHY datasheet or IP provider to obtain the model inputs. Frequency scaling can be implemented suitably by going into idle and sleep states for the various blocks based on the frequency of operation. These blocks often have wakeup times when entering active mode from idle and sleep modes, and these wakeup times can be modeled within CACTI-IO. Table IV shows example wakeup times for the building blocks in the PHY. The wakeup times fall into a few broad categories.

1) Closed-loop blocks need large (order of microseconds) wakeup times to lock the loop. Sometimes designs try to optimize lock times but tradeoffs with loop dynamics and jitter performance need careful consideration [34].
2) Mixed-signal or analog blocks may need bias setup times, which could range from microseconds to few nanoseconds, depending on the type of bias (e.g., a bandgap or a self-referenced receiver).
3) Digital settings on mixed-signal blocks, e.g., delay line settings or voltage reference settings could change from active to idle and sleep modes. Changing these often requires settling time in the order of a few nanoseconds.
4) Digital datapaths may need clock synchronization during frequency changes, and this could cause a wakeup time of a few clock cycles.

The wakeup time reported by CACTI-IO can be used by a system simulator to consider the latency associated with such frequency scaling.

The above four components of the IO and PHY power are combined as follows, according to the mode of the interface:

1) WRITE or READ

$$P_{\text{Total\_Active}} = P_{\text{dyn}} + P_{\text{dyn\_interconnect}} + P_{\text{term}} + P_{\text{static/bias}}. \quad (21)$$

2) IDLE

$$P_{\text{Total\_Idle}} = P_{\text{term}} + P_{\text{static/bias}} + P_{\text{dyn\_clock}}. \quad (22)$$

3) SLEEP

$$P_{\text{Sleep}} = P_{\text{leakage}}. \quad (23)$$

The duty cycle spent in each mode can be specified using the duty cycle input parameter.

B. Voltage and Timing Margins

The minimum achievable clock period T_{ck} depends on the voltage and timing budgets (i.e., eye diagram and/or BER (bit error rate) compliance).

Traditionally, the memory interface budgets have been based on a worst-case analysis approach shown in Figure 2, where the budgets are divided between the DRAM, the interconnect, and the controller chip or SOC (System-On-Chip). With increasing speeds there is a need for a statistical analysis approach similar to serial links [42] during detailed design analysis. However, for architectural exploration, we continue to use worst-case budgets in our initial framework, with the option of accounting for optimism or pessimism based on prior correlation between the two approaches, or with measurements. This correlation factor also helps address different BER requirements for server DIMM modules that include error correction (ECC) schemes [4], [36], [39].

1) Timing Budgets: The key interface timing equations are based on DRAM AC timing parameters in the JEDEC specification [24], [25]. There are nuances to the system timing based on the controller and PHY design, but most rely on measuring setup and hold slacks to ensure positive margins.

It is interesting to note that while the DQ bus is DDR in almost all DRAMs today, the CA bus is mostly SDR (single data rate), except for LPDDR2 and LPDDR3 where the CA bus is DDR [24], [25]. In addition, the CA bus provides an option for 2T (two clock cycles) and 3T (three clock cycles) timing to relax the requirements when heavily loaded. This is done since the CA bus is typically shared across all memories in the DIMM.

The jitter on the interface is the true limiter of the timing budget, and optimizing the interface for low jitter is the key challenge. The common sources of jitter include Tx jitter, ISI (inter-symbol interference), crosstalk, SSO (simultaneously switching outputs), supply noise, and Rx jitter [4].

Jitter can be estimated from various deterministic (DJ_i) and random (RJ_i) sources as follows [4]. Q_{BER} is a Q-function at the the desired BER [4], and σ_i is the standard deviation of the random source. The user can calculate the jitter at the desired

BER and enter it into the setup and hold timing equations as described in [6, Ch. 2.2].

$$T_{\text{jitter}} = \sum_i DJ_i + \sqrt{\sum_i RJ_i^2} \tag{24}$$

$$RJ_i = 2 \cdot Q_{\text{BER}} \cdot \sigma_i \tag{25}$$

$$T_{\text{jitter}}(\mathbb{F}_0) = T_{\text{jitter\_avg}} + \sum_i \left(T_{\text{jitter}}(F_i = F_{i0}) - T_{\text{jitter\_avg}}\right) \tag{26}$$

Here, factor F_i is a parameter that affects T_{jitter} [4]. $\mathbb{F}_0$ is the value of a set of factors $F_i = F_{i0}$ for which we calculate the jitter, $T_{\text{jitter}}(\mathbb{F}_0)$, as an estimate assuming there is no interaction between the factors F_i [4]. This is done efficiently by running a Design of Experiments (DOE) for a set of orthogonal array experiments as defined by the Taguchi method [4], [30]. $T_{\text{jitter\_avg}}$ represents the average jitter from all the experiments in the orthogonal array, while $T_{\text{jitter}}(F_i = F_{i0})$ represents the average jitter from all experiments where $F_i = F_{i0}$. For cases where F_{i0} is not part of the orthogonal array, a piecewise linear approximation is employed.

2) Voltage Budgets: A voltage budget can be developed for voltage margins as follows [2], which once again is based on a worst-case analysis, where V_N is the voltage noise, K_N is the proportionality coefficient for the proportional noise sources (that are proportional to the signal swing V_{sw}), V_{NI} is the noise due to independent noise sources and V_M is the voltage margin. Crosstalk, ISI (inter-symbol interference), and SSO (simultaneously switching outputs) are typical proportional noise sources [2], while the Rx-offset, sensitivity, and independent supply noise are typical independent noise sources.

$$V_N = K_N \cdot V_{\text{sw}} + V_{NI} \tag{27}$$

$$K_N = K_{\text{xtalk}} + K_{\text{ISI}} + K_{\text{SSO}} \tag{28}$$

$$V_{\text{NI}} = V_{\text{Rx-offset}} + V_{\text{Rx-sens}} + V_{\text{supply}} \tag{29}$$

$$V_M = \frac{V_{\text{sw}}}{2} - V_N \tag{30}$$

A DOE analysis for the voltage noise coefficient, K_N, can be performed in a similar manner as described above for T_{jitter}.

C. Area Models

The area of the IO is modeled as shown below in (31), where N_{IO} is the number of signals, f is the frequency, and R_{ON} and R_{TT1} are the impedance of the IO driver and the ODT circuit, respectively, as shown in Fig. 3. A_0, k_0, k_1, k_2, and k_3 are the constants for a given technology and design. They need to be fitted based on data from the PHY IP provider or datasheet

$$\text{Area}_{\text{IO}} = N_{\text{IO}} \cdot \left(A_0 + \frac{k_0}{\min(R_{\text{ON}}, 2 \cdot R_{\text{TT1}})}\right) + N_{\text{IO}} \cdot \left(\frac{1}{R_{\text{ON}}}\right) \cdot (k_1 * f + k_2 * f^2 + k_3 * f^3). \tag{31}$$

The area of the last stage of the driver is proportional to $1/R_{\text{ON}}$ or the drive current, and the fanout in the IO for the predriver stages is proportional to f, the frequency of the interface, to reflect the proportional edge rates needed based on the frequency. In the event that the ODT ($2 \cdot R_{\text{TT1}}$) is smaller than R_{ON}, the driver size is determined by $1/(2 \cdot R_{\text{TT1}})$. A_0 is the fixed area of the rest of the IO, which includes ESD protection.

The case studies in Section V-C and E discuss area results and the importance to keep area in mind when widening the bus. Further area tradeoffs of interest that can be explored using the tool can be found in [6].

TABLE V
TECHNOLOGY SCALING FOR DDR3

Parameter	Data rate (Mb/s)		
	800	1066	1600
vdd_io (V)	1.5	1.5	1.5
c_data_max (pF)	3.0	3.0	2.3
c_addr_max (pF)	1.5	1.5	1.3
t_ds_base (ps)	75	25	10
t_dh_base (ps)	150	100	45
t_dqsq (ps)	200	150	100

III. TECHNOLOGY PORTABILITY

The models described in Section II above are dependent on on-die as well as OFF-chip technology. As with prior CACTI versions, the IO and OFF-chip parameters that scale with process technology are taken from ITRS [45]. The underlying assumption is that the DRAM technology scales to meet the speed bin that it supports [28], since if DRAM technology is scaled, the speed bin that the IO parameters belong to are suitably scaled as well, including load capacitances [DRAM DQ pin capacitance (C_{DQ}), DRAM CA pin capacitance (C_{CA})], and JEDEC DRAM AC timing parameters [24], [25]. LPDDRx use different technologies compared with DDRx to save leakage power, so their capacitances and timing parameters are different from a DDRx memory of the same speed bin. Voltage also scales with DRAM technology, typically when a DRAM standard changes, e.g., DDR2 used 1.8-V IO supply voltage, while DDR3 uses 1.5-V IO supply voltage [28]. Sometimes, a lowered voltage specification is released as an addendum to a standard, e.g., DDR3-L [28]. Shown below in Table V are a subset of DDR3 DRAM parameters based on the speed bin.

If the user is interested in studying the impact of technology on a future memory standard, or a speed bin that is yet undefined, to first order the timing parameters can be assumed to scale down linearly with frequency.

The SoC PHY power and timing parameters scale with the technology node of the SoC, but are far more sensitive to the circuit architecture and analog components used to implement the design. It is hard to provide simplistic scaling trends for these parameters. For a given design and architecture, it would be possible to provide scaling power and timing for different technology nodes, but as speeds increase, the design and architecture for the PHY and IO are optimized and/or redesigned for the higher speed. Various design-specific trends for power and timing scaling with technology suggest around 20% scaling of analog power from one technology node to the next, or from one speed bin to the next [19].

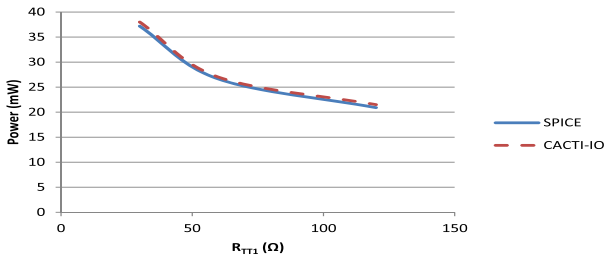


Fig. 5. DQ single-lane DDR3 termination power.

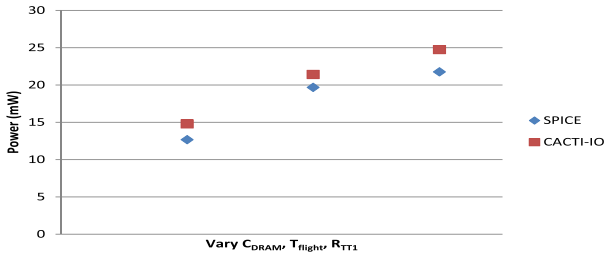


Fig. 6. DQ single-lane DDR3 total IO power.

The area of the IO directly scales mostly with the thick-oxide device of the technology. The scaling of the thick-oxide device typically does not keep pace with the core thin-oxide device as a consequence of supply voltages for external standards and reliability concerns. The constants k_0, k_1, k_2, and k_3 scale inversely with I_{dsat}/μm of the thick-oxide device.

Besides the parameters that scale with technology, the topology impacts the models for timing and voltage noise. A suitable DOE is required to fit the jitter and voltage noise coefficients for a given topology that defines the number of loads and interconnect length. When defining a topology other than the three standard configurations, a DOE analysis (as shown in Section IV) needs to be performed to be able to port the timing models for the channel.

The user can also add a new configuration into CACTI-IO to evaluate a future standard. For every new technology, voltage and timing DOE will need to be run for the given loading, as described in Section II-B. The IO area for a new technology can be obtained by curve fitting the constants in (31) using an IO datasheet. Guidance on how to modify the power models can be found in [6, Ch. 3.3].

IV. Validation

We now discuss validation of the new analytical IO and OFF-chip models added in CACTI-IO. The analytical power models are verified to be within 1%–15% of SPICE results. Models that are based on a lookup table, including the PHY power numbers, are valid by construction.

We first validate the power models for each DQ and CA bit line. Figs. 5 and 6 show SPICE versus CACTI-IO for the termination power and total IO power of a single lane of DQ DDR3. Fig. 5 shows that the worst case error between SPICE and CACTI-IO is less than 1% across different R_{TT1} values ($R_{\text{ON}} = 34\ \Omega$ for these cases). The total IO power shown in Fig. 6 for three different combinations of C_{DRAM}, R_{TT1} and T_{flight} shows a worst error of less than 14%.

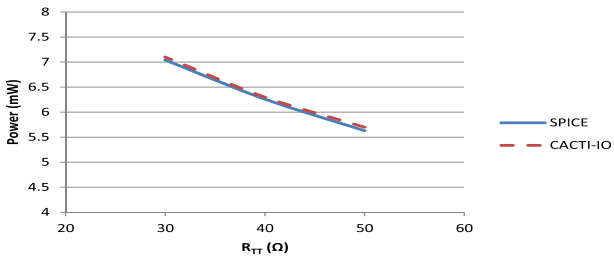


Fig. 7. CA single-lane DDR3 termination power.

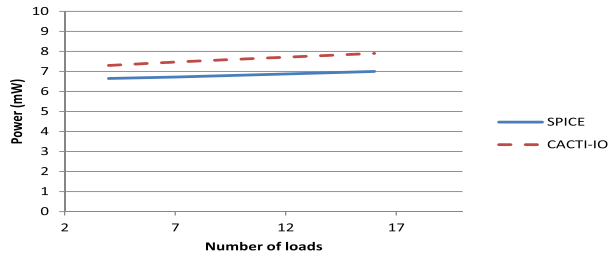


Fig. 8. CA single-lane DDR3 total IO power.

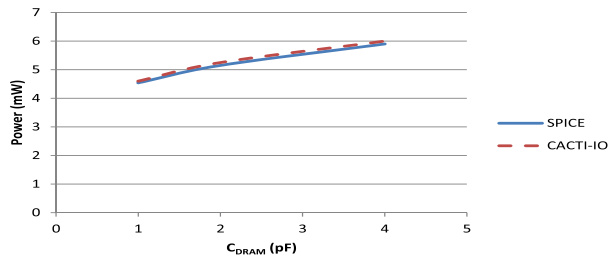


Fig. 9. DQ single-lane LPDDR2 total IO power.

Figs. 7 and 8 show SPICE versus model for the termination power and total IO power of a single lane of CA DDR3 using a fly by termination scheme. Fig. 7 shows the termination power for different R_{TT} values (the fly by termination shown in Fig. 4), while Fig. 8 shows the total IO power for different numbers of loads or fly by segments. Once again, the errors are similar to the DQ cases above, with the termination power within 1% and the total IO power within 15%.

Fig. 9 shows SPICE versus model for the switching power (dynamic IO and interconnect power) for DQ LPDDR2, where no terminations are used. In this scenario, the model is within 2% of the SPICE simulation.

To validate the power model for the entire interface, we compare it against measurements. Shown in Fig. 10 is measured versus model power for LPDDR2 WRITE obtained from a typical memory interface configuration for a 32-wide bus using a $\times 32$ LPDDR2 dual-rank DRAM. As can be observed, the model is within 5% of the measurement at the higher BWs. At lower BWs, power saving features make it harder to model the power as accurately since the duty cycle between the READ/WRITE/IDLE/SLEEP modes is harder to decipher. Here, the error is within 15%.

Shown in Fig. 11 are the results of an example DOE analysis on a sample channel for T_{jitter}. The input factors (F_i in 26) used here are R_{ON}, R_{TT1} and $C_{\text{DRAM\_DQ}}$. The simulations are performed for nine cases as indicated by the Taguchi array method explained in Section II. JMP [14] is then used to

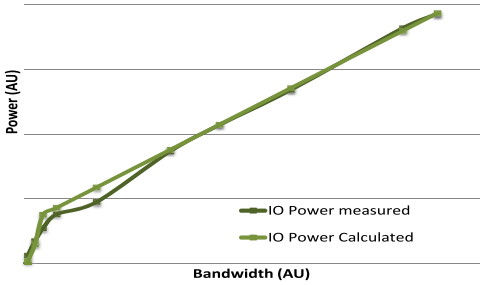


Fig. 10. LPDDR2 WRITE measurement versus model.

Ron (Ω)	Rtt (Ω)	Cdram (pF)	Pattern	Tjitter (ps)
34	30	1	−−−	45
34	40	1.5	−00	40
34	60	2	−++	66
40	30	1.5	0−0	56
40	40	2	00+	53
40	60	1	0+−	40
48	30	2	+−+	76
48	40	1	+0−	47
48	60	1.5	−0	80

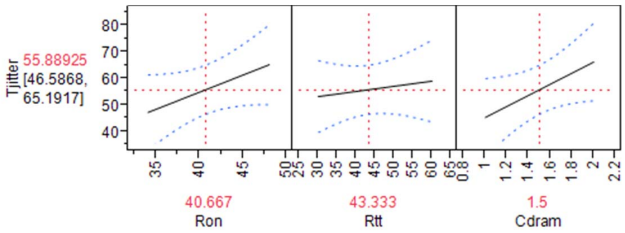


Fig. 11. DOE analysis on a DDR3 channel.

create a sensitivity profile. The table of values used for the Taguchi array and the sensitivity profile are shown in Fig. 11. The profile allows us to interpolate the input variables and predict T_{jitter}. CACTI-IO uses the sensitivity profile to perform the interpolation.

V. CACTI-IO

CACTI-IO is an extended version of CACTI [5] that includes the models described in Section II above. CACTI-IO allows for a quick search of optimal IO configuration parameters that help optimize power and performance of the IO along with the DRAM and cache subsystem.

CACTI has analytical models for all the basic building blocks of a memory [22]: decoder, sense-amplifier, crossbar, on-chip wires, DRAM/SRAM cell, and latch. We extend it to include the OFF-chip models presented in this paper. This requires modifying CACTI's global on-chip interconnect to include buffers at the PHY and drivers at the bank edge to connect to the IO circuit. Since all calculations are based on the ITRS [45] technology parameters, the energy and delay values calculated by CACTI are guaranteed to be mutually consistent. When a user inputs memory parameters and energy/delay constraints into CACTI, the tool performs an exhaustive design space exploration involving different array sizes, degrees of multiplexing, and interconnect choices to identify an optimal configuration. CACTI-IO is capable of performing an additional search for OFF-chip parameters, including optimal number of ranks, memory data width ($\times 4$, $\times 8$, $\times 16$, or $\times 32$ DRAMs), OFF-chip bus frequency, and bus width. This allows for optimal tradeoffs between OFF-chip power, area, and timing.

We present four case studies: 1) high-capacity DDR3-based server configurations in Section V-B; 2) 3-D memory configurations for high- BW systems in Section V-C; 3) buffered output on module (BOOM), a novel LPDDRx-based configuration for servers [12] in Section V-D; and 4) a PCRAM study showing separate READ and WRITE buses in Section V-G. All comparisons in the case studies are shown for one channel of the memory controller.

The IO power shown in the case studies is the peak power during activity, except in Section V-D for the BOOM case study, where we show how CACTI-IO can project the total system power as a sum of both IO and DRAM power and provide quick design-space exploration of both OFF- and ON-chip components together. The case studies show the variety of options the IO models provide, as well as the achievable range of capacities and power efficiencies, making for interesting tradeoffs for the architect.

To further highlight the utility of CACTI-IO, we study two tradeoffs in more detail for the BOOM designs. In Section V-E, we discuss optimal fanout of the data bus; and in Section V-F, we discuss the optimal fanout of the address bus.

A. Simulation Methodology

For studies of the high-capacity DDR3 configurations and 3-D configurations, we run the CACTI-IO models stand-alone to provide IO power comparisons described in Section V-B and C. For the BOOM cases, we use a multicore simulator [13] built on top of PIN [15] to provide the activity factor and idle-time information for multiprogrammed workload mixes from SPLASH2 [16]. While different benchmarks will yield different results, we expect that overall trends for IO and DRAM power will remain stable. We model a 16-core processor with two memory controllers. Each controller has a dedicated memory channel and each channel has four ranks. Number of reads, writes, activates, idle cycles, and power down cycles from this simulation are fed into CACTI-IO to evaluate the DRAM as well as IO energy averaged over the SPLASH2 benchmarks for the different BOOM configurations described in Section V-D.

B. High-Capacity DDR3 Configurations

We compare several configurations shown in Table VI for a $\times 64$ DDR3 memory channel; they all use a DIMM. RDIMM refers to a registered DIMM, where the command and address signals are buffered to allow for increased capacity. A load reduced DIMM (LRDIMM) [37] has a buffer for both address and data signals, allowing further increase in capacity at the cost of some data latency due to the buffering. The quad-rank case shown for LRDIMM uses two dual-die packages (2×2 d). The last configuration listed uses a buffer-on-board (BoB) from Intel [11] shown in Fig. 12. In this configuration, the buffer is not integrated into the DIMM, but is rather a stand-alone chip on the board. The buffer drives two RDIMMs and has two channels (four RDIMMs in all). While the interface

TABLE VI
CASE STUDY 1: SUMMARY OF POWER FOR DIFFERENT SERVER CONFIGURATIONS USING ×4 4-Gb DRAMS

Configuration	Capacity (GB)	No. of DQ loads	BW (GB/s)	P_{IO} (W)	$P_{CPU-Buf}$ (W)	P_{PHY} (W)	Efficiency (GBps/W)	Efficiency·GB (GB·GBps/W)
2 RDIMMs dual-rank	32	4	12.8	4.7	0.55	0.6	2.19	70.1
3 RDIMMs dual-rank	48	6	12.8	6.2	0.55	0.8	1.70	81.6
3 LRDIMMs dual-rank	48	2	12.8	4.86	3.2	0.8	1.44	69.12
3 LRDIMMs quad-rank w/ 2-die stack	**96**	2x2d	12.8	5.1	3.2	0.8	1.41	135.4
BoB w/ 2 channels 2 dual-rank RDIMMs	64	4	25.6	10.8	0.34	1.2	2.07	132.5

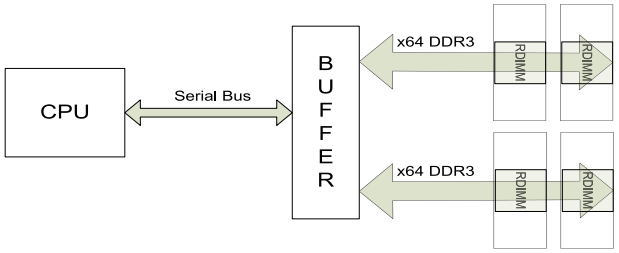


Fig. 12. BoB [11].

between the RDIMM or LRDIMM and the CPU remains a DDR3 bus, the interface between the BoB and CPU is a proprietary serial interface [11].

All configurations shown in Table VI use ×4 4-Gb memory devices. We study the interface to the DRAM as the bottleneck in the system, and the timing on the interface between the buffer and the host CPU is assumed not to be the limiting factor in this paper. The table lists the power consumed due to the IO on the DRAM interface (P_{IO}), the PHYs (P_{PHY}), and the IO on the interface between the CPU and the buffer ($P_{CPU-Buf}$). For signals that are buffered, although the P_{IO} reduces, $P_{CPU-Buf}$ goes up as it accounts for the buffered signal from the CPU to buffer. All configurations are assumed to operate at 800 MHz (DDR3-1600) and 1.5 V. As can be observed from the table, the LRDIMM offers a 50% increase in capacity (96 Gb for a ×64 channel) compared with the 3-RDIMM for a 17% decrease in efficiency. The product of capacity and efficiency is the highest for LRDIMM, at 135.4 Gb · Gb/s/W. The BoB configuration offers a 30% increase in capacity and a 2X BW improvement over the 3-RDIMM with 23% better power efficiency. Its product of capacity and efficiency is 132.5 Gb · Gb/s/W.

This case study highlights the ability of CACTI-IO to calculate IO power numbers for various configurations under consideration, and search for an optimal solution based on either total capacity (3-LRDIMM with 2-die stack), or efficiency (2-RDIMM), or perhaps a very good balance between the two (BoB). The BoB design presents a novel means of increasing capacity using a buffer on the board, while maintaining efficiency and low pin-count using a serial bus to the CPU with 2X the BW (25.6 Gb/s).

C. 3-D Stacking Using Wide-IO

In the second case study, we evaluate different 3-D stacking configurations to maximize BW. The configurations chosen include a 3-D TSS 4-die 4-Gb stacked DRAM with 4×128 channels [40], an 8-die stack with 4×128 channels, and narrower buses (4×64 and 4×32 as opposed to 4×128) with same BW, all of which connect to the CPU directly, exposing the die stack to the external pin loading. We also include the hybrid memory cube (HMC) proposed in [9], wherein the memory controller is included along with the DRAM stack, and connected by a 16×128 interconnect. A serial interface is used to connect the HMC to the CPU. The HMC 1.0 specification [10] supports various speeds (10, 12.5, and 15 Gb/s) for the serial link and supports a full-width (16 lanes) or half-width (eight lanes) option for the number of lanes. There are either four or eight such links depending on what aggregate BW is required. Since the serial interface can support up to 240 Gb/s, it is assumed to not limit the BW of the memory access, and focus is on the 16×128 interconnect within the HMC. All configurations operate at 1.2 V [47]. The data-rate on the interface is limited by the DRAM timing and voltage parameters and data-rates proposed for wide-IO [47], although CACTI-IO predicts some changes from the proposed data-rates based on the jitter sensitivity to loading and R_{ON}. Furthermore, the HMC allows for opportunity to explore timing and voltage optimization of the interface to the DRAM, as this is within the DRAM cube. We explore this by relaxing the timing and voltage parameters by 20% for the HMC. This allows the HMC to achieve better power efficiency compared with 3-D TSS.

Table VIII shows the results for these configurations calculated by CACTI-IO. As can be observed, the power efficiency varies by around 2X, with the HMC showing the highest efficiency (56 Gb/s/W), and a 3-D stack using a 4×32 bus showing the lowest efficiency (27 Gb/s/W). A peak BW of 176 Gb/s for 16×128 channels is achieved for the HMC with a 4-die stack, a 4.76X improvement over the standard 3-D TSS stack in an external connection using 4×128 channels. The isolation provided by the HMC to the CPU allows the bus to operate faster without the additional external loading.

The 4×64 and 4×32 cases shown in Table VIII represent narrower buses that achieve the same BW. The PHY power (taken from Tables I and II) goes up considerably for the ×32 case since the complexity increases at 1066 MHz; this leads to the poorest efficiency. CACTI-IO can furthermore predict V_{ddmin} based on the voltage noise parameters, as described in (27)–(30). The V_{ddmin} and the scaled efficiency at V_{ddmin} are shown in Table VIII. CACTI-IO predicts that the HMC can further scale down to 0.85 V and improve its efficiency to 100 Gb/s/W.

Table VIII also includes IO area comparison for the configurations shown, using the model discussed in (31). Of interest to note, is that for the narrower buses (4×64 and 4×32), the

TABLE VII
CASE STUDY 2: SUMMARY OF POWER FOR DIFFERENT 3-D CONFIGURATIONS

Config.	Capacity (GB)	Freq. (MHz)	BW_{MAX} (GB/s)	IO Power (W)	PHY Power (W)	Efficiency (GBps/W)	IO Area (sq.mm.)	V_{ddmin} (V)	Eff.@V_{ddmin} (GBps/W)
3-D 4-die	2	290	37	0.9	0.06	38	1.5	1	54
3-D 8-die	4	290	37	1.2	0.06	30	1.7	1.2	30
4x64	2	533	34	0.74	0.14	38	0.9	1.2	38
4x32	2	1066	34	0.84	0.44	27	0.7	1.2	27
HMC	2	350	**176**	2.96	0.29	56	6.0	0.85	100

TABLE VIII
CASE STUDY 3: SUMMARY OF POWER FOR DIFFERENT BOOM CONFIGURATIONS

Configuration	Capacity (GB)	Freq. (MHz)	No. of DQ loads	BW (GB/s)	P_{IO} (W)	$P_{CPU-Buf}$ (W)	P_{PHY} (W)	Efficiency (GBps/W)
x8 BOOM-N2-D-800	16	800	4	12.8	4.96	3.52	0.8	1.38
x8 BOOM-N4-L-400	32	400	4	12.8	2.51	3.52	0.4	2.0
x8 BOOM-N4-L-400 with serial bus to host	32	400	4	12.8	2.51	0.34	0.4	3.94

area decreases, but not by a factor of 2× or 4×, respectively. The additional area is to support the higher speed on the narrower bus. The 8-die TSS incurs an area overhead to support the larger load.

As described in Section III, it is also important to note that for the comparisons of 3-D configurations, we modeled the voltage a timing budgets for a 3-D interconnect based on running SPICE simulations to extract the timing and voltage noise coefficients described in Section II-B. C_{Total}, the load capacitance and R_{ON}, the output impedance of the 3-D IO are parameters that impact timing and voltage noise. They are used to model T_{jitter} and K_N, as described in Section II-B. The remaining timing parameters are from the wide-IO specification [47].

An important design consideration in 3-D DRAM configurations is to architect the internal banks to take best advantage of the improved OFF-chip BW with the wider interface. Unlike traditional DDR or LPDDR DRAM chips, HMC and wide-IO memories employ a number of smaller banks to improve the overall BW. When modeling both on- and OFF-chip components in CACTI-IO, CACTIs on-chip design space exploration considers the latency of individual banks, and adjusts the internal bank count to match the OFF-chip IO BW.

This case study highlights the ability of CACTI-IO to calculate IO power and timing for a new interconnect technology such as 3-D, including the novel HMC. The baseline models included in CACTI-IO can be configured for DDR3-based signaling as well as for 3-D interconnect. We see that CACTI-IO is able to identify the solution with the highest BW and efficiency (HMC) and also predict how much the efficiency would be affected when going from 4×128 to 4×64 or 4×32 due to PHY power increase for the higher data rates. CACTI-IO is also able to calculate V_{ddmin} for a given frequency and loading, predicting a 1.8X improvement in power efficiency for the HMC.

D. BOOM: LPDDRx for Servers

The BOOM architecture [12] from Hewlett-Packard relies on a buffer chip on the board that connects to lower-speed and lower-power LPDDRx memories. To match the channel BW, BOOM uses a wider DIMM-internal bus (from the buffer to the DRAMs), as shown in Fig. 13. Furthermore, BOOM has the option of grouping multiple physical ranks into a single logical rank [12]. BOOM can use commodity LPDDRx DRAMs with lower power, but achieves high BW and capacity through wider buses. As servers become more sensitive to memory subsystem power, BOOM provides a valuable means for the use of mobile DRAM to achieve a better power efficiency while still meeting server performance.

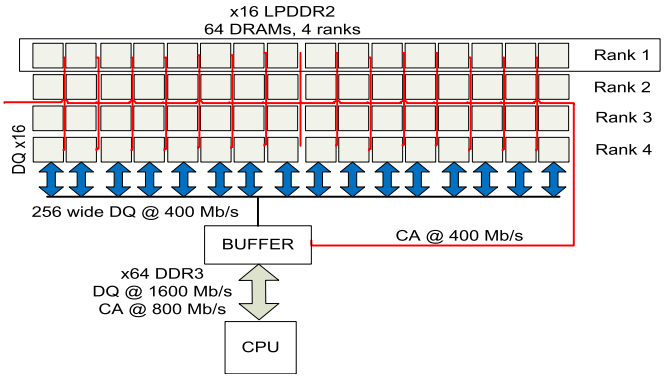


Fig. 13. BOOM-N4-L-400 configuration with ×16 devices [12].

Table VII summarizes the IO peak power for three BOOM configurations [12]. The power is shown per memory channel (equivalent of a ×64 DDR3 channel). A BOOM configuration is denoted as BOOM-Nn-X-Y, where n is a ratio of the wider internal bus to the channel's x64 bus, X is DRAM type (D for DDR3 and L for LPDDR2) and Y is DRAM data rate (typically 1600/n Mb/s). All BOOM configurations shown use ×8 memories.

Table VII clearly shows a 2X improvement in IO power (P_{IO}) from buffer to DRAM using LPDDRx memories to achieve the same BW when we compare BOOM-N2-D-800 (using DDR3 DRAM) and BOOM-N4-L-400 (using LPDDR2 DRAM).

In addition, BOOM offers the advantage of using a custom interface between the CPU host and the buffer chip. Instead of a standard ×64 DDR3 interface, a serial bus similar to the BoB [11] case in Section V-B above can be used. This further improves the total efficiency by 2X, achieving a 2.85X improvement in total power efficiency over a DDR3 design.

To highlight the ability of CACTI-IO to provide combined DRAM and IO power, we compare the three BOOM configurations with respect to normalized energy in Fig. 14.

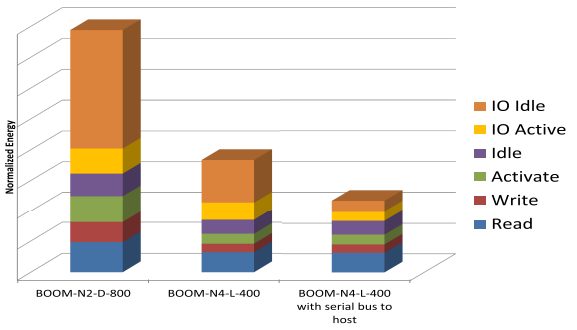


Fig. 14. Normalized system (DRAM+IO) energy for BOOM configurations.

The simulation methodology used to obtain the normalized energy is described in Section V-A. The total energy is broken down into the DRAM core power (read, write, activate, and idle), the IO active power (read and write), and the IO idle power (mainly due to terminations and the active clock). The precharge power is included in the activate power. The total idle power (also referred to as background power [12]) is got from adding the DRAM core idle power and the IO idle power.

We make the following observations.

1) The IO power is a significant portion of the combined power (DRAM+IO): 59% for the DDR3-based (BOOM-N2-D-800) configuration and 54% for the LPDDR2-based configuration (BOOM-N4-L-400). When using a serial bus from the buffer to the host, the IO power for BOOM-N4-L-400 reduces to 27% of the total power.
2) The IO idle power is a very significant contributor. The BOOM-N4-L-400 design reduces the IO idle power using LPDDR2 unterminated signaling, but since the BOOM configuration still relies on a DDR3 type bus from the buffer to the host, as shown in Fig. 13, the IO idle power for the whole channel is still significant.
3) Once the DRAM core becomes efficient, IO becomes a major contributor to the total power. Replacing DDR3 memories with LPDDR2 alone is not as efficient as further reducing the IO idle power using a serial bus instead of a DDR3 style bus to the host. The BOOM-N4-L-400 design with a serial host provides a 3.4X energy savings (DRAM+IO) over the BOOM-N2-D-800 design. While Table VII only compares the IO active power, Fig. 13 also accounts for IO idle power and projects total energy based on active and idle times. While the serial bus only provides a 2.85X savings in IO active power, it provides an 11X savings in IO idle power when compared with the BOOM-N2-D-800 design.
4) The number of power-down cycles is around 15% of the total cycles. More aggressive power-down will help reduce the IO idle power. Supply scaling is also an option at lower frequencies in the case of BOOM-N4-L-400.

This case study highlights CACTI-IOs ability to provide IO power numbers to a system simulator, which can then provide valuable insight into total system power. Only combining the IO and DRAM power brings out the right tradeoffs needed to further improve efficiency. The study also highlights how

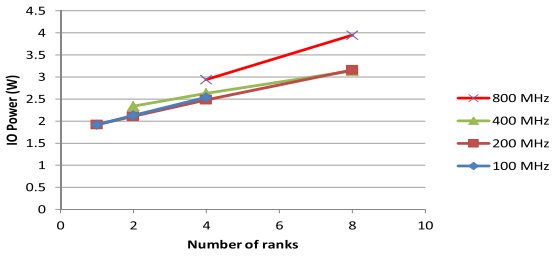


Fig. 15. IO power versus number of ranks for BOOM-LPDDR2.

CACTI-IO can be used to optimize a buffer-based topology such as BOOM, where IO choices including bus frequency and width can make a 2.85X difference in IO Active power and nearly an 11X difference in IO idle power. Furthermore, the need for aggressive power down depends on the OFF-chip configuration as well, and IO idle power is a key factor in determining how to address the power-down mode.

E. Optimizing Fanout for the Data Bus

We now illustrate how one can calculate the optimal number of physical ranks in a BOOM configuration to minimize IO power for a fixed capacity and BW. The number of physical ranks represents the fanout on the data bus. For this example, we assume that the memory density per DRAM die is fixed.

If N_R is the number of ranks, W_B the bus width, W_M the memory data width, and f the data rate, then [8]

$$N_R \cdot (W_B / W_M) = \text{Capacity} \tag{32}$$

$$W_B \cdot 2f = BW. \tag{33}$$

Fig. 15 shows the IO power as we vary the number of ranks to meet a capacity of 64 DRAMs and a BW of 12.8 Gb/s for an LPDDR2 bus. The IO power varies for different bus frequencies f, as the width of the bus and the memory data widths vary to meet the conditions in (32) and (33). The memory data width is chosen to be $\times 4$, $\times 8$, $\times 16$, or $\times 32$ for the LPDDRx memories. The number of ranks is 1, 2, 4, or 8. The bus width is $\times 64$, $\times 128$, $\times 256$, or $\times 512$, and the bus frequency is 800, 400, 200, or 100 MHz.

As can be observed from Fig. 15, the wider and slower LPDDR2 bus provides the lowest power. A 512-wide bus using $\times 8$ memories in a single-rank configuration running at 100 MHz consumes the lowest power at 1.92 W, while a 64-wide bus using $\times 8$ memories in an eight-rank configuration running at 800 MHz consumes the highest power at 3.94 W. Also to be noted are the diminishing returns of scaling down to a lower speed once the bus is scaled to 200 MHz, owing to high-impedance terminations. This frequency at which termination is no longer needed depends on the interconnect length and the loading, which change based on the topology and technology as determined by the jitter DOE analysis.

One of the downsides to having a wider and slower bus is the cost of area on the die, package, and board. CACTI-IO predicts the impact on on-die area as we scale frequency and bus width to keep the BW constant. Shown in Fig. 16 is the IO area versus frequency for low fanouts (1 or 2 ranks) in 28-nm technology, such that total BW is kept constant. Also shown is

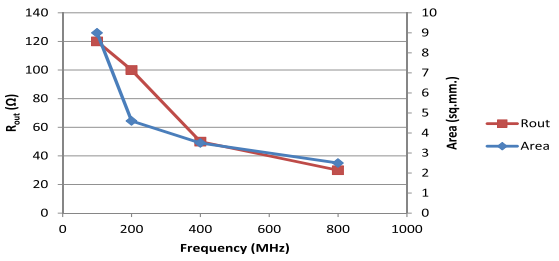


Fig. 16. Area versus frequency for a constant-BW BOOM-LPDDR2.

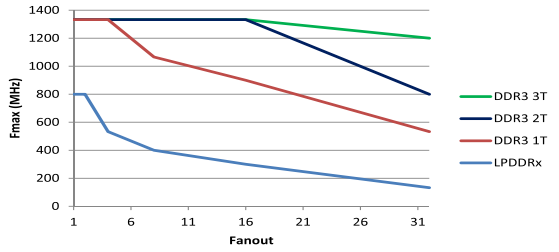


Fig. 17. Fanout versus Fmax for a typical DDR3 CA bus.

the R_{out} that is used in (31) to calculate the area. Wider buses result in a net increase in area even though they operate at lower frequencies. In a buffer chip this may be acceptable as there is less premium on area than on a CPU or DRAM die. Since there is almost a 2X increase in area going from the 200 to 100 MHz solution, while there is hardly any difference in power, it may be prudent to choose the 200-MHz solution. The optimal solution would then be $N_R = 1, W_B = 256, W_M = 4$, and $f = 200$ MHz. This example highlights CACTI-IOs ability to optimize the number of ranks based on IO power and any user-provided IO area, thus helping to optimize the IO configuration for a buffer-based design.

F. Optimizing Fanout for the Address Bus

As we increase capacity, the address bus incurs a penalty as all memories on the channel share a common address bus. The LPDDR2 and LPDDR3 standards [28] offer address buses at DDR speeds, with no option for 2T timing [25]. This idiosyncrasy in the DRAM specification is not easily exposed to architects, but CACTI-IO allows for verified configurations to be systematically provided to architects.

To calculate the maximum achievable speed for a fly by topology, as shown in Fig. 4, we need to define the sensitivity of the jitter on the CA bus to the fanout of the bus, as shown in (26). Fig. 17 shows the maximum achievable clock frequency on the CA bus for DDR3 and LPDDR2/3 as a function of the fanout for a representative channel. For DDR3, the 2T and 3T timing options allow for relaxed timing on the CA bus [24].

Given the limitation for the LPDDR2 address fanout owing to the DDR speed requirement, multiple address buses may be needed to achieve higher capacities. For instance, based on the example in Fig. 17, with a fanout of 16, we would need two LPDDR2 CA buses to support 400 MHz, while a single CA bus on DDR3 could support 1066 MHz with 2T timing.

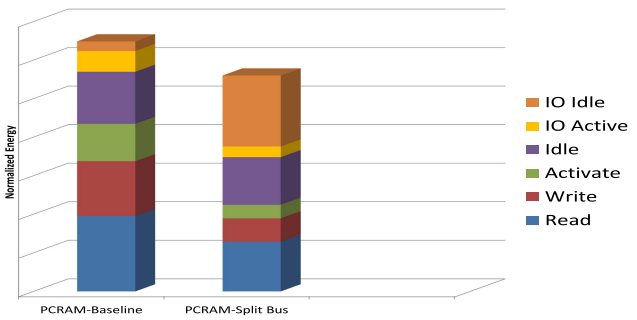


Fig. 18. Normalized system (DRAM+IO) energy for PCRAM configurations.

With a buffer-based design, it is possible to have multiple address buses for a given channel between the buffer chip and the DRAMs. This would provide a means to limit the fanout on the address bus. Architects can optimize the design for a given address speed with optimal latency and burst requirements, including subranking [12]. Understanding the limitations of the address bus allows architects to plan to overcome or minimize its impact on system performance.

G. Phase-Change RAM

PCRAM is a type of nonvolatile RAM [43]. The READ and WRITE latencies for a PCRAM are very different depending on whether the page is open or not. The READ or WRITE miss latency, when the page is closed, is an order of magnitude bigger than if it were a hit, with the page open. In addition, the WRITE miss latency is significantly larger than the READ miss latency.

In our case study, we evaluate the power and performance tradeoff of the IO bus to the PCRAM. We compare two configurations. The first configuration has a 64-wide bidirectional bus operating at 400-MHz DDR. In the second configuration the 64-wide bus is broken into two 32-wide unidirectional buses, one dedicated for READs, operating at 800 MHz, and the other for WRITE, operating at 200 MHz. This allows for the READ BW to be maintained, while the WRITE BW is much smaller owing to the significantly larger WRITE miss latency. The idea behind splitting the bus into a READ and a WRITE bus is to see if enabling READs independently can allow for the WRITE bus to be optimized for low power with higher latencies. The READ bus at 800 MHz needs terminations, while the WRITE bus at 200 MHz can be unterminated. Fig. 18 shows a comparison of the power for the two configurations. The comparison assumes $\times 8$ devices and four ranks. For a 5% performance penalty, the split bus configuration provides 10%–40% savings in power depending on how the IO idle power is managed.

There are various solutions to address IO idle power reduction, and CACTI-IO can help the user evaluate these options.

1) A suitable termination option to Voltage Supply (like DDR4 [26]) or Ground, rather than the DDR3-type midrail termination, could significantly help save IO idle power, as described in Section II-A3d.
2) Furthermore, by scaling frequency appropriately, significant further reductions of power are possible.

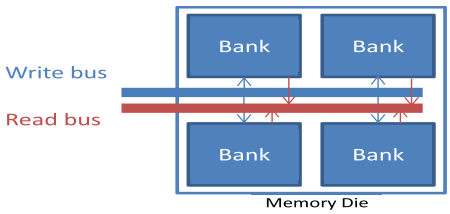


Fig. 19. Split bus PCRAM configuration.

TABLE IX
PCRAM IO AREA

Bus Config	No. of IO	IO Area (sq. mm.)
Bidirectional	64	1.75
Split	32+32	1.8

The wakeup time model in CACTI-IO described in Section II-A can help assess the latency impact of any such frequency scaling.

3) Dynamic ODT [29] options could help optimize tradeoffs between idle power and signal integrity.

This example highlights the use of CACTI-IO to study not only an emerging memory technology, but the optimal IO and bus configuration as well. It helped identify the IO idle power as significant for the split bus configuration, which could be optimized by design choices described above. Having dedicated READ and WRITE buses (instead of a conventional bidirectional DDR bus) will require changes to the bank organization and interbank interconnects, which in turn will impact READ and WRITE latencies. For example, the OFF-chip write bus can be accommodated in two different ways. First, we can buffer writes internally in a PCRAM die using a small write buffer, and use a single shared interbank bus. Alternatively, we can have a dedicated bus for read and write, as shown in Fig. 19. Both these scenarios can be modeled in CACTI. As these design choices are specific to a given architecture (in this case, a dedicated read/write bus), the architect has to manually modify the internal bus specification to simulate an architecture like this.

Table IX compares the IO area for the split bus and bidirectional bus configurations. The IO area is nearly identical. This is because the split bus benefits from a slower WRITE interface, while the READ interface is faster.

VI. CONCLUSION

We have presented CACTI-IO, a version of CACTI that models the OFF-chip memory interface for server and mobile configurations. Its models include OFF-chip power and IO area, as well as voltage and timing margins that help define the maximum achievable BW. Our framework permits quick design space exploration with the rest of the memory subsystem and provides a systematic way for architects to explore the OFF-chip design space. It also exposes DRAM signaling standards and their idiosyncrasies to architects, while still providing an easily extensible framework for customization of OFF-chip topologies and technologies.

Using CACTI-IO, we have also illustrated the tradeoffs between capacity, BW, area, and power of the memory interface through four industry-driven case studies. These clearly show the ability of CACTI-IO to calculate IO power for various configurations, including DIMMs, 3-D interconnect, buffer-based designs such as BoB and BOOM, and new memory technologies like PCRAM. CACTI-IO helps determine the lowest-power OFF-chip configuration (bus width, memory data width, number of physical ranks, address bus fanout, minimum supply voltage, and bus frequency) for a given capacity and BW requirements.

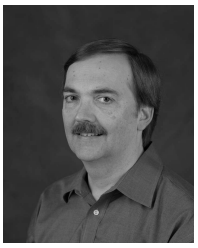
Furthermore, we have highlighted the capability of CACTI-IO to combine IO and DRAM power, which shows the significant contribution of IO power to the total (DRAM + IO) memory power (up to 59% in some cases). We have observed the relative importance of IO idle power using CACTI-IO and a system simulator together to calculate system energy in various modes (read, write, activate, precharge, and idle). A combination of a wider and slower bus to the DRAM and a faster serial bus to the CPU provides the lowest IO idle power.

CACTI-IO will be publicly available online as part of the latest CACTI release [5]. We expect that the new capabilities provided by this tool will enable improved understanding of memory interface issues, allowing architects to evaluate customized OFF-chip buffer-based designs as well as the impact of new interconnect technologies on system power and performance.

REFERENCES

[1] N. P. Jouppi, A. B. Kahng, N. Muralimanohar, and V. Srinivas, "CACTI-IO: CACTI with off-chip power-area-timing models," in *Proc. ACM/IEEE Int. Conf. Comput.-Aided Design (ICCAD)*, Nov. 2012, pp. 294–301.
[2] W. Dally and J. Poulton, *Digital Systems Engineering*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
[3] H. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI*. Reading, MA, USA: Addison-Wesley, 1990.
[4] D. Oh and C. Yuan, *High-Speed Signaling: Jitter Modeling, Analysis, and Budgeting*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2011.
[5] *CACTI* [Online]. Available: http://www.hpl.hp.com/research/cacti/, accessed Aug. 2014.
[6] N. P. Jouppi, A. B. Kahng, N. Muralimanohar, and V. Srinivas, "CACTI-IO," HP Labs, Palo Alto, CA, USA, Tech. Rep. HPL-2013–79, Sep. 2013.
[7] N. Chang, K. Kim, and J. Cho, "Bus encoding for low-power high-performance memory systems," in *Proc. Design Autom. Conf. (DAC)*, 2000, pp. 800–805.
[8] A. B. Kahng and V. Srinivas, "Mobile system considerations for SDRAM interface trends," in *Proc. System Level Interconnect Prediction Workshop (SLIP)*, 2011, pp. 1–8.
[9] J. Baloria, "Micron reinvents DRAM memory: Hybrid memory cube," in *Proc. IDF Workshop*, Sep. 2011.
[10] *HMC 1.0 Specification* [Online]. Available: http://tinyurl.com/jwzjezg, accessed Aug. 2014.
[11] *Intel's Scalable Memory Buffer* [Online]. Available: http://tinyurl.com/7xbt27o, accessed Aug. 2014.
[12] D. H. Yoon, J. Chang, N. Muralimanohar, and P. Ranganathan, "BOOM: Enabling mobile memory based low-power server DIMMs," in *Proc. IEEE 39th Annu. Int. Symp. Comput. Archit. (ISCA)*, Jun. 2012, pp. 25–36.
[13] *McSim* [Online]. Available: http://cal.snu.ac.kr/mediawiki/index.php/McSim, accessed Aug. 2014.
[14] *JMP Statistical Software* [Online]. Available: http://www.jmp.com, accessed Aug. 2014.
[15] C.-K. Luk *et al.*, "Pin: Building customized program analysis tools with dynamic instrumentation," in *Proc. ACM SIGPLAN Conf. Program. Lang. Design Implement.*, 2005, pp. 190–200.
[16] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The SPLASH-2 programs: Characterization and methodological considerations," in *Proc. IEEE 22nd Annu. Int. Symp. Comput. Archit. (ISCA)*, Jun. 1995, pp. 24–36.

[17] H. Zheng and Z. Zhu, "Power and performance trade-offs in contemporary DRAM system designs for multicore processors," *IEEE Trans. Comput.*, vol. 59, no. 8, pp. 1033–1046, Aug. 2010.
[18] H. Lee *et al.*, "A 16 Gb/s/link, 64 GB/s bidirectional asymmetric memory interface," *IEEE J. Solid-State Circuits*, vol. 44, no. 4, pp. 1235–1247, Apr. 2009.
[19] J. Poulton *et al.*, "A 14-mW 6.25-Gb/s transceiver in 90-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2745–2757, Dec. 2007.
[20] F. O'Mahony *et al.*, "A 47×10 Gb/s 1.4 mW/(Gb/s) parallel interface in 45 nm CMOS," in *IEEE Int. Solid-State Circuits Dig. Tech. Papers (ISSCC)*, Feb. 2010, pp. 156–158.
[21] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi, "McPAT: An integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Proc. 42nd Annu. IEEE/ACM Int. Symp. Microarchit. (MICRO)*, Dec. 2009, pp. 469–480.
[22] S. Thoziyoor, J. Ahn, M. Monchiero, J. B. Brockman, and N. P. Jouppi, "A comprehensive memory modeling tool and its application to the design and analysis of future memory hierarchies," in *Proc. IEEE 35th Int. Symp. Comput. Archit. (ISCA)*, Jun. 2008, pp. 51–62.
[23] *Micron DRAM System Power Calculators* [Online]. Available: http://www.micron.com/support/dram/power_calc.html, accessed Aug. 2014.
[24] *JEDEC DDR3 Specification*, Standard JESD79–3E, Jul. 2010.
[25] *JEDEC LPDDR2 Specification*, Standard JESD209–2C, Jun. 2009.
[26] *JEDEC DDR4 Specification*, Standard JESD79–4A, Sep. 2012.
[27] *JEDEC LPDDR3 Specification*, Standard JESD209–3, May 2012.
[28] *JEDEC* [Online]. Available: http://www.jedec.org, accessed Aug. 2014.
[29] *Micron Technical Note* [Online]. Available: http://www.micron.com/~/media/Documents/Products/TechnicalNote/DRAM/TN4104.pdf, accessed Aug. 2014.
[30] G. Taguchi, *Introduction to Quality Engineering*, 2nd ed. New York, NY, USA: McGraw-Hill, 1996.
[31] R. Palmer, J. Poulton, A. Fuller, J. Chen, and J. Zerbe, "Design considerations for low-power high-performance mobile logic and memory interfaces," in *Proc. IEEE Asian Solid-State Circuits Conf. (ASSCC)*, Nov. 2008, pp. 205–208.
[32] J. Ellis, "Overcoming obstacles for closing timing for DDR3–1600 and beyond," in *Proc. Denali MemCon*, 2010.
[33] A. Vaidyanath, "Challenges and solutions for GHz DDR3 memory interface design," in *Proc. Denali MemCon*, 2010.
[34] P. Mosalikanti, C. Mozak, and N. Kurd, "High performance DDR architecture in Intel Core processors using 32 nm CMOS high-K metal-gate process," in *Proc. IEEE Int. Symp. VLSI Design, Autom. Test (DAT)*, Apr. 2011, pp. 1–4.
[35] D. Wang, B. Ganesh, N. Tuaycharoen, K. Baynes, A. Jaleel, and B. Jacob, "DRAMsim: A memory system simulator," *ACM SIGARCH Comput. Archit. News* vol. 33, no. 4, pp. 100–107, 2005.
[36] *HP Memory Technology Evolution: An Overview of System Memory Technologies* [Online]. Available: http://tinyurl.com/7mvktcn, accessed Aug. 2014.
[37] [Online]. Available: http://www.micron.com/products/dram_modules/lrdimm.html, accessed Aug. 2014.
[38] (2009, May). Challenges and solutions for future main memory. *Rambus White Paper* [Online]. Available: http://tinyurl.com/cetetsz, accessed Aug. 2014.
[39] B. Schroeder, E. Pinheiro, and W. Weber, "DRAM errors in the wild: A large-scale field study," in *Proc. 11th Int. Joint Conf. Meas. Model. Comput. Syst.*, 2009, pp. 193–204.
[40] J.-S. Kim *et al.*, "A 1.2 V 12.8 GB/s 2 Gb mobile wide-I/O DRAM with 4×128 I/Os using TSV-based stacking," in *IEEE Int. Solid-State Circuits Dig. Tech. Papers (ISSCC)*, Feb. 2011, pp. 496–498.
[41] S. Sarkar, A. Brahme, and S. Chandar, "Design margin methodology for DDR interface," in *Proc. IEEE Electr. Perform. Electron. Packag. Syst. (EPEPS)*, Oct. 2007, pp. 167–170.
[42] S. Chaudhuri, J. A. McCall, and J. H. Salmon, "Proposal for BER based specifications for DDR4," in *Proc. IEEE 19th Conf. Electr. Perform. Electron. Packag. Syst. (EPEPS)*, Oct. 2010, pp. 121–124.
[43] M. Qureshi, V. Srinivasan, and J. Rivers, "Scalable high-performance main memory system using phase-change memory technology," in *Proc. IEEE Int. Symp. Comput. Archit. (ISCA)*, Jun. 2009, pp. 24–33.
[44] *HP Power Advisor* [Online]. Available: http://h18000.www1.hp.com/products/solutions/power/index.html
[45] (2011). *International Technology Roadmap for Semiconductors* [Online]. Available: http://www.itrs.net/
[46] B. K. Casper, M. Haycock, and R. Mooney, "An accurate and efficient analysis method for multi-Gb/s chip-to-chip signaling schemes," in *IEEE Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2002, pp. 54–57.
[47] *JEDEC Wide IO Specification JESD229*, Dec. 2011.
[48] M. A. Horowitz, C.-K. K. Yang, and S. Sidiropoulos, "High-speed electrical signaling: Overview and limitations," in *Proc. IEEE/ACM MICRO*, 1998, pp. 12–24.
[49] D. Oh *et al.*, "Prediction of system performance based on component jitter and noise budgets," in *Proc. IEEE Electr. Perform. Electron. Packag. (EPEPS)*, Oct. 2007, pp. 33–36.



Norman P. Jouppi (M'84–SM'02–F'03) received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, USA, in 1984 and the M.S. degree in electrical engineering from Northwestern University, Evanston, IL, USA, in 1980.

He is a Distinguished Hardware Engineer with Google, Mountain View, CA, USA. He is known for his innovations in computer memory systems, including stream prefetch buffers, victim caching, multilevel exclusive caching, and development of the CACTI tool for modeling memory timing, area, and power. He was the Principal Architect and Lead Designer of several microprocessors, contributed to the architecture and design of graphics accelerators, and extensively researched video, audio, and physical telepresence. He holds more than 75 U.S. patents. He has authored more than 125 technical papers, with several best paper awards and two International Symposium on Computer Architecture Influential Paper Awards.

Dr. Jouppi is a fellow of the ACM and a member of the National Academy of Engineering.

Andrew B. Kahng (M'03–SM'07–F'10) received the Ph.D. degree in computer science from the University of California at San Diego (UCSD), La Jolla, CA, USA, in 1989.

He was with the Department of Computer Science, University of California at Los Angeles, Los Angeles, CA, USA, from 1989 to 2000. Since 2001, he has been with the Department of Computer Science and Engineering and Department of Electrical and Computer Engineering, UCSD, where he holds the Endowed Chair in high performance computing. He has authored and co-authored more than 400 journal and conference papers, and three books. He holds 27 issued U.S. patents. His current research interests include IC physical design, the design-manufacturing interface, combinatorial algorithms and optimization, and the road mapping of systems and technology.

Naveen Muralimanohar (M'04) received the Ph.D. degree in computer science from the University of Utah, Salt Lake City, UT, USA.

He is a Senior Researcher with Hewlett-Packard Labs, Palo Alto, CA, USA. His current research interests include architecting non-volatile memories, designing and modeling next-generation memory systems, and solving reliability challenges associated with large compute clusters.

Vaishnav Srinivas (M'02) received the B.Tech. degree from IIT Madras, Chennai, India, in 2000 and the M.S. degree from the University of California at Los Angeles, Los Angeles, CA, USA, in 2002. He is currently pursuing the Ph.D. degree with the University of California at San Diego, La Jolla, CA, USA.

He is a Principal Engineer with Qualcomm, San Diego, CA, USA, working on PHY circuit design and memory architecture. He holds 14 issued U.S. patents. His current research interests include low-power interfaces, IO and memory architecture, and roadmapping interface technology.