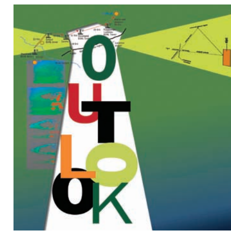
2003 Technology Roadmap for Semiconductors



This update to the 2001 ITRS Roadmap shows the industry shifting its focus toward systems on chip, wireless computing, and mobile applications.

Don Edenfeld
Intel

Andrew B. Kahng
University of California, San Diego

Mike Rodgers
Intel

Yervant Zorian
Virage Logic

Two years ago, our report titled "2001 Technology Roadmap for Semiconductors" (*Computer*, Jan. 2002, pp. 42-53) summarized our assessment of the opportunities and challenges awaiting the semiconductor industry in its continued pursuit of Moore's law. The 2003 edition updates this effort and reveals a significant shift in the industry's focus.

Today, the introduction of new technology solutions is increasingly application-driven, with products for different markets using different technology combinations at different times. General-purpose digital microprocessors for personal computers have been joined by mixed-signal systems for wireless communication and embedded applications. Battery-powered mobile devices are as strong a driver as wall-plugged servers. System-on-chip (SoC) and system-in-package (SiP) designs that incorporate building blocks from multiple sources are supplanting in-house, single-source chip designs.

SYSTEM DRIVERS

The International Technology Roadmap for Semiconductors (ITRS) provides quantified, self-consistent models of canonical products—*system drivers*—that drive the semiconductor industry. These models support extrapolation of future technology requirements for basic circuit *fabrics*—processor, analog/mixed-signal, and embedded memory—as well as the SoC products they comprise. The 2003 ITRS system drivers chapter presents an overarching SoC context for future semiconductor products, along with new discussions of technology requirements for analog/mixed-signal and embedded memory fabrics.

Systems on chip

The SoC product class is a yet-evolving design style that integrates technology and design elements from other system driver classes into a wide range of high-complexity, high-value semiconductor products. SoC manufacturing and design technologies are typically developed originally for high-volume custom drivers such as processors, field-programmable gate arrays (FPGAs), or memories. SoCs integrate building blocks from the other system driver classes and increasingly subsume the application-specific integrated circuit (ASIC) category.

SoCs exist to provide low cost and high integration. Lower implementation cost requires greater reuse of intellectual property as well as platform-based design, silicon implementation regularity, or other novel circuit and system architecture paradigms. Cost considerations also drive the deployment of low-power process and low-cost packaging solutions, along with fast-turnaround-time design methodologies. The latter, in turn, require new standards and methodologies for IP description and test,

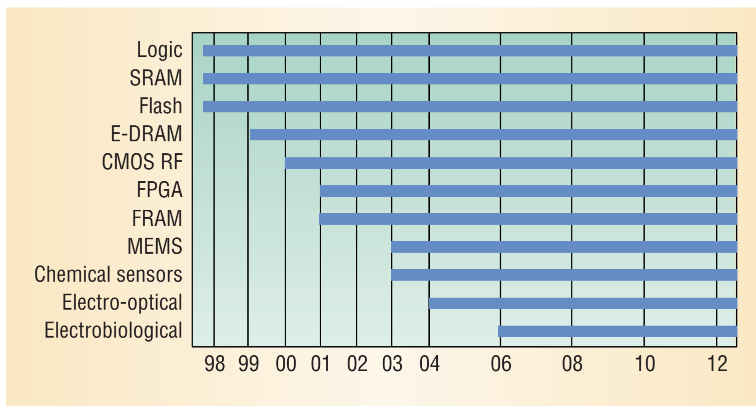


Figure 1. Existing and predicted first integrations of SoC technologies with standard CMOS processes.

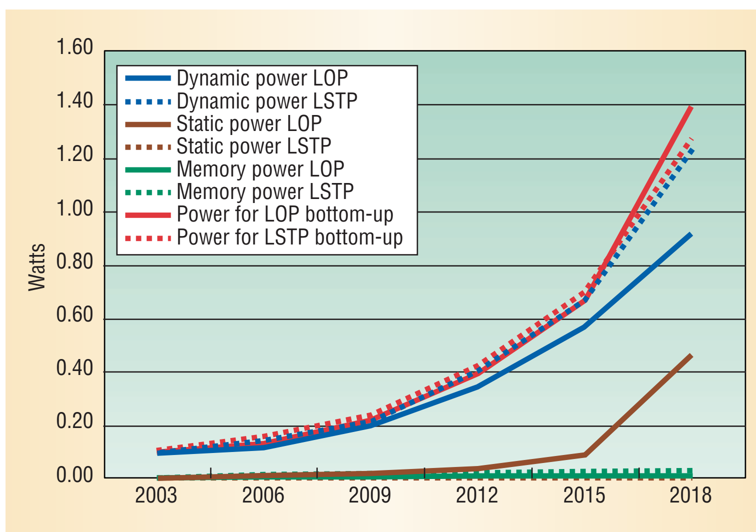


Figure 2. Total chip power trend, in watts, for a SoC-LP personal-digital-assistant application, for both low-operating-power (LOP) and low-standby-power (LSTP) modes.

block interface synthesis, and so on. Heterogeneous integration requires merging reprogrammable, memory, analog and RF, MEMS, and software elements, as well as chip-package-system cooptimization. Thus, SoCs provide the driver for convergence of multiple technologies not only in the same system package, but also potentially in the same manufacturing process. Overall, SoC designs present many challenges to design, test, process integration, and packaging technologies. Most daunting among these challenges are

- design productivity improvement of greater than 100 percent per node, including platform-based design and integration of programmable logic fabrics;
- power management, especially for low-power, wireless, multimedia applications;
- system-level integration of heterogeneous technologies, including MEMS and optoelectronics; and
- development of SoC test methodology, including test reusability and analog/digital built-in self-test (BIST).

Multitechnology integration. Considerations such as cost, form factor, connection speed and overhead, and reliability drive the need to build heterogeneous systems on a single chip. As process technologists seek to meld CMOS with MEMS, bio and chemical sensors, and so on, process complexity and production cost become major constraints. The total cost of processing is difficult to predict for future materials and new combinations of processing steps. However, at present, cost considerations limit the number of technologies on a given SoC: Processes are increasingly modular, but the modules are not generally stackable.

CMOS integration with other technologies depends not only on basic technical advances but also on being more cost-effective than multiple-die SiP alternatives. Figure 1 shows how first integrations of each technology within standard CMOS processes—not necessarily together with other technologies, and not necessarily in volume production—might evolve.

The power challenge. Recent ITRS editions project the implications of *power management*—a grand challenge for the semiconductor industry—on the achievable space of SoC designs. From the functionality of a canonical low-power multimedia PDA application, an architecture of processor, memory, and interface blocks can be inferred. Chip power requirements can then be estimated bottom up from the implied logic and memory content, and from the roadmap for process and circuit parameters.

Figure 2 shows the bottom-up *lower bound* for total chip power at an operating temperature of 100° C, even given the assumption of multiple device technologies integrated within a single core to afford greater control of dynamic power, standby power, and performance. The salient observation is that significant advances are required in architecture and design technology to meet the system targets of 100 mW peak and 2 mW standby power.

Mixed-signal chips

Analog/mixed-signal (AMS) chips at least partially deal with input signals whose precise values matter. This broad class includes RF, analog, analog-to-digital, and digital-to-analog converters—and, more recently—many mixed-signal chips in which at least part of the chip design must measure signals with high precision. These chips have design

and process technology demands that differ from those for digital circuits. Technology scaling is not necessarily helpful for analog circuits because dealing with precision requirements or signals from a fixed voltage range is more difficult with scaled voltage supplies. In general, AMS circuits and process technologies present severe challenges to cost-effective CMOS integration.

The need for precision also affects tool requirements for analog design. Digital circuit design creates a set of rules for correct logic gate function; as long as developers follow these rules, precise calculation of exact signal values is unnecessary. Analog designers, on the other hand, must concern themselves with many second-order effects to obtain the required precision they require. Relevant issues include coupling and asymmetries. Analysis tools for these issues are mostly in place but require expert users, while synthesis tools are immature. Manufacturing test for AMS circuits poses a problem that remains essentially unsolved.

Most analog and RF circuitry in today's high-volume applications resides in SoCs. The economic regime of a mainstream product is usually highly competitive: It has a high production volume and hence requires a high level of R&D investment; thus, technology requirements drive mixed-signal technology as a whole. Mobile communication platforms are the highest-volume circuits driving the needs of mixed-signal circuits.

Mixed-signal evolution. The interplay between cost and performance determines mixed-signal driver evolution, including its scope of application. Although mixed-signal *performance* is an important metric, *cost* of production is also critical for practical deployment of AMS circuits. Together, cost and performance determine the sufficiency of given technology trends relative to existing applications. They also determine the potential of given technologies to enable and address entirely new applications.

Cost estimation. In mixed-signal designs, chip area determines only one of several cost factors. The area that analog circuits occupy in a SoC typically ranges from 5 to 30 percent. Economic forces to reduce mixed-signal area are therefore not as strong as those affecting logic or memory. Related considerations include the following:

- sometimes, shifting the system partitioning between the analog and digital parts can reduce the analog area;
- introducing high-performance analog devices increases process complexity so that solutions occupy less area at the expense of increased total cost;
- technology choices can impact design cost by introducing greater risk of multiple hardware passes;
- parametric yield sensitivities can impact manufacturing cost; and
- a SiP multidie solution can be more cost-effective than a single SoC solution.

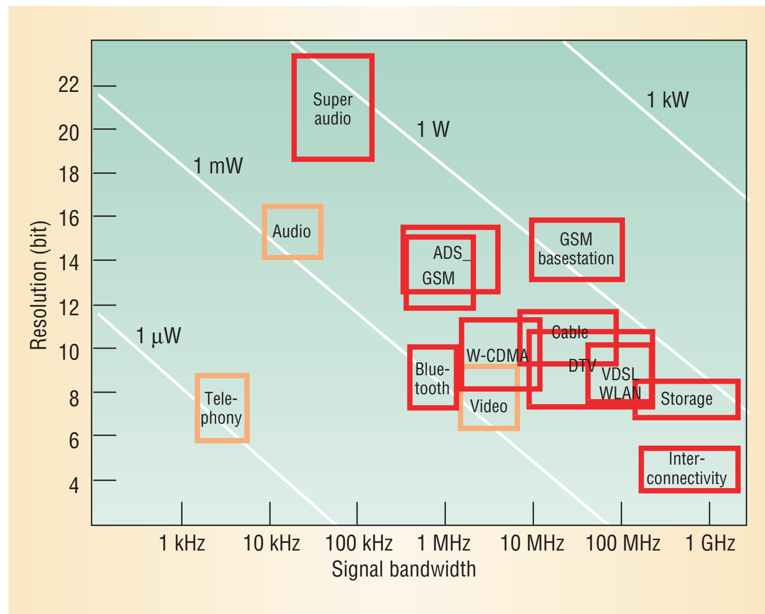


Figure 3. Recent analog-to-digital converter (ADC) performance needs for important product classes.

Such considerations make mixed-signal design cost estimation difficult. We can attempt to quantify mixed-signal cost by first restricting our attention to high-performance applications, since these also drive technology demands. Because analog features are embodied as high-performance passives or analog transistors, that area can be taken as a proxy for cost. Since improving digital density drives transistor scaling, analog transistors can simply follow, and it isn't necessary to specifically address their layout density. At the same time, embedded passives determine total area in most current AMS designs, whose area consumption dominates a system's mixed-signal costs.

Estimating technology sufficiency. Figure 3 shows requirements for analog-digital converter (ADC) circuits in terms of a power-performance relationship. Under conditions of constant performance, a straight line with slope of –1 represents constant power consumption. Increasing performance—achievable with better technology or circuit design—is equivalent to a shift of the power consumption lines toward the upper right. The data shows a very slowly moving technological barrier line for ADCs for a power consumption of 1 W. Most of today's ADC technologies lie below the 1 W barrier, and

The challenge for compound semiconductors is to increase the number of devices per unit area and integrate them with CMOS processing.

near-term solutions for moving the barrier more rapidly are unknown.

Although ADC performance has improved at a rate adequate for handset applications, it clearly has failed to do so for applications such as digital linearization of GSM base stations, or handheld and mobile high-data-rate digital-video applications. Assuming progress at recent rates, manufacturing ADCs with adequate performance in volume will not occur until 2010 or later. Silicon and silicon germanium technologies have the necessary bit resolution, but not the speed, to provide such performance now; on the other hand, III-V compound semiconductor technologies have the speed but not the bit resolution. The challenge for compound semiconductors then is to increase the number of devices per unit area and to integrate them with CMOS processing.

Enabling new applications. In the semiconductor industry, improving technology and design performance enables new applications, which pushes the industry into new markets. Researchers also can use mixed-signal design analysis to estimate design needs and feasibility for future applications and new markets.

Alternatively, when they know the specifications of a new product, researchers can design the technology needed to fulfill these specifications and estimate the timeframe in which the semiconductor industry will be able to build that product with acceptable cost and performance. The ability to build high-performance mixed-signal circuitry at low cost will continuously drive the semiconductor industry to develop new products and markets.

Mixed-signal challenges. For most of today's mixed-signal designs, a voltage difference represents the processed signal so that the supply voltage determines the maximum signal. Decreasing supplies—a consequence of constant-field scaling—mean decreasing the maximum achievable signal level. This strongly impacts mixed-signal product development for SoC solutions. Typical development time for new mixed-signal parts takes much longer than for digital and memory parts, which makes lack of design resources another key challenge.

Overall, the most daunting mixed-signal challenges are decreasing supply voltage, increasing relative parametric variations, increasing numbers of analog transistors per chip, increasing processing speed, increasing crosstalk, and shortage of design skills and productivity.

Embedded memory

SoC designs contain an increasing number and variety of embedded RAM, ROM, and register file memories. Interconnect and I/O bandwidths, design productivity, and system power limits all point to a continuing trend of high levels of memory integration in microelectronic systems. Driving applications for embedded memory technology include code storage in reconfigurable applications, data storage in smart or memory cards, and the high memory content and high-performance logic found in gaming or mass storage systems.

Opportunities and constraints. The balance between logic and memory content reflects overall system cost, power and I/O constraints, hardware-software organization, and overall system and memory hierarchy. With respect to cost, the device performance and added mask levels of monolithic logic-memory integration must be balanced against chip-laminate-chip or other SiP integration alternatives. Levels of logic-memory integration also reflect tradeoffs in hardware-software partitioning as well as code-data balance.

I/O pin count and signaling speeds determine how system organization trades off bandwidth versus storage: Memory access can be made faster at the cost of peripheral overhead by organizing memory in higher or lower bank groups. Access speed also depends on how pin count and circuit complexity are balanced between high-speed, low-pin-count connections and higher-pin-count, lower-speed connections.

In the traditional processor architecture domain, memory hierarchy is crucial in matching processor speed requirements to memory access capabilities. This has led to the introduction of several layers of hardware-controlled caches between main memory and foreground memory in the processor core. Typically, one physical cache memory is present at each layer. However, the choice of hierarchy also has strong implications for power. Conventional architectures increase performance largely at the cost of energy-inefficient control overheads by using, for example, prediction and history mechanisms and extra buffers arrayed around highly associative caches. From the system viewpoint, the embedded multimedia and communication applications that dominate on portable devices can profit more from software-controlled and -distributed memory hierarchies.

Different layers of the hierarchy require different access modes and internal partitionings. The use of page, burst, and interleaving modes, and the

physical partitioning into banks, subarrays, and divided-words and bitlines generally must be optimized per layer. Increasingly dominant, leakage power constraints also lead to more heterogeneous memory hierarchies.

Embedded memory challenges. At the circuit level, scaling issues such as amplifier sense margins for SRAM and decreased I_{on} drive currents for DRAM present two clear challenges. Smaller feature sizes allow integration of more devices into a single product, which leads to increased parametric yield loss with respect to both noise margins and leakage power. Future circuit topologies and design methodologies will need to address these issues.

Another challenge, error tolerance, becomes severe with process scaling and aggressive layout densities. Embedded memory's soft-error rate increases with diminishing feature sizes and affects both embedded SRAM and DRAM. Moving bits in nonvolatile memory can also suffer upsets. Particularly for highly reliable applications such as those in the automotive sector, error correction will remain a requirement that entails tradeoffs of yield and reliability against access time, power, and process integration. Finally, cost-effective manufacturing test and BIST is a critical SoC requirement for both large and heterogeneous memory arrays.

DESIGN

Previous ITRS editions documented a *design productivity gap* in which the number of available transistors grows faster than the ability to meaningfully design them. Yet, investment in process technology has far outstripped investment in design technology. Fortunately, progress in design technology continues: The 2003 ITRS estimates that the design cost of the low-power SoC PDA is approximately $20 million in 2003, versus $630 million had design technology improvements since 1993 not been realized. However, embedded software, manufacturing test, and design verification continue to escalate into crises, even as traditional complexity challenges persist. In the 2003 ITRS, the design chapter includes new material on AMS-specific design technology, as well as the future evolution of design processes.

Facets of the complexity challenge

As our January 2002 ITRS article described (*Computer*, pp. 48-49), design technologists are caught between two basic challenges. *Silicon complexity* refers to the emergence of nanometer-scale variability and physical effects that place long-standing paradigms at risk, while *system complexity* refers to the exponentially increasing transistor counts that are enabled by smaller feature sizes and spurred by consumer demand for increased functionality, lower cost, and shorter time to market. Together, these challenges imply exponentially increasing design process complexity. New paradigms are needed in every area of design technology—design process, system-level design, logical-physical-circuit design, design verification, and design for testability—as we move to the 65-nm technology node and beyond. Examples from the 2003 ITRS include:

Embedded software, manufacturing test, and design verification continue to escalate into crises, even as traditional complexity challenges persist.

- *Communications-centric design.* As it becomes impossible to move signals across a large die within one clock cycle, or to run control and dataflow processes at the same clock rate, design will likely shift to asynchronous or globally asynchronous and locally synchronous styles. A new focus on on-chip communication architectures and protocols will usher in an era of communications-centric design.
- *Robustness.* Network-oriented paradigms, as well as yield and synchronization issues, suggest that future SoC design will contend with potentially lossy communication and node failures, placing a new focus on robustness. For example, implementing a completely specified function in an inherently imperfect fabric demands new approaches to design, mapping, and verification.
- *Scalability.* Tools that operate at the lowest abstraction levels—such as logical and physical design and verification tools—face instance complexities that at least double with each technology node. Scalability requires new ways to manage data, search solution spaces, and map optimizations onto distributed and parallel computational resources. Construct-by-correction methodologies and the rise of reuse demand tool runtimes that are proportional not to the size of their inputs, but rather to the size of *changes* from the previous version of the inputs.
- *Cost optimization.* Multitechnology integration in SoC or SiP brings analog and mixed-signal into the mainstream, as well as codesign of die, package, and substrate levels. System cost optimization must span many degrees of freedom: multiple-die and stacked-die options, package and board interconnects, IP reuse across process generations, and the use of reprogrammable blocks.

AMS scaling and migration faces many challenges, including decreasing supply voltages and increasing relative parametric variations.

- *Cooptimization*. Implementation tools create logic and timing structures concurrently with constraint budgets and spatial embedding. For example, today's register transfer level (RTL) floorplanning and global interconnect planning tools define repeater insertion and pipelining, along with detailed layout of global signals. As logical-physical and layout-clock-test synthesis unifications continue, a near-term goal is cooptimization of timing structure, signaling strategies, logic optimization, placement, and routing in a single environment.

Analog/mixed-signal challenges

Analog and RF circuits differ from digital circuits in that they lack quantized information represented in a defined range of voltage levels for the two states of a bit and during a defined discrete time interval. Rather, analog and RF circuits process continuous signals to a much higher degree of precision or smaller tolerance both in time and amplitude. Therefore, factors such as linearity, noise, parasitic elements, and electrical nonuniformity directly cause distortion and noise in analog or RF signals.

Speed issues, or simply that a signal-recovery circuit produces more noise and distortion than it prevents, make these factors much more challenging and less straightforward in the analog domain. AMS scaling and migration faces many challenges, including decreasing supply voltages; increasing relative parametric variations; increasing numbers of analog transistors per chip; increasing signal, clock, and intrinsic device speeds; increasing leakage and crosstalk in SoC integration; and a shortage of design skills and automation.

In system-level design, the critical AMS challenges are analog circuit nonscalability and analog behavioral modeling and synthesis. Automated analog circuit synthesis and optimization is needed, along with language-level modeling methodologies that permit analysis of overall system function and interfaces with other integrated technologies. Issues include coping with vastly varying time scales in simulation, creating heterogeneous test benches and ensuring coverage, achieving systematic top-down constraint propagation, and mixing functional and structural representations.

Analog synthesis provides the key challenge in logical, physical, and circuit design. Scalable SoC design requires eliminating the analog design bottleneck and leveraging reusable, retargetable analog IP generators. More general techniques can augment today's specialized automatic circuit syntheses for particular circuit classes. Syntheses must also handle future regimes of increased manufacturing variability using, for example, hybrid analog-digital compensation for device mismatch.

In design verification, AMS circuits require checking to specification, not structure. New verification solutions must include statistical techniques, better compact models that speed up simulation while increasing accuracy, and new acceptance criteria. AMS designs also force the issue of performing hybrid-systems verification into the near term. This creates an immediate challenge to improve the current ad hoc approaches and find a path toward more powerful techniques.

Design process trends

Four major trends govern future leading-edge chip design processes and their supporting design system structures:

- *Tight coupling*. Design processes are evolving into collections of modular applications that operate concurrently and share design data in memory. In modern methodologies, optimization loops can no longer contain slow file accesses, and the plethora of design issues requires simultaneous optimization of multiple criteria. Further advances will be needed to avoid noise problems, minimize power dissipation, and ensure manufacturability.
- *Design for manufacture*. More intelligent data preparation requires new characterizations of manufacturing process and cost tradeoffs. While inspection and repair form the largest component of mask-making cost and delay, manufacturers perform the process without insight into design intent, wasting effort in satisfying identical tolerances for every shape. We need a standard framework to communicate design data criticality to the mask-making process and to feed manufacturing complexity back to the design process.
- *Increasing abstraction levels*. Most design today takes place at the gate level for greater productivity and at the RTL to specify design in a modern flow. Continued designer productivity improvements require an emerging system-level of design, well above RTL. Higher abstraction levels allow verification to discover problems much earlier in the design process, which reduces time to market and lowers costs.

- *Increasing automation levels*. Using new models to specify design intent leads to opportunities for introducing other tools, such as synthesis. This trend replaces designer guidance with constraint-driven optimization to reduce the number of iterations in later process steps. In future technology nodes, the system-level specification must include both software and hardware and become the controlling representation for constraint-driven implementation.

Detailed implications of these precepts are given in the context of a new ITRS canonical design flow, which serves as a framework for specification of future design technology innovations. For example, in system-level design the simplification of mixed hardware and software system specification, verification, and implementation requires a new level of abstraction above the familiar RTL. This necessitates the following advances:

- *Reuse-based design*. Reusable, high-level functional IP blocks offer the potential for productivity gains estimated to be at least 200 percent. Preverification and reusable tests reduce design complexity, and libraries of reusable software modules can speed embedded software development.
- *Platform-based design*. An extension of core-based design creates highly reusable groups of cores to form a complete hardware platform, further simplifying the SoC design process.
- *System-level verification*. Raising the abstraction level will require a single notation for system-level design. Several years of experimentation with C, C++, and Java variants has led to the recent emergence of SystemC as an option for building interoperable system models of hardware and software for simulation.
- *Microarchitecture synthesis*. Although system synthesis is extremely difficult, progress will likely start with automatic creation of an effective RTL specification from a slightly higher-level representation of the hardware in microarchitecture-specification form.
- *Hardware-software cosynthesis*. Achieving the best overall solution will require the ability to concurrently synthesize a hardware and software implementation. By 2007, we anticipate that an automated step will replace the manual process of mapping a behavioral specification to a software program and a hardware microarchitecture.

In general, solutions to silicon complexity challenges will entail restrictions on design rules as well as continued improvement in analyses. Potential solutions to system complexity challenges will strive for increased capacity, use of hierarchical methods, and a higher abstraction level of design. Joining these two trends is the evolution of the design process and design system architecture itself, as depicted in Figure 4. Tool integrations must minimize unproductive data translation time and data redundancy and support synthesis-analysis tool architectures that execute concurrently on distributed or SMP platforms. Some specific predictions for the future follow.

In general, solutions to silicon complexity challenges will entail restrictions on design rules and continued improvement in analyses.

Designer productivity will improve through multilevel automation, increased reuse, and freedom *from* choice. To systematically advance design technology in this direction, design must be treated as science, not art, with designers adopting a measure-to-improve mindset. Eventually, predictive models of application- or driver-specific silicon implementation will be needed, while complementary innovations should measure and improve design technology productivity.

Approaches to reduce time to market will include use of reprogrammable and structured-ASIC fabrics, reuse of predesigned cores and platform architectures, and pervasive automation. New tools will be required to support a find-and-try style of reuse-centric design-space exploration and design optimization. Development and formalization of design rules that, if followed, assure reusability, will require associated design and analysis software.

In the long term, building on the *design for manufacturability* trend, we will need restricted design rules, as well as the integrated design and manufacturability optimizations indicated in Figure 4. Even today, the focus on yield-driven layout makes manufacturability a standard design criterion.

TEST

The dramatic increase in SoC and SiP designs, largely targeted to commodity consumer applications, has consequently increased pressure to reduce the cost of test for mixed-technology designs. These designs break the traditional barriers between digital, analog, RF, and mixed-signal test equipment capability requirements, resulting in a trend toward highly configurable, one-platform-fits-all test solutions. The first generation of this equipment combined leading-edge technology from all segments, with the consequence of increasing test cost due to the high capital cost of this

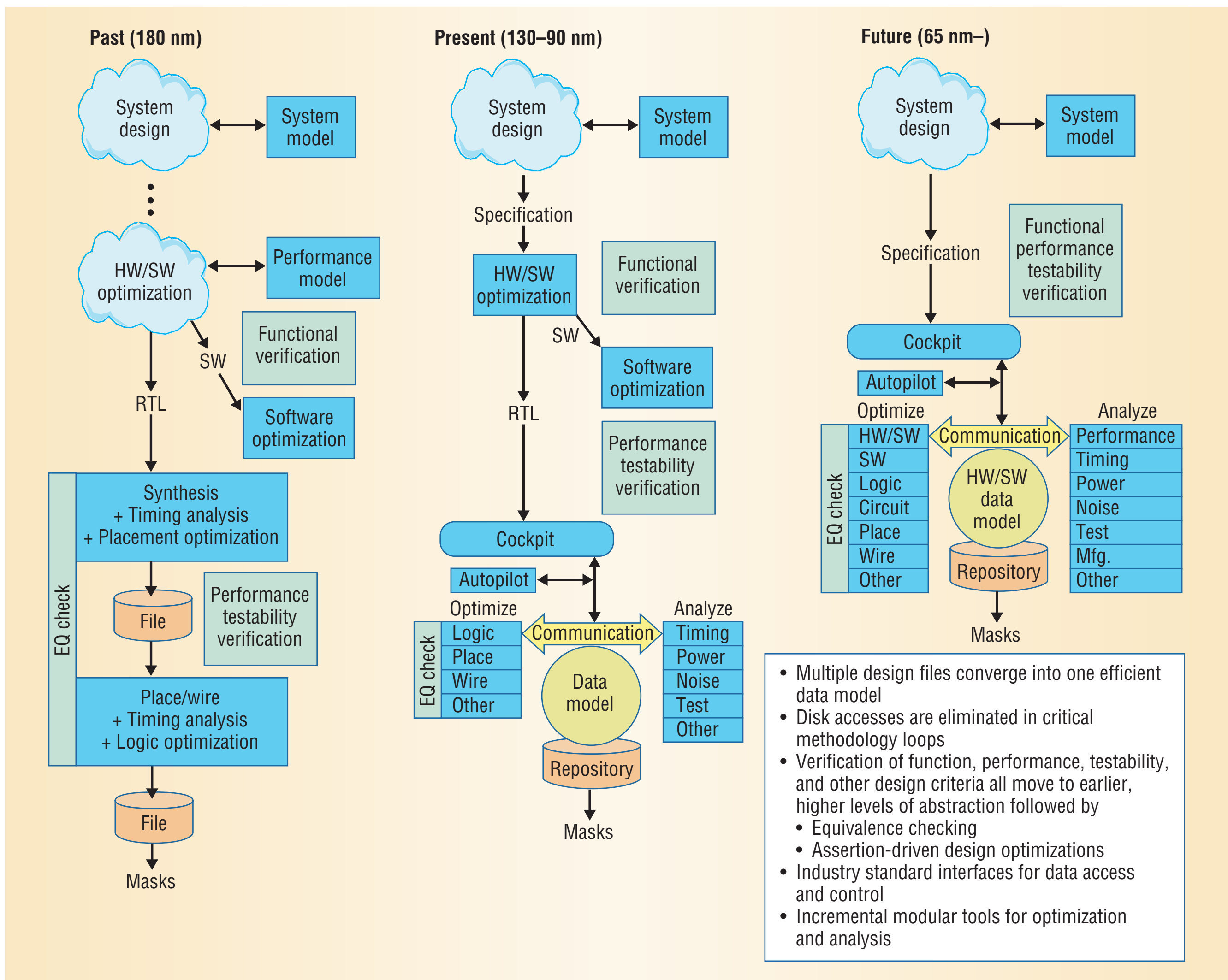


Figure 4. Required evolution of design system architecture.

approach. Low-cost-equipment solutions targeting DFT-enabled devices do not scale into the mixed-technology space today. The next logical step is to increase test system configurability and flexibility to achieve a more appropriate cost performance point—which is leading to a fundamental shift in test equipment architecture.

This revision of the Roadmap finds the test-equipment industry at the beginning of a significant shift from traditional test-architectures to *universal-slot* architectures with high levels of test-instrument encapsulation and modularity. The continued evolution of technology and increased integration level of design components has enabled this shift. In many cases, a single FPGA or ASIC can take the place of entire electronics subsystems.

Encapsulation and modularity have led to the open-architecture concept: the ability to mix and match test instruments from multiple suppliers into a single tester hardware and software environment. This concept presents significant business model challenges to the test equipment industry, which today is based on full vertical integration and proprietary platforms. However, an open-architecture approach offers several potential advantages. It would

- focus research and development investment, both dollars and effort, on the test instrument itself rather than test infrastructure;
- base differentiation on test capability rather than platform;
- focus supplier efforts on developing solutions within their particular core competency, reducing cost and speeding time to market;
- reduce the investment in reengineering infrastructural elements; and
- eliminate the need for each supplier to be everything to everyone.

Ultimately, the industry will drive this approach's success. No one can dispute, however, that the next several years represent a turning point for many test-solution suppliers. Regardless of the open-architecture approach's success, emerging platforms offer a new level of capability, flexibility, and longevity that will significantly impact the industry and test deployments. Quite possibly, the next

significant equipment selection decision could identify the test platform deployed for *all* products through the next decade.

While for many years the cost of the tester has overwhelmed all other parameters—with the single exception of throughput—significant progress has been made to address this aspect of the overall device-test cost equation. Focus is shifting to address the numerous secondary costs that now become large contributors. This analysis must encompass all aspects of test cost, including the design nonrecurrent engineering associated with design for test, handler or prober equipment, device interface hardware, and facilities cost, among many other factors.

SoC test

SoC test is dependent on a highly structured DFT methodology to enable observability and controllability of individual cores. Increasingly, SoC design will rely on a database of preexisting IP cores that encapsulate the design itself, interfaces to other blocks, and test.

A fundamental challenge of SoC test is the need to combine test requirements from multiple sources with differing testability approaches and methods. Opportunities exist to define standards for test to conform to a hierarchical methodology; these standards are most easily imposed on internally designed cores. However, when IP is purchased or licensed from a third party, it is typically the test methodology that must adapt. Many electronic design automation (EDA) tools already leverage a standard format for logic designs. This standard must be extended to other core types, such as analog circuits.

Structured use of IP core wrappers and test access mechanisms must be developed for testing of individual cores within a SoC. These should be developed carefully to enable functional, at-speed parametric and interconnect testing. Further, these methods should be standardized with the interface language for interoperability of EDA tools. One such effort is the Core Test Language development within the IEEE P1500 standard. The high complexity of SoC design creates design-and-test development productivity and test-quality challenges. EDA tools and data interchange standards must be developed to aid management of this complexity.

High-frequency I/O

High-speed serial interfaces have been used in the communications market segment for many years. While the communications market is expected to maintain a significant frequency lead, penetration of high-speed serial protocols into the microprocessor, ASIC, and SoC markets in the form of multilane buses has accelerated dramatically. This trend brings a complex test problem, previously limited to the high-speed networking environment, into the mainstream.

Key lessons from this market segment indicate the need to execute extensive testing such as jitter tolerance and jitter transfer on these interfaces. Such testing is done today in the analog domain through a rack-and-stack or mixed-signal tester approach. These solutions impose significant manufacturing cost considerations due to test time and equipment capital cost, and they support a relatively limited number of high-speed serial ports on a single device. As these interfaces proliferate to many ports on a single device, the traditional analog test approach will fail due to the scalability of analog instrumentation.

As the frequency of these interfaces and the number of interfaces on a single design continue to increase, alternative equipment solutions and test methods will need to be developed to enable cost-effective, high-coverage engineering and manufacturing test.

The high complexity of SoC design creates design-and-test development-productivity and test-quality challenges.

Reliability screens

The test process is responsible for the screening of manufacturing defects that affect device functionality, performance, and reliability to reduce customer perceived defects per million. A portion of the test flow is dedicated to the acceleration of latent defects that do not appear as test failures but would manifest as longer-term reliability failures. Traditional techniques for screening of reliability failures include I_{DDQ}, burn-in, and voltage stress.

The effectiveness of these methods is challenged by the continued device scaling with each successive process generation. Increases in device quiescent current in the off state is raising the level of background current to the milliampere and, in some cases, ampere range. These levels of background current increase the difficulty of identifying microampere to milliampere level I_{DDQ} fault currents. Continued extension of techniques such as delta-I_{DDQ} that have enabled extensions into current process generations is uncertain.

At the same time, the effectiveness of voltage and temperature acceleration methodologies used by burn-in and voltage stress is declining due to the reduced margin between operational and overstress conditions. Costs associated with burn-in techniques continue to rise and, in some cases, now

dominate manufacturing costs for high-power products.

The increasing cost and declining effectiveness of current techniques for latent defect acceleration combine to create one of the most critical challenges facing the industry for future process generations. Extensions to current techniques may prove adequate for the next several years, but fundamental research in the development of new methodologies is required.

Automated test program generation

Correct-by-construction test programs and patterns have long eluded test development teams. Increasing design complexity and demands on team productivity require significant improvement in test program generation automation to limit or reduce time-to-market impact. The EDA industry has developed and deployed many tools intended to aid the entire process of test development, content creation, and equipment translation. However, device makers seldom have a homogeneous environment provided by a single EDA supplier, and the general lack of interoperability standards among tools creates significant challenges requiring effort by the device maker to enable automation. Cooperation among EDA and test equipment suppliers is increasing, and a focus on tool interface and interoperability standards is growing. The challenges increase with every process generation and related growth in design integration.

Achieving full automation of test-program-generation will require increased standardization of the test-equipment software environment itself. Historically, each equipment supplier has taken a holistic and proprietary approach to definition and development of its specific software environment. Similar to EDA, most device makers do not have a homogeneous test environment of equipment provided by a single supplier. Increased standardization of the test-equipment software environment where appropriate would lower the entry barrier for suppliers as well as simplify the porting of test content between platforms as required by the device maker.

Today's environment of platform-unique supplier software solutions and homegrown tools for equipment programming, automation, and customization will drive unacceptable growth in test- development engineering and factory-integration efforts. Automation of common tasks and decreasing test-platform integration time demand a focus on standards to enable more efficient use of resources in line with shrinking product development life cycles. New tool development must comprehend the end use to ensure that the resulting effort is indeed reduced over existing methods and that pre- and postprocessing of data do not eliminate throughput gains.

Previous ITRS editions documented a design productivity gap, with the number of available transistors growing faster than the ability to meaningfully design them. Yet, investment in process technology has by far dominated investment in design technology.

The good news is that enabling progress in design technology continues. The bad news is that software can account for 80 percent of an embedded system's development cost, test cost has increased exponentially relative to manufacturing cost, and verification engineers outnumber design engineers on microprocessor project teams. Overcoming many existing design technology gaps will require a concerted effort by the entire industry. ■

Acknowledgments

We acknowledge the efforts of the many individuals who contributed to making the 2003 edition of *The International Technology Roadmap for Semiconductors* a successful endeavor.

Don Edenfeld *is responsible for test equipment definition and selection at Intel. He received an MS in electrical engineering from the University of Virginia. Edenfeld is a member of the IEEE Computer Society. Contact him at donald.e.edenfeld@intel.com.*

Andrew B. Kahng *is a professor of computer science engineering and electrical and computer engineering at the University of California, San Diego. He received a PhD in computer science from the University of California, San Diego. He is a member of the ACM, the IEEE, and SPIE. Contact him at abk@ucsd.edu.*

Mike Rodgers *is responsible for test technology integration at Intel. He received a BS in electrical engineering from the University of Illinois. Rodgers is a member of the IEEE Computer Society. Contact him at michael.j.rodgers@intel.com.*

Yervant Zorian *is the vice president and chief scientist of Virage Logic. He received a PhD in electrical engineering from McGill University. Zorian is a Fellow of the IEEE and a member of the IEEE Computer Society. Contact him at zorian@viragelogic.com.*