



[Boards](#) | [Chips](#) | [Devices](#) | [Software](#) | [LinuxDevices.com Archive](#) | [About](#) | [Contact](#) | [Subscribe](#)







Toradex
Embedded. Computing.

**We're
HIRING!**

**ARE YOU INTERESTED IN A
JOB WITH TORADEX?**
[Apply Now!](#)

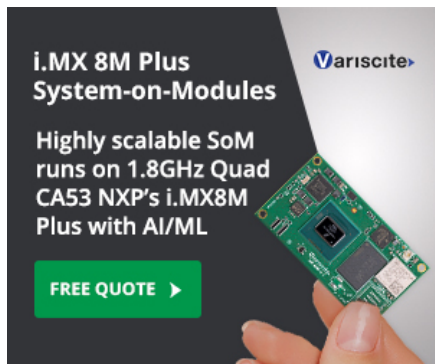
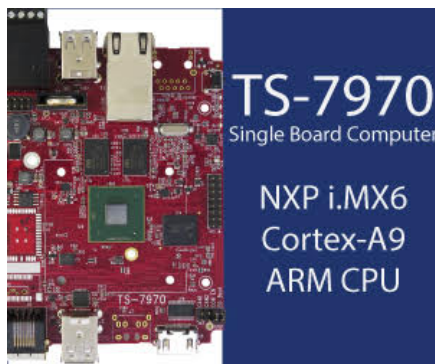


Follow LinuxGizmos:
   
[* get email updates *](#)

- **Search LinuxGizmos:**

- **Search LinuxGizmos.com + LinuxDevices Archive:**


- **LinuxGizmos Sponsor ads:**



Enhanced for IoT

COM Express Mini with Intel Atom® x6000E

- High Performance
- LPDDR4 4266 MT/s
- In-Band ECC
- 2.5GbE Ethernet
- TSN / TCC



ADLINK **COM Express** Intel ATOM Intel COLUMERON Intel PENTIUM

cicoze

Dual GPU, Double Power

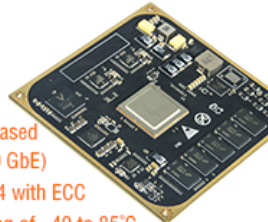
Industrial AI and Machine Vision Computer

GP-3000



Open Embed

Rugged System-On-Module



- NXP LS1046A based
- Up to 2x XFI (10 GbE)
- Up to 8GB DDR4 with ECC
- Temperature range of -40 to 85°C

DART-6UL

Variscite

- NXP i.MX 6UL/6ULL/6ULZ up to 900MHz Cortex-A7
- Dual LAN, USB, CAN
- Small size 25x50mm
- Low power < 5mA
- Certified WiFi/BT 802.11 ac/a/b/g/n

From \$24

GET A QUOTE



DFI

1st Industrial Pi + AMD R1000


1.8" SBC GHF51

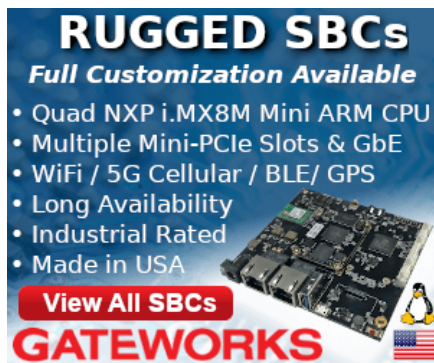
in box

Free Sample

Apply Now

EC90A-GH





Engineered to Last

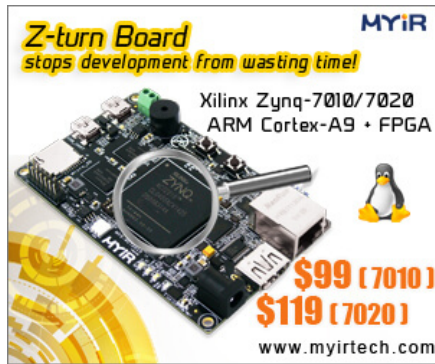
OnLogic

(advertise here)

• Top 10 trending posts...

- [Zynq UltraScale+ modules include high-end Andromeda model](#)
- [Six-axis manipulation robot based on Raspberry Pi 4B sells for \\$699](#)
- [Banana Pi with quad -A55 Amlogic S905X3 launches at \\$61](#)
- [SATA HATs support up to four drives on Raspberry Pi 4 or Rock Pi 4](#)
- [Turing Pi 2 clusters four Raspberry Pi CM4 modules](#)
- [Tiny, dual-GbE Raspberry Pi CM4 carrier sells for \\$30](#)
- [Raspberry Pi CM4 carriers boast dual LAN ports](#)
- [Rock Pi SBCs move to 2GHz RK3399 and toss in free eMMC](#)
- [Raspberry Pi-like Zynq-7020 SBC sells for \\$72](#)
- [ClusBerry for Home clusters up to eight CM4 modules with a la carte I/O](#)

• LinuxGizmos Sponsor ads:



(advertise here)

• Follow LinuxGizmos or subscribe to our posts:



• Subscribe

Enter your email address to subscribe to this blog and receive notifications of new posts by email.

Join 1,344 other subscribers

Using UEFI in embedded and mobile devices

May 29, 2013 — by Guest Contributor — 1530 views



In this guest column, two engineers with Intel's Software and Solutions Group describe the benefits of UEFI pre-boot software to mobile and embedded devices. The Unified Extensible Firmware Interface specifications are meant to facilitate emerging technologies, services, security mechanisms, and user experiences that come into play prior to loading the device's OS.

Using UEFI in embedded systems, from smartphones to in-vehicle infotainment

by Vincent Zimmer and Michael Rothman

Designed to allow for cross-functionality between devices, software, and systems, Unified Extensible Firmware Interface (UEFI) specifications encourage innovation, helping to drive the evolution of next-generation technologies, such as the expansion of embedded and non-PC systems. To accomplish this, UEFI provides a plurality of services, including console, storage, and networking services.

Hosted and implemented by the [UEFI Forum](#), this technology enables firmware innovation by promoting a standardized, extensible, and interoperable firmware interface that extends far beyond the PC and into the future of computing and the realm of embedded systems.

Embedded systems — from simple-state platforms to 64-bit micros

Decades ago, embedded systems were analog or simple-state, machine-based platforms using SSI, MSI, or PLDs. In 1970, the Intel 4004 introduced the ability for reprogrammable embedded systems, such as traffic lights, controls, and other fixed-function usages.

The microprocessor ushered in a wave of continual evolution, demonstrated by the embedded market shift from 8-bit, to 16-bit, to 32-bit, to 64-bit microprocessors. This reflects the progression from simple cell phones to smartphones, from mechanical brakes to microprocessor-controlled anti-lock units, and from handwritten directions to satellite-based navigation systems. Along with this advancement, in-vehicle infotainment and digital signage emerged, along with many other instances of hidden intelligence in the world around us — providing several new modes of utility for UEFI.

This wave of hardware evolution brought about a nearly insatiable appetite for software.

Scaling the PC ecosystem into the embedded space

With the increasing richness of the software ecosystem and the growing complexity of platforms, a layer between the operating system kernel and the platform became a necessity. As UEFI-enabled PC platforms became pervasive, a new challenge emerged: how does one take the evolving PC ecosystem and scale it down for use in the embedded space? Typical PC platforms aren't scaled to meet the advances in the embedded territory. For various embedded operating systems, user demand drives a vastly different set of requirements, such as instant power-on.



Many of these operating systems require customized firmware with OS-specific hardware interfaces in order to fit into the PC firmware ecosystem model. Ultimately, everyone — from the vendor, to the developer, to the end-user — wants to work smarter, as opposed to harder. The contemporary challenge is to provide the embedded platform firmware with capabilities comparable to the traditional model. The firmware must be OS-agnostic, scalable across different platform hardware, and capable of enhancing the efficiency of developers. With its capability of supporting scalability, UEFI specifications fulfill a unique role.

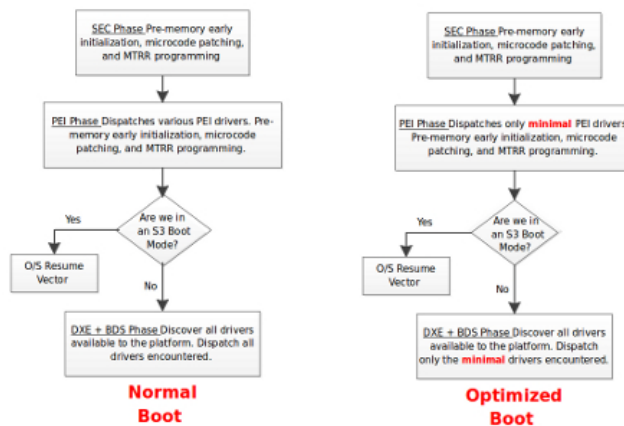
UEFI specifications provide a consistent set of OS-agnostic software interfaces that abstract the underlying details of the platform. These abstractions can be layered upon SATA, SCSI, iSCSI, USB Bulk Only Transport, or any other block device. By traveling with the system board, UEFI serves as a set of built-in drivers and capabilities upon which operating system loaders and pre-OS applications can depend.

From the software perspective, not only does UEFI work across platforms to help facilitate growth and interoperability, but it serves as a useful tool in the fight against malware in the pre-OS space. Using UEFI standards, independent software vendors (ISVs) serving the pre-OS market can write interoperable applications — such as provisioning agents, diagnostics, and full-disk encryption. Because UEFI standards enable troubleshooting and testing to be completed prior to when the OS loads, efficiency in the development process is enhanced.

From the hardware perspective, UEFI specifications promote innovation and interoperability, working across platforms to sustain technological growth. Independent hardware vendors (IHVs) support UEFI drivers that are designed to function on any UEFI-conformant system board. These driver programs can function under UEFI specifications on multiple processor architectures.

UEFI architecture and optimization of extensible boot

From its onset, UEFI firmware was designed to support extreme scalability, as shown in the diagram below. There are no design differences between the normal boot and an optimized boot. Optimizing a platform's performance does not require violating any of the design specifications. Additionally, a UEFI standards-compliant design need not encompass all aspects of the standard PC architecture; instead, its scope can be limited to the components required for platform initialization.



Architectural boot flow comparison
(click image to enlarge; source Intel)

Designed for scalability, extensibility and interoperability, UEFI is well-positioned to streamline technological evolution in both PC and embedded applications. As the complexity of the platform increases in the embedded sphere, the decoupling of concerns between the hardware and system software constituencies makes way for the flexibility and standardization provided by UEFI.

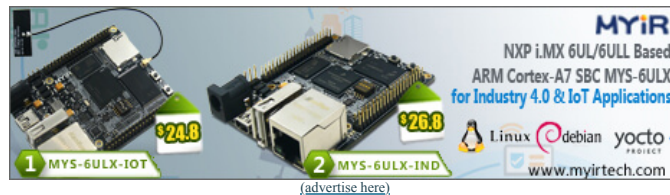
For more information about UEFI, visit the [UEFI Forum website](#).



Vincent Zimmer is an engineer in the Software and Solutions Group at Intel. Originally from Houston, he has worked at Intel since 1997. During his 20 years working in embedded systems and firmware, he has received more than 250 US patents, presented at several industry conferences and contributed to several book publications.

LinuxGizmos related posts:

- [IDE supports embedded Linux development within Windows](#)
- [Embedded Online Conference reg is free through Feb. 28](#)
- [Automotive Grade Linux dips into telematics with 6.0 release](#)
- [Open source ACRN hypervisor debuts on an industrial...](#)
- [Wind River Linux and VxWorks team up for new Helix...](#)
- [Runtime security agent tailors itself to each...](#)
- [Whiskey Lake thin Mini-ITX SBC boasts eight USB ports](#)
- [Automotive Grade Linux update clusters up -- and...](#)
- [Snapdragon 820E mini-PC supports AI on the retail edge](#)
- [A crash course in embedded Linux software deployment](#)



(advertise here)



Print Friendly

PLEASE COMMENT BELOW

2 responses to “Using UEFI in embedded and mobile devices”

1. *Rubberman* says:

[May. 30, 2013 at 4:06 pm](#)

My only problem with UEFI is how secure boot is being used to restrict user control of their purchased hardware. Of itself, UEFI is fine. Coupled with secure (sic) boot, it becomes an abomination! It should NOT be possible for a system vendor, such as Microsoft, to lock down SB on UEFI so that the hardware owner is incapable of installing other operating systems without resorting to mind-boggling (and difficult) processes. This is just unacceptable to me, and means that I will NEVER purchase such a system, EVER!

[Reply](#)

2. *ram* says:

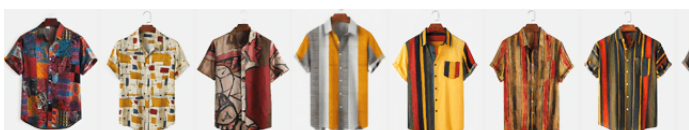
[Jun. 2, 2013 at 9:03 pm](#)

I agree with Rubberman above. UEFI is not a feature, it is a bug. My company will not develop for such platforms.

[Reply](#)

Please comment here...

Enter your comment here...



[\(advertise here\)](#)

The content on this site is copyright © 2007-2021 LinuxGizmos.com unless otherwise noted. Comments are the property of their submitters. LinuxGizmos is part of the [KCK Media Corp. network](#). Linux is a registered trademark of Linus Torvalds.

5