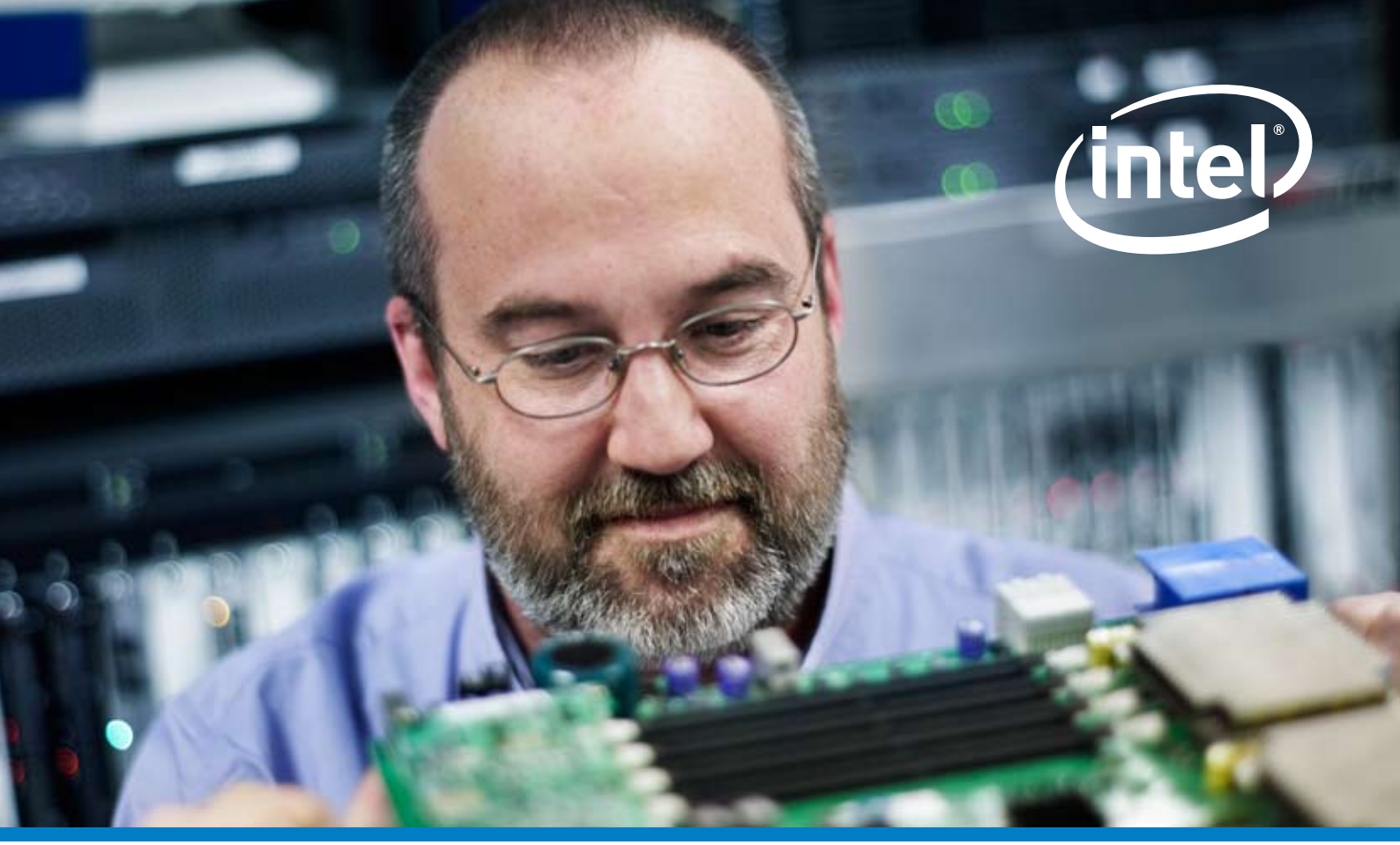


Case Study

Embedded UEFI Firmware Platform for Industrial PCs

UEFI: A Better Firmware for Industrial Platforms

Advantech* harnesses the UEFI firmware to enable remote monitoring and management of e-platforms without an Operating System.

Trusted ePlatform Services

ADVANTECH

Computing firmware change is coming to the embedded industry in the form of a new standard interface that replaces the proprietary lock between hardware and the operating system. Known as the Unified Extensible Firmware Interface (UEFI), this new and open firmware standard is a well-designed, highly versatile and long-overdue replacement for 25-year-old basic input/output system technology (BIOS).

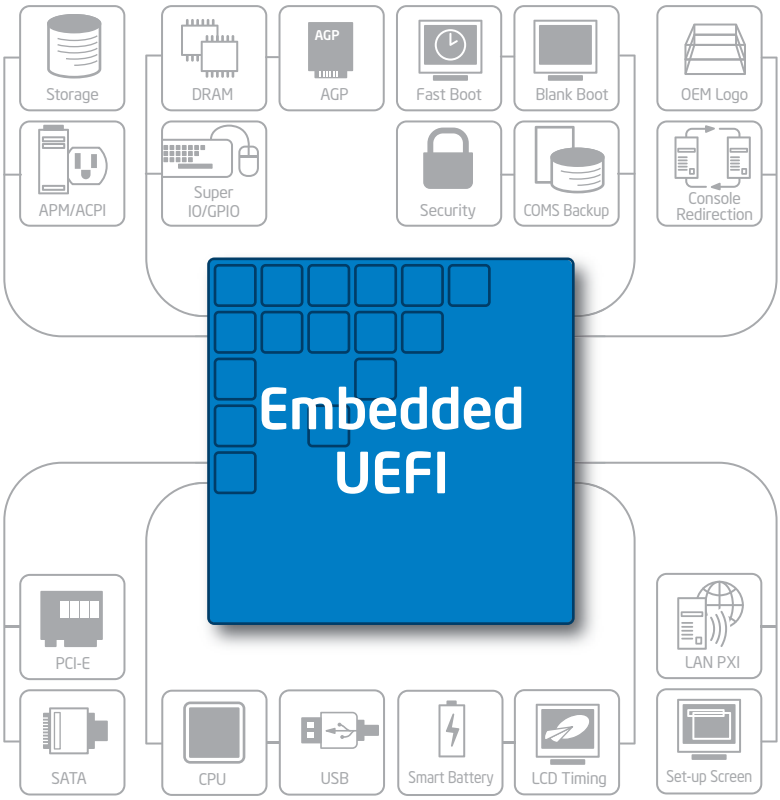
Although it is virtually unknown in the embedded industry today, the UEFI is quickly replacing BIOS technology in the PC industry, where desktops and notebooks due out in 2009 are being developed with UEFI boot technology. The embedded industry is sure to follow in the very near future. In fact, Advantech and Intel are committed to facilitate a complete transformation from legacy BIOS solutions to the UEFI standard within the next few years. Both companies are delivering UEFI solutions today, and more are planned.

Advantech's first UEFI BIOS solutions not only replace old BIOS technology but include remote monitoring and management features that can function even without a working operating system (OS). Booting directly to a UEFI shell, Advantech's e-platforms can even launch embedded applications without first booting an OS. The full potential of this capability and how you ultimately use the UEFI interface in your embedded designs is limited only by your imagination.

Background: Why UEFI is a Better Firmware

The Unified Extensible Firmware Interface (UEFI) standard is a long-overdue technology replacement for the 25-year-old basic input/output system (BIOS) used by the computing industry. It unlocks the firmware layer to give OEMs greater versatility and choice in selecting hardware and operating systems for their platforms.

The UEFI interface standard offers the versatility needed by industrial PC manufacturers to establish an open, flexible connection between the platform hardware and embedded applications. For example, UEFI standard firmware runs freely on any silicon architecture, including any x86 architecture and others. It also supports 64-bit code, something legacy BIOS cannot do. It's a standards-based specification implemented in the commonly used "C" programming language, which means there is wide availability of tools to debug the hardware. It even supports a graphics mode of pre-boot environment that enables validation and benchmarking without the need of an OS.



The UEFI can support both GUID partition table (GPT) and master boot record (MBR) disk configurations, while legacy BIOS is limited to using MBR. As a result, UEFI is much more modular and extensible to a variety of platforms. It specifies a highly portable driver model and an application with runtime API's to these modules.

And there's no need to worry about compatibility between their existing BIOS-based designs and the new UEFI because it was designed to coexist with legacy BIOS services, allowing the system to support both UEFI and legacy boot via the compatibility support module (CSM).

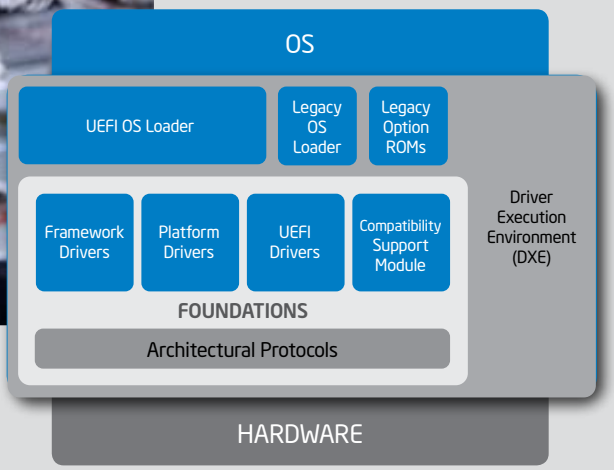
It's Easier to Differentiate with UEFI

There are dozens of pressures facing industrial PC makers, not the least of which is figuring out how to differentiate solutions and generate profits in a highly competitive industry. Customers want systems that deliver robust feature sets, high performance, energy efficiency, ruggedized operating ability and support for a wide variety of applications. Yet they also expect low prices and long product life.

One of the ways that the embedded industry sets itself apart from the commercial PC and server marketplace is through the use of a wide variety of operating systems. These can be as varied as the applications that run on them. But the old BIOS makes it even more difficult to support all of those operating systems since each one requires its own specific firmware solution.

"EFI is an evolution from legacy BIOS architectures, and Advantech Emb'Core Service provides innovative embedded EFI solutions specifically for the Industrial PC market," said Miller Chang, director of the Emb'Core Services. "We focus on embedded EFI firmware, pre-boot applications and interface development to provide an embedded EFI ready solution to the Industrial PC market."

Intel has developed a platform innovation "framework" to ensure that the UEFI interface is not OS and hardware dependent (see sidebar on page 3). Using the framework and the UEFI specification means the firmware can link the hardware to any OS or application; all that's needed is a graphical user interface (GUI), such as a web browser or other application-specific interface. It's not a replacement for your system OS, but it can work around it when necessary.



These capabilities then allow you to consider a whole range of new embedded functions that don't require a rewrite of your BIOS. You can extend platforms to add new modules or components without touching the core. Just imagine the potential EFI-based solutions now possible.

Advantech Harnesses UEFI Potential

Advantech Embedded Core (Emb'Core) Services has developed one of the very first embedded UEFI firmware solutions that enables industrial PC makers to build more reliable systems that can easily be differentiated to meet different vertical market requirements. Using Intel's "framework," Advantech EFI solutions allow customers to configure firmware features for specific vertical market demands. Advantech can support vendors' EFI solutions, and provide versatile embedded software features and customization services.

Advantech Embedded EFI solutions have remote monitoring and boot capabilities that allow for down-the-wire control of systems, even without a working operating system. The complete package of solutions includes three elements:

Embedded EFI Firmware – Modular core architecture enables flexible BIOS configuration and enhanced CPU management with power savings. The versatility of the architecture lets customers tailor their products to a wider range of designs and target markets. Boot-safe technology has OS and software images integrated into the EFI. And the BIOS security ID mechanism has a secure application program interface.

Pre-Boot Application – With the user-friendly Advantech Embedded EFI Pre-Boot Application customers can monitor systems, automatically diagnose and detect the CPU temperature, fan speed and voltage, and send the information to the remote EFI Alert Server for further action. This application can also be used to upgrade the firmware and enable the TCP/IP connection.

Remote Management Server – Provides complete, remote, real-time client monitoring of temperature, fan, voltage and alarm settings as well as crisis backup and recovery when needed. Event logging is also provided.

Intel® Platform Innovation Framework for UEFI

The Intel® Platform Innovation Framework for EFI (referred to as "the Framework") is a product-strength implementation of EFI and UEFI. The framework is a set of robust architectural interfaces, implemented in the "C" programming language that has been designed to enable the BIOS industry and our customers to accelerate the evolution of innovative, differentiated, platform designs. The framework is Intel's recommended implementation of the EFI Specification for platforms based on all members of the Intel® Architecture (IA) family.

Unlike the UEFI specification which focuses only on programmatic interfaces for the interactions between the operating system and system firmware, the framework is an all-new firmware implementation that has been designed to perform the full range of operations that are required to initialize the platform from power-on through transfer of control to the operating system. The framework differs from previous generations of firmware infrastructure typically used on IA systems in the following ways:

- It employs a purpose-built modular component design.
- It uses high-level language coding wherever possible.
- It is designed from the outset to support long-term growth of platform capabilities and innovation in the pre-boot environment.
- It is designed to provide a single code base that is equally applicable to platforms based on all members of the Intel® architecture processor family and that scales to fit the needs of everything from handheld devices to high-end servers.

Calculating the Benefits

The openness of the UEFI standard coupled with Advantech's pre-developed embedded UEFI firmware solution speeds delivery of new applications. There's no need to rewrite the BIOS or firmware for each new application, and OEMs/ODMs can now source their firmware from different vendors. Similarly, hardware vendors can more easily enable new silicon on their platforms, leaving time and resources for the board vendors (like Advantech) to dedicate to adding value and differentiating their solutions.

EFI Pre-Boot Application

- Resource of System Table
  - ST -> ConOut -> ClearScreen
  - ST ConOut EnableSursor(ST ConOut, FALSE) & ST ConOut Enable Cursor (ST ConOut TRUE)
  - ST ConOut SetAttribute (ST ConOut, XX)
  - ST ConOut SetCursorPosition (ST ConOut, x, y)
  - ST ConIn ReadKeyStroke
- Library of EDK

  Print(L".....",string1,string2....);

  PrintAt(L".....",string1,string2....);
- PCI, IO & Memory Access

  LocateHandle, HandleProtocol for gEfiPciRootBridgeLoProtocolGuid

  EFI_PCI_ROOT_BRIDGE_IO_PROTOCOL
  - EFI_PCI_ROOT_BRIDGE_D_PROTOCOL_ACCESS Mem;
  - EFI_PCI_ROOT_BRIDGE_D_PROTOCOL_ACCESS Io;
  - EFI_PCI_ROOT_BRIDGE_D_PROTOCOL_ACCESS Pci;

Sample Code for Advantech EFI Pre-Boot Application

UEFI brings other benefits as well, specifically in the pre-boot space. With the full TCP/IP network stack in the pre-boot, and a standard environment for running pre-boot applications, developers can consider using an interface other than an operating system for launching applications. The possibilities enabled by this approach are virtually unexplored but wide open.

Developers will also enjoy the C programming language and the reusability of the drivers. UEFI modules can be developed once and reused multiple times across platforms of all types.

Conclusion

The time is probably not too far off when embedded developers will stop supporting legacy, proprietary BIOS firmware solutions. Why wait for that time to come? With UEFI now available in embedded platforms, and access to a framework that enables quick delivery of versatile, scalable embedded designs, you can start using it today.

For more information:

www.uefi.org
www.intel.com/technolgoy/framework
www.intel.com/go/embedded