Embedded Intel® Solutions
Winter 2013

Security in the Connected Car

Analyze Automotive PCB Layouts Efficiently with Simulation

Performance-Portable Programming

Improve Real-Time Linux Behavior on Embedded Multicore Devices

www.embeddedintel.com

Gold Sponsors

Extension MEDIA

Scan this QR code to subscribe
Empowering the Mobile Workforce

Go the distance with your solutions NEXCOM takes your there

For more information, please contact us at sales@nexcom.com or (510) 656-2248. You may also visit us at www.NEXCOM.com

VTC 1000
In-Vehicle Computer
Intel® Atom™ processor E640-based Fanless In-vehicle Computer

- Compact/fanless design
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 6~36V
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VTC 7110-C4
In-Vehicle Computer
Intel® Core™ i7 processor-based Fanless In-vehicle Surveillance Computer

- Build-in Intel® Core™ i7-2610UE processor 1.5GHz
- 4-channel PoE and dual removal 2.5” SSD tray and CFast storage
- Optional CAN bus in support of SAE J 1939 or J 1708
- Support two SIM card slots
- 9~36V wide range DC power input
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VMC 3501
Vehicle Mount Computer
10.4” Rugged Vehicle Mount Computer with Intel® Core™ i7 processor, Touch Screen and IP65

- Built-in Intel® Core™ i7-2610UE processor. Compact and fanless design
- 10.4” XGA TFT LCD monitor with optional sunlight readable solution with 1000nits
- On screen F1~F10 function key
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 8~60V
- Optional CAN Bus with J 1939 and 1748 protocol
- Robust design with Die-cast aluminum front panel

www.nexcom.com

NEXCOM USA
2883 Bayview Drive, Fremont CA 94538

Empowering the Mobile Workforce
Go the distance with your solutions NEXCOM takes your there

For more information, please contact us at sales@nexcom.com or (510) 656-2248. You may also visit us at www.NEXCOM.com

VTC 1000
In-Vehicle Computer
Intel® Atom™ processor E640-based Fanless In-vehicle Computer

- Compact/fanless design
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 6~36V
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VTC 7110-C4
In-Vehicle Computer
Intel® Core™ i7 processor-based Fanless In-vehicle Surveillance Computer

- Build-in Intel® Core™ i7-2610UE processor 1.5GHz
- 4-channel PoE and dual removal 2.5” SSD tray and CFast storage
- Optional CAN bus in support of SAE J 1939 or J 1708
- Support two SIM card slots
- 9~36V wide range DC power input
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VMC 3501
Vehicle Mount Computer
10.4” Rugged Vehicle Mount Computer with Intel® Core™ i7 processor, Touch Screen and IP65

- Built-in Intel® Core™ i7-2610UE processor. Compact and fanless design
- 10.4” XGA TFT LCD monitor with optional sunlight readable solution with 1000nits
- On screen F1~F10 function key
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 8~60V
- Optional CAN Bus with J 1939 and 1748 protocol
- Robust design with Die-cast aluminum front panel

www.nexcom.com

NEXCOM USA
2883 Bayview Drive, Fremont CA 94538

Empowering the Mobile Workforce
Go the distance with your solutions NEXCOM takes your there

For more information, please contact us at sales@nexcom.com or (510) 656-2248. You may also visit us at www.NEXCOM.com

VTC 1000
In-Vehicle Computer
Intel® Atom™ processor E640-based Fanless In-vehicle Computer

- Compact/fanless design
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 6~36V
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VTC 7110-C4
In-Vehicle Computer
Intel® Core™ i7 processor-based Fanless In-vehicle Surveillance Computer

- Build-in Intel® Core™ i7-2610UE processor 1.5GHz
- 4-channel PoE and dual removal 2.5” SSD tray and CFast storage
- Optional CAN bus in support of SAE J 1939 or J 1708
- Support two SIM card slots
- 9~36V wide range DC power input
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VMC 3501
Vehicle Mount Computer
10.4” Rugged Vehicle Mount Computer with Intel® Core™ i7 processor, Touch Screen and IP65

- Built-in Intel® Core™ i7-2610UE processor. Compact and fanless design
- 10.4” XGA TFT LCD monitor with optional sunlight readable solution with 1000nits
- On screen F1~F10 function key
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 8~60V
- Optional CAN Bus with J 1939 and 1748 protocol
- Robust design with Die-cast aluminum front panel

www.nexcom.com

NEXCOM USA
2883 Bayview Drive, Fremont CA 94538

Empowering the Mobile Workforce
Go the distance with your solutions NEXCOM takes your there

For more information, please contact us at sales@nexcom.com or (510) 656-2248. You may also visit us at www.NEXCOM.com

VTC 1000
In-Vehicle Computer
Intel® Atom™ processor E640-based Fanless In-vehicle Computer

- Compact/fanless design
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 6~36V
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VTC 7110-C4
In-Vehicle Computer
Intel® Core™ i7 processor-based Fanless In-vehicle Surveillance Computer

- Build-in Intel® Core™ i7-2610UE processor 1.5GHz
- 4-channel PoE and dual removal 2.5” SSD tray and CFast storage
- Optional CAN bus in support of SAE J 1939 or J 1708
- Support two SIM card slots
- 9~36V wide range DC power input
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VMC 3501
Vehicle Mount Computer
10.4” Rugged Vehicle Mount Computer with Intel® Core™ i7 processor, Touch Screen and IP65

- Built-in Intel® Core™ i7-2610UE processor. Compact and fanless design
- 10.4” XGA TFT LCD monitor with optional sunlight readable solution with 1000nits
- On screen F1~F10 function key
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 8~60V
- Optional CAN Bus with J 1939 and 1748 protocol
- Robust design with Die-cast aluminum front panel

www.nexcom.com

NEXCOM USA
2883 Bayview Drive, Fremont CA 94538

Empowering the Mobile Workforce
Go the distance with your solutions NEXCOM takes your there

For more information, please contact us at sales@nexcom.com or (510) 656-2248. You may also visit us at www.NEXCOM.com

VTC 1000
In-Vehicle Computer
Intel® Atom™ processor E640-based Fanless In-vehicle Computer

- Compact/fanless design
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 6~36V
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VTC 7110-C4
In-Vehicle Computer
Intel® Core™ i7 processor-based Fanless In-vehicle Surveillance Computer

- Build-in Intel® Core™ i7-2610UE processor 1.5GHz
- 4-channel PoE and dual removal 2.5” SSD tray and CFast storage
- Optional CAN bus in support of SAE J 1939 or J 1708
- Support two SIM card slots
- 9~36V wide range DC power input
- Smarter ignition power on/off, delay-time and low voltage protection
- Certified by CE/ FCC/ e13 Mark

VMC 3501
Vehicle Mount Computer
10.4” Rugged Vehicle Mount Computer with Intel® Core™ i7 processor, Touch Screen and IP65

- Built-in Intel® Core™ i7-2610UE processor. Compact and fanless design
- 10.4” XGA TFT LCD monitor with optional sunlight readable solution with 1000nits
- On screen F1~F10 function key
- Built-in GPS receiver with option 3G/4G LTE and WLAN connectivity
- Wake on RTC/ SMS via WWAN Module (option)
- Wide range DC input from 8~60V
- Optional CAN Bus with J 1939 and 1748 protocol
- Robust design with Die-cast aluminum front panel

www.nexcom.com

NEXCOM USA
2883 Bayview Drive, Fremont CA 94538

Empowering the Mobile Workforce
Go the distance with your solutions NEXCOM takes your there

For more information, please contact us at sales@nexcom.com or (510) 656-2248. You may also visit us at www.NEXCOM.com
conga-TS77

- 3rd Generation Intel® Core™ processor-based platform
- COM Express® Type 6 Module with PCI Express®, SATA, USB, 3x HDMI / DisplayPort
- Improved Graphics Performance, DirectX® 11
Chinese Embedded Design Contest Offers Insight

By John Blyler, Editorial Director

Reviewing the list of first-, second, and third-place winners reveals the technology direction of China’s university/industrial embedded-development community.

In recent years, many have speculated about the trends in China’s home-grown technology and related global-patent problems. Few real insights have emerged. According to the often-cited 2011 Financial Times Alphaville report:

“... we suspect that these data mostly just tell the typical story of China’s rise up the technology value chain and its use of industrial policy to accelerate growth on the back of already-existing technologies.”

Perhaps another way to gauge the direction of China’s internal technology is by looking at designs coming out of the country’s university-industrial complex.

Although admittedly biased, one place to start is the annual Intel Cup Undergraduate Electronic Design Contest – Embedded System Design Invitational. The bi-yearly event was initiated by the Chinese government, hosted by Shanghai Jiao Tong University, and has been solely sponsored by Intel Corp. since 2002.

The contest provides an opportunity for undergraduate students to design a working system based on an assigned Intel embedded platform over a period of three months. Each team consists of three members and a faculty mentor.

This year’s winning project was a Chinese sign-language translation system that helps deaf individuals communicate with the hearing world.

What do the runner-up projects tell us about the direction of Chinese technology? Unfortunately, not much. As you can see from the list of first-, second-, and third-place winners (see Appendix), most designs seem typical of embedded projects in the U.S. and Europe. Perhaps more telling was the interesting wording and specific topical focus of various projects, such as the following:

- Fresh Food Every Day from Intel ATOM-Processor Icebox
- Fairy in the car
- Prison On Fire – A monitoring system based on the “Internet of things” and video-analysis technology
- Happy Chess Player

The list of university-Intel partnered projects seems little different from similar contests held in other parts of the world. This seems to confirm the findings of the earlier analysis by the Financial Times.

Still, this similarity verifies the universal importance of embedded applications in the medical, automotive, and consumer markets. For semiconductor intellectual-property (IP) system-on-a-chip (SoC) designers, this emphasizes the importance of designing chips that easily integrate with board-level hardware and software. The march toward system-level design (e.g., Cadence’s EDA360 approach) continues!

John Blyler is the editorial director of Extension Media, which publishes Chip Design and Embedded Intel® Solutions magazine, plus over 36 EECatalog Engineers’ Guides in vertical market areas.
Intel® Atom™ Processor
Small, Fanless
Embedded PCs

For your next design consider
WinSystems’ single board computers
powered with a single- or dual-core
Intel® Atom™ processor. Our Industry
Standards-based SBCs have a wealth
of onboard I/O, plus PC/104, SUMIT,
and MiniPCI(e) expansion connectors.

► Industry Standard Platforms
  • EPIC – 4.5 x 6.5 inches
  • EBX – 5.75 x 8.00 inches
  • PC/104 – 3.6 x 3.8 inches
  • SUMIT-ISM™ – 3.6 x 3.8 inches

► Software Support
  • Windows® CE, XPe, and WES7
  • Linux
  • x86-compatible RTOS
  • Quick Start Development Kits

► Stackable Input/Output Modules
  • A/D and D/A
  • Digital and Serial

► Accessories Include
  • Cables and Adapters
  • Power Supplies and Memory

► Communication Expansion
  • 802.11 a/b/g Wireless
  • 10/100/1000 Mbps Ethernet
  • Supports MiniPCI and MiniPCI(e)

► Long-life Product Availability

► Extended Temperature Operation

Our SBCs are the right choice for industrial,
pipeline, transportation, instrumentation, medical,
and MIL/COTS applications. Call us today.

Call 817-274-7553 or
Visit WinSystems.com/AtomEIS

Ask about our 30-day product evaluation

WinSystems®
An Employee-Owned Company

715 Stadium Drive • Arlington, Texas 76011
Phone 817-274-7553 • FAX 817-548-1358
E-mail: info@winsystems.com
2 Chinese Embedded Design Contest Offers Insight
By John Blyler, Editorial Director

6 Product News
By Jennifer Burkhardt

28 Automotive Ethernet Consortium takes 100Mbps Ethernet to the Street
Consortium’s objective is to help producers of BroadR-Reach devices and auto manufacturers bring their products to market faster, while ensuring there is broad market appeal devices through interoperability and conformance testing and demonstrating product readiness.
By Cheryl Coupé, Editor

31 Deep Inside Intel
Semiconductor Manufacturing & Design sat down with Mark Bohr, senior fellow at Intel, to talk about a wide range of manufacturing and design issues Intel is wrestling with at advanced nodes—and just how far the road map now extends.
By Ed Sperling, Contributing Editor

10 Performance–Portable Programming
The OpenCL Language and ecosystem could be a lot more portable than it is today. The idea of a static OpenCL runtime couple with and OpenCL–to–C translator would make OpenCL code theoretically runnable anywhere C and C++ can be compiled for.
By John Stratton, MulticoreWare

24 Improve Real–Time Linux Behavior on Embedded Multicore Devices
Linux is the prevalent operating system choice for the new real-time platforms; but, standard Linux is designed for overall throughput rather than for real time, and consequently needs to be modified or extended to meet high demands on both latency and determinism.
By Michael Christofferson, Enea

15 Message Passing is the Future of Multicore Programming
The Multicore Communications API (MCAPI) was designed to address some of the most common challenges in multicore programming, providing a platform-agnostic standard API for sending and receiving messages between cores. Combined with the right tools, it provides a powerful solution for multicore programming.
By Tamer Abuelata, PolyCore Software

19 Analyze Automotive PCB Layouts Efficiently with Simulation
Integrating electronic systems safely into a vehicle poses significant challenges that can be lessened with the help of rule-checking and EM simulation software to reduce the number of design iterations required, by pointing out SI, PI and EMC problems before development proceeds to the testing stage.
By Richard Sjiariel, CST

40 Security in the Connected Car
By Franz Walkenbach, Wind River

35 ADL Embedded Solutions Inc.
36 AXIOMTEK
37 COMMELL
37 DFI-ITOX, LLC
38 Emerson Network Power
39 MSI Computer
39 X–ES

Cover Image: At CES 2012, Lenovo’s team of DO Devils ride on top of taxis and limos driving down Las Vegas Boulevard and surrounding the Las Vegas Convention Center area, while using their Lenovo IdeaPad U300s Ultrabooks™ inspired by Intel technology. (Photo courtesy of Intel®)
Portwell’s flexible business model and optimized SFF boards & systems have helped numerous Medical, Networking, and Industrial OEMs reduce costs, save space, accelerate development schedules and extend system longevity. Contact us to learn how we can tailor our embedded solution offering to your exact needs!

- Small footprints to save space
- 120x120mm or 102x146mm
- Low power consumption
- Single-/dual-core Intel® Atom™ and Intel® Core™ processors
- Extended temp: -40 to 80°C with Intel® Atom™ processor Z510PT Z520PT
- On board dual display, GbE LAN, PCIe, IDE or SATA and more

- Compact turnkey solutions
- Fanless system with Intel® Atom™ or ULV Intel® Core™ processors
- All-in-one with LCD and touchscreen
- Ideal for digital signage, telematics, industrial automation, control and communications
- Expansion I/O options

- Wide selection of COM Express and Qseven modules
- The latest Intel® Core™ i5/i7 and single-/dual-core Intel® Atom™ processors
- Custom carrier board design and manufacturing
- Quick time-to-market solutions
- Extended temp and ECC memory support

Portwell’s extensive product portfolio includes single-board computers, embedded computers, specialty computer platforms, rackmount computers, communication appliances, and human-machine interfaces.

We provide both off-the-shelf and versatile custom solutions for applications in the medical equipment, factory automation, retail automation, semiconductor equipment, financial automation, mission critical and network security markets.

AAEON Releases NanoCOM-CV Rev.A COM Express Type 1 Module for Thermal Critical and Computing Intensive Designs

AAEON has released the new NanoCOM-CV Rev.A COM Express Type 1 module with low power BGA type Intel® Atom™ N2600 1.6GHz Dual Core processor. Complimented by the low power Intel® NM10 chipset, the NanoCOM-CV Rev.A module boasts wider temperature tolerance and good performance, making it suitable for small form factor designs in thermal critical applications such as parking systems and home automation or computing intensive applications for medical, gaming and industrial automation. A wide temperature version for harsh environments is available upon request.

ADLINK Technology Announces COM Express® Type 6 Module with High-performance Graphics and Three Independent Display Interfaces

The Express-IB is accompanied by extensive tools to accelerate development time of applications with advanced processing and graphics requirements.

ADLINK Technology, Inc. presents its latest COM Express® offering, the Express-IB. The Express-IB is a high performance COM0 R2.0 Type 6 module featuring an Intel® Core™ i7/5/i3 processor supporting Intel® HD Graphics integrated on the CPU with three independent displays. A PCI Express x16 Generation 3.0 bus is available for discrete graphics expansion or general purpose PCIe (optionally configure as 2 x8 or 1 x8 + 2 x4).

The Express-IB targets applications in Government, Military, Medical, Digital Signage and Communications and is ideal for customers with advanced processing performance and graphics requirements looking to reduce development time by outsourcing the base design of their system and focusing on application functionality. The Express-IB supports Intel® Advance Vector Extensions (Intel® AVX v1.0), with its improved Floating Point Intensive Applications, and also offers the benefits of increased bandwidth provided by USB 3.0.

Intel Delivers Broad Range of New Mobile Experiences

New Smartphones, Tablets and Ultrabook™ Convertibles to Redefine Mobile Computing

Intel Corporation executives held a press conference to outline a plan to accelerate new mobile device experiences across the company’s growing portfolio of smartphone, tablet and Ultrabook™ offerings.

The announcements included a new smartphone platform for emerging markets, details on a forthcoming 22nm quad-core SoC for tablets, and more personal and intuitive Ultrabook™ devices in innovative convertible designs were outlined by Mike Bell, vice president and general manager of the Mobile and Communications Group, and Kirk Skaugen, vice president and general manager of the PC Client Group at Intel.

DFI EC800 Palm-Size Fanless Embedded System Targets In-Vehicle Application

DFI® has launched the industrial grade EC800, DFI’s first ultra-compact palm-size fanless embedded system. Powered by the Intel® Atom™ processor, its low power consumption design offers energy-efficient performance ideal for the embedded computing market.

AAEON Introduces AIS-Series Advanced Embedded Servers to Power Modern Day Intelligent Automation Systems

AAEON has released two compact and powerful Embedded Servers, the AIS-E1 and AIS-E2 ideal solutions for Building Automation, Factory Automation, IP Surveillance, advanced Digital Signage and Kiosks. Designed with upgradable socket type multi-core Intel® Core™ iSeries processors under 45W, these embedded servers offer superb performance and multiple high definition video playbacks, while consuming very little power.

Axiomtek Rolls Out PICO830, Intel® Atom™ Processors N2800/N2600 Powered Pico-ITX SBC

Axiomtek released PICO830, its new Pico-ITX SBC based on the low-power Intel® Atom™ dual core processor N2800 1.86 GHz and N2600 1.6 GHz with the Intel® NM10 Express chipset. The system memory on PICO830 can support either 2 GB or 4 GB
of DDR3, depending on the processor. It comes with two different configurations of display interfaces: DisplayPort and 24 bit dual channel LVDS or VGA and 24 bit dual channel LVDS. The embedded board also includes a ready-to-use pin-header for I/O expansion which can help system integrators to develop solutions quick and cost-effectiveness. For convenience, we offer AX93265, the I/O board specially designed for PICO830. Additionally, it requires only +5V DC supply input into work. This extreme-compact embedded board offers an excellent solution for in-vehicle PCs, medical imaging, gaming, in-flight entertainment systems, industrial automation systems, and the portable devices.

**NEXCOM Expands NDiS Family with New Flagship Digital Signage Player M532**

NEXCOM expands its digital signage family with a new flagship digital signage modular player – the NDiS M532. The M532 is based on the 3rd generation Intel® Core™ processor family and follows the electrical and mechanical specifications of the Open Pluggable Specification (OPS). The M532 can be plugged into any OPS-compliant display devices to render rich multimedia contents. Thanks to the modular and cable-less design and advanced built-in remote management function, the M532 satisfies the need for quick deployment and hassle-free maintenance of large digital signage network dispersed in different geographical locations.

**Portwell’s New 1U Rackmount Network Appliance Combines Entry-Level Cost With Server-Grade Performance**

New CAR-3020 series supports Intel(r) Sandy Bridge-Gladden processor and latest Intel Cave Creek Communications 89xx series chipset

CAR-3020, the latest offering from American Portwell Technology, Inc. (http://www.portwell.com), is a compact 1U rackmount network security appliance that provides support for a variety of network interface cards, built-in security functions-including bulk encryption/decryption-high flexibility and high performance. The engine that drives the new CAR-3020 is based on the Intel Crystal Forest-Gladden platform of the Intel BGA1284 type processors from Celeron(r) to Xeon(r) (codenamed Sandy Bridge-Gladden) and Intel Communications Chipset 89xx series (codenamed Cave Creek).

**Mercury Systems Announces Its New InfiniBand-Based Product for OpenVPX High-Performance Embedded Computing Solutions**

Industry’s first embedded processing module using Intel’s powerful third-generation processors and dual host adapters for InfiniBand fabric support

Mercury Systems, Inc. has expanded its product line to offer the industry’s first embedded processing module using the powerful Intel® 3rd generation Core™ i7 quad-core Ivy Bridge mobile-class processor and dual Mellanox® ConnectX®-3 host adapters for a total of four InfiniBand™ fabric connections. The new LDS6523 (low-density server) is a new industry open architecture high-performance embedded computing solutions, offering unparalleled data plane bandwidth with four 40Gbps fabric ports. The product can be configured to support Double Data Rate, Quad Data Rate and 40 GigE speeds. Solutions based on the LDS6523 are perfectly suited for multi-dimensional applications requiring high throughput, determinism and low latency — such as CyberINT, IMINT, SIGINT and radar.

**ADL Embedded Solutions Announces ADLN2000PC – Intel® Atom™ N2600 Dual Core, 1.6 GHz, PCIe/104 SBC**

ADL Embedded Solutions has announced the ADLN2000PC PCIe/104 SBC with a shipment date of Q4 2012. The ADLN2000PC features the Intel® Atom™ N2600 processor with integrated HD graphics engine and memory controller functions. The processor interfaces to the ICH9M-E, is also used on the ADLD25PC, but provides the PCIe/104 I/O bandwidth (2.5GT/s) necessary to enable performance-based rugged, portable or thermally constrained applications. The ADLN2000PC has an Intel rated Thermal Design Power (TDP) maximum of only 3.5 Watts, yet has enhanced graphics including dual-channel video capability at full 1080p with full MPEG2 (VLD/iDCT/MC) and HW decode/acceleration for MPEG4 (AVC/H.264). The ADLN2000PC will be demonstrated at Embedded World 2013.

**X-ES Introduces Intel® Core™ i7 Processor-Based Small Form Factor System**

Extreme Engineering Solutions (X-ES) introduces the XPedite7450 Small Form Factor (SFF) system supporting a 3rd generation Intel® Core™i7 COM Express® module. With commercial connectors, this SFF system can be utilized as an XPedite7450 COM Express development platform, a deployable SFF system, or a demo platform for applications requiring a high-performance Intel Core i7 processor solution.

**High-End 3rd Gen. Intel® Core™ Mini ITX with Q77 chipset, Rich I/O and HDMI – MANO871**

Axiomtek is excited about the launch of MANO871 which is our newest 3rd generation Intel® Core™ processors based Mini ITX solution. This industrial grade Mini ITX motherboard is an ideal platform for developing embedded and industrial computing solution since it provides fast and smart capabilities by utilizing Intel® 3rd generation Intel® Core™ processor as well as having several new features. In addition to having a rich performance, the MANO871 features Intel® HD 2000/3000/4000 Graphics for high-definition display resolution. Having a small footprint of only 170 mm x 170 mm and powerful performance enables system developers to design extremely high-end applications.
**conga-TS77**

- Highest Performance COM Express® Type 6
- 3rd Generation Intel® Core™ processor-based platform
- Better transcode HD-HD, HD Video Conferencing
- Improved Graphics Performance, DirectX®11

<table>
<thead>
<tr>
<th>Formfactor</th>
<th>Formfactor COM Express™ Basic, (95 x 125 mm), Type 6 Connector Layout</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Intel® Core™ i7-3615QE processor (4x 2.3 GHz, 6MB L2 cache, TDP 45W)</td>
</tr>
<tr>
<td></td>
<td>Intel® Core™ i7-3612QE processor (4x 2.1 GHz, 6MB L2 cache, TDP 35W)</td>
</tr>
<tr>
<td></td>
<td>Intel® Core™ i7-3555LE processor (2x 2.5 GHz, 4MB L2 cache, TDP 25W)</td>
</tr>
<tr>
<td></td>
<td>Intel® Core™ i7-3517UE processor (2x 1.7 GHz, 4MB L2 cache, TDP 17W)</td>
</tr>
<tr>
<td></td>
<td>Intel® Core™ i5-3610ME processor (2x 2.7 GHz, 3MB L2 cache, TDP 35W)</td>
</tr>
<tr>
<td></td>
<td>Intel® Turbo Boost Technology 2.0, Intel® Hyper-Threading Technology</td>
</tr>
<tr>
<td>DRAM</td>
<td>2 Sockets, SO-DIMM DDR3 up to 1600MT/s and 16 GByte</td>
</tr>
<tr>
<td>Chipset</td>
<td>Mobile Intel® 7 Series chipset: Mobile Intel® QM77 Express chipset</td>
</tr>
<tr>
<td>Ethernet</td>
<td>Intel® 82579 Gigabit Ethernet PHY with AMT 8.0 support</td>
</tr>
<tr>
<td>I/O Interfaces</td>
<td>7x PCI Express™ GEN. 2.0 lanes, 1x PEG, 4x Serial ATA® with 6 Gb/s, 2x Serial ATA® with 3 Gb/s (AHCI) RAID D1/5/10 support, 2x ExpressCard®, 4x USB 3.0 (XHCI), 8x USB 2.0 (EHCI), LPC bus, PIC bus (fast mode, 400 kHz, multi-master)</td>
</tr>
<tr>
<td>Sound</td>
<td>Digital High Definition Audio Interface with support for multiple audio codecs</td>
</tr>
<tr>
<td>Graphics</td>
<td>Intel® Flexible Display Interface (FDI), OpenCL 1.1, OpenGL 3.1 and DirectX11 support, Three simultaneous independent displays, High performance hardware MPEG-2 decoding, WMV9 (VC-1) and H.264 (AVC) support Blu-ray support @ 40 MBit/s, hardware motion compensation</td>
</tr>
<tr>
<td>LVDS</td>
<td>Dual channel LVDS transmitter, Supports flat panels 2x24 Bit Interface, VESA mappings, resolutions up to 1920x1200, Automatic Panel Detection via EDID/EPI</td>
</tr>
<tr>
<td>Digital Display Interface (DDI)</td>
<td>1x SDVO / DisplayPort 1.1 / TMDS (DVI, HDMI)</td>
</tr>
<tr>
<td></td>
<td>2x DisplayPort 1.1 / TMDS (DVI, HDMI)</td>
</tr>
<tr>
<td>CRT Interface</td>
<td>350 MHz RAMDAC, resolutions up to QXGA (2048x1536 @75Hz)</td>
</tr>
<tr>
<td>congatec Board Controller</td>
<td>Multi Stage Watchdog, non-volatile User Data Storage, Manufacturing and Board Information, Board Statistics, BIOS Setup Data Backup, PIC bus (fast mode, 400 kHz, multi-master), Power Loss Control</td>
</tr>
<tr>
<td>Embedded BIOS Features</td>
<td>AMI Aptio® UEFI 2.x firmware, 8 MByte serial SPI firmware flash</td>
</tr>
<tr>
<td>Security</td>
<td>The conga-TS77 can be optionally equipped with a discrete “Trusted Platform Module” (TPM). It is capable of calculating efficient hash and RSA algorithms with key lengths up to 2,048 bits and includes a real random number generator. Security sensitive applications such as gaming and e-commerce will benefit also with improved authentication, integrity and confidence levels.</td>
</tr>
<tr>
<td>Power Management</td>
<td>ACPI 4.0 with battery support</td>
</tr>
<tr>
<td>Operating Systems</td>
<td>Microsoft® Windows7, Microsoft® Windows XP, Microsoft® Windows® embedded Standard, Embedded POS Ready (WEPOS), Linux 3.0</td>
</tr>
<tr>
<td>Power Consumption</td>
<td>Typ. application: tbd., see manual for full details, CMOS Battery Backup</td>
</tr>
<tr>
<td>Temperature:</td>
<td>Operating: 0 .. + 60°C Storage: -20 .. + 80°C</td>
</tr>
<tr>
<td>Humidity Operating:</td>
<td>Operating: 10 - 90% r. H. non cond., Storage: 5 - 95% r. H. non cond.</td>
</tr>
<tr>
<td>Size</td>
<td>95 x 125 mm (3.74” x 4.92”)</td>
</tr>
</tbody>
</table>
Performance-Portable Programming

The OpenCL language and ecosystem could be a lot more portable than it is today. The idea of a static OpenCL runtime coupled with an OpenCL-to-C translator would make OpenCL code theoretically runnable anywhere C and C++ can be compiled for.

By John Stratton, MulticoreWare

As microprocessors became more and more prevalent, the challenge of portability became critical. The C programming language became popular in large part because it provided the right programming mechanisms for matching common programming patterns, such as counted loops, to architecture-specific mechanisms for handling those constructs. It’s still true that many of the most highly tuned libraries are written in assembly code or intrinsics and tuned to specific pieces of hardware. But the vast majority of programmers do not write assembly code anymore; even the ones that really care about performance. The typical performance-minded programmer does take into account both algorithmic complexity and general architecture principles such as locality, stride-one memory accesses in loops and avoiding control flow in innermost loops. Today, practically the first tools available for any new processor are an assembler and optimizing C compiler.

Now, parallel computing hardware is becoming increasingly diverse and attractive. Multicore CPUs, GPUs, DSPs and many other architectures are becoming more and more parallel. The change has become so overwhelming that, for possibly the first time in decades, many people are considering rewriting working code to be scalably parallel, and portable across parallel architectures.

Accepting that working code can be rewritten opens up huge opportunities for new parallel programming models, which explains why so many have sprung up in the last few years. But the existing landscape is largely segmented. Want to run parallel code on an NVIDIA GPU? Write it in CUDA, and decompose your code into thousands of tiny threads. Want to run parallel code on multicore CPUs? Write it in OpenMP and make sure it scales to a few dozen traditional CPU threads. The only language that is widely touted as being applicable for a wide variety of parallel processors today is OpenCL.

Figure 1: Typical OpenCL software framework.

The only language that is widely touted as being applicable for a wide variety of parallel processors today is OpenCL.

Figure 1 shows how OpenCL is implemented in practice. The OpenCL application code links against a system library defined and provided by the Khronos OpenCL standardization committee. When the application runs on a system, it will locate that shared library, which will use some mechanism to discover registered vendor OpenCL platforms on the system. The application code therefore gains visibility into every available OpenCL stack, including the platform interface implementing the OpenCL API for that platform. That interface defines methods for invoking that platform’s compiler on kernel code, and for managing the execution of those kernels. The goal is that OpenCL application code, at runtime, can find vendor support for OpenCL on each particular system, making the application itself portable among all systems implementing OpenCL.

Yet the OpenCL ecosystem today does not actually provide portability to the extent many software publishers desire. First, OpenCL is only portable across supported architectures and systems. OpenCL was not installed by default on the majority of currently operating consumer x86 devices, adding a new library dependency to any applications intending to use it. Those plan-
Featuring Battery Backup Power (BBP™) module

**BBP™ Key Benefits:**
- Increases overall data center power efficiency
- Hot-swappable and easily deployed
- Advanced server design incorporates UPS protection without sacrificing performance or functionality
- Frees UPS space to increase server density deployment
- More cost-effective and higher efficiency than using traditional UPS

**Supermicro BBP™ Module**
Available in 200W and 1000W battery pack capacities, these new redundant, BBP™ (battery backup) modules strengthen Supermicro server and storage solutions with the most robust protection against power disruption, eliminating power delivery as a single point of failure. Using less than 1W at 99.9% power efficiency to maintain a full charge, the system lowers power consumption with a cost-effective infrastructure solution and saves $2k/rack/year when compared to single rack 20KVA online UPS systems running at 95% efficiency. The Supermicro BBP™ module status information is conveniently monitored and managed via IPMI.

* Supports Intel® Xeon® Processor E5-2600 product family

© Super Micro Computer, Inc. Specifications subject to change without notice. Intel®, the Intel® logo, Xeon®, and Xeon® Inside, are trademarks or registered trademarks of Intel Corporation in the US and/or other countries. All other brands and names are the property of their respective owners.

www.embeddedintel.com
The OpenCL language and ecosystem could be a lot more portable than it is today.

Regarding using an OpenCL from Intel stack are limited to CPUs at least as recent as the Intel® Core™ i7 processors. The AMD OpenCL stack is somewhat more general, but even it struggles to provide more than functional portability.

Although OpenCL is a functionally portable standard, people can only write performance-portable code if the different implementers of OpenCL agree on a set of best programming practices for good performance. Unfortunately, that agreement does not exist today. The GPU vendors teach developers to create thousands of work-items to fill their hardware thread contexts, and avoid divergence among work-items in a work-group, because that’s where your SIMD execution really comes from. To help control locality, GPU vendors promise lightning-fast, hardware-implemented barriers between work-items in a group. Yet on AMD’s CPU implementation, divergence among work-items ends up being irrelevant, because it won’t execute multiple work-items in SIMD anyway. Plus, the cost of a barrier is at least a dozen instructions per work-item per executed barrier, which fundamentally changes what use cases a barrier can reasonably be used for.

So there are two major problems that need to be solved. The first is that OpenCL is currently distributed as a system runtime library, currently not present by default on many consumer platforms. The second is that current OpenCL implementations on x86 CPUs are either insufficiently portable (Intel) or suffer in performance if given the same code that works well on a GPU (AMD). We need a solution that will enable both broad portability and a single-source, high-performance programming environment compatible with GPU acceleration and parallel CPU execution.

To solve both problems, we need a way of making the OpenCL code, both the runtime and the kernels, just another compiled module of the application, as shown in Figure 2. An OpenCL API implementation could be statically linked against the main application to mediate the standard OpenCL API calls to the precompiled kernels. To be truly portable, it would also have to be able to dynamically find the system OpenCL runtime library, and through it, any other vendor libraries that happen to be available on the system. A static OpenCL runtime with that capability would provide the same access to GPU-accelerated OpenCL implementations, for instance, but would still run on a platform without any preinstalled OpenCL implementation.

The OpenCL kernels themselves are difficult to manage portably. It would be unlikely for one company’s OpenCL runtime product to support all the different architecture variants that C can support now, much less be kept up to date. The best situations would be if the OpenCL kernels could be compiled with the same tools used to compile the main application: the target’s C or C++ compiler. An OpenCL-to-C translator for CPUs would enable this possibility, and would be generically reusable for a wide variety of architectures.

Translation from OpenCL to multithreaded C is difficult to get right. Several academic papers have been published on the topic, but the core concept is that the many small work-items of the OpenCL work groups have to be merged into a single CPU software thread. A CPU thread has too much overhead in creation and scheduling to be suitable for a typically tiny, individual OpenCL work-item. By serializing many work items, wrapping regions of the kernel code in loops over work-item indexes, as shown in Figure 3, the serialization process effectively replaces implicitly declared local work-item indexes with explicitly enumerated loop iterations. Serialization increases the task granularity to a degree much more suitable to a more coarsely threaded CPU architecture design. Figure 3 shows how care must be taken to...
note the placement of barriers in the original OpenCL code, and obey the ordering constraints imposed by those barriers. What is labeled as the second region overwrites input used in the first region. The barrier is inserted to ensure all the input has been consumed before any of it is overwritten. The barriers essentially define regions of code that can safely be serialized, while separate regions must be divided such that a region after a barrier only executes after all operations before the barrier for all work-items are completed. The technique is even generalizable to cases where barriers are inside other control flow constructs. Once the barrier-ordering constraints have been applied to the serialized code, the barrier itself has no function it must perform, and can be removed.

Once the kernels are in this translated, serialized C-code format, they can be considered just additional source files of the application and compiled using the same toolchains. This makes the translator forward-compatible, as updated C compilers are provided with every new architecture release from most companies. A suitable runtime for interfacing the OpenCL API to those precompiled C kernels would essentially treat each kernel as a plug-in identifiable by kernel name, using techniques such as LLVM’s compiler plugin mechanisms.

In conclusion, the OpenCL language and ecosystem could be a lot more portable than it is today. The idea of a static OpenCL runtime coupled with an OpenCL-to-C translator would make OpenCL code theoretically runnable anywhere C and C++ can be compiled for. This is the vision adopted by MulticoreWare in the Multicore cross-Platform Architecture (MxPA) product line. The goal is to move OpenCL away from being the duplicate, fast codepath that’s only used when it works, to being the only implementation an application needs for its data-parallel kernels.

* John A. Stratton is a senior architect for MulticoreWare, Inc. John has been teaching GPU computing since the first university course on the subject in spring 2007, and developing compilers and runtimes for kernel-based accelerated programming models since that year. He has received several awards for outstanding research, teaching and technology development, most recently given the “Most Valuable Entrepreneurial Leadership in a Startup” award by the University of Illinois Research Park for his work with MulticoreWare. 

Designing with Intel® Embedded Processors?

Embedded Intel® Solutions delivers in-depth product, technology and design information to engineers and embedded developers who design with Intel® Embedded processors.

Visit www.embeddedintel.com

Subscribe Today at www.embeddedintel.com Free!
Message Passing is the Future of Multicore Programming

The Multicore Communications API (MCAPI) was designed to address some of the most common challenges in multicore programming, providing a platform-agnostic standard API for sending and receiving messages between cores. Combined with the right tools, it provides a powerful solution for multicore programming.

The Problem

Today, a multitude of multicore chips are available for every compute application: embedded, desktop and mobile processors, DSPs, hardware accelerators, FPGAs. New chips are released every day, and with more and more features.

But how do I write my application code to take advantage of these powerful devices? And how do I avoid re-writing my code when I need to design faster, less-costly or less power-hungry silicon alternatives?

The problem becomes even more complex when the systems are running different operating systems. For example, a single device may contain an ARM processor running Linux and a TI DSP running DSPBIOS or no OS (“bare metal”). What communication should be used between cores on different operating systems, and on different chips? And how does the programmer prepare the application code to hide the platform specifics?

To add to the fun, consider the challenge of mixing and matching various transports to optimize data movement (shared memory, DMA, Rapid IO, serial, PCIe, TCP/IP). Should my application logic be concerned with this?

Fortunately, there are solutions out there, ready to tackle my questions head-on, and helping me to take advantage of the innovation in these devices.

What would make this easier?

Ideally, I would just push a magic button and transform code into a fully optimized multicore application. Unfortunately, no one has found that magic button yet. However, there are solutions that exist today that can definitely make multicore life easier.

Obviously, this challenge calls for a suitable programming method. If I want to simplify the programming of multicore systems, I need a method that can span multiple boards, multiple chips, multiple operating systems and multiple transports and scale to many cores. I also need to abstract the details of the underlying hardware from the application code, in order to address portability and code re-use and reduce time-to-market.

SMP has been a major player in the multicore world. It scales decently on desktop computers with two, four or even eight cores, running unrelated applications. However, past a certain point SMP simply doesn’t scale because it relies on shared resources. SMP also cannot be of use in heterogeneous environments. How-

Figure 1: An example of a heterogeneous topology spanning multiple types of operating systems and different types of processors.

By Tamer Abuelata, PolyCore Software
Message Passing Has the Answers

Message passing has answers to the questions that I previously raised and has been gaining momentum over the years. It simplifies application programming by scaling well (in theory, infinitely) and can help to address the challenges of heterogeneous environments. In a way, it is similar to email in the sense that when an email is sent, users must only specify a destination email address and hit “send.” I do not need to know how the message will get there and what route will be used. The underlying system and network takes care of transporting the message to its destination, and alerting users when the mission fails. One of the nice things about sending an email is that the user does not need to pay attention to what OS, device or email client the recipient is using. Those variants will not affect how the email is written.

Similarly, it would be nice, from the application programmer’s point of view, to just say, “I want to send a message from core A to core B” and then have an underlying system that would take care of delivering the message, regardless of which OS the destination is running, and regardless of the transport being used to communicate between those two cores. This way, the core type or transport type can later be changed without needing to change application code. The code would say “send this message from A to B.”

Another benefit of message passing is that it inherently provides synchronization. A message only has one owner at any given time. Even if the message doesn’t move, the ownership of the descriptor has a single owner. You can think of it as passing a token around.

Message passing is a well-known, tried-and-tested programming method that is available in most environments locally within an operating system or remotely (e.g., networking). This means that most programmers already know the programming method and require little learning.

The Internet itself is the largest message-passing network. Any device that implements TCP/IP is able to consume or deliver services on the Internet, regardless of hardware or operating system. We could say that the Internet, which for the most part runs on TCP/IP, is a living proof that message passing can scale significantly and easily.

Do Message Passing the Right Way

Standards, such as HTTP, help the Internet become ubiquitous. Similarly, for a multicore application to interoperate in many environments, standards should be used.

Whenever new technologies are adopted, designers should use a careful process of identifying and selecting solutions that will be long-lasting, based on standards and have an active user community. Open standards such as the Multicore Association’s Multicore Communications API (MCAPI) ensure that the application that is built today, will still run tomorrow and will require minimal effort to maintain or port to new platforms in the future. The Multicore Association (MCA) designed MCAPI to address common challenges in multicore communication, targeting closely distributed computing (chip, board, rack). MCAPI is message-passing-based, therefore very scalable and suitable for many-core systems that we can expect in the future. With the belief that message passing is the future of multicore, MCAPI developed a platform-agnostic standard API for sending and receiving messages between cores.
FREE ADMISSIONS! Sign up by February 28th

8th ANNUAL
Multicore Developers Conference
May 21-22, 2013
Hyatt Regency Santa Clara, CA

Resolving the Technical Aspects and Business Challenges of Designing with Heterogeneous and Homogeneous Multicore Processors

**Multicore Use in Application Specific Topics**
- Android
- Automotive
- Cloud computing
- Networking and LTE equipment
- Mil/aero
- Real-time and safety considerations
- Security
- Smart phones, and more

**Evolutionary Topics**
- Manycore
- GPGPU
- Futuristic development and optimization tools, etc.

**Business Topics**
- Predictions
- Trends, etc.

**General Multicore Topics**
- Debug
- Software frameworks
- Parallel programming
- Protocols
- Optimizations, and more
- Choosing application-specific acceleration or multicore technology
- Interconnect technology for scaling from 4, 8, or 16, to 100s of cores

Register at www.multicoredevcon.com
Therefore, in addition to graphical configuration and code-generation tools, it is essential to have a supporting runtime engine – an implementation of MCAPI, optimized for various platforms with support for different transports.

The Solution
To simplify the process of programming for multicore systems, the programming methodology should scale across homogeneous and heterogeneous environments, and support a standard that is both tested and adopted by the programming community. To improve productivity, development tools should be employed, saving time, improving code quality and moving the product to market quickly.

The model should allow for easy migration to other multicore systems, should be based on established standards and should remain flexible and agnostic to specifics like OS or processor. Message passing, together with the combination of graphical tools, code-generation and runtime are just the right prescription, and ensures that the benefits offered by multicore are being fully utilized.

Tamer Abuelata, director of engineering for Poly-Core Software, has been a part of the team since 2008 and has been a major contributor to the growth of Poly-Platform. In 2008, he earned a bachelor of arts in philosophy from San Jose State University, and currently pursues his interest in linguistics through private study.

For the sake of efficiency and flexibility, the programming model should allow the programmer to focus on application logic. The processor-specific, OS-specific and transport-specific code would be written by a systems programmer or, by using modeling tools, generated automatically after a configuration effort. Applications developers have the opportunity to minimize, and even with some specific tools, eliminate that part of the coding effort. Using graphical tools and wizards, the programmer would input the platform characteristics using a friendly user interface. From the configuration files and the model, the platform-specific code would be generated to build the underlying communication infrastructure.

Such tools are desirable for several reasons. First, tools save time by generating consistent, quality code. Consistent, quality code is key when working in large engineering teams that are spread across the globe. Code generated in China or India will be the same as the code that is generated in the U.S. and Germany for instance. These tools can perform validation of the configuration and catch errors before runtime as well as allow for rapid reconfiguration of the application so that porting to other platforms or scaling up or down can be done easily, quickly and without rewriting application code. In addition, MCAPI-compliant code can be generated to ensure that applications are being built on established standards.

Graphical tools help to visualize the overall system, giving developers the chance to examine the big picture and more easily determine changes that should be made in the coming design cycles. Analyses can be made to verify performance, and to identify problem areas that may need to be reconfigured.

Another key element in providing abstraction is having a runtime engine that transparently delivers messages across multiple transports. The runtime is responsible for balancing performance and resource utilization. Performance goals are to maximize throughput, while keeping the runtime overhead to a minimum.

“Mirror, Mira” on the Car’s IVI Screen: Two Different Standards?

By Chris Ciufo, Senior Editor

You might be hearing about a new technology called MirrorLink that mimics your smartphone’s screen on the larger nav screen in your “connected car”. Or, you might be following the news on Miracast, a more open standard now baked into Android that offers Apple AirPlay-like features to stream smartphone content to devices like connected TVs.

You’d be forgiven if you think the two similarly-named standards are trying to accomplish the same thing. I didn’t understand it either, so I did some digging. Here’s what I found out.

Read the complete interview at:
http://eecatalog.com/caciuf0/
Just twenty years ago, automotive electrical systems were relatively simple, and few of their components were computer-controlled. However, recent years have seen an explosion in the field of automotive electronics. Components that were once physically connected to the controls, such as the throttle and the brakes, are increasingly operated by electronic drive-by-wire systems. Computers also constantly monitor the state of the vehicle; as well as replacing traditional mechanical gauges like the speedometer and the odometer, these electronics offer ways of measuring the previously unmeasurable, such as road conditions, lane position and the distance to the vehicle in front. Whole new areas of automotive design have opened up, with communications, infotainment and active-safety systems standard in an increasing number of cars.

PCBs and integrated circuits are now fundamental parts of any new vehicle; however, this new-found complexity comes at a price. Anti-lock brakes, airbags, cruise control, drive-by-wire – and faults in these systems can lead to injury or death. Obviously, for the brakes or the airbags to fail to receive a signal in an accident would be disastrous, but the consequences could be just as bad if a stray signal accidentally triggered them at speed. Keeping vehicles safe, secure and operational means designing for signal integrity (SI), power integrity (PI) and electromagnetic compatibility (EMC) at the most fundamental level, while reducing electromagnetic interference (EMI) from the electronics.

Avoiding SI Problems with Rule Checking
Many factors can play a role in degrading the signal in an electronic system. On a tightly packed PCB, signals can leap from one line to another. This crosstalk can lead to timing problems and false triggering, which can lead to a complete failure of the device.

To reduce effects like crosstalk, there are a number of rules that PCB designers should follow. For example, designers can impose a minimum separation between the signal lines, they can define guard traces between signal lines, and if signal lines are located on adjacent layers, they can ensure the lines are placed perpendicularly.

On a simple circuit board, making sure a design follows these rules is easy enough, but the complexity of a modern PCB with many layers, high bit-rate data buses, multiple I/O devices and very tight dimensions makes simply checking the design by eye
impossible. Even if the engineer only focuses on the most critical parts of the board, such as the data buses or the clock signal lines, the potential for human error is too great. Mistakes that are missed in the design stage will only be noticed once a prototype has been built and tested, and fixing them means going through the design painstakingly looking for the cause of the error, removing it, and then repeating the whole prototyping process. To help cut down on such delays in the design process, the engineer can take advantage of an automated rule-checking program.

The rule-checker automatically applies chosen design rules to any nets the engineer deems important. Parts of the design that violate the rules are pointed out automatically, giving the design engineer the chance to adjust the layout if necessary. With the help of simulation, the time taken to check whether the design complies to SI and EMC rules drops from hours or days to minutes.

**Simulation Instead of Prototyping**

Of course, while following the rules reduces the risk of errors in the signal, it doesn't remove it altogether. Even a well-laid-out PCB can experience problems from radiation effects or unforeseen couplings. Detecting these without using prototypes means moving from simply checking the design to modeling and simulating the board in use.

Simulating a PCB can give the designer a far better idea of its properties and characteristics than the rule-checking process alone would. For example, a variety of digital signals (usually based on the industry-standard IBIS models) can be injected into the simulated PCB to test how well data flows from an input to an output under a wide range of conditions.

Simulations can also generate useful standard outputs such as S-parameters, eye diagrams and impedance profiles from virtual prototypes, providing engineers with a fast overview of how the system behaves and allowing them to replicate common laboratory measurements at the modeling stage.

**Maintaining Power Integrity**

Alongside signal integrity, power integrity is becoming more important as circuit boards get faster and their power consumption gets lower. This makes the electronic circuit elements very sensitive to voltage fluctuations, and even a relatively small fluctuation can be enough to cause a component to trigger falsely. A PCB with a good power-distribution network is one that can provide a stable power source to all the electronic circuit elements, keeping voltage fluctuations low so as not to jeopardize the signal quality.

The classical approach to power integrity simulation, using just the circuit simulator, can still be used for a simple circuit board with one or two power layers at lower speeds. A more complex board with several power layers, however, needs both the circuit simulator and a specialized EM simulator tool. The EM simulator tool helps to analyze the power integrity in two aspects: the DC power drop and the AC power fluctuation caused by high speed current switching.

The latter is trickier to fix, yet on a high-speed PCB these fluctuations will make up the majority of the voltage noise. With AC, capacitance and inductance become important, as do the resonances of the power plane itself, and the best way to improve the board’s AC noise characteristics is to use decoupling capacitors to reduce the board’s impedance. A simulation lets the engineer find where a decoupling capacitor would do most good; the effects of different capacitor placements can be tested without having to build multiple prototypes.
Immunity with shielding and filtering. Even less-critical devices like the infotainment and navigation systems need a thorough EMC analysis, since the fields they radiate can interfere with the operation of the most important systems.

**The Benefits of Modeling and Simulation**

Integrating electronic systems safely into a vehicle poses significant challenges to the engineer. However, these difficulties can be lessened with the help of rule-checking and EM simulation software. Both reduce the number of design iterations required, by pointing out SI, PI and EMC problems before development proceeds to the testing stage. This means shorter development times and fewer costly prototypes, and so simulation can offer a considerable advantage to designers that implement it into their workflow.

Richard Sjariel received his B.Sc. and M.Sc. degrees in electrical engineering from the University of Wuppertal, Germany. He joined CST in 2006 as an application engineer, where his main area of work involves signal/power integrity and EMC/EMI simulation and other high-frequency applications.

Even a well-laid-out PCB can experience problems from radiation effects or unforeseen couplings.

**Promoting Electromagnetic Compatibility**

Interference from the circuit itself is not the only threat to the quality of the signal. PCBs inside cars are bombarded with noise from a variety of sources: stray EM fields from other equipment, interacting in complicated ways by the car body, can induce unwanted currents in the circuits, and the spikes and fluctuations in voltage caused by the car’s ignition system can push the components well outside their working tolerances.

With a 3D simulation, the calculations are not just restricted to the PCB – the simulation can also take into account the PCB housing, the body of the car, and other potential sources of EM fields within the vehicle.

When putting together the vehicle’s electronics, it’s always important to take EMC/EMI into consideration, minimizing the system’s electromagnetic radiation and maximizing its immunity with shielding and filtering. Even less-critical devices like the infotainment and navigation systems need a thorough EMC analysis, since the fields they radiate can interfere with the operation of the most important systems.
For a FREE subscription visit: www.embeddedintel.com
Improve Real-Time Linux Behavior on Embedded Multicore Devices

Linux is the prevalent operating system choice for new real-time platforms; but, standard Linux is designed for overall throughput rather than for real time, and consequently needs to be modified or extended to meet high demands on both latency and determinism.

By Michael Christofferson, Enea

The semiconductor industry is trending towards multicore devices, and as such, the workload on many multicore processors will likely become even more multi-programmed to support a mixture of best-effort, high-throughput control applications together with low-level protocol or data-processing applications with potentially high real-time requirements. The simultaneous demand for high throughput and low latency is difficult to achieve since those characteristics are usually contradictory. So how can Linux adapt?

Real-Time in Operating Systems

A real-time OS must provide a deterministic runtime environment for the entire application execution flow so that it can respond within the specified operational deadline on a system level. This implies that the task scheduler and the resource-handling API in the OS must behave in a deterministic way. When designing a system for high-throughput performance, the goal is to keep down the average latency. But when designing a real-time system, the aim is to keep the worst-case latency under a specified limit. Consequently the design of a both high-performance and real-time-capable system must take both average and maximum latency into account. Historically, “latency” has most often referred to interrupt latency without scheduling (see sidebar). But this interpretation is becoming obsolete. Drivers are today usually implemented as POSIX applications running in user-space threads as they are easier to debug, maintain and isolate from licensing issues compared to code in kernel space. The most important measure is thus the task-response latency, which includes the scheduling latency.

It is well-known that the standard Linux kernel does not provide very deterministic interrupt/scheduling latency behavior, and that under load, there is much jitter. This is the reason for the development of Linux “real-time” extensions. These extensions, often referenced as the Linux kernel preemption patch set, are but one of the approaches analyzed in this article.

Three Approaches to Enable Linux for Real-Time

Figure 1 depicts three basic design approaches for improving standard Linux real-time behavior.

- **Kernel-preemption patch approach:** Standard Linux can be modified with the PREEMPT_RT kernel patch. This can be considered as a “horizontal layering” design approach since it implements a kind of para-virtualization of the interrupt management and the kernel locking mechanisms.

- **Thin-kernel/virtualization approach (minimize or avoid):** Use a thin RTOS kernel layer between the hardware and Linux kernel in order to create a truly isolated virtual machine for Linux. Compared to PREEMPT_RT, this approach is an even more obvious “horizontal layering” design approach since it provides virtualization on hardware level using hypervisor technology.
Vertical-partitioning approach (avoid): A Linux SMP kernel running on a multicore processor can be configured and modified to provide resource isolation between two sets of cores; one dedicated for real-time applications and the other one for non-real-time applications. Per-core scheduling becomes less important while inter-core communication needs to be more efficient and deterministic.

Kernel-Preemption Patch Approach: PREEMPT_RT
The PREEMPT_RT patch provides several modifications for real-time support in the Linux kernel by mitigating the effects of resource conflicts. This is the well-known approach for adding real-time capabilities to the Linux kernel. The PREEMPT_RT patch was originally developed by Ingo Molnar and his team, and is maintained today by Thomas Gleixner among others. Most of the patches in this set have been contributed to the main line, but the kernel must be explicitly built for PREEMPT_RT.

The changes in this patch set include re-implementing some kernel locking mechanisms to enable full preemption where otherwise the Linux design states a non-preemptible region. For example, regular spinlocks are replaced with mutexes with priority inheritance, thus many interrupt handlers need to be migrated into kernel threads to become fully preemptible. In a way, PREEMPT_RT could also be seen as a kind of horizontal partitioning, since it is a patch that “para-virtualizes” and redefines the kernel locking design of the standard Linux kernel.

The patches add overhead to the locking kernel mechanisms and will decrease the throughput performance to some degree, depending on the applications. The PREEMPT_RT patch also requires adoptions in driver code and other kernel code which is not originating from kernel.org.

The Thin-Kernel Approach: A Hard Real-Time RTOS Kernel Layer
This design alternative, depicted in Figure 2, offers truly hard real time and the lowest figures for latency.

The primary use of the thin kernel is to provide a hardware virtualization layer for the Linux kernel. If the actual CPU architecture offers true hardware virtualization support, the overhead introduced by the thin kernel can be kept down to as low as 2-3%.

DEFINITIONS & FUNDAMENTALS

What is Real Time?
“Real time” is often and incorrectly taken as a synonym to “high performance.” Real-time systems may be high-performance, but strictly speaking, real-time characteristics are expressed in other terms than performance. Real-time systems have well-defined operational deadlines from event to system response, with strict requirements on maximum and average response time. Non-real-time systems cannot guarantee the response time in any situation and are often optimized for high-throughput performance with best effort.

Hard, Firm or Soft Real-Time?
Real-time systems are often classified as hard, firm or soft. Missing a deadline in a hard real-time system means total system failure, and the impact can be critical. Firm real-time systems can tolerate infrequent misses although QoS deteriorates quickly when misses occur. More forgiving are soft real-time systems, where misses slowly degrade QoS, but the system is still possible to use. Typical hard real-time applications are pacemakers, process and engine controllers, robot control and anti-lock brakes. Baseband processing and signaling in mobile network equipment and wireless modems are examples of firm real-time applications. Less time-critical processing like IP network control signaling, network servers and live audio-video systems belong to the family of soft real-time applications.

Latency and Jitter
When discussing real-time requirements in an operating system, the term latency is frequently used. There are however several definitions of latency, therefore we need to define how latency is used in this article.

Interrupt latency – The time elapsed from the signaling of an interrupt request on hardware level until the kernel-level interrupt service routine (ISR) starts to execute. In Linux, the interrupt handling of an external event is usually split into one initial part (the kernel-level ISR triggered by the interrupt), and a second part referred for later execution (using the tasklet or work queue concept, or as a thread in kernel or user space).

Scheduling latency – The time it takes from when a piece of driver code via some kernel API signals a thread (usually a user-space thread) to be scheduled, until the kernel scheduler resumes the actual thread, for example the returning of a blocking call on a device file. A number of sources can trigger the scheduling of a thread, for example an ISR routine in the kernel or another application thread.

Task response latency – The interrupt latency plus the scheduling latency.

Jitter - The difference between minimum and maximum latency.
In a thin-kernel solution, a Linux kernel image includes the thin-kernel startup in the initial phase, hiding the fact that the RTOS kernel initially boots right before the Linux kernel. From a user’s perspective, it looks like a regular Linux kernel image is booted, but in addition to the Linux user space there are also RTOS real-time partition(s) created, with hard real-time capabilities. From Linux user space, one can load and manage the applications in the real-time partition and communicate with them using an inter-process communication (IPC) message mechanism.

The advantages of the thin-kernel approach are hard real-time support coexisting with a standard “Linux kernel. The drawbacks are that the real-time and non-real-time tasks are independent, making debugging more difficult since it adds another proprietary debug and execution environment. Also, the real-time tasks do not have full Linux platform support but they may provide a limited POSIX API that, for example, provides access to the Linux file system.

Examples of a thin-kernel approach are Xenomai and the Real-Time Application Interface (RTAI).

**The Vertical-Partitioning Approach: CPU Resource Shielding**

The strategy with the vertical-partitioning approach is to create kernel resource isolation “barriers” between sets of cores – “core clusters” – so that different applications can run on the same processor simultaneously without affecting the characteristics of each other. This is often called “CPU resource shielding.” The goal is to isolate a shielded set of cores into a real-time domain such that its local per-CPU schedulers are not in any way affected by the potential massive load that the applications outside this real-time domain generate in terms of kernel-resource locking scenarios or thread preemptions.

Figure 3 exemplifies a Linux SMP kernel, vertically divided into two types of execution partitions or domains, where non-critical applications for throughput are located to the non-shielded cores in the non-real-time partition, and applications that require real-time task response latency and a deterministic behavior are located to the shielded cores in one or more real-time partitions. This requires some changes in the behavior of the applications in the real-time partition:

- Disable the regular load balancing in the Linux kernel for shielded cores
- Explicitly bind IRQs that belong to the real-time application to a shielded core.
- Move the IRQs and kernel threads not belonging to real-time applications to the non-shielded cores.
- Disable the local timer interrupts (but this removes the ability for the local scheduler to enforce time sharing and to do some book-keeping of resources).

While the full POSIX API is present, extensive use of the POSIX API will affect both soft and firm real-time characteristics, i.e., the ability to meet deadlines in time and with minimal jitter. So on the set of shielded cores in the real-time partition, the design must explicitly avoid an extensive use of the normal POSIX API.
API that would potentially suffer from resource conflicts in the kernel call implementations. Omitting the API is not an option as existing solutions would lack any kind of OS API for services such as task management, timeout management and communication. Such a poor environment is undesirable, so a user-space real-time (RT) runtime environment that offers those services in a deterministic, low-intrusive and multicore scalable way needs to be defined.

**Vertical Partitioning with RT environment inside**

Figure 4 depicts the RT Run Time execution environment in the real-time partition. It provides a low-overhead, deterministic and OS-agnostic API to services like inter-process communication (IPC), timeout management, memory/buffer management and thread management in user space. The RT runtime environment provides necessary support to implement different kinds of scheduling environments. A light-weight task scheduler may be implemented using a restricted set of native pthreads in Linux, or as a user-space light-weight thread package.

The RT runtime environment should implement such a high-speed IPC “backplane” between the real-time partition and the “outside” by using lock-less techniques and user-space queues. This “backplane” does not only provide a channel to or from the real-time domain, it also provides a high-speed IPC network between applications spread over the cores of the entire SMP Linux processor. It is a challenge to design such an IPC backplane for high speed as well as scalability, yet not affecting the real-time characteristics in the real-time domain.

**Summary and Conclusions**

Two horizontal partitioning approaches have been outlined: the commonly accepted kernel-preemption patch PREEMPT_RT that improves the real-time responsiveness to POSIX applications, and the hard real-time thin RTOS kernel which acts as a hardware abstraction layer for Linux and provides an OS API for real-time applications outside Linux.

One vertical partitioning approach was treated, namely CPU resource shielding with IPC and RT runtime environment in user space based on standard SMP Linux and equipped with a high-performance and scalable IPC mechanism. This alternative strives to improve the overall design in order to avoid the kernel resource conflicts instead of making workarounds to minimize the effects from them.

In systems with SMP Linux on multiple cores, the vertical partitioning method will most likely provide the ability to give both high general-purpose throughput capabilities and good real-time capabilities. The fact that the standard Linux kernel design continuously improves with regard to the kernel-resource separation on core basis, i.e., the “space-partitioning capabilities,” is a promising foundation for the vertical-partitioning approach as being a good solution for next-generation, Linux-based embedded multicore platforms.

Michael Christofferson has over 30 years experience in software development for deeply embedded systems and telecom networking systems. He spent the first 8 years of his career with Raytheon in defense communications, specifically with remote-controlled signal and communications intelligence systems (SIGINT/COMINT). That was followed by 8 years in the telecom/datacom market working for such companies as GTE Telecom, Sprint, Data General Telecommunications and Pulse Communications. For the past 14 years, Mr. Christofferson has moved into sales, marketing and business development for leading industry RTOS, embedded development tools and middleware providers. These have been Ready Systems, then Microtec, then Mentor Graphics and now finally Enea for whom he has served since 1998. He is currently director of product management at Enea. Mr. Christofferson received a B.S. in physics from Bradley University in 1972, and an M.S. in physics from the University of Michigan in 1974. For more information please visit enea.com or contact us at info@enea.com.
Automotive Ethernet Consortium Takes 100Mbps Ethernet to the Street

Consortium’s objective is to help producers of BroadR-Reach devices and auto manufacturers bring products to market faster, while ensuring broad market appeal through interoperability and conformance testing and demonstrating product readiness.

By Cheryl Coupé, Editor

The University of New Hampshire InterOperability Laboratory (UNH-IOL) recently announced the launch of the Automotive Ethernet Consortium, which hopes to pave the way for semiconductor companies to address automotive industry requirements for next generation in-vehicle networking. We talked to Dave Estes, Ethernet manager and research and development engineer for UNH-IOL, and Jeff Lapak, UNH-IOL senior manager for Ethernet technologies to get more background and to understand what the announcement means for developers of embedded automotive systems.

First, some background. A year ago, the OPEN Alliance (One-Pair Ether-Net) Special Interest Group (SIG) was announced by Broadcom, NXP, Freescale, and Harman International, along with founding automotive members BMW and Hyundai, to encourage wide-scale adoption of 100Mbps Ethernet connectivity as the standard in automotive networking applications. The approach is based on Broadcom’s BroadR-Reach technology to allow multiple in-vehicle systems (such as infotainment, automated driver assistance and on board-diagnostics) to simultaneously access information over unshielded single twisted pair cable. By eliminating shielded cabling, the group expects automotive manufacturers can significantly reduce connectivity costs and cabling weight.

UNH-IOL is the first laboratory to be endorsed by the OPEN Alliance to provide neutral testing for the BroadR-Reach standard, but is no stranger to this approach: the lab was founded in 1988 to do Ethernet 10Base-T testing. The lab employs about 100 undergraduates and 10 graduate assistants, plus about 20 full-time staff members, and has 20-22 consortia doing testing for different technologies at any given time. Consortia membership varies from year to year, but the lab typically works with more than 200 companies. These organizations’ year-long memberships allow access to testing and equipment throughout the year. While the lab is affiliated with the University of New Hampshire, all projects are funded by industry through membership and testing fees. The lab is active in IEEE 802.3 working groups and the Ethernet Alliance.

For embedded developers of automotive systems, the lab offers the confidence that Ethernet-based products are interoperable and conformant to the customer’s expectations. Developers should be able to reduce time to market by having an existing set of test beds available for them to come in and test against, as well as the potential competitive advantage of an Ethernet-based system that is proven compliant and interoperable. The lab is careful to maintain third-party independence and neutrality to make sure that even direct competitors can use the lab for testing, knowing the results will stay completely confidential.
According to Jeff Lapak, auto manufacturers who are part of the Open Alliance are looking towards Ethernet as the way of the future. Most of the big automotive players are consortium members, including BMW, Mercedes, Volkswagen, GM, Ford, Hyundai and Toyota, and new members are joining all the time. Although testing is currently only available to semiconductor companies, the UNH-IOL plans to open membership to parts suppliers and automotive manufacturers as adoption of the BroadR-Reach standard progresses. Dave Estes provided the following responses via email.

**EECatalog**: What was the objective for establishing the Automotive Ethernet Consortium? What does “success” look like a year from now? Five years from now?

**Dave Estes, UNH-IOL**: The objective is to help producers of BroadR-Reach devices and auto manufacturers bring their products to market faster and to ensure there is broad market appeal for these devices through interoperability and conformance testing and demonstrating product readiness. In a year we hope to have several members including semiconductor companies, parts suppliers and auto manufacturers. Success would also mean having tested a reasonable number of products and showing proven interoperability. In five years we hope to have even more members and to be testing the next generation of in-car Ethernet being standardized by the IEEE, which is called Reduced Twisted Pair Gigabit Ethernet (RTPGE).

**EECatalog**: Are there key semiconductor (or other technology) vendors who are taking alternative approaches?

**Estes, UNH-IOL**: I am not aware of any other Ethernet technologies that are targeted for the automotive industry. There are several other in-car technologies that vendors may be invested in. Additionally, not all vendors will have become members of our consortium at this time, as it is still in an early-adopter state.

**EECatalog**: How will this group interact with other relevant in-vehicle networking standards groups, such as the IEEE 802.3 Ethernet Working Groups?

**Estes, UNH-IOL**: This group will be actively participating in the IEEE 802.3 RTPGE working group which is defining the Gigabit Ethernet PHY that is expected to be used in the automotive industry. We will also be working with the OPEN Alliance and the AVnu Alliance.

**EECatalog**: What's holding back the broad use of Ethernet in in-car networks? What issues still need to be addressed?

**Estes, UNH-IOL**: Prior to BroadR-Reach, Ethernet was not widely used in in-car networks simply because the auto manufacturers had to use shielded cables to meet the automobile EMC requirements. Shielded cables are heavier and more expensive. BroadR-Reach is able to meet the EMC requirements over a single unshielded twisted pair, saving weight and cost. Additionally there was no alliance or forum that had formed to select a single automotive Ethernet standard, therefore the market could not easily select a single approach. As for issues that still exist, this is the main reason we launched our effort. At this point a standard is selected and the technology will be adopted faster by proving that it works.

**EECatalog**: How will the migration from multiple “closed” systems to a single Ethernet-based network play out for developers? Are there challenges they’ll need to address as this evolution proceeds?

**Estes, UNH-IOL**: By using a standardized version of Ethernet and proper testing, auto manufacturers will have access to a wide variety of products from multiple suppliers while guaranteeing that the networking interfaces will work together. However as in all technology deployments, each individual auto manufacturer will need to address how best to migrate into using this technology.

**EECatalog**: Some of the advantages of this standard include more cost-effective safety and driver-assistance applications as well as advanced power savings that may be especially important for electric cars. What other opportunities for innovation do you expect to see as this approach gains momentum?

**Estes, UNH-IOL**: Some other advantages are reducing the weight of the cable harnesses in cars, which should reduce cost and increase performance (fuel-efficiency). Also the nature of Ethernet networks means that by using packet switching an appropriate amount of bandwidth can be allocated to different systems. There are far too many other potential applications...
comment on what may be actually developed (car safety systems like radar, cameras, infotainment systems, drive-by-wire, etc.).

**EECatalog:** There’s already a lot of talk about security concerns with connected cars. Is that an issue you’re addressing?

**Estes, UNH-IOL:** This issue will not be immediately addressed by our Automotive Ethernet Consortium; however this is an area that we can explore in the future because our lab has a lot of expertise in security protocols such as MACsec and IPsec. Also currently this technology is a wired solution; connected cars, on the other hand, I believe refers to wireless communication between vehicles, which is not currently part of this effort.

**EECatalog:** Are there any common misunderstandings about the consortium’s approach that you continue to run into? What questions do you find yourself answering regularly (or not often enough)?

**Estes, UNH-IOL:** The common misunderstandings are very similar to what we would have in any consortium. They generally revolve around the membership model, although we also offer pay-per-test services, and how they can access our test bed. There are also general questions about what kind of products we can test.

**EDITOR’S NOTE:** While Intel is not currently a member of the Automotive Ethernet Consortium, the company provided this comment: “Intel is excited about the opportunities for automotive use of Ethernet. While it appears simple on the surface to replace hundreds of pounds/kilos of complex wiring there is a significant effort underway to apply to automotive standards for reliability, predictability and durability. Intel is active in industry alliances that are taking an open approach to apply IEEE standards to these requirements. The alliance has supported automotive industry workgroups focused on development and adoption of these standards. For example, Intel is maintainer of a new open source project operating within AVnu called Open AVB, designed to encourage developers to take advantage of the APIs being defined within these alliances.”

Cheryl Berglund Coupé is editor of EECatalog.com. Her articles have appeared in EE Times, Electronic Business, Microsoft Embedded Review and Windows Developer’s Journal and she has developed presentations for the Embedded Systems Conference and ICSPAT. She has held a variety of production, technical marketing and writing positions within technology companies and agencies in the Northwest.
Deep Inside Intel

Semiconductor Manufacturing & Design sat down with Mark Bohr, senior fellow at Intel, to talk about a wide range of manufacturing and design issues Intel is wrestling with at advanced nodes—and just how far the road map now extends.

**SMD:** Will EUV make 10nm? And if it doesn’t, what effect will that have on Intel?

**Bohr:** For a process module as critical as lithography, Intel always has more than one option we pursue. In this era, the options are either EUV or 193nm immersion with multi-patterning.

**SMD:** How about directed self-assembly?

**Bohr:** That’s not a universally usable approach. You still need to define some layers with direct patterning, not a self-assembly technique. That’s a niche direction that will not replace these mainstream lithography techniques, but there may be some layers where it can complement the normal patterning techniques.

**SMD:** What’s your opinion about the future of the foundry business?

**Bohr:** The traditional foundry model is running into problems. In order to survive, the foundries will have to become more like an integrated device manufacturer.

**SMD:** Can even Intel afford to be an independent IDM? The cost of building state-of-the-art fabs at future nodes is astronomical.

**Bohr:** Yes. We have the volume and the products that can fill multiple fabs.

**SMD:** But you’ve also opened up your fabs to at least a couple customers. Are you planning on extending that?

**Bohr:** Our motivation is that we know we have great process technology, and partnering with other strategic companies can be a win-win situation. We can sell our technology and make more money off what we’ve developed, and they can have some very compelling products. It’s not Intel’s goal to be a general-purpose foundry, but we will be partnering where it makes strategic sense.

**SMD:** What’s your opinion about the future of the foundry business?

**Bohr:** The finFET is scalable to 14nm.

**SMD:** Is Intel sticking to bulk CMOS or will move to new materials such as fully depleted SOI?

**Bohr:** We see more advantages in bulk than SOI. I won’t say SOI won’t be in the future. There may be some device structure that is better done in SOI than bulk. But I don’t see than happening right now. When we first announced that we were making TriGate or finFET devices at 22nm, we said we’re making these devices on SOI, as well. But we think there are cost advantages to doing TriGate on bulk rather than SOI. That’s our plan for the foreseeable future.

**SMD:** What comes after the current finFET?

**Bohr:** The finFET is scalable to 14nm.
MARKET WATCH

**SMD:** But if you’re at 22nm, 14nm isn’t very far away, so you’ve got to be working on the next step.

**Bohr:** For Intel, you’re right. For other companies, it’s many years away. For 10nm, which is where I’m spending most of my time these days, I know we have a solution. I can’t elaborate at this point.

**SMD:** At 10nm aren’t you running into quantum effects?

**Bohr:** Everything gets different and tougher, but the problems are solvable—at least at that generation.

**SMD:** How far ahead can you see?

**Bohr:** I know we can get to 10nm. Beyond that, our research group is working on solutions for 7nm and 5nm. I have confidence we’ll have solutions for those. But by the time we’re down to 5nm we’ll be looking at non-familiar devices and device structures. That’s what we’ll have to do to get down to that level.

**SMD:** Where do stacked die fit into your roadmap?

**Bohr:** 3D stacked die have advantages, but only for certain market segments. You have to be very clear about what problem and what market segment you’re trying to serve. For a small handheld application where a small footprint and form factor are key and power levels are low, it probably makes good sense to use 3D stacking. For desktop, laptop and server applications where form factor isn’t as valuable and power levels are higher, 3D stacking has some problems that make it not an ideal solution.

**SMD:** Along those lines, does Intel see the smartphone and small mobile device market as a key direction?

**Bohr:** Intel is very serious about getting into the smartphone and tablet markets. We are a very different company from what we were five or six years ago. We are developing process technologies, but also products, that span a much wider range of performance and power than anywhere in our history. We’re not just at the high-performance desktop. We’re developing products that support 100-watt server chips down to sub-1 watt smart phone chips.

**SMD:** There are a number of interesting techniques Intel is working with, such as near-threshold computing. How will power management start changing inside these chips?

**Bohr:** When you’re talking about developing a smartphone chip that is ultra low power that also provides improved performance features that the market expects, you have to pull every trick out of the bag. You need great transistor technology, great package technology, great CPU architectures, the ability to turn off parts of the chip when you don’t need them so you’re saving power, the software links with the chip design so the software knows when to throttle power down. You need transistors, CPU architecture and software to be effective in that space.

**SMD:** How many cores will be required in the future?

**Bohr:** It depends on the market. In the server market, the more cores you can pack on the better. But in desktops, laptops and smartphones, there’s probably a limit to how many cores are practical. It’s not one. It’s probably several.

**SMD:** But less than eight?

**Bohr:** Yes, probably less than eight. But when you talk about the number of cores and computing engines, it depends on whether you’re dealing with traditional computing tasks where four cores are better than two cores. If you’re talking about execution engines in a graphics processor, clearly you want more cores.

**SMD:** What does this do for Intel’s platform strategy, particularly as you go after many markets with very specific needs?

**Bohr:** Even for Intel there are probably an optimal number of chip designs. It’s not like in the past where we tried to make one size fit all or have one chip serve multiple markets. But on the other hand, trying to design and manufacture dozens of very different designs in a generation is also impractical. There’s an optimal number of designs, although I don’t know what that number is, that can best meet the market requirements. You want to make as few iterations between the different designs as you can or re-use the cores or some of the circuit blocks between the different chips so you’re not completely redesigning it.

**SMD:** Are there other materials being considered for transistors?

**Bohr:** Our research group has been publishing papers about using 3-5 materials http://en.wikipedia.org/wiki/List_of_semiconductor_materials for the channels. You deposit indium phosphide or gallium arsenide layer on top of silicon to make a transistor on the surface. It’s still a silicon wafer, but you’re looking at depositing more exotic materials. That’s new and different and it may happen, but it’s not yet fully resolved how good that approach may be.
**SMD:** Has the priority for what you’re designing into a chip changed? Is it still all about performance, or has power overtaken that?

**Bohr:** Ten or 15 years ago, performance was the main goal in developing a new process technology. That really has gone away as the No. 1 priority. We still strive to provide a performance boost with each new technology, but there’s much more emphasis on improving power or efficiency on each new generation. We do that by reducing active power for the work a chip does. That’s a much more important goal for us today. Part of the reason is that the market has shifted from desktop applications to more mobile products. The first transition was from desktops to laptops. Now the move is to put things into smartphones. Today’s consumer wants computing power he can hold in his hand in the form factor of a smart phone and a tiny battery. He wants the performance he had on his laptop only three or four years ago. That’s what we shoot for.

**SMD:** That shifts the biggest challenge to the architecture, right?

**Bohr:** Yes. Whether it’s low-power, low-leakage transistors or a more efficient core architecture—or linking that with more efficient software.

**SMD:** What becomes the next big bottleneck?

**Bohr:** We have lots of challenges. Lithography is the key challenge in making transistors smaller. Whether EUV will happen on time or we have to extend immersion using multiple patterning. But when you make transistors smaller they don’t become less leaky. In fact, the opposite is true. You have to continually invent new structures and materials to allow feature-size scaling, which is critical for active power reduction and for cost.

**SMD:** But wires don’t scale well. How do you deal with that?

**Bohr:** RC delay gets worse as you scale, compared with transistors, which tend to get faster as you scale down. The industry has had 20-plus years of struggling with that problem. One way we’ve addressed that is that we’re no longer striving for very high operating frequency, especially in the phone market where 2 or 2.5GHz would probably be sufficient. That’s one advantage. The other advantage is that the average size of the chips is smaller in these laptop and cell phone applications so you don’t have interconnects traveling a long distance across a large chip. Instead, it’s a more compact chip so the signals don’t have to go as far. But even with those chips, we still have a challenge of performance from the interconnect. We have to be clever about what pitches we choose. Some of the lower layers are dense pitch, where density is important. Some of the upper layers are coarser pitch, where performance is important. We’re also continuing to drive down interconnect capacitance by employing lower-k dielectrics.

**SMD:** Is the interconnect becoming more problematic?

**Bohr:** If you talk to a designer 10 years ago you would have heard the same thing. Maybe now they’re saying, ’This time we’re really serious.’

**SMD:** How about new interconnect technology?

**Bohr:** It’s hard to replace copper and low-k other than by making lower k. But at least in the low-power cell phone market, stacking chips does help to minimize some of the interconnect issues, particularly between the logic and the memory chips.

**SMD:** You’re referring to through-silicon vias?

**Bohr:** Yes.

**SMD:** Is the interconnect becoming more problematic?

**Bohr:** If you talk to a designer 10 years ago you would have heard the same thing. Maybe now they’re saying, ’This time we’re really serious.’

**SMD:** How about new interconnect technology?

**Bohr:** It’s hard to replace copper and low-k other than by making lower k. But at least in the low-power cell phone market, stacking chips does help to minimize some of the interconnect issues, particularly between the logic and the memory chips.

**SMD:** You’re referring to through-silicon vias?

**Bohr:** Yes.

**SMD:** So if Intel is planning to get into that market, the company is experimenting with that technology right now?

**Bohr:** Yes, and we’ve been public about exploring TSV and 3D technology for a couple years. Although there are some challenging technology aspects, the real issue is cost. Doing TSVs and stacking chips—especially these custom Wide I/O chips—is expensive. So this might be a better engineering solution in terms of density, performance and power, but will the market bear the added cost? Not all markets will bear the higher cost.

---

*We still strive to provide a performance boost with each new technology, but there’s much more emphasis on improving power or efficiency on each new generation.*

---

*Ed Sperling has been an editor and writer for more than a quarter century, and has spent the past two decades writing about technology and business issues. He is a frequent moderator and speaker in Silicon Valley, and is a regular contributor to Chip Design Magazine. Sperling is the former editor in chief of Electronic News and Electronic Business. Before that, he held top editorial positions at Ziff-Davis and CMP Publications, and prior to that he was a daily newspaper investigative reporter covering crime and corruption.*
THE NEW MOBILE HORIZON

The mobile ecosystem continues to expand at lightning speed, with endless innovation and new applications of connected technology. Similarly, the industry's premier event is expanding in 2013. Mobile World Congress is relocating to a new world-class venue, Fira Gran Via, in the heart of the Mobile World Capital. With this convergence, the possibilities for global impact are endless.

Join us in Barcelona as together we explore the New Mobile Horizon

www.MobileWorldCongress.com
ADLQM67PC - Highest Performance Intel Core i7 PC/104 Platform!

The ADLQM67PC is a stackable PC/104-based form factor SBC utilizing second generation Intel® Core™ processors. This small, highly-rugged form factor is ideal for applications where ruggedness and processor performance are critical. It brings unparalleled performance to applications such as radar and sonar processing, image signal processing, tactical command and control, surveillance and reconnaissance, transportation and railway.

Features & Benefits

- Second generation Intel® Core™ processors in dual and quad core variants.
- Extended temperature -40C to +85C available.
- Broad portfolio of alternate PC/104 SBCs based on Intel® Atom™ processors Z510/Z530, D525 and N2600, Intel® Pentium® M, Intel® Core™ Duo, Intel® Core™ 2 Duo processors 2nd and 3rd generation Intel Core processors, and the Mobile Intel® GS45 Express chipset.
- Peripheral possibilities include: GSM, WLAN, WWAN, GPS, 3G cellular, storage drives, power supplies and many more functions including custom board services for specialty applications.
- Small stack footprints ranging from 95mm x 96mm for Intel Atom processors to 96mm x 115mm for Intel Core processors.
- Custom Enclosure Design Services Available!
The eBOX623-831-FL is our new cost-effective low power fanless embedded solution that features the high performance dual-core Intel® Atom™ processor N2600 1.6 GHz or D2550 1.86 GHz with DDR3-1066/1333 SO-DIMM support max up to 4 GB. Our embedded system has unique thermal solution, expandability, has a wide operating temperature range (-10°C ~ +60°C) and is IP40 protected. The eBOX623-831-FL is the reliable and robust box system which ideal for various embedded applications of gaming, digital signage, POS, Kiosk, automation, medical equipment and more.

Features & Benefits
- Intel® Atom™ processor N2600 1.6 GHz dual core / D2550 1.86 GHz dual core
- Intel® NM10 Express chipset
- Fanless operation design with full feature I/O
- High performance 204-pin DDR3-1066/1333 SO-DIMM max. up to 4 GB
- One 2.5" SATA drive bay and One CFast™ slot
- Supports VGA and DisplayPort dual display

MANO871 MINI ITX SBC for 3rd Generation Intel® Core™ i7/ i5/ i3/ Processors and Intel® Celeron® Processors

The MANO871 is our latest MINI ITX SBC model that features the 3rd generation Intel® Core™ i7/ i5/ i3/ processors and Intel® Celeron® processors (Ivy Bridge) with two DDR3-1333/1600 MHz up to 16GB, excellent for small and space limited environments.

The MANO871 features the latest Intel® vPro technology and Intel® Active Management Technology (Intel® AMT 8.0) for PC remote/security management. Triple view display ideal designed for graphic-intensive applications such as medical imaging, digital signage, Kiosk, gaming system and more.

Features & Benefits
- LGA1155 socket 3rd Generation Intel Core i7/ i5/i3/ processors and Intel Celeron processors (Ivy Bridge)
- 2 DDR3-1333/1600 MHz up to 16 GB
- PCIe x16 Gen3 supported
- 2 SATA-600 with RAID 0,1, 5 and 10 supported
- 2 SATA-300 with 1 SATA DOM supported
- 4 USB 3.0 supported
- Intel AMT 8.0 supported
- TPM 1.2 supported
- Triple view display
DFI Introduces...

Palm-Size Fanless Embedded System for In-Vehicle Application

EC800 Series

- Palm-size for space-limited application
- 9~24V wide range power input voltage
- Supports Wi-Fi and 3G/ GPRS for wireless communication

DFI-ITOX, LLC
8 Elkins Road, East Brunswick, NJ 08816 U.S.A.
Toll free: (888) 200-4869   Tel: (732)1390-2815
E-Mail: sales@dfi-itox.com  http://www.dfi-itox.com

DFI-ITOX, aka ITOX, is an affiliate company of DFI Inc.

Embedded Intel Solutions — Winter 2013 | 37
**ATCA-7370 Dual Intel® Processor-based Xeon® ATCA® server blade**

The Emerson ATCA-7370 uses dual 8-core Intel® Xeon® processors E5-2600 in a blade design optimized for compute and thermal performance.

It delivers outstanding performance, using dual Intel® Xeon® processors E5-2600 “EP”, each with four dedicated memory controllers, and two high capacity processor interconnects. For memory and I/O intensive applications, this design provides 33% more memory bandwidth and twice the processor interconnect capacity of other competitive products. The addition of an optional Intel® Communications Chipset 89xx device provides additional offload performance for encryption and decryption using Intel® QuickAssist technology.

The PICMG® 3.1 compliant fabric interface provides 10 Gigabit Ethernet (10Gbps) capability for applications requiring higher network throughput in the backplane.

Main memory configuration and mass storage options can be flexibly configured providing a perfect fit to your applications needs. Hardware RAID 0 and 1 is supported for locally and externally connected disk drives.

Designed for NEBS and ETSI compliance, and with multiple network and storage I/O interfaces Emerson’s ATCA-7370 can be easily integrated into different 1G and 10G network infrastructures such as telecommunication central offices and network data centers. It is built for maximum compatibility with commercial off the shelf software and it supports the use of higher performance processors in temperature-managed environments.

**ATCA-7470 Dual Intel® Xeon® Processor-based 40G ATCA® packet processing blade**

Emerson’s ATCA-7470 is a 40G ATCA® packet processing blade that enables the highest packet processing performance and security features. You can consolidate packet, application and control processing functions in a single blade architecture and benefit from lower development costs and the use of common tool suites. This can get you to market faster and enable you to balance workloads efficiently across available hardware resources.

Main memory and mass storage can be flexibly configured to provide a perfect fit to the needs of your application. Multiple available rear transition modules provide a flexible combination of storage and I/O, with options for high capacity redundant storage or up to 6x10G Ethernet interfaces.

**Features & Benefits**

- Two 8-core Intel® Xeon® processors E5-2648L, 1.8 GHz or E5-2658, 2.1 GHz
- Up to 128GB main memory
- Redundant 40G active/active ATCA Fabric interfaces, backward compatible with previous 10G systems
- Optional hardware off load module for encryption and compression acceleration with two Intel® Communications Chipsets 8920 devices
- Multiple 1 and 10Gbps network and storage I/O connectivity options
MSI launches the MS-9896 with the embedded Intel® Atom™ processor 3.5” Fanless Solution for Low Power and High Performance

**Features and Benefits**

- CPU: Intel® Atom™ processors D2550/N2800/N2600 (Dual Core)
- Chipset: Intel® NM10 Express chipset
- BIOS: AMI 32Mbit SPI
- RAM: Single Channel DDR3 1066MHz (Non-ECC, N2600 only support DDR3 800MHz) up to 4GB (N2600 only support up to 2GB)
- Socket: 1 x 204-pin SO-DIMM
- Graphic Controller: Intel® Graphics Media Accelerator (Intel® GMA) 3650 (N2600 only support GMA3600)
- LVDS: 2 x LVDS (LVDS1/LVDS2) LVDS1 Single Channel 18/24-bit(N2800/N2600 only Support 18-bit) LVDS2 Dual Channel 24-bit
- I/O Interfaces: 4 x COM-ports (COM1–4 x Internal) COM1 RS-232/422/485, 0V/5V/12V COM2&3&4 RS-232, 0V/5V/12V
- Mini-PCIe: 2x slots (1 supporting mSATA)
- Power Adaptor: DC in 12V

Rugged COM Express® Module with 3rd Generation Intel® Core™ i7 Processor for Small Form Factor Systems

The XPedite7450 is a rugged COM Express® module that complies with the PICMG COM Express Basic form factor (95mm x 125mm) and supports an enhanced Type 6 pinout. It is available with the quad-core Intel® Core™ i7-3610QE processor or dual-core Intel Core i7-2655LE, Intel Core i7-2610UE, Intel Core i7-3555LE, or Intel Core i7-3517UE processors. The XPedite7450 features up to 16 GB of DDR3-1333/DDR3-1600 ECC SDRAM, an integrated high-performance 3D graphics controller, five Gen2 PCI Express ports, four USB 2.0 high-speed ports, six SATA 3.0 Gb/s ports, and an Intel® High Definition Audio port. BSPs for Linux, INTEGRITY, and VxWorks and Windows drivers are available.

**Features & Benefits**

- Supports conduction-cooled and air-cooled applications with a single board design
- Utilizes tin-lead manufacturing process to mitigate tin whisker effects (RoHS-compliant process is also available)
- Provides BIT support

---

**Extreme Engineering Solutions**
3225 Deming Way Suite 120
Middleton, Wisconsin
53562, USA
608-833-1155
608-827-8171
sales@xes-inc.com
www.xes-inc.com
Security in the Connected Car

Connected, safe and reliable – this is the vision for the car of the future. To make sure these three requirements can be met, vehicle manufacturers, suppliers of electronic components and software as well as system architects have to master numerous challenges, and they must do it with close cooperation.

Demand for Sophistication
The consumer’s wish list for the car of the future is long and demanding. It includes the need for continuous connectivity with the Internet and the best possible integration between the vehicle and personal mobile devices such as smartphones. Additionally, there is an increasing interest around the concept of integrating and synchronizing personalized data from smartphones and “apps” with in-vehicle systems. The message is clear: Consumers want access to their important data at all times.

As today’s modern vehicles become more sophisticated, the amount of electronic components in each automobile continues to increase. The average car contains approximately 70 computer chips. To enable these chips and carry out their allotted tasks, up to 100 million lines of command code are required, in up to 100 electronic control units (ECUs), distributed over five bus systems. Given these statistics, there is no question that the role of software is incredibly important.

Connectivity and Risk
As car manufacturers try to set their vehicles apart from the competition, they can look to differentiating through electronics and in-vehicle infotainment (IVI) systems. Although computerization and connectivity allow consumers to enjoy greater convenience and automotive innovation, it also increases the dangers and risks. For example, imagine malicious code triggering equipment malfunctions by infiltrating the electronic control system via the unknowing use of infected MP3 files or downloaded apps.

Security Solutions
Given that connectivity opens the door for potential vulnerabilities, security is clearly a top priority. Key security aspects for the connected car can be divided into these categories:

- Cable-connected and wireless communication
- IVI system, including SSL encryption
- Electronic components within the vehicle, such as sensors or ECUs, including the certification of applications, remote management and malware control
- Services developed by car manufacturers or third parties, including apps from the cloud

The most powerful solutions encompass a range of security aspects and will demand close collaboration with the car manufacturer. Among the best security solutions is the use of embedded virtualization and hypervisors.

Today, the availability of processors with several multi-core CPUs permits new IVI architectures. Moving forward, expect greater consolidation of hardware, with several operating systems to run simultaneously, such as WinCE, VxWorks, Linux or AUTOSAR. In this case, protection is possible through the use of embedded virtualization and a hypervisor coupled with appropriate certifications (e.g., ISO 26262 standard). The trend of multicore and embedded virtualization paves the way for greater reliability, shorter boot times and cost optimizations as well as allows for brand new use cases and applications previously too difficult to achieve without hypervisor technology.

Additionally, as the auto industry increasingly turns to Android and leverages its flexibility for innovation, especially in areas of multimedia and connectivity, automotive electronics can become increasingly popular targets of attackers and malware.

A further measure to enhance security in the future may be to outsource susceptible files to the cloud and only load content or information when required via telecommunications. For this scenario to come to life, coordination is needed across the automotive ecosystem and standardization committees such as GENIVI.

The Future
Security solutions can only protect a system against the threats for which they were developed. While it’s impossible to safeguard against all attacks 100 percent of the time, there are many methodologies the industry is investigating further. Ideally, a comprehensive solution would take into consideration the issues around cloud services and over-the-air security as well as embedded security.

Security is a multifaceted issue and requires factoring in a variety of elements. Companies that have the expertise, technologies, and relationships across the embedded and automotive ecosystem, such as Wind River, are becoming even more important as the auto industry increasingly turns to the experts who understand and can connect all the pieces together.

Franz Walkembach is a senior product manager of open source platforms and automotive solutions at Wind River. Walkembach studied at Fachhochschule Giessen-Friedberg, with a specialty in mechatronics.
Nürnberg, Germany
February 26 – 28, 2013

embedded world 2013
Exhibition & Conference
... it's a smarter world

Register now for your free entrance ticket:
embedded-world.de

Programmed for success.

With some 900 exhibitors, embedded world is the top pioneering exhibition for embedded technologies worldwide. Make a note of the date now!
SEARCHING FOR Intel® Core™ i7 PROCESSOR SOLUTIONS?

X-ES delivers the latest 3rd generation Intel® Core™ i7 processor solutions on the widest range of standard and custom form factors in the industry. With our proven record of meeting aggressive schedules, you can count on X-ES to deliver Intel Core i7 processor solutions on time, and with unparalleled customer support. Call us today to learn more.

You need it, we have it! That’s Extreme.