

DESIGNERS IN THE CONSUMER MARKET  
WORK UNDER ENORMOUS PRESSURES.  
HAVE THESE CIRCUMSTANCES CHANGED  
THE WAY THEY DESIGN CHIPS AND SYSTEMS?

BY RON WILSON • EXECUTIVE EDITOR

# CONSUMER ELECTRONICS: feeling the squeeze



**T**he demands on consumer-electronics design teams are unquestionably different from those placed on other parts of the electronics industry: Have the new product on retailers' shelves by Thanksgiving, or kiss the company good-bye. Meet the cost targets, or we will cancel the design. Re-spin the silicon? No, just pull the plug on the project. And by the way, the final version of the video-compression "standard" we chose won't be ready—so, make sure you leave us enough processing head room to adapt to it with nothing but software updates.

The pressures on chip designers and system designers in the consumer world have been well-documented. All indications are that with global competition, low entry barriers to start-ups, and emerging local wireless and media standards, the pressures are just getting worse. What has not been widely discussed is how design teams are responding to these pressures. Are chip and system design soldiering forward in their traditional paths and just taking the risks? Or are we seeing a new style of design forming in the crucible of the consumer market?

Two of the forces crushing down upon consumer designers—schedule and cost—are familiar features of the arena. But today's fad-cycle consumer market has transformed them. There are two kinds of schedule pressure now, according to Arun Mittal, senior vice president and general manager of the power-management and -supply business unit of Infineon. "In some cases, the vendor is already present in the market, and [it is] enhancing [its] offering. In this case, total time to retail presence is the issue. In other cases, the customer is entering a new market area and must hit a par-

ticular window—usually the Christmas gift-buying season."

The dynamics of the two pressures are different. If the design is an enhancement or cost reduction, a schedule miss results in lost market share or reduced margins for a period of time. In the new-entry case—for instance, the original Microsoft Xbox—missing the Christmas window may mean having to wait another full generation for viable market share. In the one case, managers can make a rational trade-off between, say, a silicon re-spin and lost revenue. In the other case, missing is simply not an option.

Cost is also a more complex issue than it might appear. Often, consumer system designs will start out with a BOM (bill-of-materials) cost target, and the team will create the system around the hardware it can obtain for that price. "We see customers who start out by costing-out their BOM based on their anticipated volumes and then go from there," says Upendra Patel, chief technology officer of design-services company eInfochips.

"Consumer customers will often start with a fixed BOM cost and invest what-

ever is necessary in software-development time to make that hardware meet their system requirements,” agrees John Dixon, DSP marketing manager at Texas Instruments. “Often, they are reliant on third-party design houses to actually do that work. So, in a way, the availability of design services becomes the real power of an architecture.”

But cost is a more elusive figure than it might seem. At the very low costs typical of consumer electronics, complexity and cost become unlinked: A substantial change in the complexity of an SOC (system on chip) may have no major impact, but the resulting change in the cost of passive components or test time

#### AT A GLANCE

▣ Schedule, cost, and energy constraints have become huge factors in consumer-electronics design.

▣ Use of platforms and reference designs has grown rapidly.

▣ Many teams shift functions to software to respond to the pressures.

▣ Verification remains a major issue—even more important than in some other markets.

▣ The growing pressure could change EDA flows and fabless semiconductor companies in the consumer area.

might be quite significant. The important thing, Infineon’s Mittal emphasizes, is to look at assembled and tested system cost, not just BOM cost or—worse yet—chip cost. For example, he says, often the only readily available savings in a system are in the cost of the power supply.

Besides the traditional pressures, new issues are weighing down consumer-electronics teams. The most obvious of these pressures is power, which itself has two meanings. In ac-powered systems, less power consumption results in a less expensive power supply, less heat to expensively dissipate, and possibly greater long-term reliability. In a handheld system, the issue is not so much power—al-

## USING A REFERENCE DESIGN TO SHORTEN PRODUCT-DEVELOPMENT-CYCLE TIME

By Gregory Eslinger, Acoustic Technologies

The growing demand for more complex products in less time increases the reliance of consumer-oriented design teams on highly integrated, high-quality reference designs. These reference designs can be ideal for wireless, multimedia, automotive, and telecommunication applications. Choosing the right reference design can increase the success of on-time delivery, lowering development cost and substantially reducing product-performance risk. More important, it can considerably reduce overall development-cycle time by as many as 40 weeks. However, finding a qualified reference design can be challenging, as the intensive process to fully develop a complete system requires a range of technical knowledge and skills from various third-party developers. When selecting the best reference design, it is important to

consider the following areas:

- **Hardware platform:** Does the hardware-design BOM (bill of materials) have the right price for your needs, the ability to quickly re-spin hardware for conformance to new plastics, and reasonable options for expandability?
- **Platform support code:** Does the platform have consistent, well-defined, and well-tested interfaces allowing third-party integration?
- **System-function software:** Does the system perform all required functions, including required third-party software? Does the ability to add additional software for future functions exist?
- **Flexible user interface:** Is the user interface flexible and simple enough to customize the human interface, allowing product differentiation?
- **Robust testing mechanism:** Does a testing mechanism exist, allowing for

quick performance analysis of the final product?

You can determine cycle time, cost, and risk reduction by addressing these five areas. As an example, consider Texas Instruments and Lyrtech’s Bluetooth Hands Free Kit Reference Design, which is a compilation of expertise from third parties, including Acoustic Technologies and Adamya. This reference design has a stable hardware design and reduces cycle time, because it requires only one board spin. With system-support code already in place, designers can focus on adding value to their custom interface rather than spending months developing, debugging, and integrating what should be standard code. The Bluetooth stack and profile code alone can take months, if not years, to develop and test. Having a reference design that has already been integrated and

tested allows designers to skip the time required for this necessary development and to focus on their user interface or other value-added features.

Once the product is complete, the ability to verify the final product performance immediately means faster time to production and ultimately allows the next-generation product to begin earlier. The right reference design that incorporates the hardware platform, system software, preintegrated and tested platform support code, and a flexible interface will significantly reduce risk, cost, and development time.

**AUTHOR’S BIOGRAPHY**  
*Gregory Eslinger is vice president of engineering at Acoustic Technologies Inc. He has more than 25 years of experience in embedded programming and managing embedded development for audio and video products.*

though high-current-transient requirements can drive up the cost of a supply—as total energy per task, which is directly related to battery life. That metric is entirely different and can have very different architectural implications. For example, as Oleg Logvinov, chief executive officer of Arkados, points out, for net energy, it may make sense to move func-

tions from efficient fixed hardware into power-inefficient software (Figure 1). Lacking the ability to completely power-down a block on the chip to halt leakage current, an infrequently used function may consume less total energy over a use profile if you do it on a less power-efficient CPU than if you do it on hardware that consumes less dynamic power but is

quiescently leaking 99% of the time.

Another pressure that has been rapidly building on design teams is the number of technologies that a single system requires. Newport Media Chief Executive Officer and Director Mohy Abdelgany observes, “The complexity has grown so high that handset-chip designers are no longer circuit designers now; they

## DRM LOOMS AS NIGHTMARE FOR DESIGNERS

As compact consumer devices pack in more and more functions, SOC (system-on-chip) designers are becoming used to the idea that they will be incorporating blocks into their chips that are quite unfamiliar to them. How, after all, is one design team supposed to be expert in cellular baseband processing, two kinds of CPU architectures, three memory architectures, four audio and two video codecs, video display, and 3-D-graphics processing?

But one design requirement on the near-term horizon looms like a gallow over chip designers and won't yield to any of the complexity-management schemes that have so far kept unfamiliar IP (intellectual-property) cores at bay: DRM (digital-rights management).

“We talk to customers who know they need a secure system to handle digital media. But they don't know what they mean by security,” says Josh Kablotsky, vision fellow at Analog Devices (ADI). This situation necessitates a two-way conversation between platform vendors and system designers about the levels of security, their costs, and their practicality. ADI segment business manager Scot Robertson adds,

“Customers start out wanting to protect data against every possible threat. Sometimes we have to explain the reality of the situation. This [area] is very unfamiliar for consumer-systems developers.”

Security, according to Kablotsky, comprises two separate functions: ciphers and hashes to protect the data when it is outside the system and system-level measures to secure the operations upon the data, so that would-be copyists can't break into the system and divert the data in an unprotected state. The latter may include hardware to detect intrusion exploits, such as physically probing the chip, manipulating the power supplies, or intercepting the memory transactions.

The most powerful weapon systems designers have to protect themselves against unfamiliar technology is encapsulation—licensing the IP in such a way that they need to understand only the interfaces, not the function. But encapsulation depends upon the existence of standards, or at least commonly accepted definitions, for the function being performed. And that consensus does not yet exist for DRM. There's not even agreement on basic concepts,

such as what ciphers are strong enough to work or whether, ultimately, hardware protection in the core of the system is necessary to prevent hacking.

And details within the DRM scheme can interact with system-level decisions to create vulnerabilities. “Chip architects have to understand this stuff in detail, even if they are following a reference design,” Robertson warns.

Another attempt to work around this problem is to import a security scheme wholesale from a different market. ARM CPU director of product marketing Kevin McDermott said recently that he had seen customers licensing the company's SC100 Smart Card core for use beside a Cortex processor in some cellular-handset applications. The SC100 would provide hardware-secured authentication to protect other functions in the handset.

The second line of defense for many design teams is to learn about the technology by working intimately with a partner, and then taking over the effort. But so far, only a few design shops claim specialized knowledge of DRM. And, in any case, this strategy may never work. DRM is not a static fix to a static problem but—like intrusion detec-

tion or spam filtering—an ongoing war of measures and countermeasures, requiring full-time warriors. “There's a growing consensus that both Windows Media DRM-10 and Fairplay have already been cracked and will have to change,” Robertson states. Nor is there any likelihood that a new DRM scheme will fare better. So once DRM enters the system, it will continually evolve. “This is a requirement for training, not for fixed solutions,” Robertson says.

Altogether, the DRM scenario is near worst-case for systems developers. It is an ill-defined requirement; content providers won't even state whether they accept your level of protection. DRM cannot simply be licensed and integrated, and it requires frequent updates, maybe including revision to basic algorithms. Yet many perceive software—the typical way of permitting field upgrades—as much more vulnerable than hardware, so even a software-based approach may have to rest on a secure hardware core of some sort. And, in the end, the liability for any content that appears on the Internet could rest with the systems developer. There's a little added pressure for you.

are systems integrators. For instance, we are working now to put digital TV into handsets. The handset people don't know broadcast-television technology. They are dependent on their module vendors, and that is very uncomfortable for many of them."

Whether the module in question is for speech recognition, audio playback, 3-D graphics, digital TV, or GPS location, handset vendors are having to include functions whose details they don't understand. And handsets are just a leading indicator, not by any means an unusual situation in the consumer market. The trend is toward sweeping groups of functions you might desire in the same context—whether they are otherwise related or not—into a single package, and ultimately into a single SOC. Understandably, this situation stresses IP (intellectual-property) quality, reuse methodology, verification plans, and the nerves of design managers.

### ARCHITECTURE RESPONDS

One of the primary ways consumer-electronics designers are responding to these pressures is with new architectural strategies (Figure 2). The simplest of these approaches involves letting someone else do the system integration—to license a complete reference design (see sidebar "Using a reference design to shorten product-development-cycle time). Although this technique is suspect to many US and European design teams, it is finding increasing favor in Asia.

Greg McNeil, product-marketing director for design kits at Cadence, relates, "Part of our strategy for having application-specific design kits is to include in the kit what we call a segment-representative design. It's basically a paper reference design that illustrates how the IP and tools would be used. When we show [it] to our North American customers, their response is often, 'Uh ... Are you going to share that with the Asian companies?' The response of our Asian customers is more like, 'Great! Can I just start out with this [design]?'"

"We are hearing more and more about IP platforms," comments Terry Danzer, tactical marketing manager at ASIC vendor AMI Semiconductor. AMI has always seen customers willing to bring in IP that is necessary but not differentiating. But now, they are increasingly will-

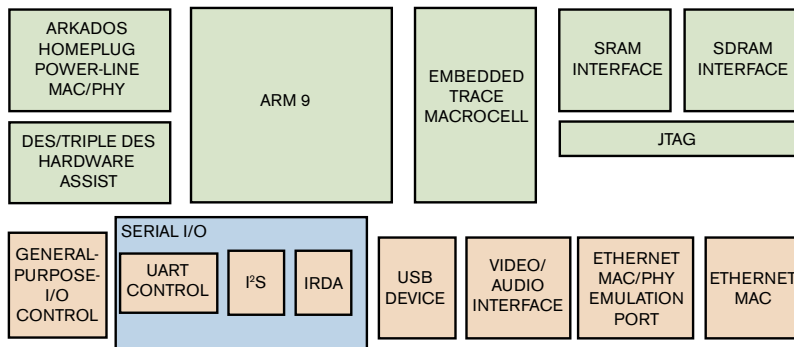


Figure 1 An early market entrant—in this case, the Arkados power-line-networking SOC—may choose to keep nearly all functions except interfaces safely in software.

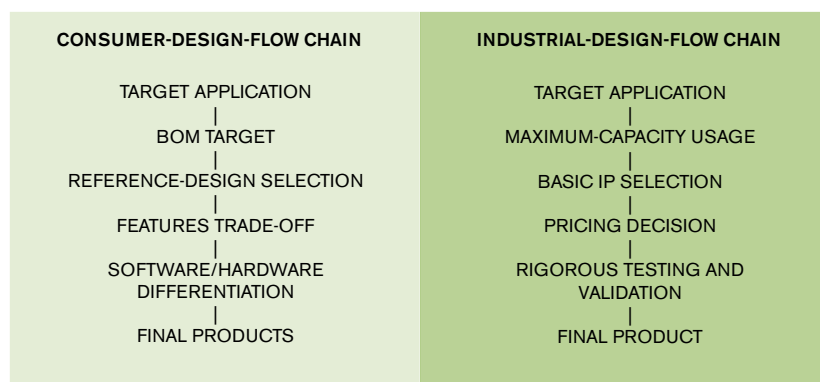


Figure 2 Design flows for consumer and industrial projects are diverging.

ing to bring in anything that is outside their specific area of expertise. And that may mean asking for a set of preintegrated IP that forms a complete platform under their specific hardware block.

But a platform ASSP may be too large, expensive, and power-hungry for long-term viability in a consumer application. It's only virtue may be time to market. To counter this effect, many design teams are adopting a hybrid strategy.

"We are seeing what I'd call a kind of platform approach," says Newport's Abdelgany. "A company may start out with a relatively full-featured chip and do a series of quick spins to make minor modifications as [it learns] the specific needs of the market."

Vendors as large as Texas Instruments have used this strategy. For a recent customer entry into the portable-video-player market, for instance, TI scaled back its flagship Davinci SOC, editing the features, adding dynamic voltage scaling, and increasing the use of clock gating to produce a software-compatible device that consumes much less

power. In this case, the effort was not small; it required a number of design teams around TI, all working to reduce the power of individual modules. But it was a much shorter effort than designing a whole new SOC.

In other cases, companies may simply remove blocks from their design, often coupling this effort with migration to a more advanced process to further improve die cost. "We are seeing a lot of derivative designs," says Cadence Vice President of Engineering Services Tim Henricks. "They are not trivial, but they are [much] cheaper than starting from scratch."

Henricks says that many customers initiate a chip design with a complete cost-reduction road map that will carry them through several generations of the chip, to the product's end of life. "This [approach] makes the availability of IP on future processes a real issue," he adds. "Especially with mixed-signal stuff, the IP vendors are reluctant to port their designs to a new process—like 65 nm today—until there's enough demand. So you have to be pretty big to be sure that

you can move to an advanced process and have all your IP come with you.”

### SHIFTING TO SOFTWARE

Another key part of a platform strategy is to isolate blocks that are subject to change—either to serve multiple products with a single design, or to reduce risk from uncertain design requirements. This strategy requires encapsulating the block in a standard interface so that internal changes cannot impact the rest of the design. You can perform this technique by simply keeping the block in question in a separate chip, by carefully planning an SOC so that you can synthesize the new RTL into a particular block without major disruption, or by moving the function to programmable hardware or to software.

Conventional wisdom says that if you may have to change something in the future, you put it in software, or at least in an FPGA. And, in fact, the people we talked with for this article said that they often see consumer designs that implement functions in software—even when that approach is less than optimal from a power or performance standpoint—just to ensure time to market and the ability to go back and fix things later. “We see a lot more designs using more programmability, so that they can make adjustments after tape-out,” says Paul Russert, Cadence’s engineering-

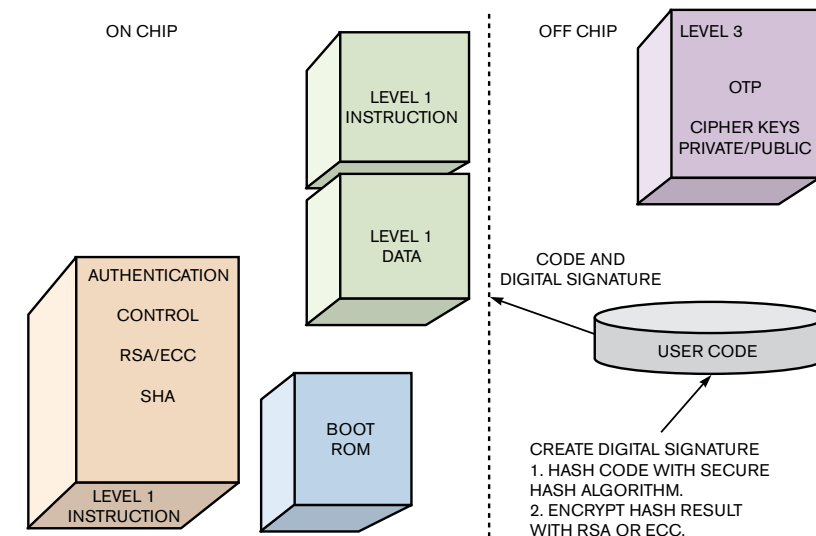


Figure 3 Functions such as digital-rights management may require complex blends of software and secure hardware.

services group director. That measure may be absolutely necessary, because the verification effort for some blocks may extend past tape-out, as well.

As to how you should implement the programmability, violent differences of opinion exist. Some architects just keep things in software if at all possible (Figure 3). The appearance of instruction accelerators, DSP blocks, and more complex instructions on embedded CPU cores has greatly advanced that option. Some architects opt for dedicated but still programmable hardware. Tony King-Smith,

vice president of marketing at IP vendor Imagination Technologies, says that in critical situations, Imagination’s designers use scalable pipeline building blocks to create datapaths. The blocks are themselves configurable, so the resulting pipeline is programmable at the microcode level. Imagination then exposes this flexibility to its customers as various levels of abstraction, from driver code all the way to microcode. The result is a programmable element with great throughput and good energy efficiency, with several flexibility-versus-programming-ease points.

Newport Media’s Abdelgany takes a different approach. “Our philosophy has been to enter a market with the lowest power and die size,” he says. “For flexibility, we rely on the ability of our design teams to incrementally redesign the chip. It’s not an easy game to play. You still end up with a fair amount of programmability, but where the customer needs it, not where it’s convenient for the chip developer.”

### CONCURRENT DESIGN

No matter what the architectural choices, unless a design team chooses to use an unmodified reference design, there will have to be an implementation cycle. Here also, consumer design is responding to pressure, mainly by employing various strategies for concurrent design.

The most obvious—and probably most frequent—strategies in this area are to make the hardware and software efforts concurrent and to modularize the hardware design, so that designers can

TABLE 1 OVERALL DESIGN CHAINS FOR CONSUMER AND INDUSTRIAL MARKET		
Parameter	Consumer market	Industrial market
Volume	Very large	Medium
Customer’s cost sensitivity	Very high	Low
Form-factor pressure	Very high	High
Royalty	Major deciding factor in price	Not a major issue
Maximum design efforts	Feature addition	Validation and hardware/software-standard compliance
Feature trade-off	Possible through phase- or model-wise approach	Not possible
Risk-taking ability of customer	Very high	Very low
Subsequent design changes	Very high	Low
Head room available for further customization	Large	Small
Selection of hardware components	Address current needs, work with recently introduced chips	Long-term availability and stability of the device in market
Time-to-market pressure	Very high	Low to medium
Design-team mindset	Time-to-market- and BOM-oriented	Compliance-oriented
Key differentiator	Supports software features and interfaces	Robust design and stringent standards compliance

concurrently design the modules. None of these ideas is even slightly novel, but design teams are applying them on a grand scale. And today, a grand scale often means implementation across global boundaries. A design-services team in California might design a chip for a shirt-pocket audio player using some IP blocks from the United Kingdom. This chip might drop into a board that a design team in India created, that runs application code that Romanians developed, on top of drivers that the Indian team wrote on a third-party kernel.

The key to this style of design is clear definition of the interfaces and specifications for each module. When you cannot encapsulate a module this way, the only viable alternative appears to be putting application engineers on site. In effect, management is matching the complexity of the interface between the blocks with the bandwidth of the interface between the design teams. It makes an intuitive sort of sense.

“You need reasonable project controls,” explains Abdelgany. “But in the end, it may come down to our engineers working at the customer’s facility. Sometimes we have won a design based on our ability to provide this level of application support.”

In a time when everyone seems concerned about design outsourcing, this kind of recent experience has made Abdelgany bullish on the United States. “We are very fortunate to have a diverse work force,” he says. “Our engineering staff includes people of Chinese, Korean, and European heritage. Almost anywhere in the world we can put an applications team on site and have someone who can serve as technical translator. But we have found that engineering is nearly an international language in its own right. Often the translation is needed mainly for interface to the marketing department.”

## VERIFICATION

Many consumer-electronics designers point to verification as the most serious issue that they face. On one hand, verification often consumes the largest share of development time. Design managers cite numbers greater than two-thirds the development time. So verification is an obvious place to respond to schedule pressure. On the other hand, in many consumer designs, a re-spin would be catastrophic. So veri-

**MORE AT EDN.COM**



[Go to www.edn.com/061109df1](http://www.edn.com/061109df1)  
and click on Feedback Loop to post a  
comment on this article.

fication must be even more thorough than it need be in markets that can meet initial orders with an FPGA-based option or can simply delay introduction by a few months.

The first response of consumer-electronics teams to this situation is planning. “We see two kinds of teams,” offers Cadence’s Henricks. “Those who come in with a verification plan, and those who are still verifying after tape-out.” That planning involves strategies for each of the blocks that will go into the system as well as a postintegration verification strategy for the system. And, increasingly, it will rely on some sort of coverage metrics. “Knowing when you are done is the biggest issue in verification,” says Cadence’s McNeil. “You may be talking about verification occupying 75% or more of the total design time, so shaving off a little bit is a big deal.”

But knowing when to stop is also an unsolved problem. Coverage metrics can help, but none is definitive. A rigorous reuse methodology can help if it allows the team to reuse testbenches for individual blocks or even to adapt a system-level testbench from a previous design.

Once again, concurrency can be a friend. Starting system-level verification at the behavioral level and working down as block-level RTL and netlist verification work up can eliminate much of the calendar time. It can also help if the really hard tests can focus on areas that are problematic or have poor coverage. In some cases, if the verification plan requires designers to use assertions properly, formal techniques can augment dynamic testing with huge savings. “This [method] has to be design-savvy,” says Henricks. “You have to know where the critical portions of the design are.” And that information in itself can be a problem when the bulk of the design is a platform you license from another source.

An additional problem that distinguishes consumer-product verification is that the ultimate outputs of the sys-

## FOR MORE INFORMATION

### Acoustic

**Technologies**  
www.acoustictech.com

**Adamy Computing Technologies**  
www.adamy.com

**AMI Semiconductor**  
www.amis.com

**Analog Devices**  
www.analog.com

**Arkados**  
www.arkados.com

**ARM**  
www.arm.com

**Cadence Design Systems**  
www.cadence.com

### eInfochips

www.einfochips.com

**Imagination Technologies**  
www.imaginationtechnologies.com

**Infineon**  
www.infineon.com

**Lyrtech**  
www.lyrtech.com

**Microsoft**  
www.microsoft.com

**Newport Media**  
www.newportmediainc.com

**Texas Instruments**  
www.ti.com

tem are often visual or auditory. There isn't an absolutely correct output—only wrong outputs and outputs that lie somewhere along a long scale of relative quality. Subjective judgments, often by golden eyes or ears, may be critical inputs into the verification process, and, without a parallel FPGA-based verification effort, these judgments may come very late in the schedule.

Unfortunately, despite all the planning and all the tools, the design manager's nightmare is all too common. "We see a fair amount of engineering-change orders coming toward the end of the design and coming from the verification process," Henricks says. To counter this situation, often consumer-electronics designers move critical functions into software, planning to use multiple software releases to fix late bugs and then reduce costs by converting the stable software modules to hardware engines. But even this strategy depends upon an art: being able to predict early in a design which functions will be the most problematic and will require post-tape-out flexibility. That ability can come only with experience.

### A PLATFORM FUTURE?

Clearly, there are many approaches to consumer-electronics design today. But all these trajectories may converge on a single trend line (Table 1). The use of platforms defines that line—whether hardware platforms in the shape of ASSP chips or IP platforms in the form of preintegrated blocks and flows. This way of thinking about consumer products has already become so pervasive that most IP vendors and many CPU and DSP vendors talk in terms of ecosystems—surrounding their products

with all the other IP, hardware, software, services, and training necessary to quickly reach the final product.

These ideas about design will necessarily impact the industry in a number of ways. First, the increasing criticalness of verification is likely to force IP vendors out of their reliance on simple testbenches and into assertion-based-verification techniques. Even standards bodies, used to producing their result as an unreadable and ambiguous 1000-page document, are facing pressure to provide executable specifications in a property language.

Second, there will be growing pressure on EDA vendors to provide flows optimized not for completely new designs, but for the incremental design changes and platform tweaks that are the reality in the consumer world. There is no reason for a design team to allow all the degrees of freedom of a complete synthesis, place, and route cycle just to modify two blocks. Similarly, verification must become incremental. It must be possible for a static tool to prove the assertion that one modification has changed nothing else in the design.

Finally, the pressures of the consumer-electronics market may change the definition of a fabless semiconductor company. Suggests Imagination's King-Smith, "Increasingly, fabless companies are about marketing and about foundry relationships. The selection of IP, integration, and verification is increasingly taking place outside, in the ecosystems that are developing around particular applications areas." In this new future, the systems integrators may be the most powerful IP vendors, or the possessors of key proprietary knowledge about a new standard—a codec, say, or a digital-rights-management scheme (see sidebar "DRM looms as nightmare for designers"). Such a move would profoundly change the way the engineering-design chain works in this evolving market. **EDN**

You can reach  
Executive Editor  
**Ron Wilson** at  
1-408-345-4427 and  
ronald.wilson@  
reedbusiness.com.

