

IBM eServer z900 high-frequency microprocessor technology, circuits, and design methodology

by 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
H. H. Smith

The IBM eServer z900 microprocessor is a seventh-generation zSeries™ (formerly S/390®) CMOS design which has achieved 1.3-GHz operation. This paper describes the 0.18- μ m bulk CMOS, seven-level copper metal process and the high-frequency circuit, integration, and design methodologies developed to achieve this operation. The microprocessor was floorplanned to closely mimic the flow of the microarchitecture pipeline and reduce the communication delay overhead between units. Novel circuit techniques were used in the implementation of the arrays and cache hit detection logic to save power and reduce circuit complexity without sacrificing performance. A four-dimensional gate library and novel synthesis algorithms were developed to yield synthesized control implementations with the performance characteristics of a fully custom circuit design.

Introduction

The z900 microprocessor is the first zSeries* design to support the z/OS* 64-bit instruction set architecture and

the first zSeries design to operate at a frequency greater than 1 GHz. Achieving this high frequency required careful integration of new custom logic and array circuit designs, high-performance gate libraries, logic synthesis, circuit-tuning algorithms, and noise-analysis techniques. This paper describes these innovations, along with the CMOS technology and z900 microprocessor die characteristics. The z900 microprocessor differs from other processors in the industry in two major aspects. First, the z900 achieves its high-frequency operation without resorting to the deep (10- to 20-stage) pipelines used throughout the rest of the industry. The z900 microprocessor pipeline is only seven stages deep [1]; although it is shallow, it yields superior commercial workload performance, but requires careful floorplanning and allocation of low-resistance wiring resources to minimize communication delay between stages. Second, the z900 includes microarchitecture and technology features to achieve industry leadership in reliability, availability, and serviceability (RAS).

Technology

The z900 microprocessor is fabricated in the IBM high-performance 0.18- μ m bulk CMOS 8S logic technology [2]. Typical chip technology parameters are shown in **Table 1**. This 1.5-V twin-well CMOS technology features a

©Copyright 2002 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

Table 1 Typical chip technology parameters.

Lithography	0.18 μm
L_{eff}	0.097 μm
T_{ox}	2.8 nm
M1 pitch	0.5 μm
M2–M5 pitch	0.63 μm
M6–M7 pitch	1.26 μm
Power supply	1.5 V

0.097- μm channel length, a 2.8-nm gate-oxide thickness, standard and low- V_t transistors, low-resistance Co silicided n^+ and p^+ polysilicon and diffusions, shallow-trench isolation, metal fuses, local interconnect, and seven levels of copper metallization. To maximize chip performance and reliability in high-end z900 systems, chiller technology [3] is used to operate the microprocessor at a nominal chip junction temperature of 0°C. This low-temperature operation improved gate reliability and reduced leakage currents sufficiently to enable a higher application V_{dd} (1.7 V) and a process shift to shorter channel lengths. All z900 microprocessor chips are stressed via a burn-in process to increase system reliability; chip paths are exercised via logic and an array built-in self-test (BIST) at high temperature. System operation at 0°C enabled a lower chip burn-in temperature (100°C) than the traditional 140°C while maintaining excellent (100°C) defect temperature acceleration. The lower burn-in temperature also reduced the chip leakage currents under test and minimized chip-over-current yield losses during burn-in operation. To further improve system reliability, all z900 systems are tested at elevated temperature (60°C). Real code sequences are executed during this run-in operation to exercise paths that are not stressed via BIST. Effective system run-in stress conditions were achieved because of the higher (60°C) defect temperature acceleration. The combination of lower temperature, higher supply voltage, V_{dd} , and a shorter than nominal channel length, L_{eff} , resulted in 20% higher frequencies of shipped microprocessors.

Integration

The z900 microprocessor chip was implemented in a 17.9-mm by 9.9-mm die. The rectangular chip size was influenced by manufacturing productivity considerations and packaging constraints. The rectangular chip improved manufacturing productivity by maximizing the number of chips per wafer and by fitting two dies onto one reticle, which is stepped across the wafer in the manufacturing process. The packaging goal was to fit twenty z900 microprocessor chips (along with second-level cache chips and supporting nest functions) on a 127-mm by 127-mm multichip module (MCM). The chip size and shape

accommodated two microprocessors on an MCM within a single chip site area to achieve the MCM processor density goal.

The z900 microprocessor physical design, like that of the S/390* G4 [4], was constructed via three levels of hierarchy. The topmost level of the hierarchy, or chip level, comprises unit-level components, unit-to-unit interconnects, I/O-to-C4-pad interconnects, and some supporting function for the macro-level components. (C4 pad structures are the solder dots which connect the chip to the MCM package.) The second, or unit, level comprises macro-level components. The chip and unit levels are the global levels which define the placement of lower-level components and the physical interconnections between these components. The third, or macro, level primarily defines the transistor devices and their interconnects. The design was physically partitioned into these three levels to enable concurrent implementation of custom, semicustom, and random logic macros with the floorplanning and wiring at the unit and chip levels. A toolset and physical design methodology supports this concurrent design effort [4]. All three levels of hierarchy adhere to a common, regular power-distribution image. The widths and periodicity of the power and ground wires are fixed for most wiring levels. Tight periodicity, minimum width constraints, and continuous power distribution across all levels of chip physical hierarchy are key to minimizing the effects of capacitive and inductive coupling noise.

The chip-level floorplan (**Figure 1**) was developed in conjunction with the microprocessor pipeline to minimize the interconnect delays within and between the pipeline stages. The data cache (D-cache in Figure 1) was centrally located between two duplicate copies of the fixed-point execution unit (FXU) and floating-point execution unit (FPU). This minimized the cache data communication delay to both of the execution units—traditional microprocessor timing-critical paths. The controls for the data cache were placed to the left of the data cache arrays. Similarly, the instruction cache (I-cache in Figure 1) and controls were centrally located with respect to two copies of the instruction unit (I-unit). The instruction and execution units were duplicated to provide 100% error detection. The branch history table (BHT) was located adjacent to the upper I-unit, and the address translation/compression coprocessor (COP) was located adjacent to the lower I-unit. The recovery unit (RU) was located on the far right side of the die near the FXU and FPUs because one of its primary functions is to compare the outputs of the duplicate execution units. The various types of macro implementations (random logic, or RLM, custom static dataflow, custom dynamic dataflow, SRAM, I/O, and decoupling capacitor) are also illustrated in Figure 1.

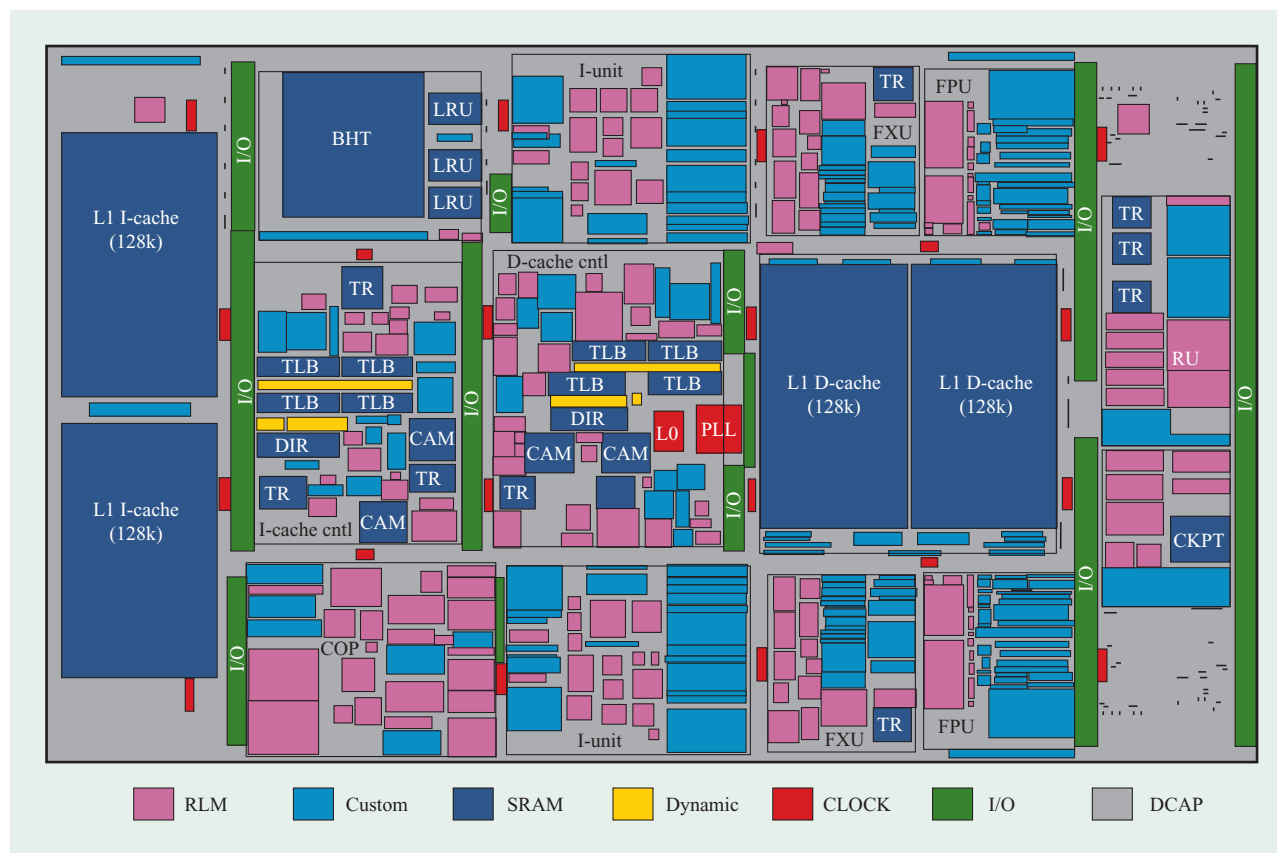


Figure 1

Floorplan of z900 microprocessor.

Wide wire/slew methodology

Achieving frequencies greater than 1 GHz required optimization of the floorplan and interconnections to achieve minimum wire RC delay and to maintain signal integrity by strictly limiting the wire far-end transition times (or slews). Far-end signal transition times were limited to one fourth of the clock period for wires whose timing was critical; these times were relaxed in proportion to the timing slack for wires with some timing margin. Convergence of the design toward timing requirements was an iterative process. During each iteration, a systematic, heuristic approach for improving far-end signal transition times employed the following strategies:

1. Reduction of total interconnect length by optimization of the subcomponent interconnect pin placements at each level of the design hierarchy.
2. Resizing of the interconnect driving circuit to provide greater drive strength.
3. Addition of one or more repowering buffers along the interconnect wire.

4. Widening the interconnection to reduce its resistance.
5. Routing the interconnection to a thicker wiring plane to reduce its resistance.

The ordering of these strategies was significant in minimizing the cost of improving the net slew and/or timing. Metrics were derived to estimate the potential improvement for each strategy, and an automated process was developed to identify the best strategy on a net-by-net basis. In some cases, the implementation of the slew fix was automated through an existing computer-aided design (CAD) tool. Lower-resistance slew fixes (strategies 4 and 5) were implemented, e.g., by passing width and level wiring constraints directly to the wire-routing CAD tool.

Circuits

Logic circuit design

The vast majority of logic circuits were implemented with static complementary logic gates and complementary transmission gates. These logic circuits could have been

Table 2 Array macro characteristics.

<i>Macro name</i>	<i>Array type</i>	<i>Macro density (Kb)</i>	<i>Access time (nominal, ns)</i>	<i>Power (nominal, mW)</i>	<i>Macro area (mm²)</i>
L1 cache	SP SRAM	1152	<1.2	1100	8.03
DIR/TLB	SP SRAM	34/17	<0.7	340/210	0.37/0.28
BHT-LRU	DP SRAM	12	<0.8	130	0.33
TLB-CAM	SP CAM	7	<0.8	170	0.32
Store buffer	Dynamic register file	2.3	<0.6	80	0.14
DIR LRU	Static register file	1.5	<0.7	70	0.15

However, the L1 array and its L1 address-generation/lookup logic are cycle-limiting paths because of its large cache size and relatively shallow processor pipeline. To support the greater than 1GHz frequency objective and to provide a simpler, more robust design to enable an easier technology remap, several new array design techniques were introduced. **Figure 2** is a functional block diagram illustrating the key attributes of the processor SRAM design:

1. *Modular building blocks.* Common array circuits are maintained in an array library and shared by multiple macros. This approach reduces design resources and risk.
2. *High-performance dynamic design.* All array macro inputs and outputs have a static interface typically implemented via master/slave boundary latches. Internal circuits are dynamic in nature (pseudostatic) with local programmable timing control.
3. *Fully programmable internal timing.* Each array macro contains extensive internal timing control to ensure hardware functionality. The input boundary latches, the bit and word decode paths, the data-in path, and the sense amplifiers are all supplied with separate flexible timing engines for delay, pulse width, and skew adjustments. Mode blocks containing four or five bit-scan registers provide the programmable control facility. **Figure 3** illustrates this highly flexible array timing concept.
4. *Pseudostatic circuit approach.* Pseudostatic array circuits are used extensively in the z900 processor in lieu of self-resetting CMOS circuits [5]. The pseudostatic design [6] has the advantages of simpler circuit topology, simpler timing control, lower power, and less chip area. Since the pseudostatic circuit style is less complex than self-resetting CMOS, initial design effort and risk are significantly reduced. This style also allows the arrays to be more readily remapped into a future CMOS technology. **Figure 4** shows a comparison between the self-resetting CMOS (SRCMOS) and pseudostatic circuit techniques.

The z900 processor arrays utilize several SRAM types. These include six-device single-port SRAMs for the L1 cache, directory and translation lookaside buffers (TLBs), content-addressable memories (CAMs), dual-port SRAMs, and multi-port register files for least-recently-used (LRU) arrays. All are designed to support a typical cycle time of 1.0–1.2 ns with a single-cycle read/write latency. To satisfy the z900 robust product reliability, the array macros are all custom designed and verified through extensive circuit simulations to guarantee performance and functionality over the entire manufacturing process range. Array circuits are simulated at twelve corners, which include nominal process and typical system environment, three-sigma worst-case, five-sigma best-case, skewed p-MOSFET/n-MOSFET device process, burn-in environment, and high- and low-voltage stress testing environment. A summary of the representative processor array macros is shown in **Table 2**.

L1 cache hit detection logic

The z900 L1 cache is accessed in a single processor cycle. The cache address is generated in a C0 cycle just prior to the cache access (C1) cycle. In the C1 cycle, four doublewords (8 bytes) of cache data are accessed in parallel with the directory TLB lookup; then one of the four doublewords is selected late in the cycle. To perform all of these functions in a single cycle, the directory and TLB array access times are significantly smaller than the processor cycle time. The lookup path circuitry (comparator and hit logic) were also implemented with novel pseudostatic circuits to attain the highest possible speed. **Figure 5** shows a simplified functional diagram of the cache hit detection logic critical path. The directory array, the absolute/logical TLB arrays, and the cache array are accessed in the C1 cycle. Results from the directory/TLB read are forwarded to two groups of address comparators, a sixteen-way 26-bit absolute TLB/directory compare and a four-way 52-bit logical TLB compare, to determine whether the requested data resides (i.e., hits) in the cache. If there is a hit, one of four late-select signals is activated to select the appropriate doubleword late in the C1 cycle.

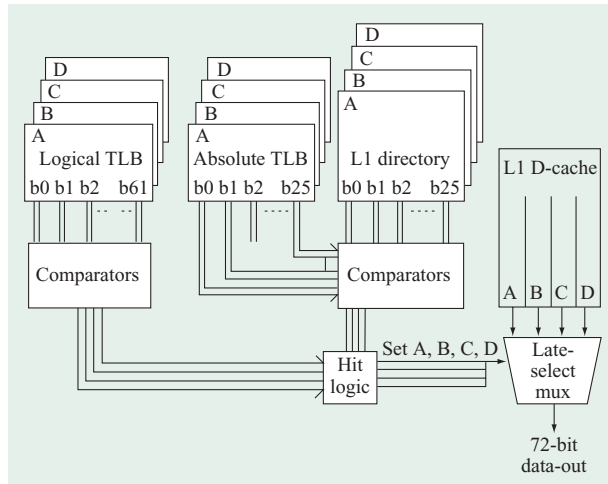


Figure 5

L1 directory hit logic function (C1 cycle).

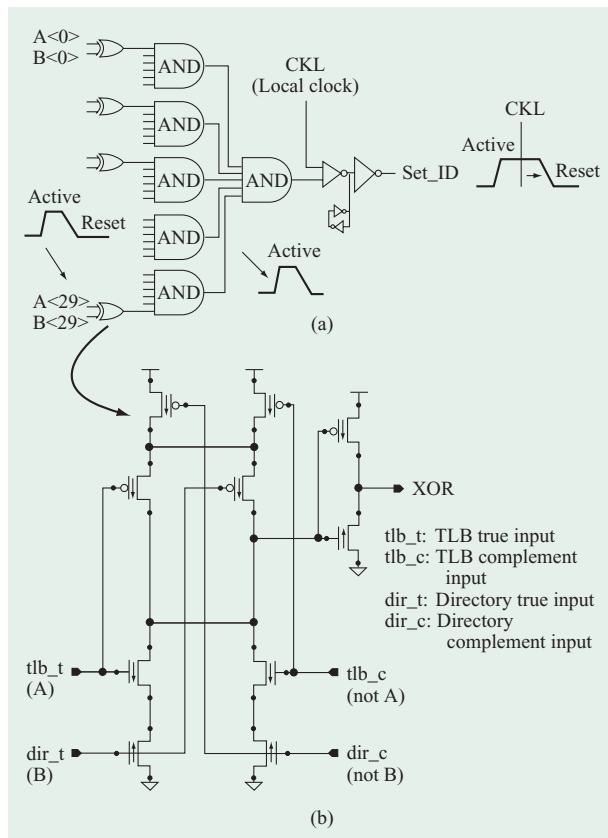


Figure 6

A 30-bit-wide pseudostatic comparator: (a) Logic diagram; (b) transistor-level diagram of bit comparator (XOR) circuit.

Pseudostatic circuits are also utilized in the hit logic path [7]. **Figure 6** demonstrates a 30-bit-wide comparator employing this circuit technique. Dual-phase dynamic outputs from the directory and TLB arrays are forwarded to the bit comparators. Since the dynamic outputs of the array are pulses containing both the active (leading edge with active high level) and reset (trailing edge) timing information, no explicit reset timing generators are needed for the comparator circuitry, as in conventional dynamic design. The overall circuit topology is therefore greatly simplified. Timing complexity and circuit power are also improved. The pseudostatic bit comparator resembles that of a skewed static circuit, with a key difference being that it works with dynamic pulses instead of static signals. Its pull-up to pull-down device strength ratios are designed to emphasize the leading edge (forward path) speed, and its delay performance is comparable to that of a conventional dynamic design. Outputs from the 30-bit XOR comparators are funneled down through two levels of six-way and five-way pseudostatic AND gates. The resulting hit signal is latched via a holding latch for pulse width reshaping.

PLL

The z900 microprocessor includes a single phase-locked loop (PLL) placed near the center of the chip to minimize the global clock distribution delay. The PLL supports frequencies higher than 1.5 GHz with cycle compression less than 35 ps. The worst-case phase difference between the launching clock on the sending chip of a synchronous interface and the capturing clock on the receiving chip is less than 250 ps. This supports 500+ MHz synchronous interface communication. The PLL also operates over the wide range of frequencies and voltages required for chip testing and stressing.

The PLL employs an analog voltage regulation technique [4] to reduce susceptibility to power-supply-induced jitter. Another key feature is the inclusion of empirical tuning/control bits to independently optimize performance for low-temperature, high-performance, air-cooled midrange-performance systems.

I/O

The off-chip I/O circuits were designed to support the high-frequency operation of the z900 processor. The multichip module (MCM) interconnects between the microprocessor and the other system (nest) chips are synchronous to minimize MCM communication latency, I/O circuit area, and power consumption. The most timing-critical MCM interconnect paths are the synchronous bidirectional (bidi) latch-to-latch data paths between the microprocessors and L2 chip data chips. Bidi I/O allows storage data to be propagated in either direction. Approximately 95% of the microprocessor I/Os

are bidi data I/Os. To minimize these interconnect path delays, a novel integrated latch bidi was developed with 10% lower propagation delay than the previous latch and bidi. The integration of the latch and driver eliminated two buffer stages and some global wire delay from the driver path. The latch bidi circuit also integrated the receiver with the master latch to improve the I/O setup time.

The driver output slew improvements in 0.18- μm bulk CMOS are unwelcome, since they would result in larger coupled and MCM delta-I noise. In conventional I/O driver designs, the pre-driver circuit transition is slowed down to reduce output stage slew. However, this severely affects the delay through the I/O driver circuit. A dynamic pulse injection slew rate control was implemented in the driver to reduce the output slew rate. A pulse of a specified duration is generated and fed back into the predriver to slow down the ramping of the gate voltage of the output driver. The tolerances of the I/O driver circuit delay and output slew are significantly reduced with this circuit, since the pulse width tracks with the process variation and power supply.

The I/O circuits were designed to operate in a (industry-standard) 50- Ω package environment. The drivers are source-terminated to reduce power; it was determined that a 30- Ω source impedance gives the best performance with acceptable overshoots. The operation of the source-terminated driver interface can be described as follows. A driver switching from ground to V_{dd} initially switches the near end of the net to a voltage of $[(Z_0/(Z_0 + Z_{\text{driver}}))]V_{dd}$, or approximately $(5/8)V_{dd}$. This voltage wave propagates to the far end of the net and doubles to V_{dd} . The wave reflection then propagates back to the driver, and the driver output current is reduced to zero as the driver output reaches V_{dd} . A driver switching from V_{dd} to ground initially switches the near end of the net to a voltage of $(3/8)V_{dd}$ (with complementary operation). An active clamp circuit limits the over- and under-shoot at the far end of long nets and thus provides controlled reflections. The fastest driver rise performance is achieved by quickly turning the output stage n-FET off and controlling the output stage p-FET to obtain a $(5/8)V_{dd}$ level. This technique is feasible when the interconnection round-trip propagation delay is less than the cycle time, so that the net has time to settle. However, at z900 short cycle times (under 1 ns), there is insufficient time for the net to settle; the net is still actively being driven at the beginning of the next cycle when it must switch again. Shutting off the driver too early creates a very sharp change in voltage and current that can create excessive noise. To accommodate the short cycle time, an adjustment was made to slow down the shutoff of the output devices. This adjustment tunes the wave shape of

the net to obtain the optimum performance and to control the noise level.

After I/O cells are placed and wired within the microprocessor, an automated routine computes the wire resistance between the I/O driver and C4 pad. An n^+ diffusion resistor is then automatically inserted by software to tune the total impedance of every I/O to C4 wire to $30 \pm 1 \Omega$.

All I/O cells are electrostatic discharge (ESD)-protected to a minimum of 4 kV HBM (human body model). This was achieved by using a double-diode ESD device, which was proven to be the most robust ESD device after many destructive ESD experiments in multiple CMOS technologies. The breakdown and breakthrough voltages are constantly being reduced in each new technology. The ESD design improvements for the z900 microprocessor make more efficient use of the active perimeter and area of the diode device. An ESD event always destroys the weakest link; thus, all paths from C4 pad to I/O device were designed to be equally robust.

Design methodology

Gate library

A custom library was developed for use in the synthesis of random logic macros (RLMs). This library contained latch and clocking components along with a large set (>1000) of conventional CMOS static gates for combinational logic. While the circuit style of the combinational logic was the same as for a conventional ASIC standard cell library, the sizing and cell partitioning were developed to yield synthesized gate solutions which mimicked full-custom designs. This was achieved by limiting the cells to only a single (inverting) level of logic, and then offering an extensive range of sizing options for each of the resulting small number of logic functions. **Figure 7** illustrates the four-dimensional library parameter space. The first dimension is power level (uniform scaling of all devices), which is the only dimension exploited by traditional RLM libraries. The next is beta ratio, which provides a tradeoff between rising and falling delays through the gate and gives synthesis better control over equalizing the rising and falling delays of a critical path. The third dimension is taper ratio, which skews the relative size of FETs in a series stack (e.g., the n-FETs in a NAND gate), thus improving the delay through one input at the expense of the other inputs. This can improve critical path performance when there is significant difference in the arrival times at the inputs of a gate. The fourth dimension is device threshold, for which there were two threshold voltages in this 0.18- μm bulk technology; a complete set of physically interchangeable gates were constructed using the low- V_t and nominal- V_t devices. The

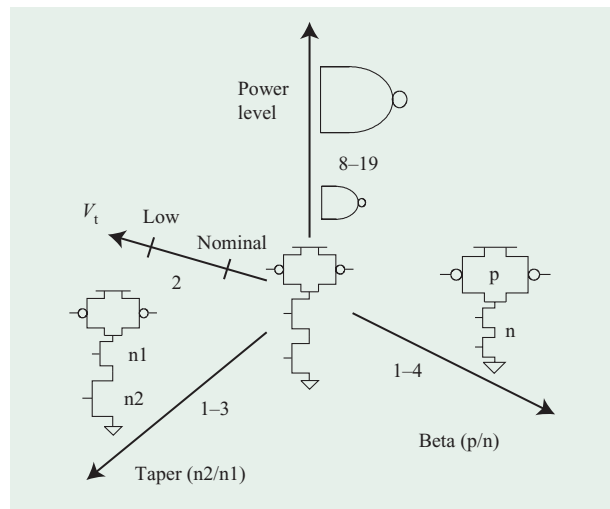


Figure 7

Four-dimensional RLM library.

numbers on the four axes of Figure 7 indicate the range of discrete points in which library cells were implemented.

Logic synthesis/tuning

Nearly half of the processor logic circuits are synthesized. To achieve the aggressive cycle time and schedule, BooleDozer* [8] was enhanced to mimic many of the circuit-optimization techniques employed by highly skilled custom circuit designers. A new synthesis engine was built upon a continuous gate-sizing technique to speed up the process of finding the optimal gate size [9]. New tapered and low- V_t gate-mapping algorithms were integrated into the synthesis. The new synthesis engine, which utilized continuous optimization techniques, was able to fully exploit the enhanced four-dimensional gate library [10–12]. All gate types (inverter, NAND2, NAND3, NOR2, etc.) were timing-characterized via a continuous-gain-based (device-size-independent) technique. Synthesis transforms utilized the gain-based timing models to make quick, delay-centric decisions during both the mapping and timing correction phases. Designs are synthesized into continuous gates, which are later discretized to the closest power level that meets cycle time, load, and slew constraints [13]. By comparison, the original synthesis engine would try each power level, one at a time, in a search for a solution. A circuit designer would normally balance the capacitance gain between stages in a path in an effort to optimize the path delay, and the gain-based synthesis engine explores this same technique.

Tapered and low- V_t gates were used extensively in critical custom-circuit paths in prior zSeries

microprocessors [7]. These gates were utilized by synthesis for the first time in the z900 microprocessor. New synthesis transforms were written to explore the new third and fourth dimensions of the gate library, and typically yielded 15 to 20% reduction in critical path delay. Excessive use of low- V_t gates would result in high chip leakage currents during burn-in stress; thus, synthesis permitted the user to specify a maximum percentage of low- V_t gates.

The random logic macro descriptions (VHDL) were first synthesized into gate-level netlists by the traditional means (BooleDozer). If the resulting implementation met the timing requirements, the gates were placed and routed. Implementations which did not meet timing requirements (i.e., were too slow) were further optimized via synthesis postprocessing low- V_t and tapered gate algorithms. This enhanced RLM synthesis flow thus employs the same techniques and exploits the same circuit optimizations as traditional full-custom static circuit design. Combining the new gain-based algorithm with the low- V_t and tapered gate transforms yielded significant improvements, as shown in Figure 8. The timing slack of the worst paths through nineteen random logic macros in the instruction unit is illustrated for the traditional-synthesis and enhanced RLM-synthesis processes. The enhanced-synthesis process yielded more positive timing slacks, which indicate shorter path delays.

A significant group of custom circuits, concentrated in the instruction unit, were implemented using a semicustom design methodology that employed circuit tuning, automated standard-cell generation, and place-and-route. The process involves schematic entry of a gate-level design using a set of parameterized cells, flattening of this entered structure, tuning the flat gate-level design via EinsTuner [14, 15], generating cells corresponding to the sizes output from the tuning process, and assembling and wiring the final gate-level design. EinsTuner applies a formal optimizer, LANCELOT [16], to maximize the timing slack as computed by the static transistor-level timing tool, EinsTLT. EinsTuner treats the gate parameters (and associated device widths) as free continuous parameters. Initially, gate-level designs with estimated wire parasitics (typically fan-out-based capacitance estimation) are tuned. After placement and routing, the actual wire parasitics are extracted, and the gate-level design is retimed via EinsTLT. Retuning with extracted wire then produces designs which are highly optimized to the specific timing constraints.

In addition, early production of layouts with improved timing accuracy, and rapid late fixes of timing problems helped greatly in timing closure, particularly for paths containing both dataflow and random logic (controls). Functions such as adders and incrementers saw the greatest benefit because of their diverse implementations,

low-symmetry cones, and often less than uniform timing assertions. Circuit tuning in the presence of parasitics generally resulted in delay improvements of 5–15% in designs that had already been through a significant manual circuit designer sizing effort.

Physical design

The synthesis suite of tools included two postsynthesis physical-design-driven phases termed second- and third-pass synthesis. Clocks and scan connections were optimized in the second pass; signal repowering and early-mode padding were performed in the third pass. During the second pass, a design model with disconnected clock and scan nets was physically placed. Such a placement optimizes the data signal paths, since no clock or scan path information is included in the model. The scan and clock net connections were then completed to yield close to minimum total scan and clock wire lengths. This resulted in different ordering of latches on the scan chain and different physical clock-to-latch groupings than specified in the original synthesized netlist. Next, proximate latches were collected into groups of sixteen, and local clock blocks (LCBs) were physically placed in the center of each group to minimize the average distance between an LCB and each associated latch. A clock pin optimization algorithm was developed to swap latches between the initial latch groupings to further minimize this average distance. Minimizing the distance between a latch and the LCB yields shorter clock wires, smaller clock wire *RC* delay, and smaller clock (arrival) skew across the group of latches. Once clock blocks were placed, small perturbations were performed on the original gate placements to eliminate physical placement overlaps of the LCB with other gates. Finally, the random logic description (VHDL) is back-annotated with the modified scan connections.

In the third pass of synthesis, gate-placement data is used to create a minimum spanning (Steiner) tree for each net. These Steiner trees provide more accurate wire-length and net-capacitance estimates than those based simply on fan-out. Repowering and early-mode padding algorithms were then invoked with these improved capacitance estimates. The user specified whether early-mode pad books were to be inserted at latch inputs or outputs. (Pad books are special buffer gates which incorporate non-minimum-channel-length devices.) The gate library included ten pad books of various delays; a third-pass synthesis selected the smallest delay book that fixed the early-mode path. Late-mode timing was reanalyzed to ensure that pad books were not inadvertently placed in a (late-mode) timing-critical path.

This suite of synthesis tools were highly effective. Clock skews were kept at or below the target of 30 ps. The second- and third-pass algorithms permitted wiring of

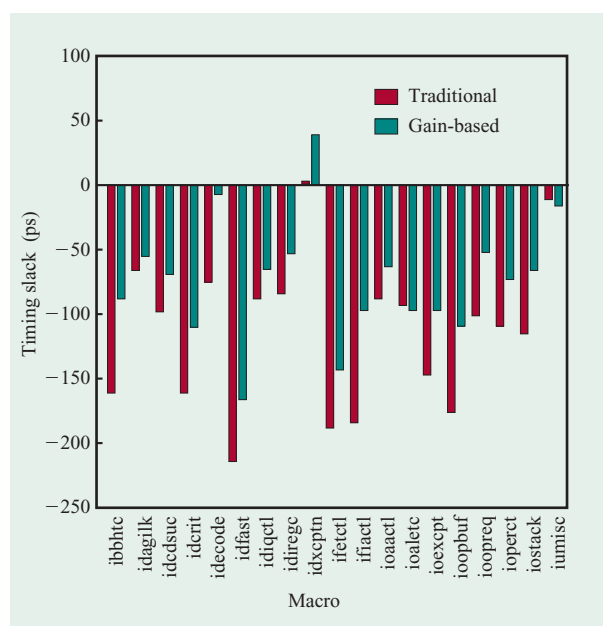


Figure 8

Comparison of traditional and gain-based synthesis algorithms.

macros with latch counts as high as 1400. Late-mode slacks were improved by repowering, and early-mode problems were automatically fixed with limited user intervention.

Noise containment and verification methodology

It is well known that technology scaling trends and performance objectives in VLSI are increasing the importance of accurate assessment of coupling noise for all circuits and nets on the chip. This assessment is needed to ensure circuit functionality and performance for a given cycle-time objective. The primary electrical effect is capacitive coupling between adjacent wires. However, there are cases in which inductive effects in the presence of the local return path resistance dictate the noise profile on a given net. In addition to these electrical quantities, characterization of circuits (such as noise margins and driving impedance) is also required for accurate prediction. Beyond the deterministic evaluation of noise waveforms for a given circuit topology, timing relationships between “aggressor” and “victim” nets determine whether a circuit actually fails. The failure criterion is the timed overlap of the aggressor nets, superimposing and generating a composite noise signature in the sampling window of the victim net. This sampling window is dictated by the required arrival time (i.e., the latest possible signal arrival to make path timing) for the net as computed by the static timing tool. Therefore, the

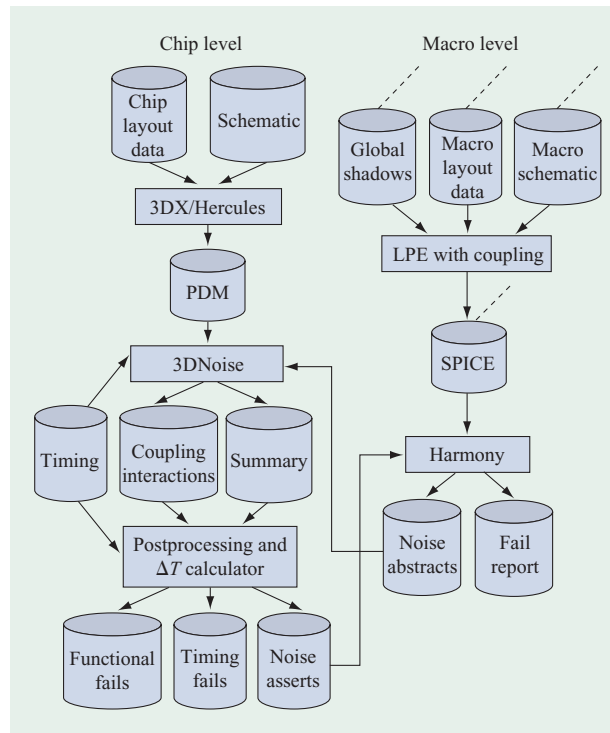


Figure 9

z900 microprocessor noise verification process. (LPE: layout parasitic extraction; SPICE: Software Process Improvement and Capability Emulation, an ISO standard simulation process.)

assessment of noise impact on functionality and timing is a multifaceted operation requiring the following elements for accurate prediction:

- Resistance inductance capacitance (RLC) extraction.
- Circuit characterization.
- Circuit simulation of interconnects.
- Assessment of timing overlap.
- Evaluation of total noise.
- Functional fail assessment.
- Assessment of noise impact on timing.

An outline of the noise verification methodology is shown in **Figure 9**.

The z900 microprocessor chip global net RLCs are obtained via 3DX, an IBM-developed tool for full-chip electrical extraction. 3DX was recently enhanced to perform pairwise high-frequency inductance extractions in the presence of a local return path. This extraction feature is used in conjunction with precharacterization of the return-path proximity effects to create a frequency-dependent equivalent RLC circuit. Noise is evaluated

through the response of this RLC circuit [17]. An example of the equivalent RLC circuit and its return-path frequency response (as represented by R_{12}) is shown in **Figure 10**. The elements in the equivalent circuit are characterized as functions of the 3DX extracted values. 3DX is applied to the flattened unit and chip RC interconnect. (Flattening is a procedure which eliminates the global wiring hierarchy levels.) Wiring within macros is simply modeled as lumped capacitance. This extraction procedure allows independent noise evaluation of the global interconnect and circuits within macros.

The driver and receiver are characterized at the macro pins before noise evaluation of the global interconnect. The output resistance is the key parameter for the driver circuit. The noise margin as a function of input noise pulse width as well as pin capacitance is important for the receiver. The macro drivers and receivers are characterized via Harmony [18], an IBM-developed program for checking static noise. In addition to providing electrical characterization at the macro I/Os, Harmony performs noise-propagation, internal-coupling, and charge-sharing calculations for each circuit within the macro [19].

Noise simulation of the global interconnect is performed using 3DNoise, a netlisting and simulation tool which merges the PDM database output of 3DX with the macro noise abstracts output of Harmony. When the PDM database includes inductance, 3DNoise merges the equivalent-circuit models as determined by the layer information in the PDM. Detailed chip timing data (signal near-end transition times, far-end arrival and required arrival times) is used in the circuit simulations to assess peak noise, noise overlap, and noise sensitivity windows [20]. This assessment is performed using the RICE4 [21] waveform analyzer.

3DNoise outputs the peak noise and pulse width for each receiver in the design and determines which receivers contain noise in excess of the Harmony-computed noise margins (functional fails). In addition to the functional failure report, 3DNoise output includes macro noise asserts files and ΔT data (noise-induced timing deltas). The noise asserts data is further analyzed by Harmony, which propagates the noise at the receiver input into the macro. If the noise does not propagate beyond the input stage, the global net is declared functional and does not require rework. The ΔT analysis incorporates an equation-based approximation to evaluate the shift in timing due to noise. The inputs into the ΔT equation are the signal transition time at the receiver and the 3DNoise-computed peak noise at the instant the signal is switching (i.e., the earliest signal arrival time for early mode and the latest signal arrival time for late mode). The output is a ΔT value for each receiver pin which is subtracted from the late-mode or early-mode timing slack for that pin. Any slack

which becomes negative indicates that the computed noise may cause a timing failure on the path through this pin.

The noise methodology described focuses on identifying “potential” functional and noise jitter problems given the physical, electrical, and timing information provided. The process focuses on minimizing the number of false fails (problems identified by the methodology that would not result in a hardware problem) while ensuring “zero” noise fallout in hardware. Given this context, the number of nets requiring rework was minimal, less than 100 for both functional and ΔT . This small quantity of rework was due to the robust design of the power and ground distribution (grid) as well as electrical analysis of signal nets within the power grid early in the design. Such fails were fixed by the use of solutions such as rebalancing receiver pull-up and pull-down strengths, powering up the victim nets driver, rerouting victim nets to avoid long adjacent runs with aggressor nets, rerouting victim nets adjacent to power or ground distribution wires, and breaking victim nets into two or more sections via repeaters. The functional failures identified were thus addressed individually, and the associated fixes did not significantly disturb the unit and chip wiring. Timing fails were also individually addressed; the methodology uncovered ΔT values of up to 50 ps. Some timing fails were eliminated via the same solutions identified for functional fails. Other fails were fixed by traditional timing solutions: wide wires, low- V_t devices, and tapered gates.

Summary

The z900 microprocessor is a seventh-generation zSeries (formerly S/390) CMOS design. It is fabricated in an 0.18- μm bulk technology with seven levels of copper metal. The microprocessors and second-level cache of the high-end z900 system are chilled to 0°C to achieve high system reliability and increase performance; z900 microprocessors have been operated at 1.3 GHz. The chip was integrated hierarchically to permit concurrent macro design and unit and chip wiring. Custom wiring solutions (wide wires and repeaters) were employed to achieve the tight net slew constraints required for high-frequency operation. Low-power pseudostatic circuits were widely utilized in the arrays and cache hit detection logic. This circuit style is less complex than the dynamic self-resetting circuits used in previous zSeries systems, yet yields the same performance. The z900 PLL has been enhanced with the inclusion of empirical tuning/control bits to independently optimize performance for low-temperature chilled and air-cooled systems. A new reduced-noise I/O driver was developed specifically to support the microprocessor 1+ GHz operation. The random logic gate library incorporated gates of multiple power levels, multiple device thresholds, multiple beta ratios and, for the first time, gates with tapered stacks. New synthesis algorithms were coded to

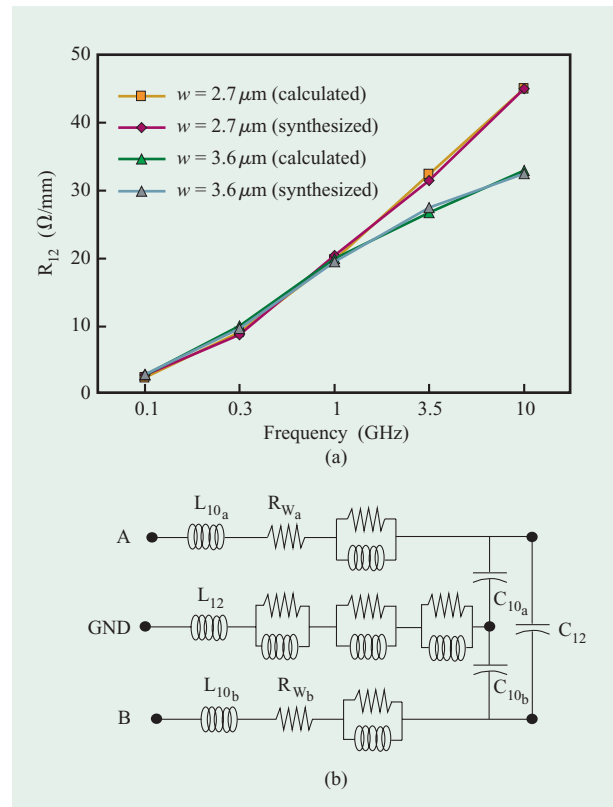


Figure 10

Top-level-metal interactions: (a) Frequency response of R_{12} for wire widths equal to 2.7 μm and 3.6 μm ; (b) synthesized circuit form.

fully exploit all four dimensions of this gate library. Many timing-critical circuits were further tuned via a formal optimizer. Two post-synthesis phases were developed to automate physical design of the local clock and scan connections and early-mode pad insertion. All global wires within the z900 were electrically characterized and analyzed for potential noise (functional) fails and noise-induced timing violations. This sophisticated analysis incorporated capacitive and inductive coupling effects, receiver circuit noise margins, driver circuit impedance, and (for timing violations) aggressor and victim net switching windows.

Acknowledgments

The authors would like to acknowledge Lakshmi Reddy, Chuck Hines, and Peter Chan for their contributions on second- and third-pass synthesis; Mike Bowen, Pat Williams, and Allan Dansky for their noise methodology contributions; Peter Loeffler, Rainer Clemen, Sean Carey, Yiu-Hing Chan, Rocco Crea, Tim Koprowski, Mark Mayo,

Leon Sigal, and Frank Tanzi for their contributions to timing and circuit design; Oliver Rettig, Guenther Hutzl, Nany Kollesar, Tom Wohlfahrt, Azmat Sharif, Adam Jatkowski, and Dave Hillerud for their contributions to unit integration; Leon Stok, David Kung, Prabhakar Kudva, Ruchir Puri, Nathaniel Hieter, David Geiger, Arjen Mets, Andrew Sullivan, and Marianne Knirsch for their contributions to logic synthesis; Chandu Visweswariah and Ee Cho for their contributions to circuit tuning; Tony Pelella for his contribution to L1 cache design; Otto Wagner for his contribution to CAM array design; Pradip Patel for his contribution to array BIST; Jerry Scharff for his contribution to register file design; George English for his contribution to PLL design; Scott Crowder, Steve Luce, and Moe Hamel for their contributions to technology; Rick Rizzolo and Bill Huott for their contributions to test; and Tom Morris, Paul Moore, Bob Hanson, Cyril Price, and Stel Tsapepas for their contributions to project management.

*Trademark or registered trademark of International Business Machines Corporation.

References

1. B. Curran, P. Camporese, S. Carey, Y. Chan, R. Clemen, R. Crea, D. Hoffman, T. Koprowski, M. Mayo, T. McPherson, G. Northrop, L. Sigal, H. Smith, F. Tanzi, and P. Williams, "A 1.1 Ghz First 64B Generation Z900 Microprocessor," *IEEE International Solid-State Circuits Conference, Digest of Technical Papers*, February 2001, pp. 238–239.
2. S. Crowder, S. Greco, S. Ng, H. Barth, E. Beyer, K. Biery, G. Connolly, J. DeWan, C. Ferguson, R. Chen, X. Hargrove, M. Nowak, E. McLaughlin, P. Purtell, and R. Logan, "0.18 μ m High-Performance Logic Technology," *Symposium on VLSI Technology, Digest of Technical Papers*, 1999, pp. 105–106.
3. U. Ghoshal and R. Schmidt, "Refrigeration Technologies for Sub-Ambient Temperature Operation of Computing Systems," *IEEE International Solid-State Circuits Conference, Digest of Technical Papers*, 2000, pp. 216–217.
4. L. Sigal, J. D. Warnock, B. W. Curran, Y. H. Chan, P. J. Camporese, M. D. Mayo, W. V. Huott, D. R. Knebel, C. T. Chuang, J. P. Eckhardt, and P. T. Wu, "Circuit Design Techniques for the High-Performance CMOS IBM S/390 Parallel Enterprise Server G4 Microprocessor," *IBM J. Res. & Dev.* **41**, No. 4/5, 489–503 (July/September 1997).
5. A. Pelella, P. F. Lu, Y. Chan, W. Huott, U. Bakhru, S. Kowalczyk, P. Patel, J. Rawlins, and P. Wu, "A 2ns Access, 500MHz 288Kb SRAM Macro," *Symposium on VLSI Circuits, Digest of Technical Papers*, 1996, pp. 128–129.
6. R. Joshi, S. Kowalczyk, Y. Chan, W. Huott, S. Wilson, and G. Scharff, "A 2 GHz Cycle, 430ps Access Time 34Kb L1 Directory SRAM in 1.5V, 0.18 μ m CMOS Bulk Technology," *Symposium on VLSI Circuits, Digest of Technical Papers*, 2000, pp. 222–223.
7. W. Reohr, J. Navarro, Y. H. Chan, Y. Chan, M. Mayo, B. Curran, B. Krumm, A. Pelella, P. F. Lu, U. Bakhru, S. Kowalczyk, J. Rawlins, S. Carey, and P. Wu, "Precharged Cache Hit Logic with Flexible Timing Control," *Symposium on VLSI Circuits, Digest of Technical Papers*, 1997, pp. 43–44.
8. J. A. Darringer, W. H. Joyner, Jr., C. L. Berman, and L. Trevillyan, "Logic Synthesis Through Local Transformations," *IBM J. Res. & Dev.* **25**, No. 4, 272–280 (July 1981).
9. M. Iyer, L. Stok, and A. Sullivan, "Wavefront Technology Mapping," U.S. Patent 6,334,205, 2001.
10. F. Beftink, P. Kudva, D. Kung, and L. Stok, "Gate Size Selection for Standard Cell Libraries," *IEEE/ACM International Conference on Computer-Aided Design (ICCAD), Digest of Technical Papers*, 1998, pp. 545–550.
11. F. Beftink, P. Kudva, D. Kung, R. Puri, and L. Stok, "Combinatorial Cell Design for CMOS Libraries," *Integration, the VLSI Journal*, pp. 67–93 (February 2000).
12. D. Kung and R. Puri, "Optimal P/N Width Ratio Selection for Standard Cell Libraries," *ACM/IEEE International Conference on Computer-Aided Design (ICCAD), Digest of Technical Papers*, 1999, pp. 178–184.
13. D. Kung, "A Fast Fanout Optimization Algorithm for Near-Continuous Buffer Libraries," *IEEE/ACM International Conference on Computer-Aided Design (ICCAD), Digest of Technical Papers*, 1998, pp. 352–355.
14. C. Visweswariah and A. R. Conn, "Formulation of Static Circuit Optimization with Reduced Size, Degeneracy and Redundancy by Timing Graph Manipulation," *ACM/IEEE International Conference on Computer-Aided Design (ICCAD), Digest of Technical Papers*, November 1999, pp. 244–251.
15. A. R. Conn, I. M. Elfadel, W. W. Molzen, Jr., P. R. O'Brien, P. N. Strenski, C. Visweswariah, and C. B. Whan, "Gradient-Based Optimization of Custom Circuits Using a Static-Timing Formulation," *IEEE Design Automation Conference, Digest of Technical Papers*, June 1999, pp. 452–459.
16. A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *LANCELOT: A Fortran Package for Large-Scale Nonlinear Optimization (Release A)*, Springer Verlag, New York, 1992.
17. H. Smith, A. Deutsch, S. Mehrotra, D. Widiger, M. Bowen, A. Dansky, G. Kopcsay, and B. Krauter, "R(f)L(f)C Coupled Noise Evaluation of an S/390 Microprocessor Chip," *Custom Integrated Circuits Conference, Digest of Technical Papers*, May 6–9, 2001, pp. 237–240.
18. K. L. Shepard, V. Narayanan, and R. Rose, "Harmony: Static Noise Analysis of Deep Submicron Digital Integrated Circuits," *IEEE Trans. Computer-Aided Design Integrated Circuits & Syst.* **18**, 1132–1150 (1999).
19. K. L. Shepard and V. Narayanan, "Noise in Deep Submicron Digital Design," *IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers*, 1996, pp. 524–531.
20. A. Dansky, H. Smith, and P. Williams, "On-Chip Coupled Noise Analysis of a High Performance S/390 Microprocessor," *Electronic Components and Technology Conference, Digest of Technical Papers*, 1997, pp. 817–825.
21. C. Ratzlaff and L. Pillage, "RICE: Rapid Interconnect Circuit Evaluation Using AWE," *IEEE Trans. Computer-Aided Design Integrated Circuits & Syst.* **13**, 763–776 (June 1994).

Received February 15, 2002; accepted for publication April 24, 2002

Brian W. Curran IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (curranb@us.ibm.com). Mr. Curran is a Distinguished Engineer in the IBM Server Group. He received a B.S. degree in electrical engineering from University of Wisconsin and an M.S. degree in computer engineering from the National Technological University. Mr. Curran joined the IBM Large Systems Division in 1984 and has worked on ten zSeries (formerly S/390) bipolar and CMOS systems. He has held numerous design and technical leadership positions in memory system architecture, processor architecture and logic design, and high-frequency circuit design. Mr. Curran holds 22 U.S. patents and has co-authored many papers relating to microprocessor design. He received an IBM Corporate Award for S/390 CMOS high-frequency techniques and is currently an architect of a future (pSeries) eServer microprocessor.

Yuen H. Chan IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (chany@us.ibm.com). Mr. Chan is a Senior Technical Staff Member at the IBM Poughkeepsie Development Laboratory, working on custom VLSI circuit and SRAM designs. He received a B.S. degree in electrical engineering from Union College in 1977 and an M.S.E.E. degree from Syracuse University in 1984. He joined IBM at the East Fishkill, New York, facility in 1977, and has worked on high-performance bipolar, biCMOS, and CMOS logic and array development. Mr. Chan is currently a technical team leader of the zSeries microprocessor array design team. He has received an IBM Corporate Award for S/390 CMOS high-frequency techniques, and several IBM Outstanding Technical Achievement and Outstanding Innovation Awards for high-performance array development. He has reached the seventh IBM Invention Achievement Plateau. Mr. Chan has authored and coauthored many technical papers, and he holds 18 U.S. circuit patents. He is a member of the IEEE.

Philip T. Wu IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (ptwu@us.ibm.com). Mr. Wu received a B.S.E.E. degree from the University of Michigan in 1974, an M.S.E.E. degree from Stanford University in 1976, and an M.B.A. from Rensselaer Polytechnic Institute in 1994. Since joining IBM in 1976, he has been involved in all key areas related to VLSI circuit technology, test, and tools development. He is currently a Senior Technical Staff Manager at IBM Poughkeepsie, managing the chip technology and test strategies for the IBM eServer microprocessors. Mr. Wu has received seven U.S. patents and several technical disclosures in the field of integrated circuits and arrays. He is also a PMI Certified project manager.

Peter J. Camporese IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (pcamp@us.ibm.com). Mr. Camporese received a B.S. degree in electrical engineering from the Polytechnic Institute of New York in 1988 and an M.S. degree in computer engineering from Syracuse University in 1994. In 1998 he joined the IBM Data Systems Division in Poughkeepsie, New York, where he has worked on system performance, circuit design, physical design, and chip integration. He was the technical team leader for the physical design and integration of CMOS zSeries microprocessors (G4 and G7). Mr. Camporese is currently a Senior Engineer in the IBM Server Division working on integration of future CMOS pSeries microprocessors. He holds 12 U.S. patents and is a coauthor of several papers on high-speed microprocessor design.

Greg A. Northrop IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (gnorth@us.ibm.com). Dr. Northrop received a Ph.D. degree in physics from the University of Illinois at Urbana-Champaign in 1982. In 1984 he joined the Physical Sciences Department of the IBM Research Division in Yorktown Heights, New York, and spent ten years doing optical spectroscopy of semiconductors. He then joined the Yorktown VLSI Design Department, where he has since worked on the Alliance series of microprocessors, leading the circuit design of the instruction unit for three years, as well as contributing to Server Division tools and methodology development. Dr. Northrop's technical interests include layout automation, library development, and synthesis and circuit optimization.

Robert F. Hatch IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (rfhatch@us.ibm.com). Mr. Hatch is a Senior Engineering Manager of a CAD application and methodology department in the IBM Server Group. He received a B.S. degree in engineering from Southern Illinois University and an M.S. degree in electrical engineering from Purdue University. Mr. Hatch joined IBM in 1978 at the East Fishkill facility, where he did custom PLA circuit design for a custom VLSI bipolar CPU chip set. His work has been in the areas of bipolar circuit design, MOS circuit design, macro design, VLSI chip design, and VLSI design tools. Mr. Hatch is currently working on the next generation of eServer processors.

Lisa Bryant Lacey IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (lbl@us.ibm.com). Mrs. Lacey received a B.S. degree in computational mathematics from the Rochester Institute of Technology. She joined IBM in East Fishkill, New York, in 1988 and moved to the Thomas J. Watson Research Center in Yorktown Heights, New York, in 1995. She has worked first in simulation and then in logic synthesis.

James P. Eckhardt IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (jpe@us.ibm.com). In 1984 Dr. Eckhardt joined IBM, where he is now a Senior Technical Staff Member. He received a Ph.D. degree from the Georgia Institute of Technology in 1990 for his work in biCMOS circuit development. He has worked in the IBM eServer group for the past seven years designing PLLs for clock generation in microprocessors.

David T. Hui IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (huid@us.ibm.com). Mr. Hui received a B.E.E.E. degree from City College of New York in 1977 and an M.S.E.E. degree from the Polytechnic Institute of New York in 1979. He was with the Rockwell International Corporation in Ohio from 1977 to 1978. He joined IBM in 1979 and has since designed many I/O circuits and ESDs for bipolar high-end systems and CMOS large server systems. Mr. Hui has received four IBM Outstanding Technical Achievement Awards.

Howard H. Smith *IBM Server Group, 2455 South Road, Poughkeepsie, New York 12601 (smithh@us.ibm.com).* Mr. Smith received B.S. and M.S. degrees in electrical engineering from the New Jersey Institute of Technology in 1984 and 1985, respectively. He joined IBM in 1984 as an integrated circuit engineer at the semiconductor development laboratory in East Fishkill, New York, working in the area of high-performance masterslice designs. Mr. Smith is currently a Senior Engineer in the IBM Server Group in Poughkeepsie, New York, where he is responsible for electrical analysis issues associated with high-density CMOS circuit technology and package-related products. His recent assignments include the development and coordination of on-chip noise verification processes for the eServer processor designs. His expertise lies in the area of system-level computer electrical noise modeling and prediction. Mr. Smith has co-authored papers on system-level noise prediction, on-chip interconnects, and electromagnetic characterization of connectors and antennas. He holds several patents on circuit designs and methodology techniques related to his area of expertise.