Embedded Systems Outlook 2012

Key Technologies and Methodologies Impacting the Embedded Systems Market

technologies & architectures ✧ business strategies & processes

ni.com/embeddedsystems

NATIONAL INSTRUMENTS™

Supporting Accelerated Innovation in Embedded System Design

National Instruments provides graphical system tools for engineers and scientists developing next-generation control and monitoring systems across a broad scope of industries, including energy, industrial control, life sciences, and transportation. With NI reconfigurable I/O (RIO) hardware systems featuring embedded field-programmable gate array (FPGA) technology and NI LabVIEW system design software, smaller design teams can prototype and deploy embedded systems that require advanced control and signal processing faster.

2012 Embedded Systems Outlook Trends

More than 30,000 companies around the world use National Instruments tools. Additionally, NI collaborates with leading technology providers such as Intel, Xilinx, and Analog Devices to ensure that NI embedded systems take advantage of the latest and greatest technologies. Using this ecosystem of customers and partners, NI created this outlook to highlight some of the most pressing trends and challenges facing design teams building embedded control and monitoring systems.

technologies & architectures

Embedded Platforms

Integrated platforms combine hardware components and a software framework for embedded system development.

Reconfigurable Computing

An increasing number of advanced control and monitoring designs use programmable logic.

Mobile Devices and the Cloud

Design teams can take advantage of the proliferation of mobile and cloud technology within embedded systems.

business strategies & processes

Innovating With Smaller Teams

Smaller design teams create a more efficient way for companies to innovate.

Future Proofing Through Software

Ever-changing system requirements are demanding new approaches to upgrade systems over time.

National Instruments is proud to be supporting engineers who are designing incredible embedded applications.

Our customers are working to solve the most difficult engineering problems facing society, such as the 14 engineering grand challenges identified by the National Academy of Engineering. These challenges include such monumental tasks as reverse-engineering the human brain, acquiring energy from fusion, engineering better medicines, restoring urban infrastructure, and providing worldwide access to clean water.

The vision of National Instruments is to provide a platform-based approach for engineers, scientists, and design teams to accelerate innovation and discovery within applications that require measurement and control.

We believe that a tightly integrated software and hardware platform provides the most productive way for engineers to quickly implement and realize new ideas.

An analogy I use to portray our vision in the embedded market is "to do for embedded what the PC did for the desktop." The PC created a standard hardware architecture and operating system that allowed consumers to use desktop computers. The standards that the PC created proliferated an ecosystem of hardware and software to let users quickly augment the PC to solve a variety of problems. Today, the embedded market contains fragmented and complex design toolchains that make it difficult for the

“Do for embedded what the PC did for the desktop.”

engineering community to build embedded systems that combine advanced measurements and control. We want to help solve that challenge and provide a standard software and hardware platform, along with an extensive ecosystem, to empower any small design team to quickly experiment and solve new problems more efficiently.

The work of our customers has gone a long way to prove that a platform-based approach is highly valuable and productive compared to traditional design toolchains. Using graphical system design tools, customers in fields like renewable energy, life sciences, and robotics have solved some extremely difficult embedded control and monitoring problems. From designing 3D optical coherence tomography (OCT) medical imaging devices to bringing remote villages power through innovative renewable technologies, our customers are proving that with the right tools, small design teams can have the confidence to tackle more complex control and signal processing tasks, and innovate faster.

I’m excited about what the future holds for NI in embedded applications. Whether it’s working with companies like LEGO® and universities to attract the best minds of the next generation to engineering or helping physicists solve cold fusion, NI wants to support engineers in meeting the toughest engineering challenges.

Best regards,

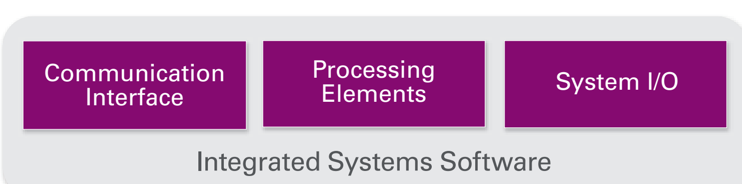
Dr. James Truchard
CEO and cofounder, National Instruments

Embedded Platforms

Technology providers are developing embedded platforms that combine hardware components and an integrated software framework to help design teams build complex embedded systems faster.

Take a moment and think back on your day. How many embedded systems have you touched or used? Consider appliances, clocks, televisions, tablets, phones, street lights, elevators, wireless keyboards, mp3 players, or exercise equipment. Intelligence is being added to more systems every day. Smart power metering, home automation, and in-home medical devices are a few examples that will be ubiquitous in just a few years. Not only are there more embedded systems around you, they are becoming more complex. The mobile phone of 10 years ago was a device that had a single processor and cellular radio that was used to make calls away from the home or office. This device has evolved into a smartphone with cellular, Bluetooth, and Wi-Fi radios and multiple processors to run applications for emails, calendars, videos, music, games, photos, and phone calls. Automobiles have also evolved into complex systems with as many as 100 processors to control the engine, brakes, traction, trip computer, seat and mirror memory, music players, and navigations systems. The same trend in the consumer space can be seen in industry as well. Electronic systems and machines are adding control and monitoring systems that improve performance, quality, and differentiation.

An ever-increasing number of design starts and escalating complexity are forcing embedded design teams to be more efficient and influencing the technology they choose to use. To address the embedded design market needs and help teams get to market faster, technology providers are developing components, modules, or even complete embedded platforms with higher levels of integration and increased functionality. Ultimately, they are working toward a complete platform for embedded design containing communications, processing, system I/O, and integrated system design software.



A complete embedded design platform—containing a communication interface, processing elements, system specific I/O, and integrated system design software—is well integrated with all hardware elements and highly flexible for application development, analysis, control, and communications.

Technology Examples

This trend began with technology providers offering more systems-on-chip (SOCs) and systems-on-module (SOMs) that target specific embedded needs and applications. SOCs and SOMs include all the electronic circuits required for a system in a single package or module. SOCs and SOMs often include the three primary elements of an embedded system: a communications interface, processing, and system specific I/O. Common examples include video digital signal processing (DSP), audio DSP, radio solutions, networking solutions, or a complete computing platform in a single chip or module. Computers-on-module (COMs) make up a special subcategory of SOMs. By integrating an entire computer or embedded subsystem into a single device, companies are providing more value to embedded designers through increased functionality, better integration, a more thoroughly tested design, a smaller package, and lower power consumption. SOCs and SOMs are commonly offered as a standard part and can be either general purpose or target vertical applications. They tend to be sold at high volumes to a wide range of design teams, driving down costs and ensuring higher quality.

SOCs and SOMs offer the clear and proven benefits of reduced cost, power, and size, while providing higher quality and capability. Today, many design teams use SOCs and SOMs. In some application areas, nearly all teams are using the same SOC or SOM. With many teams using the same SOC or SOM, it is difficult to maintain differentiation with the final design. To maintain differentiation, most design teams augment the SOC or SOM with additional discrete components and programmable logic. With the addition of programmable logic, such as a field-programmable gate array (FPGA), to the design, teams can add specialized processing, improve performance, and future proof the design with the ability to update the logic at any time during development or even once they have deployed the embedded system. The addition of FPGAs to embedded systems has become so common that new SOCs are being released that contain both a complete microprocessor and an FPGA in a single package. The most interesting part being released in 2012 is the Xilinx Zynq-7000 Extensible Processing Platform (EPP). The Xilinx Zynq integrates a dual-core ARM Cortex-A9 processor with Xilinx 7 Series programmable FPGA fabric across an industry-standard AXI interface. These parts provide an extremely high level of performance with increased flexibility to give embedded design teams the ability to maintain differentiation within their design while benefiting from the reduced cost, size, and power of an SOC.

Embedded Platform Challenges

Although the advancements in SOCs and SOMs are exciting, most fall short of offering a complete embedded platform. In the upcoming decade, software tools will play a more critical role in system design and development. In the past, many embedded designs were dictated by embedded hardware capabilities and mapping them to the system requirements. Due to the reduction in power, cost, and size in embedded hardware over the last decade, hardware will no longer limit or dictate many embedded design choices. Productivity will. Embedded design productivity will be driven by tightly integrated software design tools that can use off-the-shelf hardware capabilities with an environment intuitive enough to be used by nearly all engineers and scientists, not only those trained in embedded software, firmware development or hardware description languages.

The Apple Genius

To understand the impact that better software development tools can have on embedded design, draw from the example of the smartphone. Although smartphones have been on the market for well over a decade, Apple's introduction of the iPhone in 2007 radically evolved the smartphone and personal entertainment technology. Within a few short years, Apple has developed a dominate market position. The iPhone hardware met the needs of most smartphone customers. However, it is clear that the iPhone was truly differentiated by the overall software experience —not only the iOS on the phone, but also the surrounding ecosystem of software including iTunes and the now hundreds of thousands of apps available in the Apple App Store. Integrated embedded design platforms should reach for the same level of integration, quality, and extensibility that Apple has achieved with the iPhone ecosystem.

A completely integrated embedded platform must include a single software development environment that programs the heterogeneous processing systems, includes a large library of analysis and control algorithms, and has tight integration with communication and application-specific I/O, while giving the design team the ability to choose from a variety of programming approaches based on the specific application needs. The embedded platform should also be flexible and modular enough to give design teams the ability to evolve the system during the entire design flow, from first prototype to final deployment, while using the same code throughout. Although the goals of an embedded design platform differ from that of the iPhone, the spirit is the same: improve the overall design experience with a tightly integrated hardware/software approach to embedded design.

Reconfigurable Computing

Advanced embedded control and monitoring systems have driven an increase in designs that use programmable logic.

In the past, the decision to use a low-cost microcontroller or a higher performance CPU was fairly straightforward and based on the expected performance needs of the embedded system. As control and monitoring systems evolve to become more feature rich, additional embedded processing and programmable logic is becoming commonplace for advanced capabilities such as the following:

- Faster and more reliable responses to I/O
- Machine monitoring to predict failures and improve safety
- Audio and image processing
- Wireless communications and Internet connectivity
- Filtering of analog and digital signals to get more accurate measurements
- Digital communications to intelligent sensors and other subsystems
- I/O level preprocessing for data reduction

Technology	**Benefits**	**Considerations**
Microcontrollers	Low cost, small form factors, easy to program	Not enough horsepower for high-performance applications
Microprocessors	High clock rates for higher performance applications, easy to program	Higher power, sequential processing architecture
DSPs	Dedicated components for signal processing, floating-point arithmetic	Inherently sequential processing
GPUs	Parallel processing engines for CPU acceleration	Higher power, requires CPU to be present
FPGAs	Flexible hardware through software-defined, reprogrammable circuitry—inherently parallel for processing	Complexity of programming in hardware description languages
ASSPs	Fast and optimized for specific applications, offered standard, commercially available chips	No flexibility to modify designs
ASICs	Completely custom chips, optimized down to a single package for a single application	High initial investments, and only feasible in high volumes

The vast majority of embedded designs begin with a processor-based component, using a microcontroller or microprocessor as command central to schedule and process basic control and monitoring tasks, communicate with user interfaces, and supervise all other parts of the design. For traditional embedded systems, this architecture provided enough processing power to service all control loops and update log files. For more complex systems that integrate advanced control and signal processing tasks, teams are forced to use additional processing components like field-programmable gate arrays (FPGAs), digital signal processors (DSPs), and graphics processing units (GPUs) to achieve higher speed data processing and more deterministic control.

With all the performance and flexibility that FPGAs offer, they are nowhere near replacing the need for microcontrollers and microprocessors in embedded designs. Comparatively, processors are still lower cost and come with a well-established ecosystem of software abstraction, including OSs, standard hardware drivers, and libraries for signal processing with easy floating-point arithmetic. The adoption of FPGA technology has been the result of higher performance systems that combine both processors and FPGA fabric to divide and conquer complex processing needs through both sequential and parallel architectures. Integrating reprogrammable hardware into designs is the fastest way to iterate without having to spend time and cost on redesigning PCBs.

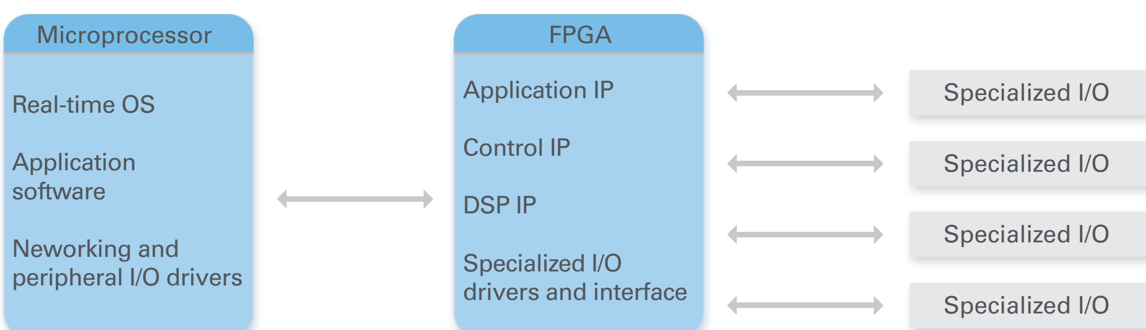
Programmable Logic

FPGAs have been used for over 30 years as digital glue logic between different components on the same printed circuit board (PCB). The reconfigurable logic within the FPGA fabric has been ideal for implementing complex state machines and application-specific digital circuitry that operate independently from processor clock cycles, with higher reliability and determinism. Over the years, the performance of FPGAs has increased dramatically, with significant reductions in power and cost. For this reason, the use of FPGAs in embedded measurement and control designs has expanded from simple glue logic to handling signal processing tasks, such as custom digital filters, fast Fourier transforms (FFTs), and logic for proportional integral derivative (PID) control. A primary benefit of FPGAs for processing is that several algorithms can now run in parallel, unlike the sequential architecture of a processor.

Industry Examples

The combination of processors and FPGAs in embedded system designs is prevalent in many industries. One example is ePower Technology, an engineering company in Denmark, which designed an embedded control system for muscle testing and training equipment. The SYGNUM energy training system used a patented five-phase motor to offer smoother control when applying force resistance during exercise motions. They used FPGAs to handle the high-speed control loops that maintained velocity and position setpoints, and then used a processor with a real-time OS for other lower frequency control loops.

From the energy industry, Xtreme Power is a US company that designed a distributed energy storage system made up of several processor and FPGA-based embedded control and monitoring nodes. They use FPGA technology to take



This standard reconfigurable computing architecture is used in advanced embedded control and monitoring systems.

accurate, high-speed measurements on three-phase power grids and run advanced algorithms to determine how to best respond to instabilities on the power grid. The processor portion of their systems provides the Ethernet communication link to various other distributed nodes, so they can remotely access data for system management and diagnostics.

Lastly, the French company O'Mos designed an embedded system to monitor crushing machines installed in quarries. They could continuously watch and process sensor values to predict machine failure and detect misuse, thereby reducing machine downtime and maintenance costs. They used FPGA-based I/O to do inline processing of various physical parameters, including pressure, temperature, vibration, and flow. They dedicated the processor part of the system to log data files and communicate with the local human machine interface.

Early adopters of reconfigurable computing designs have taken advantage of parallel processing to meet performance needs without a large impact to system costs or development time. Due to the reconfigurable nature of FPGA circuitry, there are many examples of embedded designs getting deployed sooner by skipping the long lead times associated with making revisions to PCBs. The end result is a higher performance control and monitoring system that is developed in less time and with higher confidence that it can serve demanding applications precisely and reliably.

A New Era of Embedded Systems Innovation

Embedded microprocessors aren't new in the FPGA world. In fact, FPGA vendors have been offering various types of embedded processors for their FPGAs ever since FPGA transistor counts grew big enough to accommodate them. In the late 1990s, FPGA vendors started offering soft cores (8-bit, 16-bit, and then 32-bit processor cores in Verilog or VHDL, or as prerouted netlists) that hardware designers could program into FPGAs with synthesis and place-and-route tools. Then, in the early 2000s, Moore's law made enough transistor real estate available to allow FPGA vendors to implement microprocessors in the silicon itself, next to programmable-logic blocks. Implementing cores in the fabric itself saves space on the chip for programmable logic, speeds processing and overall performance of the chip, and lowers power. Xilinx used this method when it implemented PowerPC processors in derivatives of its Virtex-4 and Virtex-5 devices. Those FPGAs continue to be successful in some markets, but the company felt the need to take the concept to the next level by creating a fully encapsulated processing system that appeals to a wider audience than just hardware designers.

In March of 2011, Xilinx officially announced the first four devices of its new 28nm Zynq-7000 Extensible Processing Platform (EPP) family. Each device merges an ARM dual-core Cortex-A9 MPCore processing system with Level 1 and Level 2 caches, memory controllers, large programmable-logic blocks, and a slew of commonly used peripherals. EPP is the product of many years of research and lessons learned from previous efforts to merge processors and programmable logic on the same device. The result is that Xilinx has created an entirely new class of device. What differentiates an EPP from a traditional FPGA that has a processor on it, is that the processor runs the show. Therefore, software designers can start developing with the EPP without being versed in hardware description languages such as Verilog or VHDL.

Programming the programmable logic portion of an EPP requires at first some degree of hardware design knowledge or assistance from a hardware engineer. However, Xilinx and valued customers and partners such as National Instruments are working diligently to develop new tools that further simplify the programming of the hardware portion of EPPs and open even more possibilities for embedded innovation.

Larry Getman
Vice President,
Processing Platforms,
Xilinx, Inc.

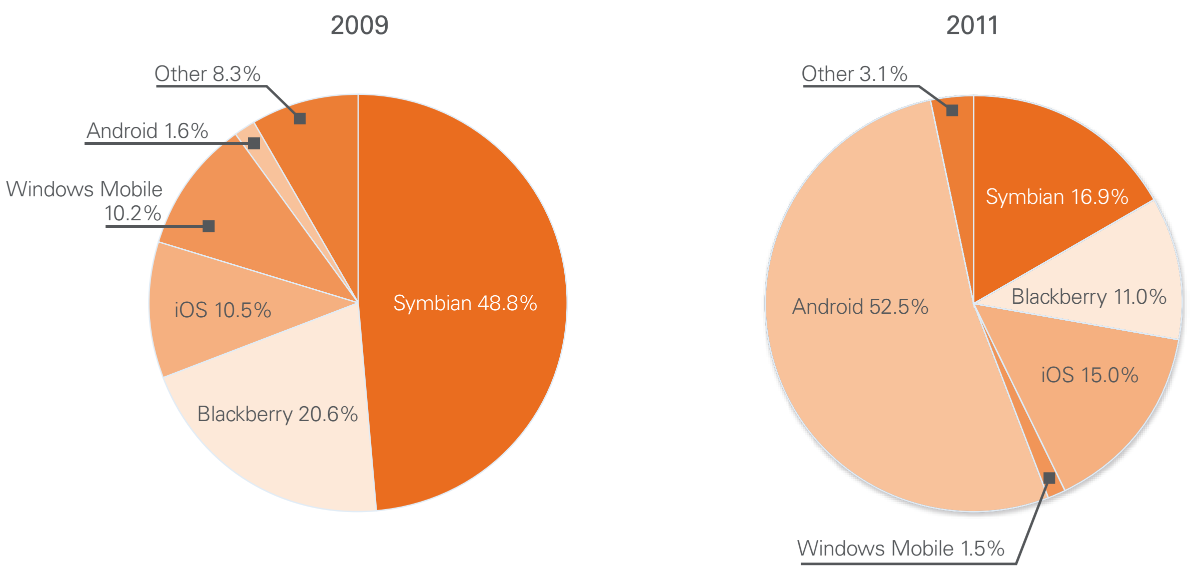
Mobile Devices and the Cloud

Design teams can take advantage of new mobile devices and cloud technologies within next-generation embedded systems.

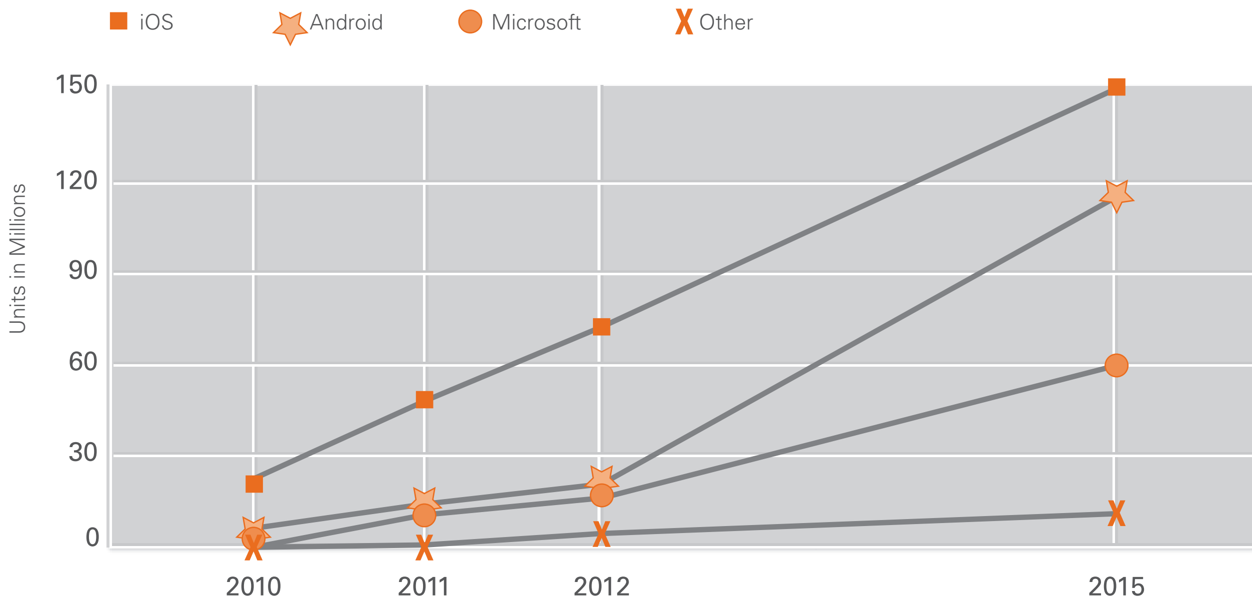
Mobile computing is not a new concept. From early handheld calculators to the Osborne 1 portable computer, the promise of portable data, computation, and communication has been slowly moving from concept to reality for nearly 40 years.

In the past four years, the right combination of technology and innovation has transformed the smartphone from a business tool to a highly functional and accessible mobile computer. Furthermore, the ecosystem of mobile software (both Web and native) has allowed for more capable extensions and raised the general expectations for phones. The relatively recent rise of tablets has further fragmented an already complex technology landscape. With no clear winner yet emerging and more disruptions on the horizon, the mobile computing field is far from settling on a "Wintel-like" common architecture.

Along with the challenges come opportunities of new capabilities that engineers building embedded control and monitoring systems can provide. Without any additional hardware cost, you can provide users with a specialized and rich user interface that they can access from anywhere. This article discusses some of the trends and important platform decisions that you need to make when extending your embedded system to mobile devices.



There has been a dramatic shift in market share of smartphones, but fragmentation is not going away.
Source: http://blog.seattlepi.com/microsoft/chart-mobile-os-market-shares



A look into the future shows the ever-changing mobile tablet market.

Technology Trends and Fragmentation

Fragmentation in the mobile market is not going away any time soon. Looking only at the smartphone market, there has been a drastic change in market share since 2009.

In this time, Android and iOS have gained market share and Blackberry OS, Symbian, and Windows Mobile have all but disappeared. The tablet market is even more volatile, with iOS maintaining a massive market share and several large vendors including Google, Microsoft, RIM, and HP developing a new round of competitors.

Ultimately, with all of the information available today, the choice seems to be one of two dominant platforms. With such an immature market, it is impossible to predict what the landscape will be in just two years. Another disruptive force could easily change the game again. Knowing that most embedded measurement and control systems have a life span of more than five years, the best you can do is employ short, iterative, and flexible design cycles while monitoring the market for disruptive change.

Cloud Computing

In addition to the ubiquity of connected networks, a primary enabling technology in the mobile revolution is cloud computing. In the context of embedded measurement and control applications, cloud computing generally provides one of the following benefits:

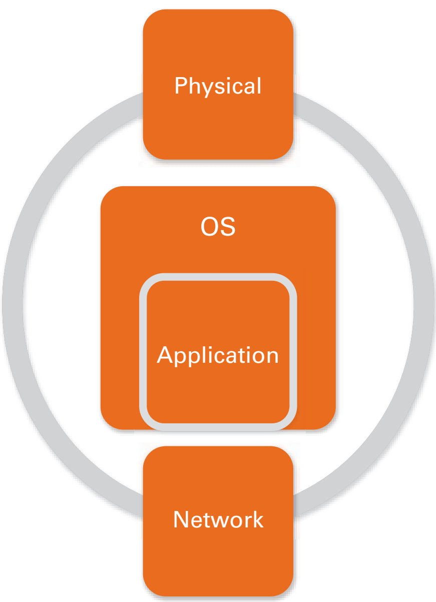
- **Aggregation of Data:** If the distance between elements of your system is measured in kilometers as opposed to millimeters, you may want to consider cloud data storage. For example, if you are monitoring the condition of each of the gear boxes in a wind farm with hundreds of turbines, collecting data can become extremely costly and cumbersome. With cloud storage, such systems can store data in a common location so that you can easily collect, analyze, and compare it.
- **Access to Data:** In some cases, the embedded control or monitoring system that you are designing is difficult to access physically. For example, if you are monitoring the health of a pipeline in a remote stretch of Alaska, you would ideally not need to send a technician to log the information and check the status of the system. If that data is being stored to the cloud, you can access it from anywhere, including connected PCs and mobile devices.
- **Offloading:** The near infinite computing resources that are available in the cloud provide an opportunity for software to offload computationally heavy tasks. These can be sophisticating image or signal processing or even compilation and development. For example, National Instruments recently introduced a cloud compile service for NI LabVIEW FPGA where you can offload and parallelize field-programmable gate array (FPGA) compilations.

Many companies offer cloud storage and services. Most charge a small setup fee and are set up on demand so that your cost is proportional to the amount of resources you use. National Instruments also provides a Technical Data Cloud that is designed specifically to store and access measurement data.

Embedded System Security

Unfortunately, allowing remote access to embedded systems introduces additional risks that must be addressed, the foremost of which is related to the security of the system. Definitions of "secure enough" can differ vastly, as such definitions are extremely subjective. Security, fundamentally, comes at a trade-off; more secure systems require greater time and cost investments and sacrifice convenience. Therefore, it is necessary to evaluate the proper investment in security for each application based on hazard and risk of failure.

For embedded monitoring and control systems, security measures can be classified at one of the following four layers: application, OS, network, and physical. It's important to have some protection at each layer.



It's important to integrate security into every layer of your networked embedded system.

Some best practices for security on embedded systems can be recommended for all basic needs, while others are only necessary for the most advanced needs. The following is a list of areas where you can secure your OS and network:

Basic

- Disable any services that leave open network ports (like FTP)
- Enable SSL support for any Web services
- Install security updates and patches from your OS vendor
- Set up antivirus and firewall software

Advanced

- Change all default network ports
- Set up a VPN-enabled firewall
- Enable third-party application whitelisting
- Encrypt all internal and external communication signals

Supporting Multiple Platforms

There are emerging technologies that make cross platform support a reasonable option. There are also some drawbacks to the general concept of cross platform applications, as well as issues to consider with each implementation approach.

Cross Platform User Experience

When looking only at Android and iOS, each platform has slightly different user experience models and conventions. What an Android user might see as normal may be jarring to an iOS user. This makes cross platform development limiting, as you must avoid those operations that are not common between the two major platforms. Here are a few examples:

- Most Android developers shy away from a global tab bar for navigation, but this practice is common on the iPhone.
- Some iOS elements are redundant with Android hard buttons. Some examples include Back and Action buttons, as well as search bars on top of list views. Most users can still use these, but they immediately notice them as peculiar, which negatively affects their perception of the app.
- Some iOS elements need to be replaced by Android conventions. A prime example of this is the Detail Disclosure button, which does not exist on Android.

HTML5

The most robust approach for supporting multiple platforms is basing a Web application on HTML5. This technology has become the preferred way to present dynamic content to users in mobile browsers. Even Adobe has recently adjusted its strategy away from mobile Flash and toward HTML5, and Microsoft has announced that their Metro OS will not support browser plug-ins in favor of HTML5. This technology shows promise, but the tools are immature and there are questions about the richness of the experience that you can provide with HTML5 compared to desktop technologies like Silverlight and Flash.

Innovating With Smaller Teams

By the nature of the economic situation today, businesses cannot just spend their way into innovation.

When searching for new sources of innovation, businesses must go beyond the walls of traditional R&D departments and decade-old engineering practices. Smaller, more agile competitors are emerging around the world and are proving that innovation can happen just as fast, if not faster, in smaller design teams. They are delivering disruptive innovations to market in shorter time periods and with equivalent, and in some cases better, quality than their traditional competitors. This is happening in established economies and equally, if not more impressively, in emerging economies as well.

> *"Leveraging our strategic partnership with NI and its technologies, our small team was able to propose and win business at FedEx Express, delivering a quality solution in half of the time and at about 50 percent of the cost of our competition."*

Jeremy Snow,
President and CEO,
Ventura Aerospace

System-Level Tools and Small Teams

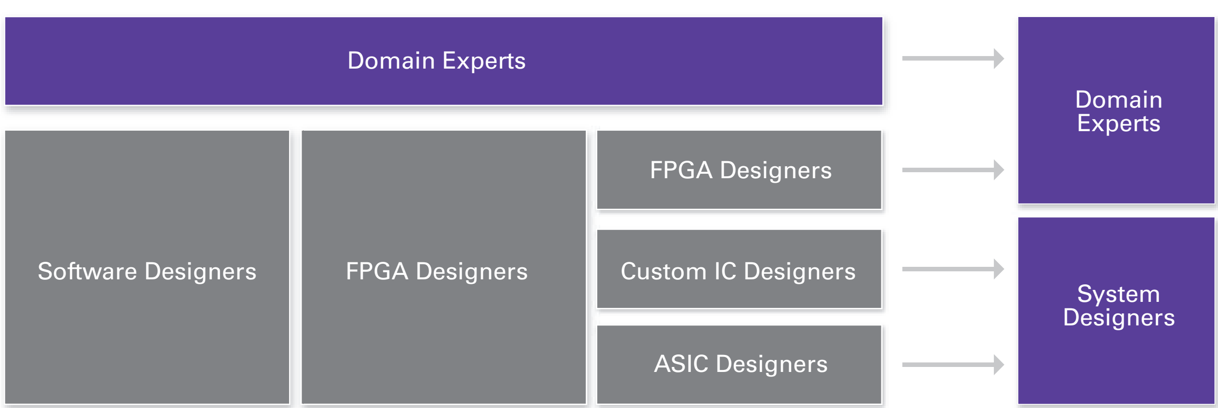
Looking at technology platforms and tools available to businesses today, there are two predominant competing design approaches. One is the traditional way of thinking, which means for a business to deliver innovation, it has to vertically integrate the hardware and software stack itself and "own" the design of all these components to deliver the quality and experience customers expect. This approach normally requires a larger design team and seems to fit better in industries where product volumes are astronomical or the design team is iterating on an existing custom design, like the consumer electronics market. The alternative methodology is the emerging approach of using off-the-shelf, system-level tools with a standard ecosystem of technology that can help accelerate the process of innovation without compromising the ability to differentiate. This approach has proven to be a better fit for areas of aggressive innovation, such as renewable energy and life sciences, as it uses standard off-the-shelf hardware components and prebuilt software tools that are friendlier and more tolerant to quick changes.

System-level software tools can reduce the need for specialized software and hardware description language tool expertise so that fewer team members can implement more of the system. Several design tools available today address this need, including NI LabVIEW graphical system design tool, Xilinx AutoESL C-to-Gates high-level synthesis tools, and Mentor Graphics Catapult C Synthesis. Commercial off-the-shelf hardware platforms optimized for control and monitoring systems integrate connectivity, signal conditioning, and A/Ds and digital-to-analog converters (DACs) with a high level of modularity to meet the needs of most applications. In many cases the need for discrete analog and digital design may be completely eliminated, further reducing the size of the design team.

The Small Team Advantage

An integrated design toolchain provides a high level of abstraction and productivity to make it possible for engineers and scientists, not trained in traditional software or digital design tools, to participate in larger portions of the system design without sacrificing performance. Market and scientific domain experts can also take on a more active role in the design process, implementing prototypes and features alongside other members of the team, reducing the number of iterations needed to finalize the design. The amount of diverse expertise needed could be reduced, providing increased collaboration and efficiency when mapping requirements to features. A single engineer on the design team could implement application, DSP, measurement, and control software executing on both the MPU and FPGA, blurring lines between hardware and software engineering; creating the new role of a system architect or designer.

The design team of domain experts and system architects is a sharp contrast to traditional large design teams using traditional tools. Large design teams can be inefficient and encounter difficulty executing all design needs in parallel without creating an ever-changing list of requirements. Even specifying system requirements can be challenging. Large design teams often have difficulty mapping market requirements into system features. For example, the team's market or scientific domain experts may have trouble communicating the precise performance, accuracy, or system behavior to other team members. Converging on the correct design and feature set may require multiple iterations and revisions of the system. Domain experts and system architects using a common design tool work more closely together, iterate more quickly, and better map market requirements to implementation. With off-the-shelf, system-level tools, domain experts and system architects can collaborate with a common design tool to implement a better system more efficiently, get to market faster, and reduce costs.



The shift to a smaller design team gives companies the ability to innovate more efficiently and better meet market needs.

Industry Examples and Acquisitions

Several industry examples illustrate the shift toward smaller design teams. For example, NexGEN Consultancy in India brought an Advanced Meter Reading (AMR) system to market to help utilities access accurate and sufficient data from metering devices and enhance the current electrical distribution network situation in India. Businesses, like NexGEN, in emerging countries have a unique challenge and opportunity. They not only lack the massive resources available in more established countries but also lack one of the greatest barriers of innovation: an existing mentality of doing things in a certain way. Herein lies the big opportunity for innovation. They can look for new ways of doing things without having to convince a large number of people why they are not doing it the "normal" typical way.

Ventura Aerospace Inc., a small company that services the airfreight industry with rigid cargo barriers and fire suppression technology, is an example of a smaller company and design team innovating more quickly in the transportation and aerospace industries. Using a small team of domain experts and system designers and NI graphical system design tools, Ventura Aerospace developed an innovative fire suppression control system for FedEx, and did it faster, cheaper, and with better quality than a much larger, entrenched competitor in this space. This is great proof that small teams can gain market share and be more efficient with the right tools.

Success from small teams can also be seen in a growing number of smaller, more innovative companies and assets being acquired by larger, more traditional competitors. One recent example within the embedded system space is OptiMedica Corp. selling its retina and glaucoma assets to Topcon Corp. OptiMedica, a small startup company based in California, has been successful in building an innovative ophthalmic medical device using graphical system design tools from National Instruments. Topcon Corp., based in Tokyo, Japan, and established in 1932, is one of the world's leading manufacturers of ophthalmic, optometric, GPS, and positioning control systems.

Future of Innovative Teams

The shift to smaller, more agile development teams will gain stronger momentum over time as established industry players observe smaller players out-innovating them. Giving the power of innovation to domain experts gives small businesses the opportunity to solve problems using lower cost structures than their traditional competitors. In addition to the cost advantage, these small businesses can also provide better quality solutions developed by the people most familiar with the problems.

In the application areas of embedded control and monitoring, there is a strong need for accelerated innovation to solve urgent challenges in industries like energy and life sciences. Many business leaders understand that innovation can be a powerful competitive differentiator. Entrepreneurs, inventors, and startup CEOs are looking for new methodologies and technology platforms that can help accelerate the cycle of innovation in their companies. Smaller design teams could be a strategic way to achieve that goal without compromising the quality of their solutions or the profitability of their business.

Future Proofing Through Software

With ever-changing system requirements and standards, teams are adopting a "software-first" approach for upgrading products.

In the last decade, there has been a flood of embedded technologies incorporated into consumer devices, from GPS systems to autonomous vacuum cleaners to mobile tablets. This growth in the market brings both opportunities and new challenges to designers of embedded systems. The available capabilities of performance and communication at a low cost are exploding but so are customer expectations for data visibility, connectivity, and performance. In addition, for most embedded systems, the expectations on longevity of design and service life remain well over a decade.

Software First

To address these constraints, many successful embedded developers have adopted a software-first philosophy where software is the first option to address system requirements. This software-first approach can be a huge asset during the design phase, giving designers the ability to more easily take advantage of modern 32-bit processors, rapidly optimize their designs for conflicting requirements, and reduce time to market. In addition, a software-first philosophy can provide a strategic advantage throughout the life cycle of a product through the capability to future proof designs.

With a future proofing capability, embedded designers can create products that not only meet customer requirements today but also meet the expectations of the market a decade from now. These new requirements may include the following:

- Integrating other systems through communication protocols or Web technologies
- Protecting against cyber security threats
- Adding new features such as better control algorithms or lower power consumption
- Addressing a system bug or failure such as correcting date register overflow or working around a hardware or sensor failure

The technical capability to future proof designs can also create a strategic business advantage. Companies can support and troubleshoot customer problems at a lower cost and increase customer loyalty through customization options and feature enhancements.

Consumers are already familiar with software updates for the purposes of future proofing. Stories such as a Toyota Prius software update to fix a brake issue or an Android smartphone over-the-air OS update are common in the news. These same approaches are being applied in embedded control and monitoring systems such as electrical power monitoring and software-defined radios.

The Smart Grid

Embedded systems currently undergoing a major technology refresh are control and monitoring systems supporting the US electrical distribution grid. In the original incarnation of the electrical grid, there was no distributed monitoring or feedback from remote sections of the distribution network, and the control mechanisms to handle over current conditions were simple fuses or mechanical circuit breakers. Today, utilities across the nation are in the process of upgrading the electrical sensing and control mechanisms on the grid to provide better visibility into power flows and grid health. This "smart grid" trend is a foundational investment that will provide greater reliability, the ability to integrate green and renewable power, and hooks for better consumer visibility and energy savings.

However, the challenge electrical utilities are facing is that while they are making massive investments to deploy embedded monitoring and control systems, new technologies, power sources, and loads are being developed and the final requirements and desired capabilities of the smart grid are unknown. Rapid research and developing innovation promise future capabilities such as grid-integrated home area networks, self-healing distribution topologies, and fault-prediction algorithms to prevent outages. The final implementations necessary to support these capabilities, such as how to get smart meters to talk to car chargers and water heaters, how to integrate new real-time event-based communication protocols, or what algorithms to apply to measured voltage and current waveforms, are still being developed; therefore, many utilities are looking to software-updatable systems. By deploying measurement and control nodes that can be remotely updated to support new algorithms and communication protocols, utilities are proceeding with investments in grid infrastructure while future proofing their installations for future capabilities.

Software-Defined Radio

Radio design traditionally involves the careful selection and integration of discrete components for capabilities such as modulation/demodulation, filtering, and amplification. Today with the explosion of wireless devices, increased spectrum use and crowding, and the rapid innovation around wireless protocols, designers are investigating a new paradigm in radio design where processors and software handle more of the functionality.

The idea for this approach originally came out of military applications intended to reduce the number of radios soldiers carried in the field. In a software-defined radio, the analog and specialty hardware components in the design are minimized and a digitized signal is passed to a CPU for signal processing. Because the protocol is no longer defined by specific hardware, multiple radio protocols can be supported without hardware changes. Software-defined radio, along with a related area of research in software-defined antennas, could unlock numerous future capabilities ranging from support of next-generation communication protocols (imagine if your 3G phone could support LTE with only a software update) to cognitive radio where radios automatically adjust their band usage and power output to maximize spectrum use while minimizing interference. While this is still largely an area of research, the potential to have a universal design that is future proofed through software and can quickly support new protocols and capabilities has prompted an active community of developers who are working on a software-first approach to radio design.

How to Design to Future Proof

Designing a future proof system requires three key criteria: a software-definable hardware platform, field update capabilities, and forward compatibility in the toolchain.

Software-Definable Hardware

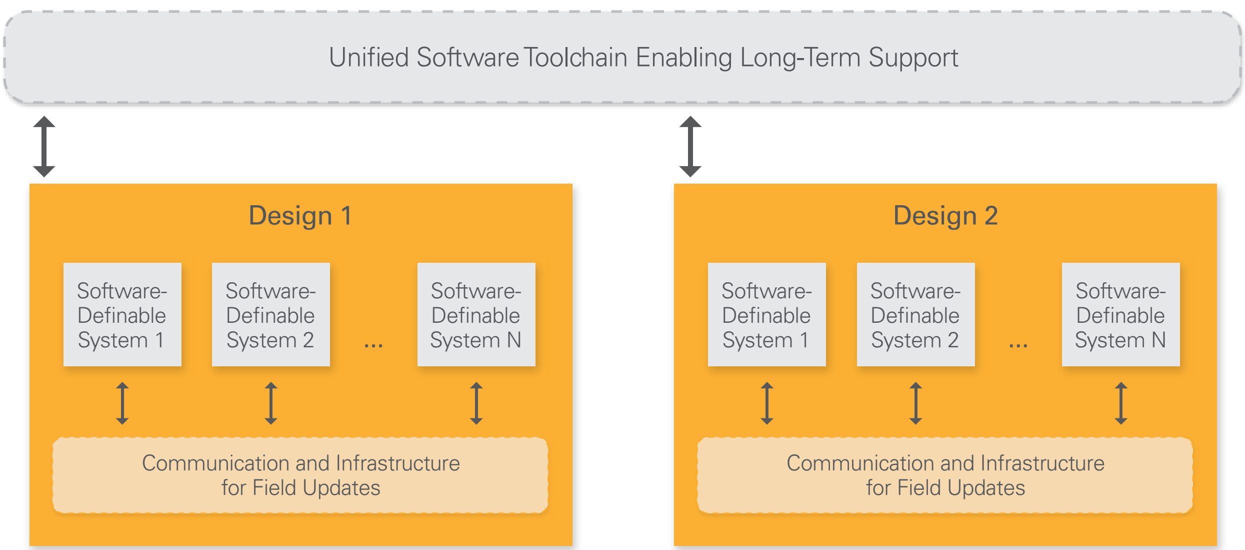
A software-first design paradigm is predicated on a system architecture that minimizes fixed-function hardware. This includes obvious fixed-functionality devices such as application-specific integrated circuits (ASICs) and hardware filters. Although these fixed-function devices offer a lower per-piece component cost, they achieve that cost at the expense of future scalability. Software-defined hardware platforms, such as processors, digital signal processors (DSPs), and field-programmable gate arrays (FPGAs), give system designers the flexibility to more completely change a device's behavior without new electrical work. While these platforms have higher component costs, they can dramatically reduce design costs, increase market share through faster time to market, and over time increase volume and drive down cost by making it possible to use one design across multiple devices.

Field Update Capabilities and Infrastructure

Although it may be acceptable to have end devices returned to a repair facility, in many cases embedded devices are permanently mounted in the field and it is desirable to field upgrade the devices. For instance, in the power industry the costs associated with a service call to a remote pole-mounted monitoring system can rival the up-front cost of the embedded controller. Embedded designers typically implement field update mechanisms for their devices. As they create these systems, they need to decide on the appropriate infrastructure and mechanisms for their specific design including physical medium, security, reliability, and system management. A USB stick is a popular updating method and the requirements for physical access and user intervention help to address security and system management concerns. Another popular, and more flexible, option is remote updates using Web services via a wireless or cabled connection. In this design, the embedded device uses Web services and security mechanisms such as SSL to call home periodically for status or update messages. The home server manages device versions and capabilities deployed in the field. If an update is desired the device can download a new set of software commands or FPGA files, run a cyclic redundancy check (CRC) or other checks to validate proper receipt of the files, and then reload the new version. These mechanisms can either be stand-alone or incorporated into existing enterprise asset management tools.

Deciding on a Toolchain

Because embedded devices live on the market for years or decades, it is essential that the design and development software tools offer long-term support and raise business issues that system designers may not typically consider. For instance, although open-source software tools offer a low price point, a designer may not be able to get appropriate development tools or technical support in the future when the tools have continued to evolve. Likewise, it is unlikely that the original development team will be available when device updates are desired and the future availability of other competent/trained developers should be considered. The choice of established software tools and reputable vendor selection for hardware and software tools will help mitigate the challenges of long-term support of embedded systems.



System design teams should use software-based toolchains that give them the ability to easily upgrade systems and solutions over time.

US Corporate Headquarters
11500 N Mopac Expwy Austin, TX 78759-3504
T: 512 683 0100 F: 512 683 9300 info@ni.com

International Branch Offices – **ni.com/global**

©2012 National Instruments. All rights reserved. LabVIEW, National Instruments, NI and ni.com are trademarks of National Instruments. LEGO is a trademark of the LEGO Group. ©2012 the LEGO Group. Other product and company names listed are trademarks or trade names of their respective companies. A National Instruments Alliance Partner is a business entity independent from National Instruments and has no agency, partnership, or joint-venture relationship with National Instruments.
351334B-01 03879

100%
From well-managed forests
FSC www.fsc.org FSC® C000000