Power-constrained high-frequency circuits for the IBM POWER6 microprocessor

B. Curran
E. Fluhr
J. Paredes
L. Sigal
J. Friedrich
Y.-H. Chan
C. Hwang

The IBM POWER6™ microprocessor is a high-frequency (>5-GHz) microprocessor fabricated in the IBM 65-nm silicon-on-insulator (SOI) complementary metal-oxide semiconductor (CMOS) process technology. This paper describes the circuit, physical design, clocking, timing, power, and hardware characterization challenges faced in the pursuit of this industry-leading frequency. Traditional high-power, high-frequency techniques were abandoned in favor of more-power-efficient circuit design methodologies. The hardware frequency and power characterization are reviewed.

1. Introduction

The IBM POWER6* microprocessor core is fabricated using the IBM 65-nm silicon-on-insulator (SOI) process and provides a significant boost in frequency and performance to pSeries* systems. Core operating frequencies of more than 5 GHz have been demonstrated. The processor chip contains two cores, 8 MB of on-chip level 2 (L2) cache, a directory for a 32-MB L3 cache, two memory controllers, a GX I/O controller, and nest support circuitry for a 128-way symmetric multiprocessor (SMP). The chip shown in **Figure 1** has an area of 341 mm^2 and contains 790 million transistors, 1,953 signal I/Os, 5,399 power and ground I/Os, and more than 4.5 km of wire.

The on-chip circuits are connected via ten levels of copper wire and are powered through multiple voltage domains. The core logic, array, and I/O circuits are designed to operate at nominal voltages of 1.15, 1.3, and 1.2 V, respectively. However, the actual logic and array voltages delivered to each chip vary between 0.85 V and 1.3 V and between 1.0 V and 1.4 V, respectively, depending on the speed of the part. Chips with shorter channels typically run faster but use considerably more power because of higher leakage. In previous-generation processors, these parts would have been discarded because of excessive power dissipation but now are usable by operating at lowered voltages. In addition, chips with longer channels typically run slower, so some of these parts also would not have been used in earlier generation processors because of their low operating frequency, but now they also are made usable by increasing their operating voltages.

Frequency

Various frequency/cycle-time targets were evaluated during an exploratory phase. A cycle time corresponding to 13-FO4[1] inverter delays was selected based on the fastest known techniques to achieve back-to-back execution of 64-byte dependent, fixed-point instructions. Performance analysis indicated a loss of ~10% IPC (instructions per cycle) if a dependent fixed-point instruction had to wait an additional cycle before executing. **Figure 2** illustrates the relative frequency, IPC, and performance as a function of FO4 cycle times. The POWER6 processor frequency is more than double that of the POWER5* processor without a doubling of the pipeline depth. **Table 1** compares the fixed-point and binary floating-point functional unit pipeline depths of the POWER5 and POWER6 processors [1].

Traditional high-frequency cores rely on a super-deep pipeline and/or aggressive dynamic circuits. Unfortunately, both of these techniques add significant power, because super-deep pipelines require more latches that consume power and dynamic circuits significantly increase data-switching power as well as clock load and power. The POWER6 core has uniquely achieved a low

[1] *Fanout of 4* (FO4) is a technology-independent metric for measuring processor cycle times in which 1-FO4 delay corresponds to the delay of a single CMOS inverter gate loaded with four identical gates.

0018-8646/07/$5.00 © 2007 IBM

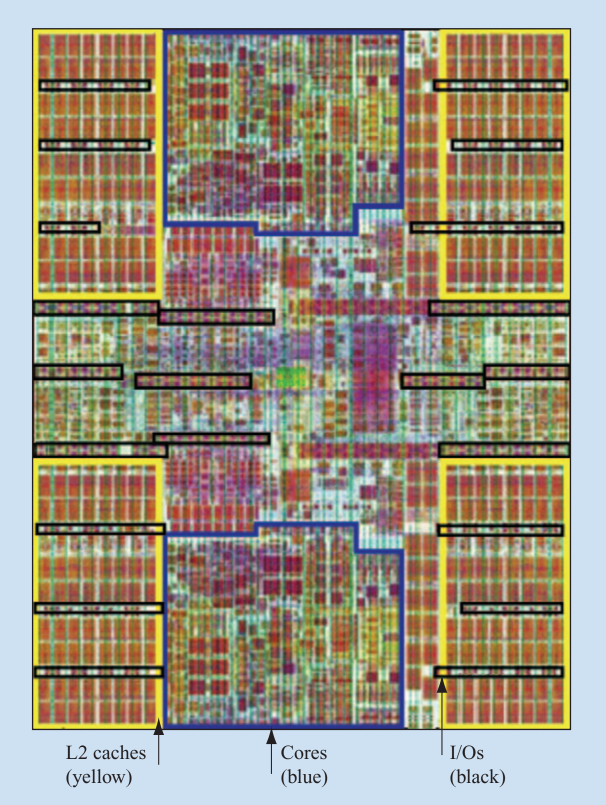

Figure 1

Die photograph of the POWER6 microprocessor.

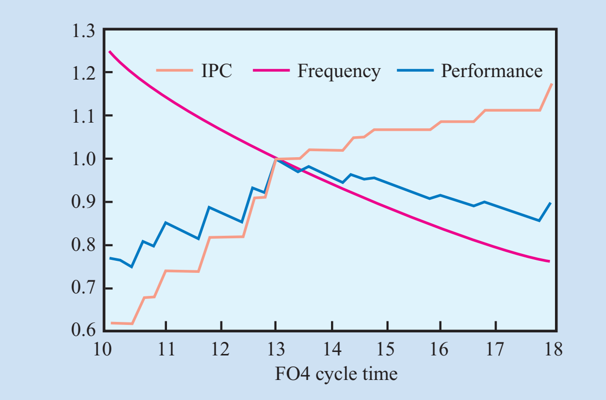

Figure 2

Relative instructions per cycle (IPC), frequency, and performance.

Table 1 Instruction latency comparison.

Microprocessor	*Fixed-point instruction latency*	*Fused multiply–add binary floating-point instruction latency*
z900	25 FO4	Not applicable
z990	28 FO4	140 FO4
POWER5	44 FO4	132 FO4
POWER6	13 FO4	78 FO4

FO4 cycle time without resorting to either of these traditional design approaches. Instead innovative logic circuit co-designs were used extensively throughout the core to achieve 13 FO4. The basic philosophy is to make each circuit do more logic work. Many circuits perform double and triple duty by implementing parallel logic functions [1] that have historically been implemented serially [2]. Logic functions have also been split into nontraditional parts (e.g., multiplexers, or muxes) to allow part of the logic to be completed earlier in a non-critical-timing part of the cycle [2].

Overview

The 65-nm complementary metal-oxide semiconductor (CMOS) process technology is described in Section 2. The innovative circuit styles and designs are presented in Section 3. Section 4 focuses on the clock distribution and custom circuit design methodologies. Power is the most significant concern in the design of high-frequency cores, and Section 5 describes the power-reduction methods. Section 6 focuses on laboratory (lab) characterization including array built-in self test (ABIST), logical built-in self test (LBIST), and frequency shmoo testing.

2. Technology

The POWER6 processor chip is fabricated using the IBM high-performance 65-nm partially depleted SOI process with 40-nm gate length n-FETs, 35-nm gate length p-FETs, and 1.05-nm gate oxides [3].

Device threshold voltages (V_t) and the polysilicon gate length (L_{polySi})

Core logic circuits are implemented with three distinct voltage thresholds (VTs) for both p-FETs and n-FETs. These VTs provide different levels of device-switching performance and subthreshold leakage. Device VT optimization for power is described in Section 5. In addition, the n-FET pass device thresholds and cross-coupled inverter n-FET and p-FET thresholds of the array six-transistor (6T) cells were each independently optimized for read performance and cell stability [4]. These array devices were not available for use by logic circuits. An additional orthogonal device dimension, L_{polySi}, was available to further reduce subthreshold leakage power. A new CMOS process compensation mask (XR) is placed over non-performance-critical devices to provide a 2-nm channel length increase for n-FETs and a 4-nm channel length increase for p-FETs. This mask was used sparingly in the core circuits but extensively in the nest and array circuits.

Back end of line

The on-chip circuits are connected via ten levels of hierarchical back-end-of-line (BEOL) copper wire. The first level (M_1) is an aggressive 180-nm pitch wire built in low-*k* plasma-enhanced chemical vapor-deposited (PECVD) SiCOH interlevel dielectric that serves short, local interconnections within circuit macros. The next three levels (M_2, $M_{3,}$ and M_4), referred to as *1X levels*, have a slightly less-aggressive 200-nm pitch and are also used to connect circuits within macros. The next two levels (M_5 and M_6) have a 2X wider pitch and twice the thickness. The next two levels (M_7 and M_8) have a 4X wider pitch and ~3X thickness. Macro-to-macro interconnects within core units and between core units predominately use the 2X and 4X levels. The 1X, 2X, and 4X levels are built with second-generation, low-*k* SiCOH interlayer dielectric (ILD). The final two levels (M_9 and M_{10}) are 1.2 μm thick built in F-doped tetraethylorthosilicate (TEOS) ILD at 1.6-μm pitch. These layers are predominately used to distribute global power, global clocks, and I/O signal wires. Scaling theory predicts that the delays of wires (scaling constant κ) do not improve between technology generations (**Figure 3**).

These nonscaling wires are problematic because 65-nm device delays are shorter than 90-nm device delays, and if no specific actions were taken in 65 nm, the POWER6 core frequency would have been severely limited by wire delays. One primary goal of the POWER6 core was to limit the wire contribution to the total path delay to no more than 35%. This required the use of hundreds of thousands of repeaters. Repeaters break a long wire segment into two or more pieces and cut the total wire delay in a path by a factor of 2 or more. Tools were developed to automatically place repeaters in long wire segments [5]. Unfortunately, each repeater adds ~2-FO4-inverter delays and many paths had insufficient timing slack to tolerate this repeater delay. The design of the POWER6 core provided two primary solutions to these timing-critical wire paths. The first and more preferred solution adds staging latches along the wire whenever the logic allows delaying of signals by one or more cycles (e.g., trace and error signals). The second solution upgrades the wire from a lower layer to a higher layer (e.g., 1X to 2X). The higher-layer wires have significantly lower resistance and resistance·capacitance (*RC*) delay. This reduces wire delay by a factor of 4 in the case of upgrading from a 1X to a 2X layer, but even more importantly it can reduce the number of repeaters along a timing-critical wire path.

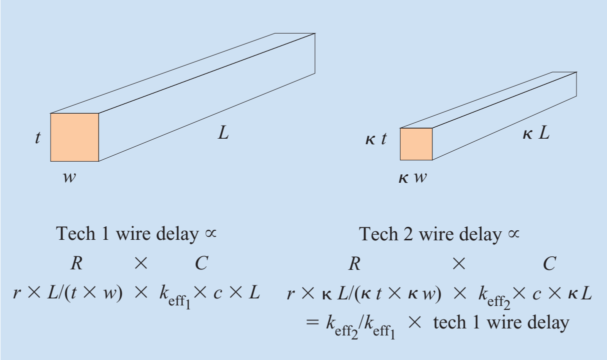
t
L
w
$\kappa\, t$
$\kappa\, L$
$\kappa\, w$
Tech 1 wire delay ∝
$R \quad \times \quad C$
$r \times L/(t \times w) \times k_{\mathrm{eff}_1} \times c \times L$
Tech 2 wire delay ∝
$R \quad \times \quad C$
$r \times \kappa\, L/(\kappa\, t \times \kappa\, w) \times k_{\mathrm{eff}_2} \times c \times \kappa L$
$= k_{\mathrm{eff}_2}/k_{\mathrm{eff}_1} \times$ tech 1 wire delay

Figure 3

Self-loaded, scaled wire delay.

3. Circuits

The POWER6 processor macro circuit design is accomplished using a strictly enforced methodology to ensure proper electrical operation at the required 13-FO4 frequency. Everything from latch designs to circuit styles to clocking was tightly controlled, tuned, and checked using sophisticated tools, which were applied on both custom and synthesized (random logic module [RLM]) circuits.

Local clock and latch design

The POWER6 processor employs a dual-clock system for synchronization; the two phases of a cycle are c1 and c2. This design choice gives circuit designers more granularity for dividing logic into pipeline states. For low-power operation, some logic paths can be configured to operate in "pulsed" mode in which the c1 clock is held active (high) and the c2 clock is pulsed at the end of the cycle [**Figures 4(a)** and **4(b)**]. For a more detailed discussion of clock distribution, refer to the section on global clock distribution.

A single global clock is distributed from a central phase-locked loop (PLL) to each macro on the chip. Inside the macros, the global clock is buffered and shaped by a local clock buffer (LCB). Three types of local clocking are supported:

- c2_chop "true pulse" clock.
- Full-phase dual c1/c2 clocks.
- c2 "pseudo-pulse" clock, in which c1 clock is held active while c2 clock pulses.

These clock waveforms are illustrated in **Figures 4(c)**, **4(d)**, and **4(e)**, respectively. There are two motivations to use pulse clocks. First, they reduce ac clock power because only a single clock is switching. Second, pulse clocks allow

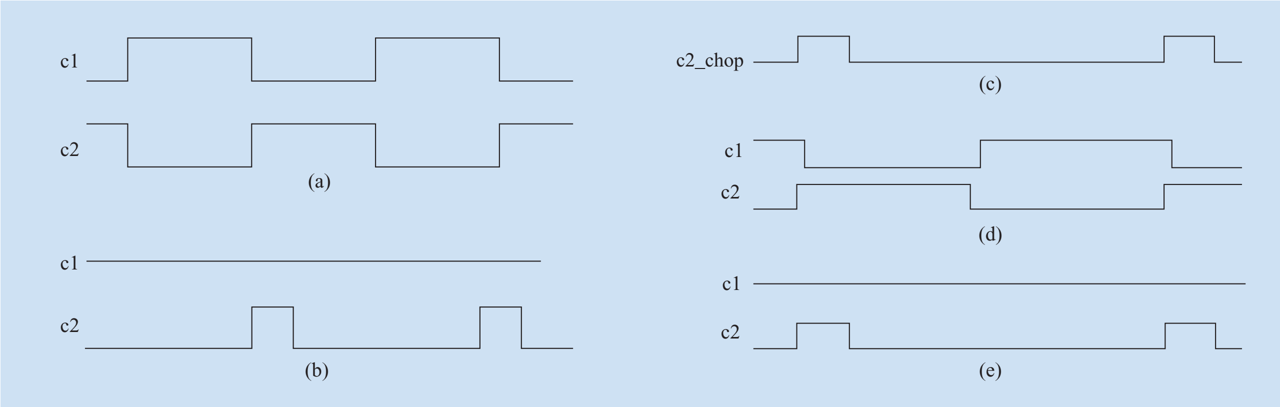

Figure 4

Clock waveforms: (a) regular dual clock; (b) pulsed mode; (c) c2_chop "true pulse" clock; (d) full-phase dual c1/c2 clocks; (e) c2 "pseudo-pulse" clock.

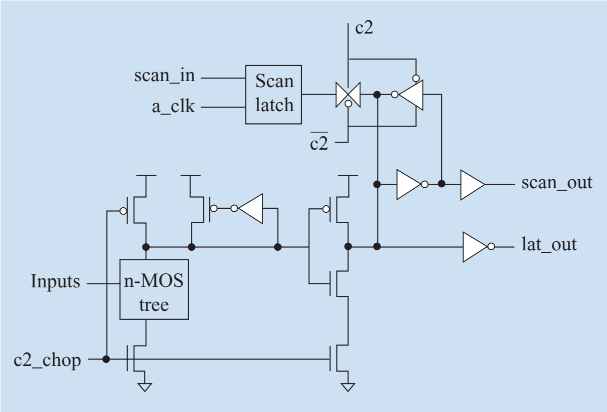

Figure 5

Dynamic multiplexer latch. (n-MOS: n-type metal-oxide semiconductor.)

the use of a single L2 latch (as opposed to an L1–L2 latch pair) in the datapath, thus reducing latch insertion overhead. The main drawback of pulse clocks is that they require data to be held stable longer at latch input. In the POWER6 processor, the rising edge of the c2 clock, the falling edge of the c1 clock, and the rising edge of the chopped c2 (c2_chop) clock are all closely aligned to occur at the cycle boundary. Data is launched from the level-sensitive scan design (LSSD) slave latch on the rising edge of the c2 or c2_chop clock and is captured in the LSSD master latch on the falling edge of the c1 or c2_chop clock. Figures 4(c) through 4(e) illustrate that the extra hold time equals the width of the c2_chop pulse clock. Thus, pulsed clocks become impractical for circuits in which there is little logic between adjacent stages of the pipeline.

The c2_chop "true pulse" clock drives a dynamic mux latch, as shown in **Figure 5**. All inputs to the dynamic pull-down network, as well as the latch output, are static and work seamlessly with the static circuit families used in this processor. The latch is scanned at reduced speed with clka/c2 clocks. The LBIST sequence of *N* cycles starts with an "at-speed" c2 clock, followed by *N* c2_chop clocks.

Dual-phase c1/c2 clocks drive either a scannable master–slave latch [**Figure 6(a)**] or a split scannable L1–nonscannable L2 latch [**Figures 6(b)** and **6(c)**]. The LCB that generates the c1 and c2 clocks for the scannable LSSD master–slave latch is programmable either to generate dual-phase dual c1/c2 clocks or to keep the c1 clock in the active state while pulsing the c2 clock. This leads to an overall chip power reduction, as explained above.

The dynamic mux latch was used sparingly because there is no means to recover from a fast (hold)-path fail. For this reason, the hybrid pulsed latch [**Figure 6(d)**] was used extensively in timing-critical custom circuits. This latch is designed to run with the c2_chop pulse clock. In this configuration, we incur an insertion delay of only a single L2 latch. In addition, there is a "safety mode" in which we can also run the latch in dual-phase c1/c2 clock mode (at reduced frequency). This safety mode was used during bring-up of early hardware. It is also the default mode used in burn-in stressing of the chips.

All LCBs are designed to operate in several clock modes that delay local clock edges under the control of scannable clock tuning bits:

- *Delay c1 falling-edge mode* provides timing-critical (slow)-path relief or stresses the fast path when the latch is in full-phase dual c1/c2 mode.

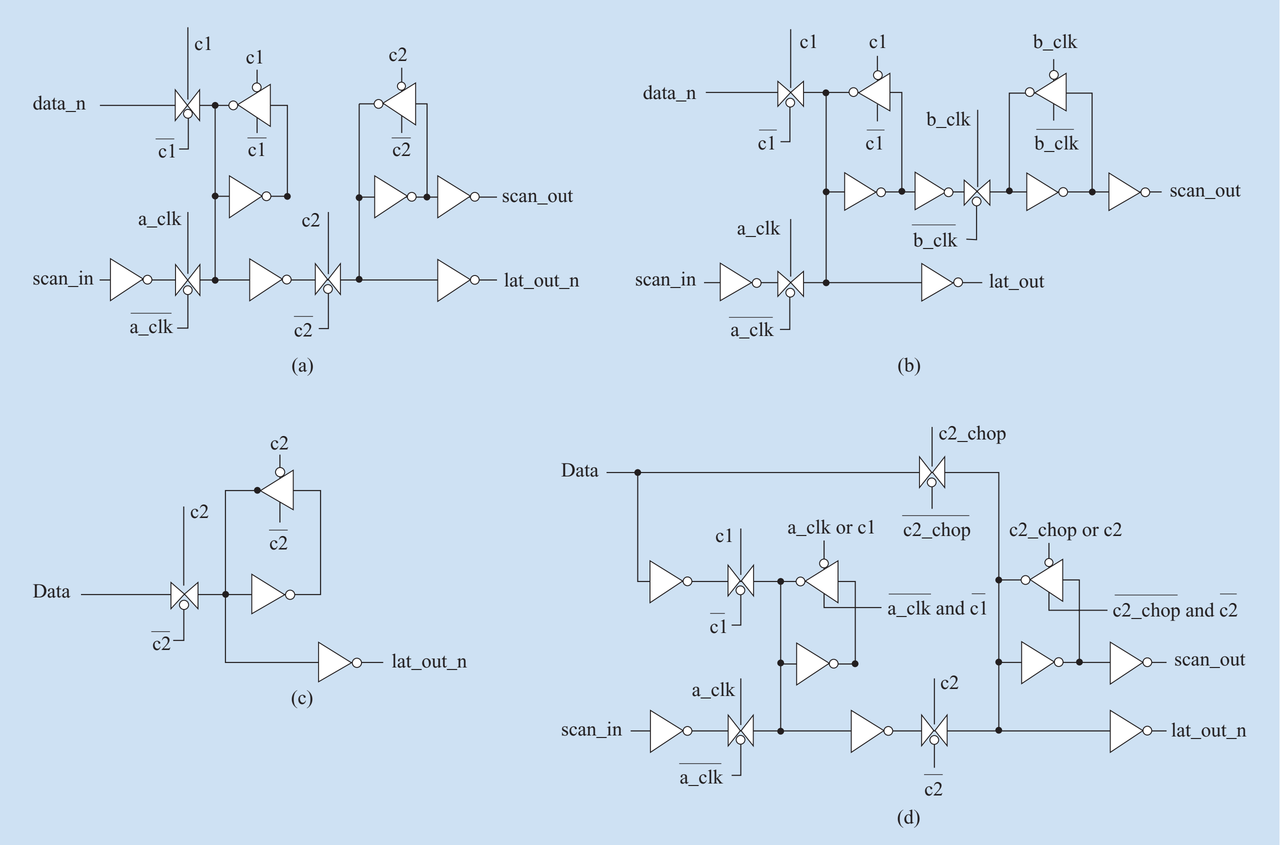

Figure 6

(a) Scannable master–slave flip-flop; (b) scannable L1 latch; (c) nonscannable L2 latch; (d) pulsed hybrid latch.

- *Delay c2 rising-edge mode* provides fast-path relief or stresses the timing-critical (slow) path.
- *Delay c2_chop/c2 falling-edge mode* varies the width of the clock pulse, provides latch writability relief or stress, or stresses the fast path when the latch is in pulse mode.

The granularity of these LCB controls was placed at the macro level via a series of scan-only latches so that all LCBs inside a given macro would similarly stress the clock edges.

Circuit styles

The POWER6 processor was designed almost exclusively using static CMOS circuits. Only the static random access memory (SRAM) and register file macros were permitted to use other circuit design styles, for example, a dynamic circuit design.

Similarly, a typical set of circuit blocks was designed for the latches using a highly tunable cell methodology that dramatically reduced the need for a custom layout of components. The predefined set includes inverters, NAND circuits, NOR circuits, AOI circuits, OAI circuits, and other specialized topologies such as fast XORs, XNORs, and transmission gates used mainly for wide muxes in which area was a critical concern.

When the predefined circuit blocks did not meet the circuit requirements, a full custom design was performed following strict guidelines. For example, scannable register file cells and SRAM cells were carefully designed by a specialized array and register file team [4].

Custom circuit flow

The POWER6 processor is designed using a strict design methodology, which includes regular power and ground connectivity, as well as consistent clean signal routing. Every aspect of the macro design is checked using a set of sophisticated circuit tools, which guarantees that the macro achieves functionality, electrical integrity, and timing specifications.

Most macros adhere to an 18-track vertical dataflow bit image. On the 1X (M_2 and M_4) vertical wiring planes,

Table 2 IBM circuit tools.

Tool	*Use*
IBMmlsa (macro-level signal analysis)	Noise checking
EinsTLT	Timing and frequency analysis
EinsCheck	Topology and circuit guidelines
Common power analysis methodology (CPAM)	Power analysis
EinsTLT electromigration	Electromigration
EinsTuner	Device-width tuning
LAVA	VT substitution

each bit consists of, from left to right, three signal tracks, followed by two ground tracks, followed by another seven signal tracks, followed by two power tracks, followed by another four signal tracks. The pattern is then stepped to the right. This bit pattern gives a total of 14 signal and 4 power tracks; these power tracks align with the unit and core vertical power distribution. On the 1X (M_3) horizontal wiring planes, most macros adhere to a 20-track image. Each 20-track pitch consists of, from bottom to top, four signal wires, followed by two power wires, followed by eight signal wires, followed by two power wires, followed by four signal wires. The horizontal bit dimension that is used throughout the custom and random logic macro designs is 3.6 μm and the vertical power distribution grid is 4 μm. All predefined circuits (latches, LCB, book cells) were designed to this 18-track bit image. Higher-level metals were similarly engineered to keep contiguous routing and power distribution at the macro and unit boundaries.

The macro circuit design flow is initiated by placing, from left to right, 16 latches, followed by a bank of LCBs occupying 4-bit spaces, followed by 32 latches, followed by another LCB, followed by another 16 latches. This 16-4-16-16-4-16 pattern is repeated as needed along the y-dimension of the macro. After the latches and logic books are placed within a custom macro, the internal interconnects are routed via a combination of automated routing and skilled layout personnel. Generally, the timing-critical signals were routed manually before any other wires were routed. Then, automated in-house routing tools were used to finish the remaining routes. Design rule checking (DRC) and logical-versus-schematic (LVS) checking, respectively, guaranteed that the layout adhered to technology ground rules and was functionally equivalent to schematic. Macro abstracts were generated directly from layout and provided a condensed description of macro size, wire blockages, and pin locations for unit integration.

Electrical correctness checking and tuning

The electrical correctness of the POWER6 processor was checked extensively with in-house tools (**Table 2**). More-detailed descriptions of the circuit-checking tools can be found in another paper in this issue [5].

The aggressive POWER6 processor frequency target was achieved through extensive circuit tuning. The IBM EinsTuner tool [5], an EDA (electronic design automation) transistor-level device width tuner, was used throughout the design to tune device widths for minimum delay (maximum speed). Critical-path circuit cross sections were tuned prior to any macro definition. Macro schematics were tuned based on estimated internal parasitics, primary input (PI), primary output (PO), boundary timing, and load assertions. When layout was complete, a final layout-aware device-width-tuning pass was performed. During layout-aware tuning, the physical area allocated to each circuit gate or bounding box constrains the sizes of the devices within the gate. The IBM LAVA (leakage avoidance and analysis) application [5] was used to tune the device VTs; specifically, it identified gates and devices on timing-critical paths and switched those devices to lower VTs (typically regular VT to low VT). LAVA was also used to reduce leakage power; it identified gates with sufficient timing slack and switched the devices for those gates to higher VTs (typically regular VT to high VT).

4. Integration

Global clock distribution design

A low-skew, high-frequency global clock distribution network was designed to support the high operating frequency. The basic clock tree structure is based on a proven grid-tree methodology employed in prior server processor chip designs and high-end game chips [6]. The clock output of PLL is distributed by a clock tree consisting of inverters and shielded high-level wires to local clock sector buffers that are evenly distributed throughout the chip, as depicted in **Figure 7**. The sector buffer, in turn, drives a part of a large-area clock grid through a local H-tree. Low local clock skew was achieved by individual width tuning of the local H-tree on the basis of actual local clock loads. The connections from LCBs to the clock grids were made using reserved tracks in order to facilitate an incremental update without affecting other existing signal wires.

Using standard RC modeling and buffering practices, the high-frequency clock requirements would reduce the maximum wire length between one stage of clock buffer to the next, resulting in many more levels of buffering.

Using more buffers generally results in higher power, more delay, and more sensitivity to process, voltage, and temperature uncertainties. To achieve the desired performance and minimum power, accurate frequency-dependent transmission-line models were used for all critical clock wires. The inductance effects permitted designers to use fewer buffers.

With the improved modeling, optimization, and design methodology, the POWER6 chip actually uses fewer inverter stages for gain than the POWER5 chip. On POWER6 chips, there are seven levels of inverter buffering between the PLL and sector buffers. The total delay from PLL output to LCB is nearly two full clock cycles. Assuming the same variability to power supply and across-chip linewidth variations (ACLV) as in previous designs, higher variability in terms of percentage of cycle time is expected. To tackle this potential issue, additional efforts were made to optimize the distribution tree from PLL to local sector buffers. Specialized tuners were employed to optimize stage-to-stage distance, buffer sizes, wire widths, and wiring structures. The end result is reduced total distribution delay and, more importantly, lower sensitivity to power supply and ACLV.

Power supply noise can vary significantly across the chip; the edge of the chip can experience high V_{dd} while the center experiences low V_{dd}. Consequently, little benefit would result from designing the V_{dd} noise response of the clock distribution to produce longer cycles, specifically during large V_{dd} droop events.

Another feature of the POWER6 processor clock design is the more balanced duty cycle of the clock [7]. Because of the transmission line nature of the high-level shield wiring used, its reflection effects may be used to correct clock duty cycle with careful design. For a certain operating frequency range, optimum wire lengths may be determined for the distribution. In addition, a duty-cycle adjustment circuit was implemented at the PLL output stage for static adjustment as required.

The POWER6 chip has five separate clock grids, one for each core, one for the nest, and one for each memory controller. Communication among circuits on different grids is generally synchronous, except for communications between nest and memory controller. Since the clock grids are not joined, and because in some configurations, the core and nest can run at different voltages, there are potential static clock skews across the grids. A high-resolution, high-linearity programmable-delay clock buffer was designed to alleviate static clock skews. The programmable-delay buffer used in the POWER6 processor has a total range of 40 ps and step value of 2 ps. These delay buffers were strategically placed to allow flexible controls through service elements. The optimal delay settings can be determined empirically or from on-chip measurement circuits such as the Skitter circuit [8].

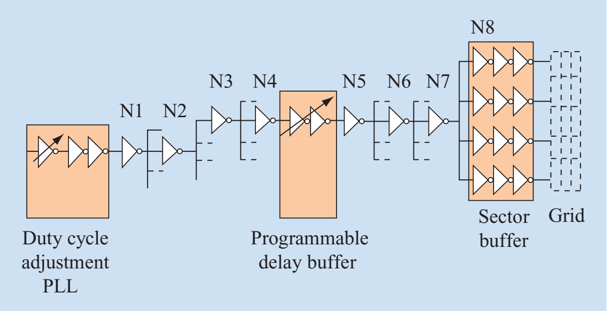

Figure 7

Global clock distribution topology. (PLL: phase-locked loop.)

Another contributor to the high-frequency clock design is the low-parasitic–high-aspect ratio clock buffer design. Because of the high loads driven by clock buffers, macro internal wiring parasitics must be very low in order to prevent degradation in clock signals. The buffers are also designed to tightly couple to power grids in order to minimize variability when placed at different locations of the chip. The high-aspect ratio allows the buffers to be placed in reserved, regularly spaced column stripes that were preallocated early in the POWER6 processor design, specifically for clock optimization. Special care was taken to minimize blockage to horizontal wiring layers.

Power planes

There are two major power planes for the two operating voltages of the chip: V_{dd} with nominal voltage set at 1.15 V at a module pin for the nest, two cores, and non-timing-critical array circuits; and V_{cs} with nominal voltage set at 1.30 V for timing-critical circuits and voltage-sensitive memory cells in all the arrays of the chip.

Custom macro design methodology

Traditionally, wire parasitics in most custom macros are estimated manually on the basis of rough placement of a small amount of timing-critical components from the macro schematic during the pre-physical-design (PD) schematic design phase. The estimated wire parasitic is then manually back-annotated as electrical elements into the schematic for timing analysis. Custom macro size is estimated by summing the area of all the leaf cells in the schematic with some contingency for area increase resulting from design changes over time during the pre-PD schematic design phase. Macro detail placement and routing begin when the logic becomes stable and when macro schematic timing and size are in accordance with

Table 3 The 130-nm and 65-nm metal layer and via resistances.

Metal layer and metal-to-metal via	*Resistance in 130-nm technology*	*Resistance in 65-nm technology*	*65-nm to 130-nm resistance ratio*
PolySi	120 Ω/μm	320 Ω/μm	3X
M_1	0.07 Ω/μm	0.22 Ω/μm	3X
M_2 to M_4	0.06 Ω/μm	0.17 Ω/μm	3X
M_1 to M_2 via	<1 Ω	6 Ω	>6X
PolySi to M_1 via	<1 Ω	44 Ω	>44X

the cycle-time and area objectives. Manual wire parasitics estimation and placement are time consuming and can be incomplete and inaccurate. Because metal layer resistance is 3X higher in 65-nm than in 130-nm technology, via resistance is 44X higher, and BEOL technology scaling lags behind front-end-of-line (FEOL) technology scaling, parasitic wire delay becomes a larger portion of the cycle time in 65-nm than in 130-nm technology. In 65-nm technology, post-PD-extraction base timing can be different from schematic base timing with manually estimated parasitics by as much as 30%. The large timing discrepancy can lead to an enormous amount of rework and difficulties in timing closure during post-PD design phase. **Table 3** lists the metal layer and via resistances of the two technologies.

POWER6 processor macro design methodology was developed to be placement based during the pre-PD schematic design phase so that wire parasitics, macro size, and macro pin locations could be estimated accurately. Two new design tools, PIP (placement by instance parameters) and STEP (Steiner estimated parasitics), were developed to support the placement-based methodologies. These tools can be used to aid detail placement in custom macros during the pre-PD schematic design phase, to support automatic wire parasitic estimation and modeling in custom macros on the basis of detail placement in the macro layout, and to provide a layout with components and pins for custom macro abstract generation. On average, custom macro design effort for schematic design plus detail placement with PIP requires about 1 to 2 weeks for a small macro and 4 to 6 weeks for a large macro. Custom macros designed with PIP and STEP during the pre-PD schematic phase have a timing error within 2% to 7.5%, compared with actual timing with post-PD-extracted data. The result is a single-pass custom macro design process with very little to no timing or physical design surprise during the post-PD design phase.

During the pre-PD schematic design phase, POWER6 processor custom macro design methodology required the schematics to be designed with detail placement. This can be accomplished with the aid of PIP to provide a more accurate placement-based estimate on pin locations, macro size, form factor, and track utilization. During the pre-PD schematic design phase, timing methodology requires all parameterized standard cell schematics used for custom macro design to include input and output wire and via parasitic resistances. It also uses STEP to estimate wire parasitics of all nets in the macro for more accurate timing model generation. Integration methodology during the pre-PD schematic design phase requires all abstracts of the custom macros to be created from layouts with detail placement. Macro abstract pin locations are based on actual macro driver and receiver placements. Macro abstract input pin capacitance is estimated by STEP during this "bottom-up" pre-PD design phase. As the design progresses, custom macro I/O pin placements are refined by macro designers working together with a unit integrator for timing and wire-ability optimization. A new integration-verification methodology checks on macro layout with placed components and pins created by PIP to ensure that post-PD custom macro layout can be routed in the unit floorplan without causing any design rule conflict between macro and unit.

POWER6 processor custom macros [9] are designed with static parameterized standard cells that are similar to those used in the POWER5 processor and the IBM System z990 system. I/O resistances are included in the parameterized standard cells schematics to improve timing accuracy. PolySi gate resistance, polySi to M_1 via resistance, and M_1 pin resistance between gate input pin and polySi gate are modeled as input resistance. Diffusion to M_1 via resistance, M_1 wires resistance for strapping the multiple output fingers into a single pin, and M_1 to M_2 via resistance between gate output pin and source diffusion are modeled as output resistance. Input and output resistances depend on the number of fingers of the gate. Extracted resistance data from samples of cell layouts are curve-fitted to Equation (1):

$$R_{\mathrm{in/out}} = \mathrm{K}_0 + \mathrm{K}_1/(N_{\mathrm{finger}} + P_{\mathrm{finger}}). \qquad (1)$$

K_0 and K_1 are constants and are different in input and output resistance models. N_{finger} and P_{finger} are number of n-FET and p-FET fingers in the gate. **Figure 8** shows the input and output resistances as a function of FET fingers for a two-input NAND gate.

Placement with PIP

PIP is used to aid detail placement in custom macro design during the pre-PD schematic design phase. During the placement process, PIP attaches five parameters to each instance in the schematic. The values of these instance parameters define the relative placement of each instance in terms of horizontal bit position and relative vertical stacking position. PIP calculates the absolute coordinate for each instance on the basis of instance parameter values, instance names, and PIP cell-view properties. PIP then automatically places each instance in the layout according to the calculated absolute coordinates. Net connectivity is also copied from the schematic into this layout for use by STEP to subsequently estimate wire parasitics. The five PIP instance parameters are fipBit for horizontal bit position, fipYlevel for vertical position, fipPcSkip for vertical spacing between instances, fipRot for rotation, and fipSnap for aligning instances horizontally along a line. For simple placement, designers have to specify placement values only for fipBit and fipYlevel; the other three PIP parameters can retain their default values. To place an array of instances with consecutive horizontal placement across multiple bits, designers only have to specify a single fipBit value for placement of the first bit of the array instance. PIP interprets the array notation in the instance name, extrapolates fipBit values for the rest of the array instances, and places the entire array instance accordingly. In addition to the instance parameter, PIP uses a special symbol ("...") to indicate a numeric pattern for PIP values ending with the symbol. The three-dots symbol "..." signals PIP to extrapolate placement values for the rest of the array instances on the basis of the given numeric pattern and to place instances with certain patterns horizontally or vertically. **Figure 9(a)** illustrates a schematic with two sets of inverter instances, I6 and I9, and the placed layout created by PIP. Both I6 and I9 are array instances. Each array is 4 bits wide. I6 has a single fipBit value of 3 that directs PIP to place the array of inverter instances consecutively across to the right starting from bit position 3, as shown in the bottom row of instances of the layout in **Figure 9(b)**. I9 has a fipBit value ending with the three-dots symbol, which instructs PIP to place the array of inverter instances across from left to right on the even bit positions starting from bit position 6, as illustrated in the top row of instances of the layout.

If the leaf cell contains pins, designers have the option of using a PIP routine called *generate layout pins* in order

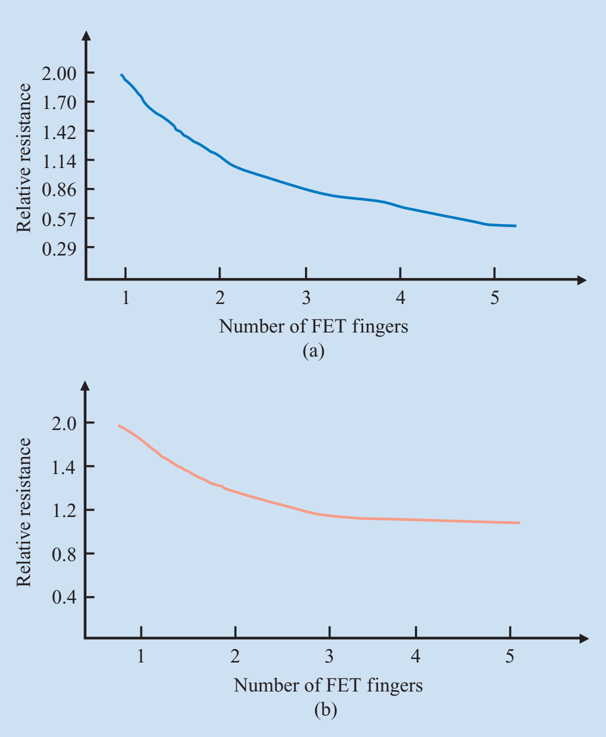

Figure 8

Parameterized standard cell (a) input resistance and (b) output resistance.

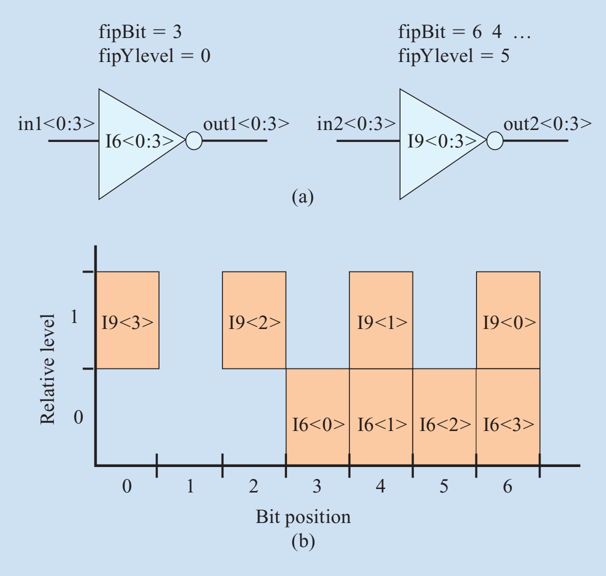

Figure 9

(a) PIP schematic; (b) placed layout.

to propagate the lowest-level (or leaf-cell) pins up the hierarchy to higher level (e.g., macro I/O) pins. The layout with components and pins placed is called a *placed layout*. A macro abstract is then generated from the placed layout and used for unit floorplanning during the pre-PD schematic design phase.

Wire parasitic estimation with STEP and timing with parasitic VIM

STEP is used to estimate Steiner graph lengths of all the nets in the placed layout and to add parasitic models with the estimated Steiner lengths into a schematic VLSI integration model (VIM), an IBM internal format of netlist. In the beginning of the schematic design process, STEP attaches default net attributes to all signals in the schematic to represent metal layer, wire width, wire spacing, neighbor hostility, and contingency for non-ideal routing. The circuit designers can optimize these attributes on the basis of timing requirements and STEP will use modified attributes to update parasitic models. During netlisting, STEP calculates the Steiner graph length for each net on the basis of the pin locations of the components in the placed layout. STEP then uses the calculated Steiner graph length, together with optimized net attributes, to create parasitic models and stitches the models into the schematic VIM. The schematic VIM with estimated wire parasitic is called *parasitic VIM* (PVIM). The PVIM can be generated in 5 to 10 minutes for a small macro with fewer than 10K transistors and 20 to 30 minutes for a large macro with 50K transistors. The IBM EinsTLT [5], an EDA transistor-level timer, can be used to generate a macro timing rule from PVIM. Accurate optimization of the circuits can be obtained either manually or with EinsTuner, together with PVIM by optimizing device type and size, component locations, pin placement, wiring layers, wire width, and wire spacing.

Pre-PD macro design optimization

During the pre-PD schematic design phase, a custom macro design is iterated through five design steps until macro timing is within a certain predefined range of the target. These five design steps include the following:

1. Update schematic topology for functional changes.
2. Update schematic topology and device width for timing optimization on the basis of feedbacks from timing analysis and EinsTuner.
3. Update detail placement with PIP for PVIM generation with STEP.
4. Generate macro abstract from placed layout for unit integration.
5. Update macro timing rule with new PVIM for timing.

Macro routing

The custom macro routing phase begins when logic becomes stable and macro timing is within a certain predefined range of the target. Four routing techniques are used by the various POWER6 processor design groups to route their custom macros. They are (1) complete custom hand-route for optimum track utilization and with ~99% redundant via for better yield; (2) complete custom route with custom software for results similar to those of the first routing technique; (3) a mixture of custom hand-route for timing-critical and dataflow nets and auto-route for the remaining nets with an auto-router called WRoute, created by Cadence Design Systems, Inc.; and (4) complete auto-route with auto-router. Each routing technique has a different turnaround time and produces slightly different results in terms of track utilization, route quality, and ease of updates. However, they all produce routed layouts that meet POWER6 processor custom macro timing, checking, and yield requirements.

5. Power

The four primary components of power in the POWER6 core are clock-switching power, data-switching power, gate leakage, and device subthreshold leakage. All components are very sensitive to operating voltage. A 1% reduction in voltage yields approximately a 3% reduction in total power. Adequate array cell stability and array read performance dictate a higher $V_{\min}$ (minimum operational voltage limit), which could prevent the core from achieving its power objective. For this reason, logic and array support circuits are decoupled from array cells. Array cells are operated on a separate (V_{cs}) voltage supply.

Clock-switching power

The clock-switching power is minimized by using several complementary techniques. The first technique is fine-grained clock gating supported by an LCB circuit to "hold" the local c1 and c2 clocks. This differs from previous designs [10], which turned off both local c1 and c2 clocks. In the POWER6 core, the c1 clock is held high so the L1 (master) latch remains open. This provides two benefits over the previous local clock-gating design: It eliminates the timing-critical half-cycle path to intercept the rising edge of the c1, and it eliminates the extraneous turning off (switching) of the c1 clock. Power is eliminated by reducing clock and signal switching; turning off clocks and signals saves power only when these signals are already off. Traditionally, clock gating is coarse grained, whereby clocks of an entire unit are turned off when the unit is not doing any useful work. The POWER6 core achieved even higher power savings through extensive fine-grained clock gating. In order to achieve fine-grained clock gating, logic is designed to determine those registers and latches that will not or do not have to change state in a future cycle and to generate corresponding local clock-gating signals. These clock-gating signals are recomputed each cycle on the basis of the current logic state.

The second clock-switching power reduction technique is called *latch sizing*. The POWER6 core was designed with six distinct master–slave latch power levels (or sizes). In order to minimize latch power, we chose the smallest-size latch that would meet the constraint that all logic paths sourced by that latch achieve the 13-FO4 cycle-time requirement. Code was developed to identify "overpowered" latches and opportunities to reduce latch sizes without breaking the timing requirement.

The third technique is to modify clock frequency. Certain portions of the chip are operated on at a lower frequency. The POWER6 core does not exploit this technique since the microarchitecture requires all pipeline stages to run in lockstep; however, the POWER6 processor nest operates at half the frequency of the core. Pulsed latches are the fourth clock-switching power reduction technique. Pulsed latches have several positive attributes: They reduce the latch delay overhead by eliminating one of the half latches in a conventional master–slave design (as described in the section "Circuit styles") and they eliminate the c1 local clock signal and its associated switching power. The c2 clock is converted from a half-cycle signal to a chopped pulse. This is necessary to eliminate the race condition or flushing of data through the latch. However, this is also the downside of the pulsed latch. It requires additional short-path padding of datapaths feeding into the latch. The final power-savings technique is called *static circuits*. The precharge and footer devices of dynamic circuits introduce a significant clock load. Clock load power is, thus, minimized by implementing all logic functions and all small register files in static circuits.

Data-switching power

Data-switching power is minimized primarily by logic gate sizing. Similar to the latch-sizing methodology described above, the logic device widths are sized as small as possible within the constraint that the logic paths through the device meet the 13-FO4 cycle-time requirement. Clock gating also affects data-switching power because the cone of static circuit logic downstream of a set of clock-gated latches will not switch. This is not always true of dynamic circuits because downstream data-switching power is eliminated only if the dynamic gates precharge and evaluate clocks are gated.

Device leakage power

The technology offers three device thresholds for n-FETs and three device thresholds for p-FETs. Circuits in paths with a sufficient timing margin (>3-FO4-inverter delays) were designed with the highest-threshold (high-VT) devices. These devices have the slowest switching speed but yield the lowest subthreshold leakages. Circuits in paths near the cycle-time limit (~13 FO4) were designed with regular-threshold (regular-VT) devices. These devices switch faster than high-VT devices but at the expense of higher subthreshold leakage. Circuits in the "ultra-timing-critical paths" were designed with lowest-threshold (low-VT) devices. Ultra-timing-critical paths are defined to be only those paths that could not achieve the 13-FO4 cycle-time objective even after applying all known timing delay optimization techniques. Low-VT devices switch faster than regular-VT devices, but because of their high amount of subthreshold leakage, the use of a low-VT device was severely restricted; only 1% of the devices in the POWER6 processor core are low-VT devices.

6. Lab characterization

The design, technology, and product engineering teams extensively tested POWER6 processor hardware at wafer level, after packaging, and then in system-oriented environments. The development team focused initially on the correctness of the design and the ability to test the processor. The next step was to evaluate voltage, frequency, and temperature operating ranges. Gradually, this evaluation included variations in the devices of the 65-nm technology that were designed to stress the POWER6 processor circuits in ways that would identify the weakest points. The results of these stresses are often included in later design revisions, thereby improving the overall robustness (yield, voltage, frequency, temperature ranges) of the microprocessor.

The tools used to evaluate POWER6 processor-based hardware are wafer and module testers, which allow an automatic pattern test mode as well as an interactive "engineering" mode. The automatic mode allows gathering of large test samples for statistical evaluation. The latter interactive mode allows an engineer to manipulate voltage, frequency, clock tuning bits, and software, among others, to isolate problems. In some cases, the problems can be eliminated immediately. Occasionally, the problem becomes a limiting issue that must be fixed in the next chip release. After the wafers are diced, the resulting chips are packaged into modules for further testing and sorting.

For the physical tools to be useful, the design team employs several testing strategies for broad evaluation of the POWER6 processor, including specific features designed into the microprocessor. The test strategies include LBIST, ABIST, functional exercisers (i.e., test programs), and PLL range, jitter, and yield tests, among others. Each test strategy is affected by voltage,

Table 4 Laboratory stress tests: Baseline settings and actions.

Test name	*Frequency start* (GHz)	V_{dd}, V_{cs} *start* (V)	*Action*
Low-voltage f_{max}	3.0	0.9, 1.05	Increase frequency
Midrange f_{max}	4.0	1.1, 1.25	Increase frequency
High-voltage f_{max}	5.0	1.3, 1.45	Increase frequency
Absolute V_{min}	1.6	1.1, 1.25	Reduce V_{dd}, V_{cs} by equal offset
Speed V_{min}	3.5	1.1, 1.25	Reduce V_{dd}, V_{cs} by equal offset
V_{diff} high	1.6	0.9, 1.05	Increase V_{cs}
V_{diff} low	1.6	0.9, 1.1	Decrease V_{cs}

Table 5 Device sample points within the window of process variation.

Name	*Definition (typically ~25-mV V_t shifts)*
Beta low	Stronger n-FET, weaker p-FET
Beta high	Weaker n-FET, stronger p-FET
Strong high V_t	Increase high-V_t n-FET/p-FET performance
Weak high V_t	Reduce high-V_t n-FET/p-FET performance
Weak low V_t	Reduce low-V_t n-FET/p-FET performance
SRAM strong PU/PG/PD	Strengthen array cell pull-up, pass-gate, and pull-down devices individually and in pairs
SRAM weak PU/PG/PD	Weaken array cell pull-up, pass-gate, and pull-down devices individually and in pairs
SRAM stability (strong PD, weak-weak PU)	Shift voltage at which cell contents are stable
SRAM vs. logic (fast, nominal, or slow logic vs. fast, nominal, or slow SRAM)	Increase or decrease SRAM cell performance relative to standard logic performance
Decoupling capacitance (decap) high	Increase decap for switching noise
Decoupling capacitance (decap) low	Decrease decap for switching noise

frequency, and temperature differently when stressing the chip. The results often have to be correlated against each other to identify the correct ones for sorting the POWER6 processor into several targeted power/performance buckets.

As standard practice, POWER6 processor-based hardware is stressed across frequency as well as below and above the nominal operating voltages. This investigation is intended to look for circuit problems in the processor and define the range in which the chip is fully functional for production. The two voltage planes of V_{dd} and V_{cs} add complexity and, therefore, required special tests for proper handling. Several types of such tests are listed in **Table 4**.

The tests of maximum frequency, or f_{max}, identify peak chip frequency as a function of technology speed, as well as slow paths in the POWER6 processor design. Absolute V_{min} is designed to find the minimum voltage at which transistor-switching behavior is functional without restriction due to frequency. For example, this test can show a condition whereby the differences between VT of regular- and high-VT transistors on different power planes prevent a signal from properly switching to the full voltage rail. The speed V_{min} is a minimum voltage test with a frequency component that normally corroborates slow paths identified in the f_{max} tests. V_{diff} tests stress array voltages against read and write performance, looking for the weak points in those structures. As V_{cs} rises above V_{dd} in the V_{diff}-high test, read performance increases, but the array cells become more difficult to write. As V_{cs} drops closer to and below V_{dd}, writing is easier, but read performance slows down and the cell stability degrades (i.e., the cell can be "disturbed" and lose its data).

Temperature is another variable in lab testing of the POWER6 processor. On the wafer testers performing the above tests, the temperature range is limited from a low of −10°C to a high of +70°C. After packaging, POWER6 processors are further stressed in burn-in ovens to +105°C.

Technology has a major impact on processor operation and performance. Any silicon device process has an allowed tolerance range for device performance, and the POWER6 processor circuits must operate across that entire range. So that the operation can be verified, the most important parameters were specifically stressed and the POWER6 chips were evaluated on that hardware. **Table 5** lists key device points used to test these process changes. *Beta* refers to the ratio of electrical conductivity between p-FETs and n-FETs. *SRAM cell pull-up* (PU) refers to the two p-FETs in the feedback inverters of a 6T cell. *SRAM cell pull-down* (PD) refers to the two n-FETs in the feedback inverters of a 6T cell. *SRAM cell pass-gate* (PG) refers to the two (or more) n-FET pass transistors of a 6T cell. Each process split was evaluated against voltage and frequency as previously defined.

Along with identifying circuit problems, the characterization team manipulated various tuning bits and verified these settings across all splits to optimize the circuit yield and performance. Examples of these tuning bits include array clock-chopper pulse width and delay, local clock duty cycle and local clock delay, dynamic circuit pulse width, and others. The effect of tuning can be dramatic. **Figure 10** shows, across a sample of parts, an average gain of ~500 MHz that is directly due to the tuning process.

LBIST is based on the traditional technique of LSSD, in which most latches can be scanned into or out of, in order to set or read the contents. An LBIST sequence starts by scanning the chip latches to a pseudo-random, repeatable value. This is followed by scanning a specific number of system clocks, commonly just one clock. Finally, the latches are scanned and checked for correctness against a calculated result. The POWER6 processor includes a highly advanced and configurable LBIST engine. To simplify debug, the core and nest latches are broken into 71 subsections. Each subsection includes random-value generators and compression latches to facilitate rapid evaluation of results that allow POWER6 processor LBIST to perform tens of thousands of tests per second. Masking logic in the engine allows the engineering team to restrict testing to latches within a scan subchain in order to isolate an individual failing latch. Because of the ubiquitous use of scannable master–slave latches in the POWER6 processor, LBIST covers the highest percentage of circuits in the shortest possible time of any of the other used test methods. While LBIST in the POWER6 processor has a few weaknesses, it provides the broadest look at the microprocessor circuits compared with any of the available tests.

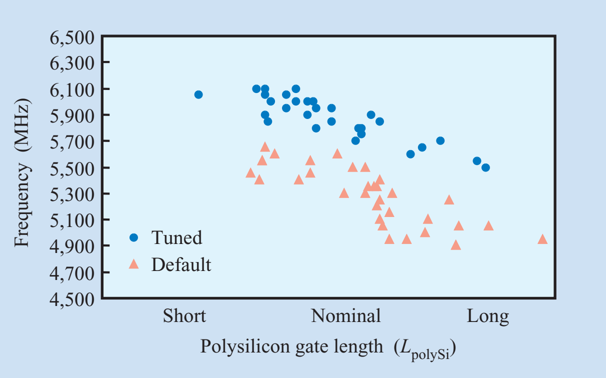

Figure 10

Frequency gain from timing bit tuning. (Republished with permission from [11]. ©2007 IEEE.)

The characterization team defined 11 varieties of LBIST to incrementally cover POWER6 processor circuits. These 11 varieties split between static (dc) and frequency-sensitive (ac) tests. The dc tests remain independent of chip frequency by only clocking the capturing latch after all latches are scanned and the downstream circuits have been evaluated. This result provides baseline data on chip functionality. The ac tests are run at a wide range of frequencies oriented toward isolating defects and design problems that limit chip speeds. At low-speed, the ac test matches the dc result. At high speed, the ac test can deviate from the dc result because of failure of the circuits to evaluate in the chosen clock period, incorrect synchronization between clock domains, and improper gating of control signals, among other problems. Any systematic failures discovered are identified and fixed.

Because of the latch types used in the POWER6 processor, additional tests were needed in the ac and dc groups to check for correct operation of half-cycle latches. Similarly, special controls were added to facilitate testing of on-chip caches and array structures. These circuits can be forced into "write-through" behavior or read-and-write behavior. This allows for limited testing of the array read-and-write circuitry. Array structures with multiport write capability have to limit LBIST to protect against random scans causing device contention. These structures rely on extra circuitry designed to drive a single port during LBIST and vary that port throughout the test to maximize coverage. The enhancements to the engine, extra control circuits, and variety of tests created by the POWER6 processor design and lab teams allowed extensive LBIST testing and enhanced LBIST coverage beyond that of the POWER5 processor. This ability to test nearly all pipeline structures makes LBIST vital in the

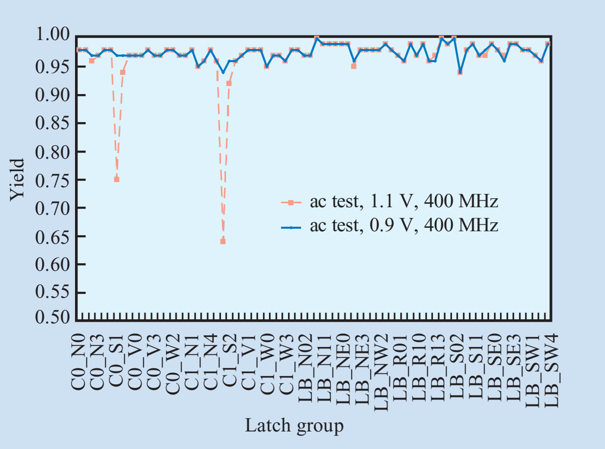

Figure 11

POWER6 microprocessor LBIST single test yield.

POWER6 processor for verifying circuit yield and performance.

Figure 11 depicts an example of reduced yield in a particular LBIST that was due to a systematic problem affected by voltage. At 0.9 V, an average of 95% of the chips that were tested passed the test; this result is typical. At 1.1 V, two areas of the chip experienced higher than normal failures, resulting in yield degrading to 75% and 65% accordingly. With embedded masking logic included in the POWER6 processor LBIST engine, the failing path was isolated to the capturing latch. At that point, clock tuning bits and other stresses were used to manipulate the fail until the cause was understood. Very often, the fail may be fixable by using clock tuning bits, as described in the section "Local clock and latch design." Occasionally, the failure is queued by the design team so that a fix can be included in the next chip revision.

Although LBIST evaluates the POWER6 processor execution pipeline well and has some visibility of cache circuits, it does not test the caches thoroughly. To comprehensively evaluate the cache structures, each POWER6 processor array is connected to a programmable hardware engine, which will perform ABIST. This hardware engine runs at frequencies higher than planned system operating frequencies so it can measure the maximum frequency of the array, in addition to identifying bad or weak cells and other test faults.

A typical ABIST involves a series of write and read operations. Each read cycle is intended to match with a calculated result. Compression registers implemented for each cache in the POWER6 processor capture many such cycles worth testing as a single result, so the entire cache can be evaluated in a small, fixed number of cycles. For some caches, additional registers and address capture logic in the ABIST engine log up to five failing points in the array, and extra SRAM cells designed into those arrays can replace the failing cells. In this way, a cache that has some damage that is due to physical defect or is weak because of process variation can be repaired to operate to system specifications.

Many varieties of tests are used to stress the caches. The standard test is a simple write-and-read pass through the array, with walking bit patterns to stress every circuit uniquely. Another test can write and read between all address combinations to look for sensitivity to switching patterns. "Stability" tests evaluate the ability of the SRAM cells to hold their state at a variety of voltages, particularly at lower voltages as transistor performance decreases.

Each POWER6 processor array contains a set of tuning bits to allow the characterization and design team to debug and stress the array. The characterization team looks for sensitivity to voltage, temperature, frequency, and the SRAM technology parameters previously identified. These sensitivities can often be mitigated or improved by using the tuning bits. Specific examples of tuning in the POWER6 processor include widening local clock pulses to increase the amount of time for cache writes, narrowing clock pulses to increase the frequency at which a cache reads, and aligning certain clock pulses differently to mitigate negative effects of raising the array voltage supply relative to the system power supply.

Figure 12 shows an example of how voltage affects yield against four POWER6 processor wafers: two with the standard design (STD0 and STD2) and two with a device enhancement (EVAL3 and EVAL6). In this example, *yield* refers to the percentage of arrays tested that are functional versus the total number tested. The arrays are tested at three logic voltage points ($V_{\mathrm{dd}} = 0.8$ V, 0.9 V, and 1.1 V) and at six or seven array voltage points (e.g., $V_{\mathrm{cs}} = V_{\mathrm{dd}}$, $V_{\mathrm{dd}} + 0.05$ V, $V_{\mathrm{dd}} + 0.10$ V, and so on). As array voltage (V_{cs}) rises above logic voltage (V_{dd}), and as overall chip voltage (V_{dd} and V_{cs}) increases, the yield increases—a typical response due to the increase in transistor performance. The red ellipses highlights a significant change in the voltage response at 0.8-V V_{dd} and 1.2-V V_{cs}, where the standard design (STD0 and STD2) fails dramatically resulting in lower yield (<30%). The enhancement being evaluated (EVAL3 and EVAL6) improves the yield to 80% to 90%. The POWER6 processor characterization team extensively used ABIST in this manner to validate design or process changes in order to maximize performance, voltage margins, and manufacturability.

With a programmable ABIST engine, a suite of tuning bits for each array macro, and repair circuits included on the large arrays, the POWER6 processor is equipped to test the caches extensively in order to optimize the design. Most problems can be immediately improved, and the voltage and frequency operating ranges enhanced. POWER6 processor arrays currently operate over a range of voltages exceeding 0.8 V to 1.3 V, with performance exceeding 5 GHz.

While LBIST is the most complete coverage tool, ABIST specifically tests the arrays. POWER6 processor characterization relies on functional exercisers to stress the chip as would happen in a complete system. Functional exercisers are programs designed to emulate worst-case system code behavior. In addition to standard exercisers, the lab team has created software targeted to stress certain circuit and logic structures that are not well tested otherwise, or that are unique to the POWER6 processor design. Some of these software exercisers cover array "hit-logic" circuits that combine cache and logic in ways that LBIST and ABIST cannot effectively evaluate. The POWER6 processor also required special multithreaded mixes of specific code and broad-coverage code. This code was designed to maximize power consumption and exacerbate local heating and supply noise to affect peak frequency.

Since wafer-level testing does not allow for the POWER6 chip to access memory, initial testing was required to operate solely from the 8-MB L2 cache. The POWER6 processor implements features in the L2 cache and the memory controller to facilitate processor operation within the L2. These changes allow full system frequency evaluation in real time on the wafer probe. Being able to execute code in this way enables extensive detailed frequency, voltage, and temperature measurements very early in the design and manufacturing cycles.

The POWER6 processor also implements features to enable cycle-accurate reproduction of a given code sequence through extensive clocking and chip control logic. This mechanism is used to stop the machine across many cycles before and after a failure event in order to identify and isolate the problematic circuit. The code sequence proved effective in debugging a circuit problem related to the L1 data cache and is now commonly used for resolving logic bugs.

Correlating the performance of these test methods against each other improves the sorting effectiveness of the POWER6 processor since minimum frequency targets must be met by every chip. **Figure 13** shows a peak frequency comparison of ABIST with various timing settings against a functional exerciser. Additionally, it provides another example of the improvement available by using the extensive tuning bits built into the POWER6 processor arrays. With the default setting, the ABIST maximum frequency generally stays within 200 MHz, yet below the exerciser. With the ability to fine-tune clock timings, the ABIST maximum frequency increases by ~200 MHz, moving it consistently above the functional exerciser peak. Being able to identify such specific frequency limitations and improve such problems increases the yield of chips in the manufacturing and sorting process.

With built-in hardware support for various test methods, the number of technology variations in which it works without performance loss, as well as extensive voltage, frequency, and temperature evaluation, the design of the POWER6 processor has proven to be robust and tunable.

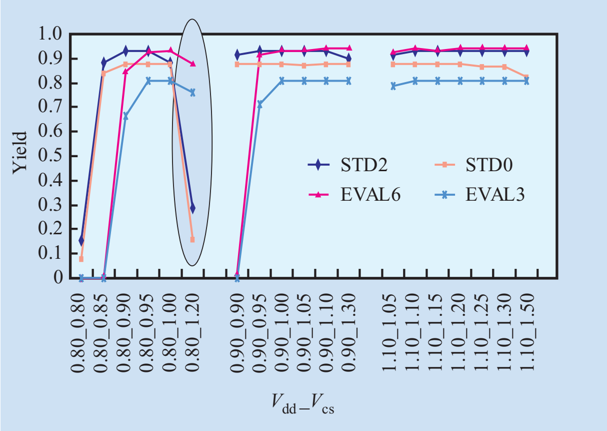

Figure 12

POWER6 microprocessor array macro, yield vs. V_{dd} and V_{cs}, standard design (STD) vs. device enhancement (EVAL).

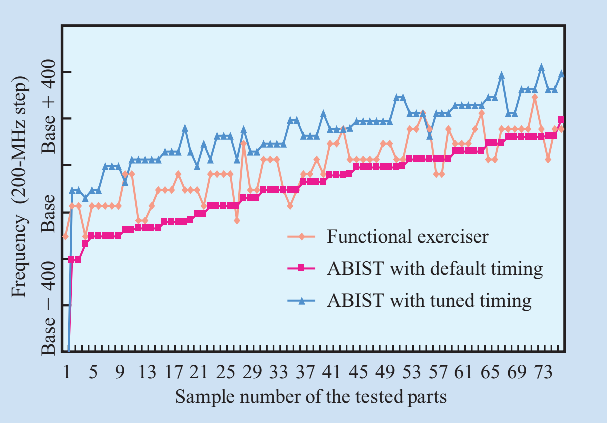

Figure 13

POWER6 microprocessor segment look-aside buffer ABIST f_{max} with tuning bits at 1.1 V/ 1.25 V.

7. Summary

The POWER6 chip has been fabricated in IBM 65-nm SOI process. This process technology incorporates multiple device thresholds and ten layers of copper wiring with a low-*k* dielectric. The logic circuits were predominately implemented in static CMOS circuits in order to reduce power. The POWER6 chip employs three distinct latch designs: a scannable, dynamic front-end latch that incorporates logic function into the latch; a scannable, master–slave latch that can be operated in pulsed mode to save power; and a scannable, hybrid pulsed latch that can be operated in an L2-only mode in order to minimize latch insertion delay, or in a safety mode for burn-in. The low-skew high-frequency POWER6 processor global clock distribution network was described.

The POWER6 processor used a new custom macro design methodology to estimate parasitic resistances and capacitances earlier in the design flow. This methodology reduced the layout rework, extraction, and timing iterations needed to close all custom paths to a 13-FO4 cycle time.

The POWER6 processor parts have been demonstrated to operate in excess of 5 GHz and within the power constraints established for the chip. Chip power dissipation is reduced through modulation of operating voltages, fine-grained clock gating, latch and logic gate sizing, VT optimization, pulsed latches, and half-frequency operation of portions of the chip.

The POWER6 chip has been extensively tested at wafer, first-level package, and system levels. The evaluation was accomplished via LBIST, ABIST, and (real code) functional exercisers across wide voltage, frequency, and temperature ranges as well as process technology variations. These tests identified potential functionality and frequency weaknesses in array, logic, and latch circuits. The robustness and speed of the identified circuits with weakness were modified on subsequent chip manufacturing releases.

*Trademark, service mark, or registered trademark of International Business Machines Corporation in the United States, other countries, or both.

References

1. B. Curran, B. McCredie, L. Sigal, E. Schwarz, B. Fleischer, Y. H. Chan, D. Webber, M. Vaden, and A. Goyal, “4GHz+ Low-Latency Fixed-Point and Binary Floating-Point Execution Units for the POWER6 Processor,” *Solid-State Circuits, 2006 IEEE ISSCC Digest of Technical Papers*, February 6–9, 2006, pp. 436–437.
2. B. W. Curran, Y. H. Chan, P. T. Wu, P. J. Camporese, G. A. Northrop, R. F. Hatch, L. B. Lacey, J. P. Eckhardt, D. T. Hui, and H. H. Smith, “IBM eServer z900 High-Frequency Microprocessor Technology, Circuits, and Design Methodology,” *IBM J. Res. & Dev.* **46**, No. 4/5, pp. 631–644 (2002).
3. E. Leobandung, H. Nayakama, D. Mocuta, K. Miyamoto, M. Angyal, H. V. Meer, K. McStay, et al., “High Performance 65 nm SOI Technology with Dual Stress Liner and Low Capacitance SRAM Cell,” *Symposium on VLSI Technology, 2005. Digest of Technical Papers*, June 14–16, 2005, pp. 126–127.
4. D. W. Plass and Y. H. Chan, “IBM POWER6 SRAM Arrays,” *IBM J. Res. & Dev.* **51**, No. 6, 747–756 (2007, this issue).
5. R. Berridge, R. M. Averill III, A. E. Barish, M. A. Bowen, P. J. Camporese, J. DiLullo, P. E. Dudley, et al., “IBM POWER6 Microprocessor Physical Design and Design Methodology,” *IBM J. Res. & Dev.* **51**, No. 6, 685–714 (2007, this issue).
6. R. J. Restle, T. G. McNamara, D. A. Webber, P. J. Camporese, K. F. Eng, K. A. Jenkins, D. H. Allen, et al., “A Clock Distribution Network for Microprocessors,” *J. Solid-State Circuits* **36**, 792–799 (2001).
7. M. G. R. Thomson, P. J. Restle, and N. K. James, “A 5GHz Duty-Cycle Correcting Clock Distribution Network for the POWER6 Microprocessor,” *Solid-State Circuits, 2006 IEEE ISSCC Digest of Technical Papers*, February 6–9, 2006, pp. 1522–1529.
8. P. J. Restle, R. L Franch, J. K. Norman, W. V. Huott, T. M. Skergan, S. C. Wilson, N. S. Schwartz, and J. G. Clabes, “Timing Uncertainty Measurements on the POWER5 Microprocessor,” *IEEE ISSCC Digest of Technical Papers*, February 2004, pp. 354–355.
9. X. Yu, Y. H. Chan, B. Curran, E. Schwarz, B. Fleischer, and M. Kelly, “A 5 GHz+ Binary Floating-Point Adder for the POWER6 Processor,” *ESSCIRC 2006*, pp. 166–169.
10. J. D. Warnock, J. M. Keaty, J. Petrovick, J. G. Clabes, C. J. Kircher, B. L. Krauter, P. J. Restle, B. A. Zoric, and C. J. Anderson, “The Circuit and Physical Design of the POWER4 Microprocessor,” *IBM J. Res. & Dev.* **46**, No. 1, pp. 27–52 (2002).
11. J. Friedrich, B. McCredie, N. James, B. Huott, B. Curran, E. Fluhr, G. Mittal, et al., “Design of POWER6™ Microprocessor,” *Proceedings of the International Solid-State Circuits Conference (ISSCC)*, San Francisco, CA, February 11–15, 2007, *Solid-State Circuits, 2007 IEEE ISSCC Digest of Technical Papers*.

Received January 8, 2007; accepted for publication August 15, 2007; Internet publication October 23, 2007

Brian Curran *IBM Systems and Technology Group, 2455 South Road, Poughkeepsie, New York 12601*. Mr. Curran is a Distinguished Engineer at the IBM Poughkeepsie Development Laboratory. He received a B.S. degree in electrical engineering from the University of Wisconsin in 1984 and an M.S. degree in computer engineering from National Technological University in 1999. He has designed many bipolar and CMOS, high-performance server processors since joining IBM in 1984. Over the last decade, he has been a driving force behind the high-frequency efforts of IBM. Most recently, he led the industry-leading POWER6 core high-frequency circuit design. Mr. Curran has received two corporate awards, one for S/390* CMOS high-frequency techniques and one for z900 CEC development, in addition to numerous Outstanding Technical Achievement and Outstanding Innovation awards. He has reached the seventh IBM Invention Achievement Plateau and is currently leading the design of a future z microprocessor.

Eric Fluhr *IBM Systems and Technology Group, 11400 Burnet Road, Austin, Texas 78758*. Mr. Fluhr graduated from Georgia Institute of Technology with a B.S.C.S. degree in 1994 and an M.S.E.E. degree in 1996. He then began work at IBM Austin on the POWER3* microprocessor in custom and array circuit design and circuit logic verification. Late in the POWER4* microprocessor development, he switched to load/store logic design through the development of the POWER5 microprocessor. He led the load/store physical design team for the POWER6 microprocessor and is currently the microprocessor technical leader for the POWER6+* microprocessor.

Jose Paredes *IBM Systems and Technology Group, 11501 Burnet Road, Austin, Texas 78758*. Mr. Paredes is a Senior Engineer at the IBM Systems and Technology Group in Austin, Texas. He received a B.S. degree in computer engineering in 1995 and an M.S.E.E. degree in 1996, both from the Georgia Institute of Technology. He has been part of the microprocessor teams since the POWER3 microprocessor, and he has worked in the areas of design for testability, array and custom design, and logic and circuit design. He also led the POWER6 microprocessor recovery unit circuit design.

Leon Sigal *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (sigal@us.ibm.com)*. Mr. Sigal received a B.S. degree in biomedical engineering in 1985 from the University of Iowa and an M.S. degree in electrical engineering in 1986 from the University of Wisconsin at Madison. He worked at the Hewlett-Packard microprocessor development laboratory between 1986 and 1992. He joined IBM Research in 1992 and has since worked on several CMOS S/390 and POWER6 microprocessors.

Joshua Friedrich *IBM Systems and Technology Group, 11400 Burnet Road, Austin, Texas 78758*. Mr. Friedrich was the circuit design leader for the POWER6 microprocessor memory subsystem. After the completion of the first-pass design, he led the circuit characterization of the POWER6 microprocessor, optimizing both frequency and yield. On past POWER* processors, he led the design of several core units and has been instrumental in power reduction efforts. Mr. Friedrich is currently a chip power leader focusing on power reduction efforts and energy efficiency.

Yiu-Hing Chan *IBM Systems and Technology Group, 2455 South Road, Poughkeepsie, New York 12601 (chanyiu@us.ibm.com)*. Mr. Chan is a Senior Technical Staff Member working on the development of a future high-frequency microprocessor. He received a B.E.E. degree from City College of New York in 1977 and an M.S. degree in electrical engineering from Syracuse University in 1983. In 1977, he joined IBM in Endicott, New York, where he worked on circuit and chip design for high-speed impact printers, midrange mainframe systems, and hardware verification engines. In 1994, he joined the advanced CMOS microprocessor team in Poughkeepsie, New York, where he worked on circuit design, physical design, and circuit design productivity tools and methodology for zSeries* servers and POWER6 microprocessors. He holds several U.S. patents and has received several Outstanding Technical Achievement awards.

Charlie Hwang *IBM Systems and Technology Group, 2455 South Road, Poughkeepsie, New York 12601*. Mr. Hwang received a Ph.D. degree from Yale University in 1994. He was involved in commodity DRAM technology development and design, embedded DRAM design, high-speed serial link design, and high-speed clocking design on server processor chips. He is currently working on embedded DRAM design for processor chips.