

Embedded COMPUTING DESIGN

Connecting Silicon, Software, and Strategies for Intelligent Systems

DECEMBER 2014 #8
VOLUME 12
EMBEDDED-COMPUTING.COM

WHAT'S HOT
2015
PG. 30



**OVER-THE-AIR UPDATES
PUT AUTOMOTIVE SYSTEMS
ON THE ROAD TO V2X**
PG. 25

PLUS

SOFTWARE
SECURING DATA FOR CONNECTED
AND IOT SYSTEMS
PG. 14

RESEARCH REVIEW
ENERGY HARVESTING:
CREATING USEFUL POWER OUT
OF PROCESSOR WASTE HEAT
PG. 10



30 MINUTES OF 4K VIDEO CONSUMES OVER 10GB OF MOBILE MEMORY.

Get high-performance storage
for an Ultra HD future.

With Ultra HD on the rise, smartphones and tablets will need highly responsive storage to keep up. That's why for over 25 years, SanDisk has been expanding the possibilities of storage. The result is more than just incredible consumer experiences. It's enabling the next generation of mobile devices. sandisk.com/storage

©2014 SanDisk Corporation. All rights reserved. SanDisk is a trademark of SanDisk Corporation, registered in the United States and other countries.

SanDisk®

Annapolis Micro Systems

The FPGA Systems Performance Leader

WILDSTAR OpenVPX Ecosystem

FPGA Processing Boards
1 to 3

Altera Stratix V or
Xilinx Virtex 6 or 7
FPGAs per Slot

Open VPX Storage
Up to 8 TBytes Per Slot

4 - 8 GBytes
Per Second

**Input/Output
Modules**

Include:
Quad 130
MSps
thru
Quad 550
MSps A/D
1.5 GSps thru
5.0 GSps A/D
Quad 600
MSps D/A
Dual 1.5
GSps
thru
4.0 GSps D/A

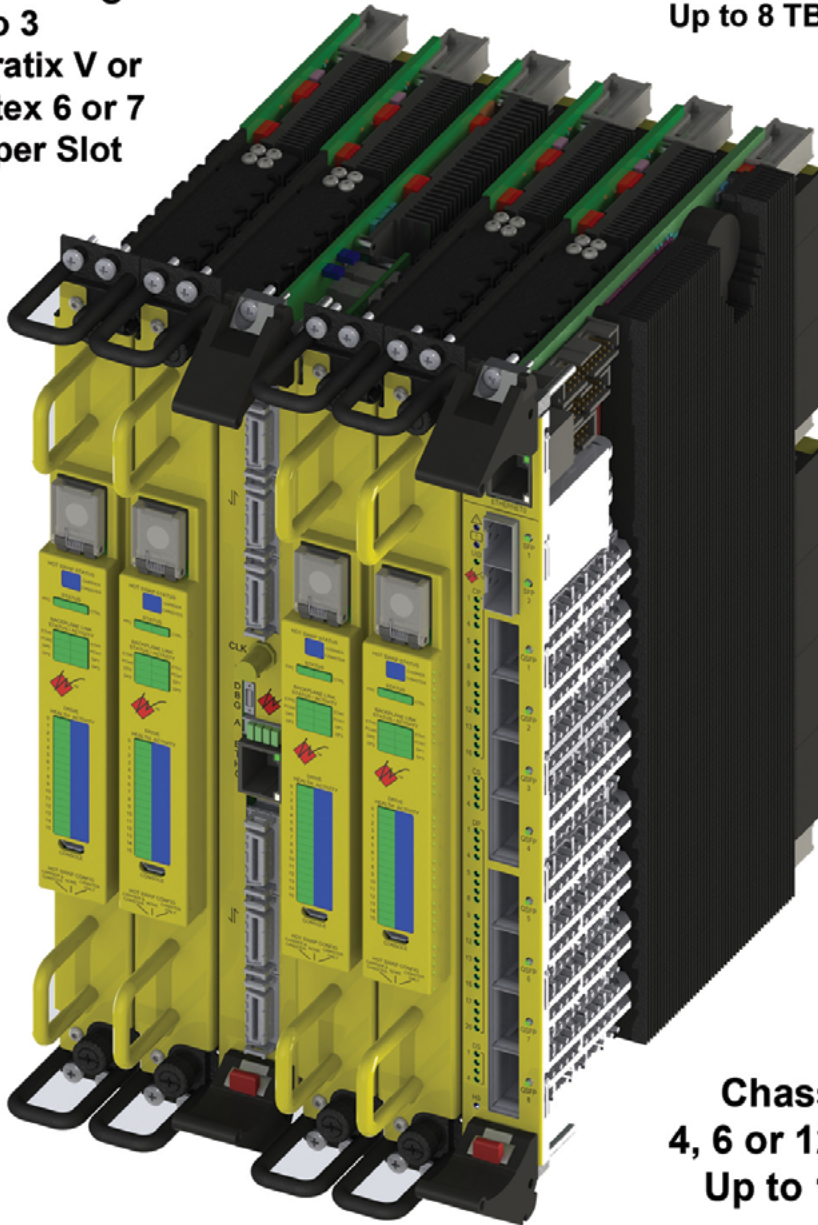
1 to 40 Gbit
Ethernet
SDR to FDR
Infiniband

GEOINT,
Ground Stations,
SDR, Radar,
Sigint, COMINT,
ELINT, DSP,
Network
Analysis,
Encryption,
Image
Processing,
Pattern Matching,
Oil & Gas
Exploration,
Financial and
Genomic
Algorithms,

**Open VPX
Switch**

1 to 40 Gbit
Ethernet
SDR to FDR
Infiniband

Chassis
4, 6 or 12 Slot
Up to 14G



**High Performance Signal and Data Processing
in Scalable COTS FPGA Computing Fabric**

190 Admiral Cochrane Drive, Suite 130, Annapolis, Maryland USA 21401
winfo@annapmicro.com USA (410) 841-2514 www.annapmicro.com



Photo courtesy of QNX Software Systems

Silicon

- 11** Opening up multicore implementation tools
Interview with Markus Levy, The Multicore Association

Software

- 14** Securing data on connected embedded devices
By Warren Kurisu and Felix Baum, Mentor Graphics
- 18** Ensure IDS/IPS and application layer protection beyond the RTOS
By Alan Grau, Icon Labs

Strategies

- 22** Reshaping vehicle insurance with telematics systems
By Cyril Zeller, Telit Wireless Solutions
- 25** OTA update possibilities put automotive on the road to V2X
By Brandon Lewis, Assistant Managing Editor
- 29** SSDs store added security for OTA
By Brandon Lewis, Assistant Managing Editor

What's Hot

- 30** Smart Vision Systems Poised for Takeoff
By Imagination Technologies
- 31** Real-Time Frameworks, Agile Modeling, and Code Generation
By Quantum Leaps, LLC
- 32** 2015: When security concerns meet safety concerns, Formal Methods become increasingly attractive
*By Cyrille Comar, Co-Founder/
Managing Director of AdaCore Europe*

Departments

- 5** Tracking Trends
Rory Dear, Technical Contributor

Industrial IoT: Robust connected embedded devices

- 7** IoT Insider
Brandon Lewis, Assistant Managing Editor

Full circle with Jeremy Rifkin's
"The Zero Marginal Cost Society" – Book review

- 8** DIY Corner
Monique DeVoe, Managing Editor

A new way to look at debugging

- 10** Research Review
Monique DeVoe, Managing Editor

Energy harvesting: Creating useful power out of processor waste heat

- 9** Community Outreach
Monique DeVoe, Managing Editor

Open book for secure systems

- 33** Editor's Choice

- 34** Web Wire



APP EXCLUSIVE CONTENT

Download the
Embedded Computing Design app:
iTunes: [itun.es/iS67MQ](https://itunes.apple.com/us/app/embedded-computing-design/id967767000?mt=8)
Kindle Fire: [opsy.st/kindlefireamaz](https://www.amazon.com/dp/B00K1QZ8Y4)

- » Preventing Linux rootkit threats through secure boot design using flash-based SoC FPGAs
By Tim Morin, Microsemi
- » Simplifying digital signage systems with multicore processors
By Ajay Misra, AMD
- » Multicore goes mainstream
By John Min, Imagination Technologies



Industrial IoT: Robust connected embedded devices

By Rory Dear, Technical Contributor

rdear@opensystemsmedia.com

The term "Internet of Things" is in danger, certainly in our industry, from those who struggle to comprehend its (concededly vague) definition. It's emblazoned across a plethora of exhibition stands at any relevant shows, with marketing appearing to be driven more by the number of global searches than a real passion for the possibilities the technology provides.

Particularly in the embedded space, from our personal interests we've all been wooed by what is now coined the human Internet of Things (HIIoT) in the guise of wireless fitness trackers and the like, but professionally we have struggled to find clear example applications for this technology to gain equivalent excitement from.

This is all about to change. Beyond the hype, what we're really interested in is the industrial Internet of Things (IIoT).

The key difference, beyond the obvious I read beautifully summarized as IIoT "brownfield" vs HIIoT "greenfield" – the former building on centuries of infrastructure, the latter necessitating fresh infrastructure deployed. The significant variations are essentially the same arguments as many a "commercial vs. industrial" solution debate.

Longevity

IIoT products must by definition offer an industrial lifecycle; HIIoT devices are often faddish and whimsical with consumers demanding functional and aesthetical upgrades well before even the suggestion of component driven obsolescence. Persuading Production Managers to risk any change in a tried and tested manufacturing environment

will demand that long-term availability commitment from manufacturers.

Reliability

Consumers today have smartened up to the importance of build quality, evidenced by the success of premium brands in all areas of the retail industry – but few will use any device for a long enough period to even experience what us embedded professionals would consider the bare minimum acceptable operational lifetime.

IIoT devices will be deployed in a "fit and forget" approach, though with the advantages of "Industrie 4.0" (the integration of Internet connectivity to industrial machinery) they'll be able to self-diagnose and self-report any failure or servicing needs

Environment

HIIoT devices are likely to be on your wrist or in your pocket. IIoT devices, on the other hand, will find themselves in the very worst conditions known to man, those well known to today's embedded computing devices. Actually, due to their invariably more compact size, they're likely to find themselves located in even further inhospitable corners.

The reality that they're likely to be performing fundamentally mission-critical tasks places greater emphasis on their environmental versatility than ever before.

The vision

As HIIoT experienced during its concept period, IIoT too struggles to gain the excitement it deserves through lack of real-world examples to adequately capture the imagination.

At the IHS Industrial Automation Conference this year, I was enthralled to hear Siemens' Dr. Dieter Wegener's vision of the connected and virtualized factory of the future made possible by IIoT.

His vision consists of a virtualized model of a manufacturing plant, though this is not just any 3D simulation that is available today. Dr. Wegener's virtual plant serves two purposes: training and management.

Training

The virtual plant, with optional virtual employee, enables on-the-job training from the comfort and safety of an office chair, avoiding placing the employee (and others) in any danger from industrial machinery with the kind of real-life simulations typically only found for commandeering monstrous vehicles.

Management

The management aspect enthused me yet further. The ability to have a live virtualized representation of your manufacturing plant, an exact mirror of your real world factory floor – made possible through universally connected peer-to-peer IIoT devices – enables true smart manufacturing and a level of autonomy that is currently a pipe dream.

These visions patently require significant capital investment so they may only be within the reach of the largest manufacturers today. But the challenge of IIoT, beyond the obvious of managing security, is convincing those below that top tier to make that investment today for tomorrow's benefit. **ECD**

ADVERTISER INDEX

- 32 AdaCore Technologies** — 2015: When security concerns meet safety concerns, Formal Methods become increasingly attractive
- 3 Annapolis Micro Systems, Inc.** — WILDSTAR OpenVPX ecosystem
- 17 COMPELL Systems Corporation** — Intel Celeron J1900, N2930, and Atom E3845 SBC
- 19 Creative Electronic Systems** — Flexibility and ruggedness
- 23 Dolphin Interconnect Solutions Inc.** — Device to device transfers
- 16 Elma Electronic** — Elma has the broadest selection of storage solutions in the embedded computing industry
- 35 embedded world** — The gathering of the embedded community
- 30 Imagination Technologies** — Smart vision systems poised for takeoff
- 15 Micro Digital, Inc.** — SMX RTOS is IoT ready
- 31 Quantum Leaps** — Real-time frameworks, agile modeling, and code generation
- 2 SanDisk** — 30 minutes of 4K video consumes over 10 GB of mobile memory
- 36 WinSystems, Inc.** — 3Accelerate your product development cycle



Get your free digital edition at
embedded-computing.com/emag



Subscriptions
embedded-computing.com/subscribe
subscriptions@opensystemsmedia.com
opensystemsmedia.com/subscriptions

Advisory Board

Jack Ganssle, consultant, Ganssle Group

Dave Kleidermacher, CTO, Green Hills

Jean LaBrosse, Founder/CEO, Micrium

Rob Oshana, Global Director of Software R&D, Freescale

Shelley Gretlein, Director, National Instruments

Dominic Pajak, Senior Embedded Strategist, ARM

Kamal Khouri, Director of Embedded Product Management, AMD

Rich Pugnier, Vice-President of Global Marketing, Kontron

Kamran Shah, Director of Corporate Marketing, Silicon Labs

Andrew Girson, CEO, Barr Group

Jim Ready, Chief Technical Advisor for Embedded Systems, Cadence

Bill Gatliff, Independent Consultant

Ian Ferguson, VP of Segment Marketing, ARM

Niall Cooling, Principal, Feabhas International

Adrian Valenzuela, Marketing Director, Texas Instruments



2014 OpenSystems Media®
© 2014 Embedded Computing Design
All registered brands and trademarks within Embedded Computing Design magazine are the property of their respective owners.
iPad is a trademark of Apple Inc., registered in the U.S. and other countries. App Store is a service mark of Apple Inc.
ISSN: Print 1542-6408, Online: 1542-6459



ECD Editorial/Creative Staff

Rich Nass, Brand Director
rnass@opensystemsmedia.com

Curt Schwaderer, Editorial Director
cschwaderer@opensystemsmedia.com

Monique DeVoe, Managing Editor
mdevoe@opensystemsmedia.com

Brandon Lewis, Assistant Managing Editor
blewis@opensystemsmedia.com

Rory Dear, Technical Contributor
rdear@opensystemsmedia.com

David Diomede, Creative Services Director
ddiomede@opensystemsmedia.com

Konrad Witte, Senior Web Developer
kwitte@opensystemsmedia.com

Sales Group

Tom Varcie, Sales Manager
tvarcie@opensystemsmedia.com
(586) 415-6500

Rebecca Barker, Strategic Account Manager
rbarker@opensystemsmedia.com
(281) 724-8021

Eric Henry, Strategic Account Manager
ehenry@opensystemsmedia.com
(541) 760-5361

Kathleen Wackowski, Strategic Account Manager
kwackowski@opensystemsmedia.com
(978) 888-7367

Shannon Alo-Mendoza, Strategic Account Manager
shannona@opensystemsmedia.com
978-501-9116

Asia-Pacific Sales
Elvi Lee, Account Manager
elvi@aceforum.com.tw

Regional Sales Managers
Barbara Quinlan, Southwest
bquinlan@opensystemsmedia.com
(480) 236-8818

Denis Seger, Southern California
dseger@opensystemsmedia.com
(760) 518-5222

Sydele Starr, Northern California
[sstarr@opensystemsmedia.com](mailto:ssstarr@opensystemsmedia.com)
(775) 299-4148

Reprints and PDFs

republish@opensystemsmedia.com

EMEA

Rory Dear, Technical Contributor
rdear@opensystemsmedia.com

James Rhoades-Brown – Europe
james.rhoadesbrown@husonmedia.com

Christian Hoelscher, Account Manager – Europe
christian.hoelscher@husonmedia.com

Gerry Rhoades-Brown, Account Manager – Europe
gerry.rhoadesbrown@husonmedia.com

OpenSystems Media Editorial/Creative Staff



John McHale, Group Editorial Director
Military Embedded Systems
PC/104 and Small Form Factors
PICMG Systems & Technology
VITA Technologies

Joe Pavlat, Editorial Director
PICMG Systems & Technology
jpavlat@opensystemsmedia.com

Jerry Gipper, Editorial Director
VITA Technologies
jgipper@opensystemsmedia.com

Monique DeVoe, Managing Editor
DSP-FPGA.com
mdevoe@opensystemsmedia.com

Steph Sweet, Creative Director
Joann Toth, Senior Designer

Lisa Daigle, Assistant Managing Editor
Military Embedded Systems
PC/104 and Small Form Factors
ldaigle@opensystemsmedia.com

Sally Cole, Senior Editor
Military Embedded Systems
scole@opensystemsmedia.com

Brandon Lewis, Assistant Managing Editor
Industrial Embedded Systems
PICMG Systems & Technology
blewis@opensystemsmedia.com

Amanda Harvey, Assistant Editor
Military Embedded Systems
VITA Technologies

Joy Gilmore, Assistant Webcast Manager
jgilmore@opensystemsmedia.com

Corporate

opensystemsmedia.com

Patrick Hopper, Publisher
phopper@opensystemsmedia.com

Rosemary Kristoff, President
rkristoff@opensystemsmedia.com

John McHale, Executive Vice President
jmchale@opensystemsmedia.com

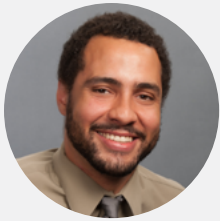
Rich Nass, Executive Vice President
jmchale@opensystemsmedia.com

Wayne Kristoff, CTO

Emily Verhoeks, Financial Assistant

Headquarters – ARIZONA:
16626 E. Avenue of the Fountains, Ste. 201
Fountain Hills, AZ 85268
Tel: (480) 967-5581

MICHIGAN:
30233 Jefferson, St. Clair Shores, MI 48082
Tel: (586) 415-6500



Full circle with Jeremy Rifkin's "The Zero Marginal Cost Society"

By Brandon Lewis, Assistant Managing Editor

blewis@opensystemsmedia.com

In a column earlier this year on hardware commoditization (opsy.st/HardwareCommoditization) I opened with a brief synopsis of Jeremy Rifkin's recent book, "The Zero Marginal Cost Society: The Internet of Things, the Collaborative Commons, and the Eclipse of Capitalism." In it, I summarized the major theme of the book, namely that the Internet of Things (IoT) will help usher in a never-before-seen age of economic efficiency and drive the marginal cost (or per-unit OPEX) of goods and services down to nearly zero. But with the year coming to a close, I thought it would be a shame not to circle back with a look at a couple of the adjacent/underlying technologies Rifkin predicts will aid in the expansion of the IoT and help push us into a zero marginal cost society.

Renewable energy

The Internet as it currently exists consumes up to 1.5 percent of the world's power, but according to consulting firm McKinsey & Company only 6-12 percent of the electricity consumed in datacenters is used to power servers during actual computation (the rest simply keeps them functioning and air conditioned). Although there are several power management schemes being developed to reduce energy usage within the datacenter, the biggest boon to players in the data storage arena will come through the implementation of renewable energy.

While the initial implementation costs of renewable energy systems are quite high, they offer the potential for long-term savings, and companies such as Apple and McGraw Hill have already begun implementing clean energy technology in their next-generation facilities. For example, Apple's datacenter in Maiden, North Carolina is powered by a 20 megawatt solar facility and 5 megawatt biogas fuel cell system, and includes a heat exchange

system that incorporates night time air into water that is used to cool the data-center. The marginal cost of generating electricity with such systems is nearly zero, Rifkin explains, and as increasing amounts of Big Data are archived in the IoT, reduced energy costs will have a direct correlation with the price of data storage.

The (F)OSS debate

The controversy over open-source software (OSS) and free and open-source software (FOSS) is one that has been going on for decades, and one that draws a line between hobbyists and the business community at large. The contention is underpinned by Stallman's "free speech, not free beer" view of code and the four freedoms of the GNU General Public License (GPL) of the free software movement on the one hand, and the and the subtle yet distinct paid licensing options for OSS on the other.

While Rifkin advocates both forms of open-source licensing within a zero marginal cost society as undermining copyrights and patents that inhibit collaborative technology development over time (as seen in the evolution of Linux), FOSS models better coincide with a new generation of developer that grew up freely copying content from the Internet under the assumption that "sharing information is little different than sharing conversation." In Rifkin's view, FOSS implementations and the free information movement also dovetail with the escalation of Big Data, as, "Just as information wants to be free, 'Big Data wants to be distributed.' What makes Big Data valuable is the information inputted from millions of individual contributors and sources that can be analyzed and used to find patterns, draw inferences, and solve problems. In a distributive, collaborative society, the millions of individuals whose data contributes to the collective

wisdom are increasingly demanding that their knowledge be shared in open Commons for the benefit of all, rather than being siphoned off and enclosed in the form of intellectual property owned and controlled by a few."

3D printing and the Maker Infrastructure

A particular area of interest for open-source technology is in Maker communities, where hobbyists are beginning to employ 3D printers as a low-cost, sustainable method of manufacturing. Similar to Stallman's four freedoms, the Maker Movement was built on four principles that include the open-source sharing of new inventions, the promotion of collaborative learning, a belief in self-sufficient communities, and a commitment to sustainable production. As 3D printing is committed to collaboration through OSS and is an additive (rather than subtractive) manufacturing process that even allows printers to create their own spare parts or additional 3D printers, the marriage of these two sets of ideals is a harmonious one. However, of specific importance to Rifkin is the fact that 3D printers can be quickly and easily connected to the IoT infrastructure to allow anyone in the world to become a prosumer (producer and consumer) of products, even to the extent of printing renewable energy technologies for the creation of local microgrids that scale out laterally into a distributed power network. These could, perhaps, help complete the circle by generating nearly free electricity for tomorrow's datacenters, which form the backbone of the communications infrastructure across which Makers collaborate and distribute.

If you're interested in checking out the book or finding out more about Rifkinomics, visit The Foundation on Economic Trends (www.foet.org). **ECD**



A new way to look at debugging

By Monique DeVoe, Managing Editor

mdevoe@opensystemsmedia.com

Usually when we look at DIY/maker things we focus on boards and projects, but those are just the ingredients list and recipe instructions. Boxed cake mix/how-to's and tried-and-true family recipes/project walkthroughs are great, but makers are also like creative chefs coming up with completely new dishes, which means there are a lot of failures and revisions before they get it right. Makers do a lot of debugging to get to that polished end result they're proud to show off or sell.

"Being an engineer or a maker means that you will spend 75 percent of your time debugging; solving problems is what we as engineers/makers do," says Jamie Bailey, Co-founder of Initial State (www.initialstate.com). "The tools that you have in your toolbox to figure out what is going on are critical to the success of your project and your sanity. Having debug tools that are super easy to use beats the heck out of guessing, swapping parts, or being frustrated because you can't figure out why your logic analyzer won't do what you want it to."

Oscilloscopes have been the traditional test and measurement tool for embedded hardware debugging for some time, but they're really only useful for analog/hardware testing. Logic analyzers are great for checking software and hardware simultaneously now that there's more of a software focus in electronics design, but they cost thousands of dollars, putting them out of reach for most individuals (Figure 1).

"Traditional test and measurement tools have been unaffordable for most makers," Bailey says. "Makers who are buying single board computers for \$35 aren't going to buy a \$2,000 logic analyzer to troubleshoot their design."

Not only are these tools expensive, they just don't provide a good user experience or keep up with what makers need for their projects.

"Makers are building a new generation of products, and they need a new generation of tools to help them with debugging and analysis. Makers need tools that give them super easy access to any information they need, including software events, hardware connections, sensor outputs, and whatever is going on inside their designs. The era of the physical probe is coming to an end," Bailey explains.

Bailey and his team at Initial State – which also includes David Sulpy, Director of SW Dev and Co-founder; Raymond Jacobs, Director of Biz Dev and Co-founder; Adam Reeves, Front-End Dev Lead; Vanessa Magalong, Software Dev; and Rachel-Chloe Gibbs, User Specialist

– have designed visualization tools for test and measurement. The inspiration for Initial State came out of Bailey's experience as an ASIC/system engineer, where sifting through a backlog of bugs from a mess of interacting complex sub-systems was hard to understand. Using log data was easier and almost always provided the information needed to solve a problem, but even a few seconds of data could generate log files of more than a hundred thousand lines.

"I had an epiphany that if instead of reading this data I could see it, I could figure out things much faster," Bailey says. "Having the right data visualization tools made the difference between hitting product schedules and being months late. I realized that we needed better tools, and the test and measurement industry wasn't keeping pace with the evolution of technology for the very products we were building."

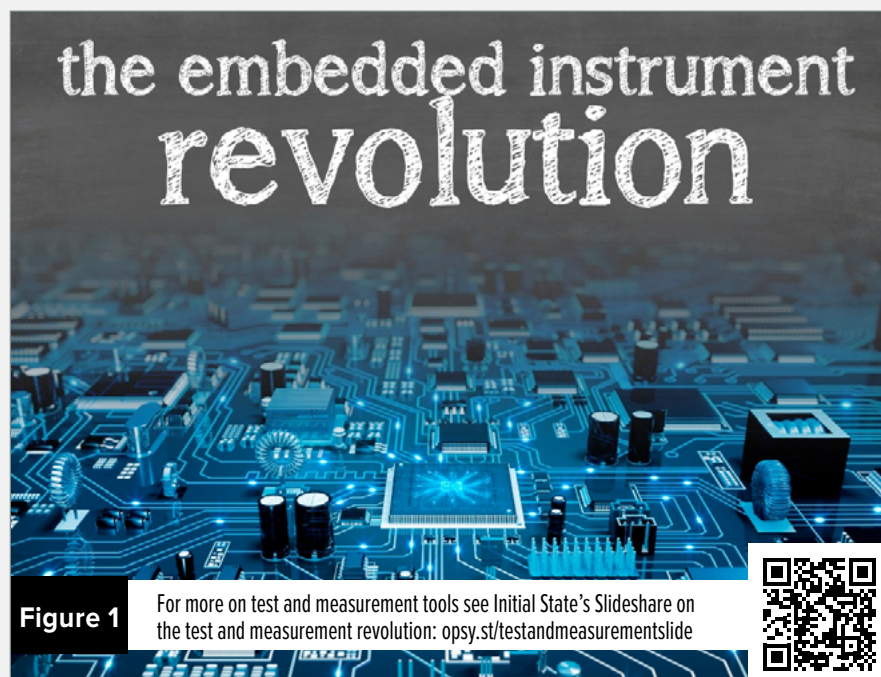


Figure 1

For more on test and measurement tools see Initial State's Slideshare on the test and measurement revolution: opsy.st/testandmeasurementslide



The Initial State team's visualization tools include Waves and Lines, among others that assist in visualizing, manipulating, and sharing data. Waves transforms and mines data in waveforms and statistics, from which you can collect targeted statistics. Lines let you interactively visualize numerical log data in stacked line graphs for measuring variables like time and magnitude, and to collect statistics. Data can be brought into these visualization tools through Initial State's IoT data streamer or by uploading log files.

"When you are debugging, you are often discovering things you didn't know you needed to discover," Bailey says. "Seeing dozens of concurrent hardware and software events at the same time is a great way to accomplish this. Being able to capture when a software function is called, the value of a variable, seeing a GPIO pin toggle, or seeing how your code reacts to a hardware interaction are frequent, common requirements during design. A great application of both Lines and Waves is being able to 'see' what happened in the past that led up to an issue. Your robot just started banging into a wall – what happened in the past that led up to the issue? It is amazing how the right visualization allows your mind to quickly catch an anomaly in your data."

Initial State's tools have been in beta since April 2014, and the feedback they've gathered so far has been very positive even ahead of the December 2014 launch.

"The feedback we have received for our tools has been unbelievably awesome and beyond even our expectations," Bailey says. "The most common feedback we get is 'this is so cool!' No one has said that about an o-scope since 1970." **ECD**



COMMUNITY OUTREACH

Open book for secure systems

By Monique DeVoe, Managing Editor

mdevoe@opensystemsmedia.com

It seems that almost every day we hear about security breaches, from relatively small exploits to massive breaches at major corporations and even the U.S. Postal Service. Cyber security is now a universal concern. Embedded developers need to be extra vigilant with the ubiquity of Internet-connected devices and the ever-growing Internet of Things (IoT). Many security techniques exist at hardware and software levels, but the best way to combat these attacks is by preventing them through built-in security – no more security afterthoughts.

In "Platform Embedded Security Technology Revealed: Safeguarding the Future of Computing with Intel Embedded Security and Management Engine" from Apress (www.apress.com) and Intel (www.intel.com), seasoned security researcher Xiaoyu Ruan of Intel's Platform Engineering Group thoroughly explains embedded system security. He walks readers through the embedded engine, security models, threat mitigations, and design details of algorithms, protocols, and interesting applications, with Intel's security and management engine as the basis of security techniques used. It's a thorough read to strengthen your security fundamentals.

This book is part ApressOpen, publisher Apress's open-access program for free and shareable ebooks (www.apress.com/apressopentitles). Other ApressOpen book topics include data virtualization, cloud security infrastructure, sensor technologies, and the IoT. **ECD**

Making and the Internet of Things

"Makers will shape the future of the IoT space," Bailey says. "The giant corporations who react at the speed of molasses cannot match the speed of innovation that comes from makers. We are seeing this firsthand everyday. It was pretty awesome to see Gartner Research come out and say that 50 percent of IoT solutions will originate from young startups and makers.[1] If the IoT opportunity is worth the trillions of dollars that Cisco predicts,[2] makers are truly going to spearhead a revolution."

[1] <http://www.gartner.com/newsroom/id/2869521>

[2] <http://www.forbes.com/sites/connieguglielmo/2014/01/07/cis-live-cisco-ceo-chambers-to-deliver-keynote/>



Energy harvesting: Creating useful power out of processor waste heat

By Monique DeVoe, Managing Editor

mdevoe@opensystemsmedia.com

Embedded systems are often mobile or deployed in remote locations off the grid, and must run reliably for years. The small size of these embedded computing systems combined with performance demands creates small, localized processor hot spots. How can designers power mobile electronics and better address the concerns of hot spots? Harness the heat energy for power. Arizona State University School of Computing, Informatics, and Decision Systems Engineering Assistant Professor Carole-Jean Wu is investigating the use of energy harvesting capabilities of thermoelectric modules in processors. This technique harvests waste heat and converts it to electricity, which can be used to enhance system cooling or be stored for future use.

"The heat distribution of modern computing platforms offers an interesting opportunity for waste heat energy harvesting," Wu says. "In particular, the unique heat distribution enables the use of thermoelectric materials in embedded applications."

Thermoelectric coolers (TECs) are often used for active cooling of CPU hot spots, and thermoelectric generators (TEGs) can be used in other areas of the CPU to turn remaining heat waste into useful electricity.

"Thermoelectric modules operate based on the phenomenon where a difference in temperature creates an electric voltage difference and vice versa," Wu says. "When a voltage is applied to a thermoelectric material, the splitting and combination of electron hole pairs results in a temperature difference on the material, called the Peltier effect. Conversely, if the material is subjected to a difference in temperature, a voltage difference is created, called the Seebeck effect."

The energy harvesting technique used in Wu's research exploits the spatial temperature difference between hot and cold components in a three-step process: perform system temperature and heat distribution characterization; identify thermal points and apply thermoelectric devices to generate electricity from temperature differences; and find native applications that exist in the system to use the harvested energy (Figure 1).

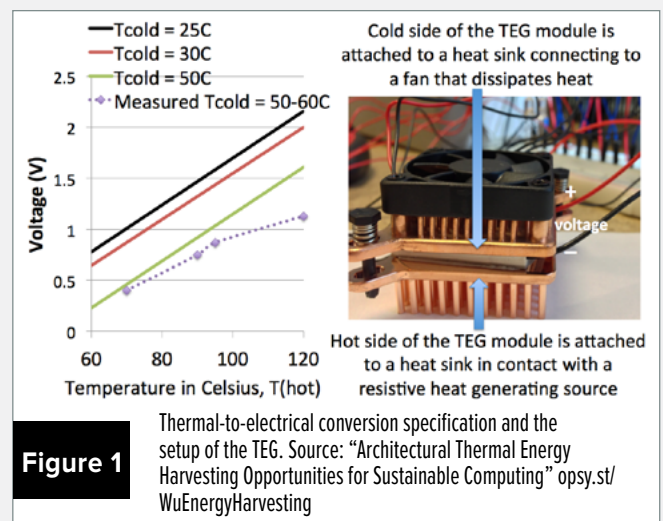
Wu and her team were able to recover 0.3 W to 1 W of power with an Intel Ivy Bridge processor running at 70 °C to 105 °C with a thermoelectric device on the CPU. The recovered energy when three TEG modules were used was at least enough to

power a fan, and can be a significant amount of power for mobile and wearable applications.

Though preliminary studies show promise for thermoelectric modules, there are still challenges to overcome, Wu says. Material efficiency and additional thermal resistance introduced to embedded systems by the energy harvesting materials are two critical challenges that must be addressed for energy harvesting to become more widespread in embedded systems.

"We are currently investigating important applications using thermoelectric modules at the processor architecture granularity," Wu says. "Our preliminary results indicated that, if managed intelligently, the temperature increase caused by thermoelectric generators can be tolerated and will not increase the overall temperature of the processor. The harvested energy is then used to lower the operating temperature of processors, which will, in turn, improve the chip reliability and the total cooling cost of the chip. We have filed a provisional invention closure on this work and are working on the first prototype of the design."

Generating power for cooling from the waste heat that already exists is a very promising solution for embedded designers. Users' battery complaints could be addressed and that bothersome excess heat from the processor could be mitigated and put to use. Designers should definitely keep an eye on where this research is going. **ECD**





Q&A

OPENING UP MULTICORE IMPLEMENTATION TOOLS



Markus Levy,
President, The
Multicore Association

Multicore architectures are increasingly useful for meeting the performance demands of increasingly complex embedded systems. However, implementation challenges are still widespread, and designers continue to struggle to come up with solutions in part due to the prevalence of proprietary tools and interfaces. The Multicore Association (MCA) aims to get vendors to work together and create industry-standard approaches to ease multicore integration. *Embedded Computing Design* discussed these issues with MCA President Markus Levy. Edited excerpts follow.

Q What do you see as the biggest challenges embedded designers are facing in the multicore space (such as scaling, optimization, communication, programming, etc.)? How does The Multicore Association (MCA) address those challenges?

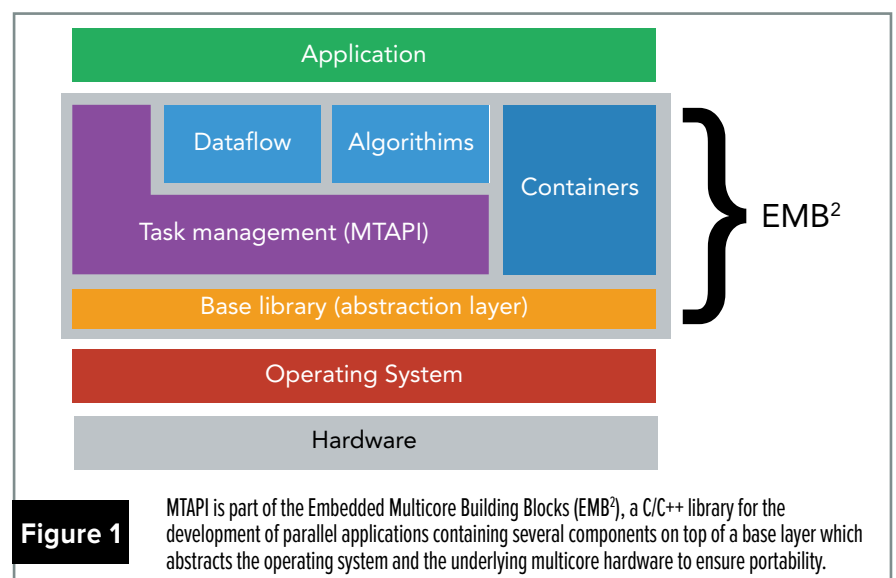
There are many challenges, but it really depends on the application space being targeted. For example, in networking, a big portion of the challenges are resolved by using Linux and all the support that's associated. There are new areas in networking that are causing challenges, but these are not necessarily multicore specific (e.g., SDN). On the other hand, in the mobile space the challenges include scaling; since the trend in mobile is 4-8 cores (because that's what marketing wants to see), the issue is developing code that really takes advantage of all the cores.

Multicore is used in many other applications that rely on heterogeneous computing, and here the challenges are all that you mentioned above. The MCA has produced specifications, such as its Multicore Communications API (MCAP)

that addresses the communication part and to some extent the programming and portability aspect. We are also soon going to release our Software-Hardware Interface for Multi-many-core (SHIM) that will make it easy for processor and software tool vendors to collaborate.

Coincidentally, the MCA has also just made an announcement that researchers at Siemens have developed an implementation of our Multicore

Task Management API (MTAPI). It's available as open source under BSD license at GitHub. Siemens' release is part of a bigger software package that they have named Embedded Multicore Building Blocks (EMB²). EMB² is a C/C++ library for the development of parallel applications containing several components on top of a base layer, which abstracts the operating system and the underlying multicore hardware to ensure portability (Figure 1). Besides



MTAPI, EMB² provides basic parallel algorithms, concurrent data structures, and skeletons for implementing stream processing applications. EMB² comes with C++ wrappers for MTAPl, which simplifies development in object-oriented environments.

Q Tell me about MCA, its working groups, and its mission.

The overall mission of the MCA is to develop standards to speed time-to-market for products with multicore processor implementations. The MCA board of directors has strategized to concentrate on some key areas of embedded multicore development. These include specifications for communications, resource management, task management, and a SHIM. In addition, we had a working group that produced a multicore book, which we call the Multicore Programming Practices guidebook. At the moment, active working group efforts are still going on with SHIM.

After several years of inactivity, we are also about to rekindle the working group efforts of our Communications API (MCAPI). MCAPI has grown in popularity over the years, mostly behind the scenes in proprietary implementations, but also in commercial products such as PolyCore Software's Poly-Platform. The new MCAPI working group efforts will include adding new features to the existing specification, as well as creating official MCAPI subsets that will address more resource-constrained applications such as the Internet of Things (IoT).

Q What is the state of multicore industry open standards? How much of multicore development is currently standardized vs. proprietary?

I don't know percentages, but I'd guess that a huge part of multicore development is still proprietary, at least in the embedded space. Industry standards take a long time to adopt because developers don't want to change their status quo. But we see a growing number of developers adopting our

“Clearly, multicore has become the de facto standard for embedded systems, and the industry continues to invest in optimizing the efficiency of multicore.

MCAPI standard, once they comprehend the benefits, which include portability and ease of use. Also, to reiterate, Siemens has done a great job with its EMB²; I think it will give many embedded developers a great head start. By the way, EMB² is sort of analogous to Intel's building blocks targeted at higher performance applications.

Q How does MCA go about creating standards and APIs? What open tools have been developed? How do they address multicore integration challenges?

The first step in MCA creating standards is based on input from our board of directors. Additionally, any member can take a proposal to the board and if it aligns with our general strategy, it will get discussed at a task group (the precursor to an official working group). Once it is realized that this proposal has legs, we walk it into a working group and invite members to participate. Similar to most consortium work, all development of standards is done in a democratic manner, which ensures increased value for a larger group. There are several open tools that have been developed, including the Poly-Platform and EMB² mentioned above. Other tools and implementations from vendors such as Mentor Graphics and Express Logic are described on the MCA website.

Q What will be the future challenges of multicore integration for embedded systems in the next 5 or 10 years?

Although we first started the Multicore DevCon almost 10 years ago to uncover development challenges, similar challenges still exist today and into the

foreseeable future. Clearly, multicore has become the de facto standard for embedded systems, and the industry continues to invest in optimizing the efficiency of multicore. The answer for today and the future is all in heterogeneous computing, whether it be CPU-GPU or complex SoCs with a variety of hardware accelerators.

The biggest challenge going forward will be standardizing the software that runs on proprietary and unique hardware. Significant improvements in software tools are necessary – something that the MCA SHIM specification plans to address. Essentially, SHIM describes the hardware from a software perspective (unlike IP-XACT, which focuses on the chip design level). SHIM will increase the adoption of new hardware by allowing tool vendors to more easily provide support for tools such as parallelizing compilers, simulators, performance analysis, and OS/middleware configurators.

Q What is next for MCA?

I'd say the biggest challenge for Multicore Association is to continue convincing companies to be leaders, rather than followers, and join our organization to help develop next generation specifications. By participating in the organization, it will help avoid re-inventing the wheel and use standards to accelerate product development, allowing companies to focus on their strengths.

The Multicore Association

➤ www.multicore-association.org
 @MulticoreMCA
 in [opsy.st/LinkedinMulticoreAssoc](https://www.linkedin.com/company/multicore-association)
 ▶ www.youtube.com/user/multiassoc



Protecting the IoT and automotive systems

By Curt Schwaderer, Editorial Director
cschwaderer@opensystemsmedia.com

By now we've all heard the astounding statistics – Gartner is forecasting that the Internet of Things (IoT) will include 26 billion units installed by 2020. IoT product and service suppliers will generate incremental revenues exceeding \$300 billion with a \$1.9 trillion global economic value add.[1] IHS estimates that in 2019 about 5 billion of these devices will be business-critical devices. Lofty numbers indeed.

The Internet of Things will shape the next generation like the web shaped our lives. Everything will be connected from wearables monitoring my health to the assembly lines cranking them out.

Security is paramount

As more business-critical IoT devices become connected, the greater the risk of "purpose-built" attacks will become. Software probing to identify connected devices will identify areas of vulnerability. Attempts to hack into these devices to gain access to the software that controls these systems can have far-reaching consequences.

For example, Freescale Semiconductor (www.freescale.com) recently announced a comprehensive hardware/software development system for enabling automotive-grade Ethernet connectivity for next-generation infotainment, instrument cluster, camera telematics, and rear-seat entertainment designs. Freescale's SABRE (Smart Application Blueprint for Rapid Engineering) for automotive infotainment development uses i.MX 6 series application processors for Ethernet audio video bridging (AVB). This kind of silicon/software/connectivity capability promises to transform the car into an

interconnected LAN that can integrate smart devices and in-vehicle infotainment and instrumentation. However, only a few feet away resides the telematics software environment – the brains of the vehicle that control the safety-critical and operational systems of the vehicle.

Many makes of automobiles are now promoting Wi-Fi hotspot capability within the car. Mix these three ingredients – Wi-Fi connectivity, infotainment interconnects of IoT devices, and the mission-critical telematics software – and you have an environment ripe for purpose-built attacks.

A combination of isolation, silicon, and software security capabilities are required to achieve the convenience of the "app store infotainment" paradigm where smart devices are welcome to interoperate and utilize the infotainment resources while still protecting the safety-critical elements of the car.

Software security

Allegro Software (www.allegrosoft.com) is an example of a company that started out as an embedded web server technology that has identified and embraced the security concerns IoT represents. Allegro incorporates RomDTLS for SSL 3.0 and TLS 1.2-like encryption over UDP communications. This provides the capability to provide authentication and encrypted tunnels between endpoints to increase security of connected systems. The Allegro RomCert provides public key infrastructure (PKI) and certificate-based authentication to enable authenticated access to application data as well as communications to perform firmware updates via the Internet. RomPager even provides an Allegro Cryptography Engine (ACE)

for software encryption and decryption services. The ACE also includes a harness API to offload cryptography calculations to silicon if needed.

Silicon security

The telematics microcontroller environment must have safeguards to prevent unauthorized access or access to critical data areas. AURIX microcontrollers from Infineon (www.infineon.com) utilize a built-in hardware security module (HSM) to protect software and data within the vehicle. This provides a measure of protection against hackers attempting to infiltrate the onboard systems. The HSM features AES128 encryption implemented in hardware with the performance to encrypt/decrypt Ethernet traffic. Secure key storage is provided in a separated HSM-DFLASH area for protection.

The AURIX architecture was developed according to an audited ISO 26262 compliant process. The architecture includes secured internal communications buses and a distributed memory protection system.

These kinds of silicon security features make the AURIX processor family a good fit for power train, engine management, injection systems, and hydraulic control functions in the car.

Securing a connected world

Convenience abounds and the explosion of IoT devices is coming. A critical blend of security capabilities from encryption to authenticated access must be incorporated in order to prevent potentially life-threatening issues from purpose-built attacks on these systems. **ECD**

[1] Forecast: The Internet of Things, Worldwide 2013, Gartner Group.



Securing data on connected embedded devices

By Warren Kurisu and Felix Baum

Securing data on connected embedded devices is a top priority among software developers and architects today. It seems with every passing week another major data breach is announced. Embedded software developers are quickly gaining the tools and technology to design safer, more secure connected devices. Developers can use ARM TrustZone technology to secure data in their designs in addition to embedded virtualization via a Type 1 Hypervisor to combine secure data, connectivity, and real-time operation through a real-time operating system (RTOS) and open source software to take full advantage of emerging multicore and heterogeneous system-on-chip (SoC) architectures.

The functionality of connected devices is rapidly increasing as is the value of the information stored on these devices, or information accessible through these devices. Most of the devices we use today are connected to at least one type of network or service. Cars are commonly connected to devices via Bluetooth or mobile data networks; portable medical devices connect to each other, to the hospital network, to the cloud, and beyond; and the smart energy grid connects the power utility to numerous consumer devices inside the home.

Protecting data on multiple fronts

When addressing how to secure an embedded connected device, it's important to first take into account the surface area vulnerable to attacks. The area of attack varies from device to device, but generally, the more sophisticated the device, the greater the area of attack. Second, it's important to understand most of the threats today target data not for the sake of data, but for the ability to manipulate the data. And third, it's critical to design and develop a device that is both robust and secure by layering various secure capabilities.



Manipulating data

An example of manipulating data might be an attack on an algorithm that affects the operation of the very system it depends on for operation, such as a banking application at an ATM terminal or the parameters that govern how a device functions within the automobile. When it comes to protecting data, developers need to be aware of the three critical stages: data at rest, data in use, and data in transit.

Data at rest is best described as when the device is powered down. Considerations in this stage include:

- Where is the bootable image stored?
- Are there anti-tampering methods used to inform the device if it's being tampered with and a means to prevent it from booting into a vulnerable state?
- Have the executables been encrypted, or could anyone who gains access remove EEPROM, dump the memory, or attempt to reverse-engineer the application?

- Have obfuscation methodologies been used for sensitive data?
- Is the device executing in a validated state? Has a chain of trust been established?

- How is data being protected if it is hijacked?
- Are encryptions or tunneling protocols in place?
- Have firewalls been deployed and what are the strategies for denial-of-service attacks?

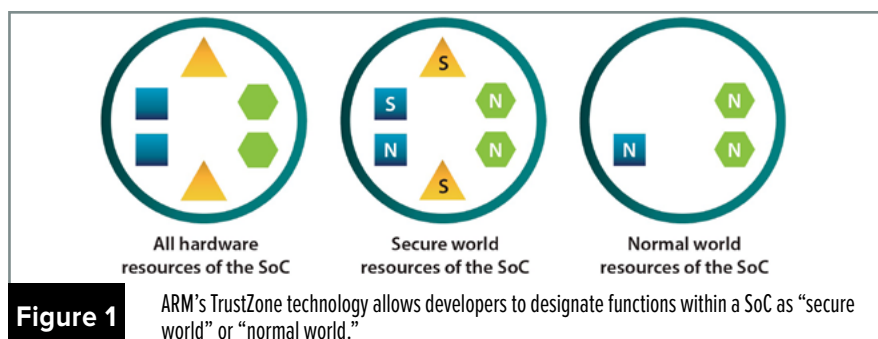
When it comes to protecting layers within your device, you might hear “defense in depth” or “layered security.” Regardless of the terminology, it boils down to creating layers of security that can defend against attacks, or, at the very least, delay the attack from penetrating subsequent layers. The layered security model might include:

- **Policies and procedures** – rules governing access and usage of a device
- **Physical** – literally, a physical barrier such as a fence, guard, or locked door
- **Network** – securing the connectivity to the outside world
- **Application** – ensuring malicious applications cannot compromise the system
- **Data** – ensuring the integrity of data that is used or stored in the system

ARM TrustZone architecture provides a solution that is able to carve out or segregate a hardware subset of the full SoC. It does this by defining processors, peripherals, memory addresses, and even areas of L2 cache to run as

The normal world (non-secure world) created and enforced by TrustZone is typically a defined hardware subset of the SoC. TrustZone ensures that a non-secure processor can access only non-secure resources and receive only non-secure interrupts. For example, a normal world hardware subset might include the UART, Ethernet, and USB interface, but exclude controller area network (CAN) access. The CAN might

Unlike the hardware subset in which normal world software runs, software running within the secure world has complete access to all of the SoC hardware. Thus, from the perspective of the secure software's execution, the system looks and acts nearly identical to what would be seen on a processor that does not have TrustZone. This means that secure software has access to all resources associated with both the secure and normal worlds.



SMX® RTOS is IoT Ready.

- smxWiFi 802.11 a/b/g/i/n Wi-Fi stack with P2P, WSC, SoftAP
- MediaTek/Ralink USB chipset drivers
- TCP/IP: IPv6, mDNS, SNMPv3, SNTP, Web, and many more protocols
- Security: SSL/SSH, SNMPv3, WPA2 Personal & Enterprise
- smxUSBH USB host for Wi-Fi dongles
- smx multitasking kernel
- Full source code – No royalty

Micro Digital
YOUR RTOS PARTNER

www.smxrtos.com/iot

A Trusted Execution Environment (TEE) refers to a software stack running within the secure world and the communications that allow secure software to interact with normal world software. TEE software typically consists of a small microkernel and applications, and APIs that allow secure software to communicate with the larger, user-centric software (e.g., Android). One of these specifications defines a TEE offering that some might call “typical RTOS” APIs and functionality, as well as additional

capabilities and APIs that are well suited to the TEE use cases.

Many of us regularly use secure world and normal world processing without realizing it. For example, online shopping typically requires a username and password. When prompted to enter this authentication information, the mobile device (Android phone, Apple iPad, etc.) will switch into secure world mode where the data is entered on a secure keyboard and securely processed

before anything else is allowed to occur. The banking application may itself run entirely in the secure world, but the device can also switch out of the secure world into the normal world to access other applications such as a browser, email, or to perform other non-secure tasks.

Securing SoCs in multicore architectures

A single ARM-based core can execute normal world context or secure world context, but what happens in a SoC with multiple cores? Developers can potentially run into situations where more than one core is accessing the same secure application, which not only extends the surface of the attack, but potentially exposes the code to the nasty timing issues that are difficult to debug. To correct this, developers can configure their device as described in the Figure 2, where only one core is allowed to execute the secure world content. In this design, when an application running on any one core needs to launch a secure application, it would have to reach out to core 0, where the transition to the secure world happens. This would make multicore designs simpler and more robust.

Running one application in the normal world and hiding secure keys and algorithms in the secure world does make a lot of sense, but, unfortunately, it's not very practical. A recent trend has developed in which silicon manufacturers are shipping more of the multicore parts. In many designs, more than one operating system (OS) is used and this is where hypervisor mode and virtualization extensions in the SoC become useful. A more complete architecture can be built on top of a hypervisor that incorporates support for ARM TrustZone technology. Designers can then partition applications and peripherals between virtual machines while at the same time secure keys and proprietary algorithms inside the secure world.

Putting it all together

A design deployed on an ARM Cortex-A15 device, for example, would look like the diagram in Figure 3. In the



ELMA
Your Solution Partner



“Elma has the broadest selection of storage solutions in the embedded computing industry.”

Our high performance, feature-rich products are used in all sorts of applications that require reliable and tested storage.

Available in air and conduction cooled, featuring SATA or SAS rotating or SLC, MLC and eMLC solid state drives for virtually any application. Features such as Secure-Erase, Write-Protect, RAID and NAS available in board and system level configurations.



Find out why Elma is the
**authority in embedded computing
platforms, systems & components.**
www.elma.com | 510.656.3400



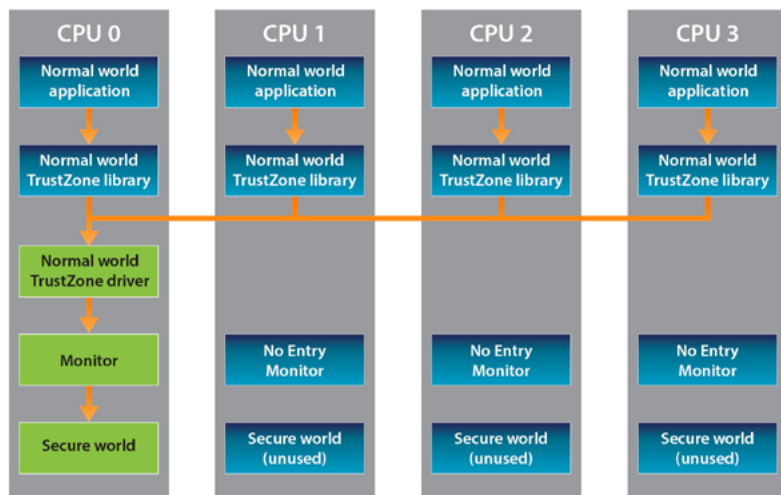


Figure 2 Using a designated core (CPU0) to execute secure world content in a multicore environment.

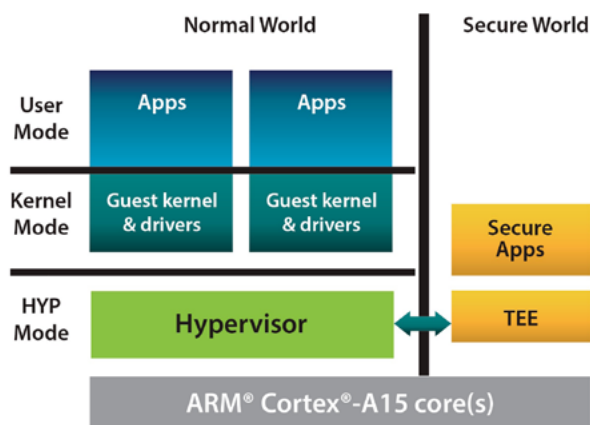


Figure 3 Incorporating a hypervisor into ARM's normal and secure worlds.

normal world we have the hypervisor executing with two virtual machines. Two instances of Linux running virtual machines within the normal world space are set up to have kernel and drivers to execute in the kernel mode context, while user applications are mapped to the user mode. TEE and secure applications are mapped to the secure world space. There are currently many ARM-based SoC processors built around the Cortex-A15 architecture that can support this type of configuration; TI OMAP5 and Jacinto6 reference platforms are two examples.

Securing connected devices

Connected embedded devices are becoming more functionally rich not only in capabilities, but in the data they generate and transmit. As these devices blend seamlessly into our daily lives, it's incumbent upon software developers to

design each new device with security as a paramount concern. Through the use of ARM's TrustZone technology, together with a Type 1 hypervisor, developers can provide a strong, robust, and secure base for SoC designs that meet the demands of our ever-expanding connected world.

Warren Kurisu is the Director of Product Management in the Mentor Graphics Embedded Systems Division.

Felix Baum is working in the Product Management team of the Mentor Graphics Embedded Systems Division.

Mentor Graphics

- www.mentor.com
- 🐦 [@mentor_graphics](https://twitter.com/mentor_graphics)
- 🔗 opsy.st/LinkinMentorGraphics
- 👤 opsy.st/MentorGraphicsGooglePlus
- 📺 opsy.st/MentorGraphicsYouTube

Advanced and reliable IPC products

Intel® Celeron® J1900, N2930 & Atom™ E3845 SBC New

LE-370 3.5" SBC

LP-173 Pico-ITX

LV-670 Mini-ITX

- * Intel® Celeron® N2920, J1900 & Atom™ E3845 SOC
- * DDR3L up to 8GB
- * DVI, DP(LV-670&LE370)
- * VGA, LVDS, Giga LAN
- * HD Audio, SATA
- * RS232/422/485
- * USB, DC9~30V IN

Intel® 4th generation Core™ SBC
LV-67N & LV-67M Mini-ITX, LE-37C 3.5" SBC

- Intel® 4th Gen. Desktop Core™ i3/i5/i7(LV-67N)
- Mobile Core™ i7-4700EQ, Celeron® 2002E(LV-67M, LE-37C)
- Intel® Q87/QM87 chipset, DDR3L up to 16GB or 8GB
- VGA/DVI/DP/LVDS, Giga LAN, HD Audio, SATAIII
- USB3.0, RS232/422/485, PCIe x 16(Mini-ITX), Mini-PCIe

FS-A78 Full-size & HE-B71 Half-size PICMG 1.3

- Mobile Core™ i7-4700EQ, Celeron® 2002E (HE-B71)
- Intel® 4th Gen. Desktop Core™ i7/i5/i3 (FS-A78)
- Intel® QM87/Q87 chipset, DDR3L up to 16GB
- VGA/DVI/DP/LVDS, 2 x Giga LAN, HD Audio, SATAIII
- USB3.0, USB2.0, GPIO, RS232/422/485, Mini-PCIe

MS-C78 & ME-C79 Micro-ATX Mainboard

- Mobile Core™ i7-4700EQ, Celeron® 2002E (ME-C79)
- Intel® 4th Gen. Desktop Core™ i7/i5/i3 (MS-C78)
- Intel® QM87/Q87 chipset, DDR3L up to 32GB
- VGA/DVI/DP/LVDS, 2 x Giga LAN, HD Audio, SATAIII
- USB3.0, USB2.0, GPIO, RS232/422/485, PCI, PCIe

Mini-PCIe Card & Backplane

MPX-210D(2)

■ Giga LAN card
Intel® I210-AT
IEEE 802.3
IPMI, MCTP
Win 7, 8, 2012

MPX-2515

■ CAN 2.0B card
Microchip 2515
ISO-11898
1 Mb/s
API & SDK

CBP-6P3X2

New
■ PICMG 1.3 Half-size Backplane
3PCI + 1 PCIe x 16 + 1 PCIe x 1

www.commell.com.tw

General Information: info@commell.com.tw
sales@tcommate.com.tw

Welcome to be commell Distributor



Ensure IDS/IPS and application layer protection beyond the RTOS

By Alan Grau

Seventy percent of cyber attacks target the application layer. While a secure RTOS provides features that are critical for security in embedded devices, that's just the foundation, not the complete solution. Some questions that need to be asked are: Why are IoT and embedded devices, even those with secure RTOS, still vulnerable to attack? What exploits are possible and what vulnerabilities have been reported? What's missing to protect these devices and how do design engineers ensure that their devices are safe? I'm glad you asked.

Advanced security capabilities are a major selling point for many RTOS vendors. Modern RTOSs provide capabilities such as multiple, independent layers of security (MILS) architecture, built-in resource provisioning, and support for security certification such as IEC 62304, IEC 61508, IEC 50128, DO-178B/C, EAL, and ARINC 653 certifications. In addition, many provide security services such as authentication, access controls, data encryption, and security protocols. Without question, embedded design engineers now have a much richer set of security tools and a stronger security foundation available in the RTOS than just a decade ago.

As impressive as all of this is, it's still just a foundation. A device running an insecure OS and communicating over an encrypted data channel is clearly insecure. The converse is not true. Securing the OS and adding security protocols is only first step to building a secure device.

Even with these pieces in place, there are still important security challenges to be considered. With security



implemented at the RTOS level, a successful attack against a protocol or application may not enable the attacker to gain full control, but that doesn't mean he won't be able to inflict considerable damage.

Data or communication that's valid and passes through the RTOS security may still present security threats to the application layer. Examples are attacks on web services or against supervisory control and data acquisition (SCADA) systems. A recent SCADA attack involved turning a water pump on and off repeatedly until the pump motor burned out. This attack focused on the application layer and resulted in a device failure despite the use of a secure OS. The OS security was never breached and yet the device was compromised. And with as many as 70 percent of all cyber attacks targeting the application layer, it's clear that security must extend to the application layer.

Application layer attacks

In 2013, security researcher Craig Heffner discovered a backdoor within the firmware found in a number of D-Link routers. The HTTP server in these routers includes a backdoor that bypasses the standard authentication process. The web server examines the browser user agent, and if it matches "xmlset_roodkcableoj28840ybtide," authentication checks are skipped. The string, read backward, "edited by 04882 joel backdoor," shows this is an intentionally planted backdoor.

The backdoor provided access to the device's configuration capabilities. The web server used in this same D-Link router already contains a number of vulnerabilities, some of which can be used in certain circumstances to allow for remote code execution.

In Australia, Vitek Boden waged a three-month war against the SCADA system of Maroochy Water Services beginning in January 2000, which resulted in millions of gallons of sewage spilling into waterways, hotel grounds, and canals around the Sunshine Coast suburb. It's an interesting case study because not only did the perpetrator cause pumps to not run when they should have, he also was able to prevent alarms from being reported, further complicating the problem. This example also shows the danger of insider attacks, as Boden was a former contractor of Maroochy Water Services.

Other widely reported attacks against application layer services include attacks on web-enabled IP cameras and nanny cams, which have notoriously weak security. A quick Google search will reveal multiple reports against web-based security cameras, nanny cams, and IP cameras. These vulnerabilities allow unauthorized users to view the video streaming from the camera, allowing them to spy on whatever the camera is set to watch. Even worse, in some cases, they can even instruct the "camera on" light to not activate, leading the victim to not know that they're being spied upon.

FLEXIBILITY AND RUGGEDNESS FROM CES

RSL-5222 - serial I/O PMC

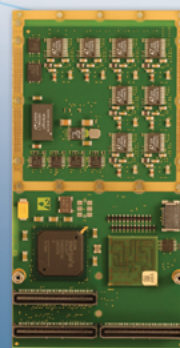
The latest rugged high-performance serial I/O solution from CES offers up to 8 channels for synchronous and asynchronous protocols in a PMC form-factor.

FPGA based, RSL-5222 supports most serial I/O protocols and matches any PMC pinout thanks to its FlexIO™.

Delivered with a SW driver, RSL-5222 is ready for your mission computer and fixed ground-based installation.

Headquartered in Geneva, Switzerland, CES - Creative Electronic Systems SA has been designing and manufacturing complex high-performance avionic, defense and communication boards, subsystems and complete systems for thirty years (such as ground and flight test computers, ground station subsystems, radar subsystems, mission computers, DAL A certified computers, video platforms, as well as test and support equipment). CES is involved in the most advanced aerospace and defense programs throughout Europe and the US, with a world wide sales presence.

For more information: www.ces.ch



Application layer security

What all of these attacks have in common is that they didn't target vulnerabilities in the underlying OS, but rather relied on vulnerabilities at the application layer. Another thing that they all have in common is that they exploit the system's standard interfaces. In each of these cases, the application layer allowed legal commands to be executed by unauthorized parties.

To protect the embedded devices' application layer from cyber-attacks requires a set of capabilities to ensure that the application only processes commands from authorized users, ensure that all processed commands are valid (e.g., contain legal data) and that all commands are appropriate (e.g., changing the ratios of ingredients or the processing temperature in a chemical processing plant). Additional capabilities that will provide a higher level of security for the device are the ability to detect and report suspicious commands or activity, a command historian to allow auditing when a problem does occur, and data protection to ensure that device data is protected.

Application security for an industrial control system

Industrial control systems are in many ways typical of modern embedded and IoT devices. They're frequently built using a secure RTOS, provide a customer application that performs a critical function, and can be controlled via messages received over an Ethernet or Wi-Fi network.

For our purposes, consider the example of an industrial control system used in the production processes of a chemical manufacturing plant. These systems frequently use Ethernet-based control protocols such as EtherNet/IP or Modbus TCP for configuration, control, and reporting. The control protocols specify the operation of a wide array of parameters

involved in the chemical processing. These can include the temperature at which the processing is performed, the ratio and the ingredients, the timing of the various processing stages, flow rates, etc. In addition to the control protocol (Modbus TCP, EtherNet/IP, etc.), the device may include a web interface for viewing configuration and processing information, and an FTP interface for downloading new firmware files.

While most cyber-attacks against the application will attempt to exploit weaknesses in the application interfaces, they may also attack the application implementation, or the interactions between interfaces/applications supported by the device (Table 1).

Protection against application-interface and -implementation attacks is provided by application protocol filtering. If the device includes an embedded firewall, it may be possible to extend the firewall to perform protocol filtering. Otherwise, application-guarding APIs can be implemented to perform protocol filtering for the device. Application-specific protocol filtering should provide:

- **Protocol validation** – ensuring that all messages conform to the protocol specification and verifying that all data is valid and in range.
- **Policy enforcement** – the protocol filter should support user-defined policies to restrict data-range values to device or installation specific ranges. For example, the protocol may allow a range of values for 0 to 100, but the device's operation may only allow values in the range of 40 to 60. The protocol filter should support this more constrained set of values.
- **Access control** – industrial protocols such as ModbusTCP don't provide any mechanism for access control. Hence,

Attack type	How it works
Application interface attack	Exploit weaknesses in the interface itself. Many legacy protocols, including Modbus TCP and EtherNet/IP have no built in security mechanisms. The protocols accept and process any commands they receive regardless of who the sender is. If a hacker can gain access to the network and send commands to the device, they can change the settings on the device, modifying how the control device is performing.
Application implementation attack	These attacks exploit weaknesses in the implementation of the applications by sending illegal data to the application in an attempt to compromise the application. Examples include out of range data and buffer overflow attacks. Well-designed applications will not be vulnerable to these attacks, but in many cases such attacks can cause unpredictable behavior.
Cross application attack	These attacks attempt to exploit relationships between multiple interfaces and applications on the system. For example, if a device provides an FTP interface, the hacker can use this to try and download system settings (configuration files), operational data (log files) and device firmware (which they can then try to reverse engineer to find weaknesses in the device for future attacks). The hacker may also use the interface to change configuration information, device firmware or other information stored on the device.

Table 1 Types of cyber attacks and how they exploit weaknesses in a system.

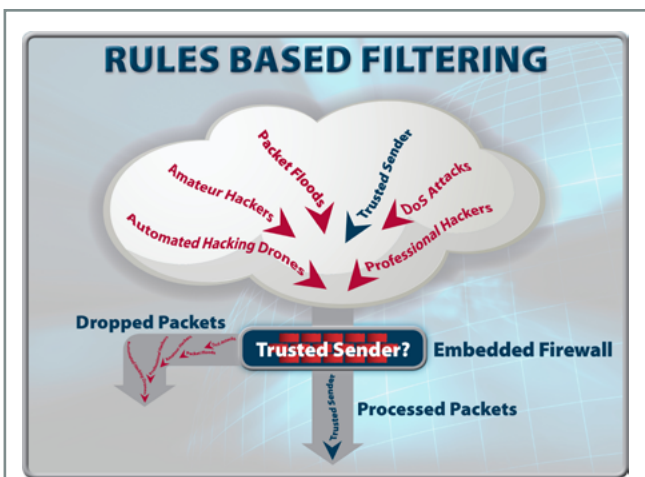


Figure 1

Rules-based filtering controls the packets processed by the embedded applications, providing the foundation for application security and intrusion detections capability.

any legal Modbus command received by the device is processed. Access control policies can be implemented in an application filter to control what devices are allowed to send commands to a device. For example, a white list of IP addresses can be configured and Modbus commands blocked if they aren't from a machine on the whitelist. Additional controls can be provided for finer-grain control.

- **Semantic filtering** – a significant challenge for industrial control devices is encoding rules to answer the question “does this command make sense.” While protocol enforcement, policy enforcement, and access control enforcement ensure that the commands received are legal and are from a trusted device or machine, they still don't solve the problem of an accidental or malicious change from an authorized insider. Semantic filtering attempts to prevent things like rapid cycling of commands or changing values in ways that are operationally incorrect, such as setting an inflow rate that exceeds an outflow rate for an extended time period.
- **Command audit logs** – record all commands executed by the application for later analysis in case a problem occurs.
- **Intrusion detection API** – an API allowing the device engineers to log and report each access, authentication attempt, or any other intrusion event. For example, if the web interface includes authentication with a username/password, each login attempt should be logged using this API. The intrusion detection API will then report this event to a management system, which would analyze the received data and detect attempts to probe a single device or multiple devices in a way that's

not possible on the device itself. The device may not have the intelligence or information to distinguish between repeated access attempts by a sysadmin who forgot a password and systematic probing by a hacker.

- **Event reporting** – provide a mechanism to send alerts when unusual behavior is detected.
- **Data anti-tamper detections** – this is achieved using a secure hash of static configuration data which lets the system detect when unauthorized changes have been made. Data anti-tamper can detect cross application attacks, such as a change to configuration data via the FTP or Web interface by an unauthorized user (Figure 1).

An application layer security framework

An application layer security framework, such as the Icon Labs Floodgate Defender, provides a framework for application security in embedded devices (Figure 2). This framework includes:

- Floodgate Defender
- Application protocol filtering engine and embedded IDS/IPS
- Floodgate Secure
- Secure file hashing for anti-tamper protection
- Audit task for run validation of secure hashes
- Floodgate Aware
- User API for event reporting and command audit logging
- Floodgate Agent
- Interface to cloud based management system supporting
- Events and audit logs
- Management and enforcement of security policies
- System level firewall filtering and intrusion detection capabilities

Alan Grau is the President and cofounder of Icon Labs, a provider of security solutions for embedded devices.

Icon Labs ➤ www.iconlabs.com

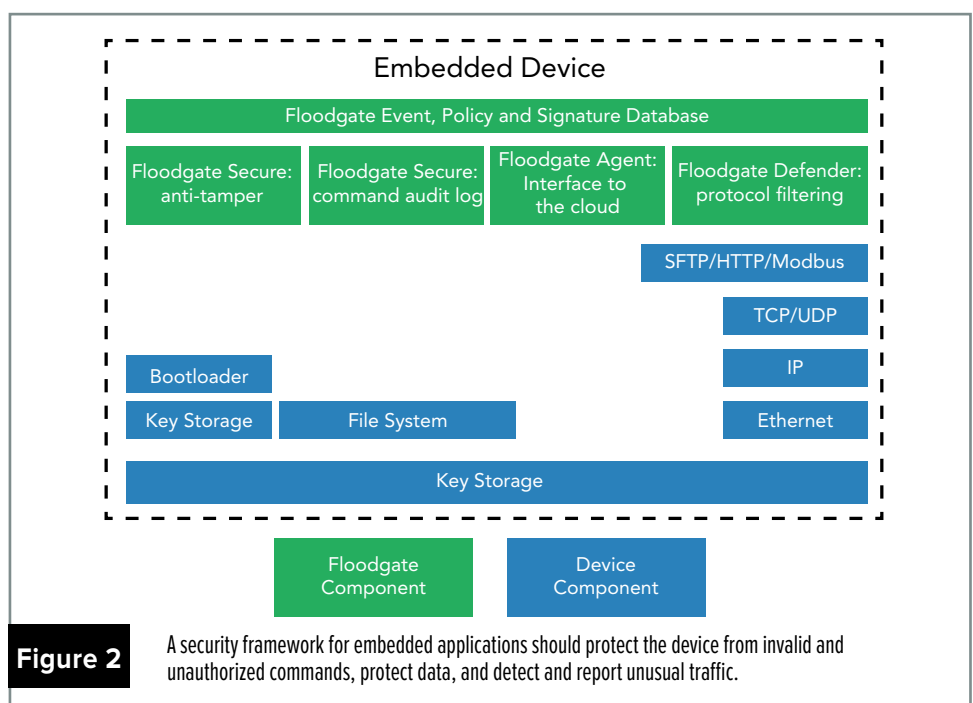


Figure 2

A security framework for embedded applications should protect the device from invalid and unauthorized commands, protect data, and detect and report unusual traffic.



Reshaping vehicle insurance with telematics systems

By Cyril Zeller

Usage-based insurance (UBI) is a hot topic: it delivers tangible benefits to consumers, the insurance industry, and society, but despite the huge potential the market is largely untapped. Right now the number of UBI policies in the world is a mere 5.5 million, with the majority in North America. However, this figure is set to rise to 100 million worldwide by 2020, according to PTOLEMUS Consulting Group.

The way motor insurance risks are assessed has stood still for many years – based on static, statistical data like age, gender, car model, etc. Telematics technology allows objective assessments of risk profiles to be based on real-time, dynamic data like mileage, area, time of day, keeping to speed limits, engine RPM, fuel level, and driver behavior. It can also be paired with publicly available data to identify road type and weather conditions.

Insurers benefit from the ability to detect and retain the majority of the lowest risk drivers. On the other hand, drivers, particularly young drivers, can get significant discounts on their premiums. Moreover, careful drivers cause fewer accidents, which is, of course, a big benefit for society at large.

The way ahead

The stats indicate very healthy growth for UBI, with ABI Research expecting a 12x increase by 2019. On the surface, there seems to be a rock-solid business case for both drivers and insurers. There are vocal advocates for smartphone-based UBI, but the insurance industry and



regulators – the organizations that matter – have a number of valid concerns: They are highly critical of the reliability and accuracy of data that is delivered: phones can be removed accidentally or run out of battery, users would need to start the UBI app manually, phones could be dropped or become airborne during an impact, and driving behavioral data such as braking, turning, accelerating is likely to be inaccurate and unreliable, among other issues.

A fragmented market also causes uncertainty and low adoption rates. The market comprises a mix of hardware vendors, insurance companies, vehicle manufacturers, and regulatory authorities whose mandated timelines are different in each country. Significant benefits are being realized for telematics systems such as fleet management, usage-based insurance, emergency calls, stolen vehicle recovery, diagnostics, and toll payments. However, these are fixed-point solutions that perform a single function. This indicates the need for open platforms running on robust hardware that will enable the development of consolidated solutions.

The development of an open, global standard is challenging, and needs a fair degree of adoption to be successful. Various government regulations are being proposed, such as the eCall emergency response system in Europe, ERA GLONASS in Russia, and the Denatran anti-theft recovery system in Brazil. However, apart from ERA GLONASS that aggregates location-based services in the basic system, these regulations

only address a single vertical issue. As such, they perpetuate market fragmentation.

The in-vehicle network

The in-vehicle network of a connected car has three domains: the physical network and the electronic control units (ECUs), the communications portal at the company that delivers services, and the communications link between the vehicle and the portal. ECU embedded computer systems control one or more of the car's electrical system or subsystems. ECUs exchange data over this network using the controlled area network (CAN) standard. Every CAN packet is broadcast to all the elements on the same bus, which means each node can interpret them.

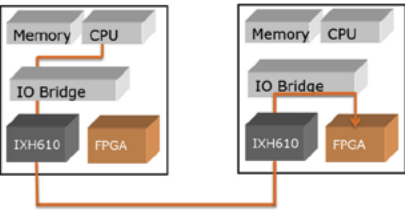
OBD-II is a standard that provides almost complete engine control and also monitors parts of the chassis, body, and accessory devices, as well as the diagnostic control network of the car. Vehicles have an OBD (onboard diagnostics) connector, an interface to the CAN network, that garages use for reading information when they service vehicles. This interface to the CAN network is also used to obtain relevant vehicular data for insurance services using OBD dongles.

The hardware

Insurance telematics embedded systems are fitted into vehicles as they are made, though right now only by luxury carmakers such as Audi, Cadillac, Daimler-Benz, and BMW. UBI should be seen as a subset of a comprehensive system that includes


Device to Device Transfers

PCI Express® Networks




Fast Data Transfers

Need to access FPGA, GPU or CPU resources between systems? Dolphin's PCI Express Network provides a low latency, high throughput method to transfer data. Use peer to peer communication over PCI Express to access devices and share data with the lowest latency.



Learn how PCI Express® improves your application's performance

www.dolphinics.com



navigation and in-car entertainment and information. There is also an increasing trend among insurers to employ M2M technologies that reduce losses due to car theft and, in addition to tracking location, devices can disable engines if an unauthorized access has been determined.

So-called “black boxes” provide similar functionality, but they are an after-market product that can be fitted into vehicles after they have been manufactured. They are normally housed in a secure part of the vehicle that is not easy to access.

In the U.S., systems based on OBD dongles have become the preferred option for eight of the top ten personal motor insurers. These robust devices are unobtrusive and, because they have a semi-permanent wired interface to the vehicle's electronic system, they provide the precise driving data that is needed for a UBI policy.

For example, data loggers like the one from DanLaw (Figure 1) have the same form factor as a regular dongle but they provide small, self-installed, cost effective, OBD-connected telematics solutions for monitoring, logging, and transmitting vehicle network message and position data.

The hybrid wireless communication device enables data communication and connectivity through GSM and Bluetooth wireless connections. Optionally it can capture timestamped accelerometer and GPS position information.

Smartphones: Confusion in the market

For insurers, smartphone apps are a tool to collect data to use for risk assessment. Free UBI trials allow smartphone services to be employed as a “teaser” that introduces the concept, allows drivers to see their driving behavior at the end of the trial, and informs them about the potential reduction in their premium if they drive carefully.

Due to concerns by the insurance industry and regulators, the great majority of the insurance industry is only offering premium reductions on dedicated in-vehicle devices like OBD dongles. It is worth noting that drivers can use smartphones in conjunction with a dedicated in-vehicle device in order to get real-time feedback on their driving behaviour. For example, if they are driving too fast a warning could be given about a potential rise in the premium.

End-to-end solutions

End-to-end solutions are needed in order to allow telematics service providers (TSPs) to focus on the applications, which are their core competence. That is the key that will unlock UBI's potential. In turn it will allow hardware vendors to ship aftermarket devices that can be demonstrated almost immediately, thereby accelerating time-to-market. IMETRIK recently announced a solution based on a telematics device that starts gathering information on driving behavior as soon as it is connected to the OBD interface. IMETRIK also manages its own mobile network infrastructure, which enables seamless connectivity to the insurer's back-office infrastructure.

IMETRIK's device provides cellular and positioning modules specifically created for deployment in dongles, where real-estate space is very limited, and enables connectivity to an open platform known as m2mAIR that runs a robust set of value-added services hosted in the cloud.

Creating telematics devices for the insurance industry

Right now the market for UBI is largely untapped, but the number of policies is set to rise to 100 million by 2020. This represents a very attractive opportunity for new entrants – hardware vendors as well as TSPs. Established TSPs are also developing added-value services that capture the demands of the increasingly connected world.

However, new entrants need to be aware that UBI's use of cellular networks means that the devices have to be certified by the mobile network operator before they can be deployed. It's clear that established TSPs have everything they need to operate a service, but in many cases their back-office system would have been designed three or more years ago – a long time in high-tech circles. Issues can arise, such as scaling the solution or increasing the security.

Cyril Zeller is Vice President of Global Telematics at Telit Wireless Solutions.

Telit Wireless Solutions

www.telit.com
[@Telit_WS_corp](https://twitter.com/Telit_WS_corp)
www.linkedin.com/company/telit-wireless-solutions
www.youtube.com/telitcommunications



Figure 1

Data logger OBD dongle from DanLaw.

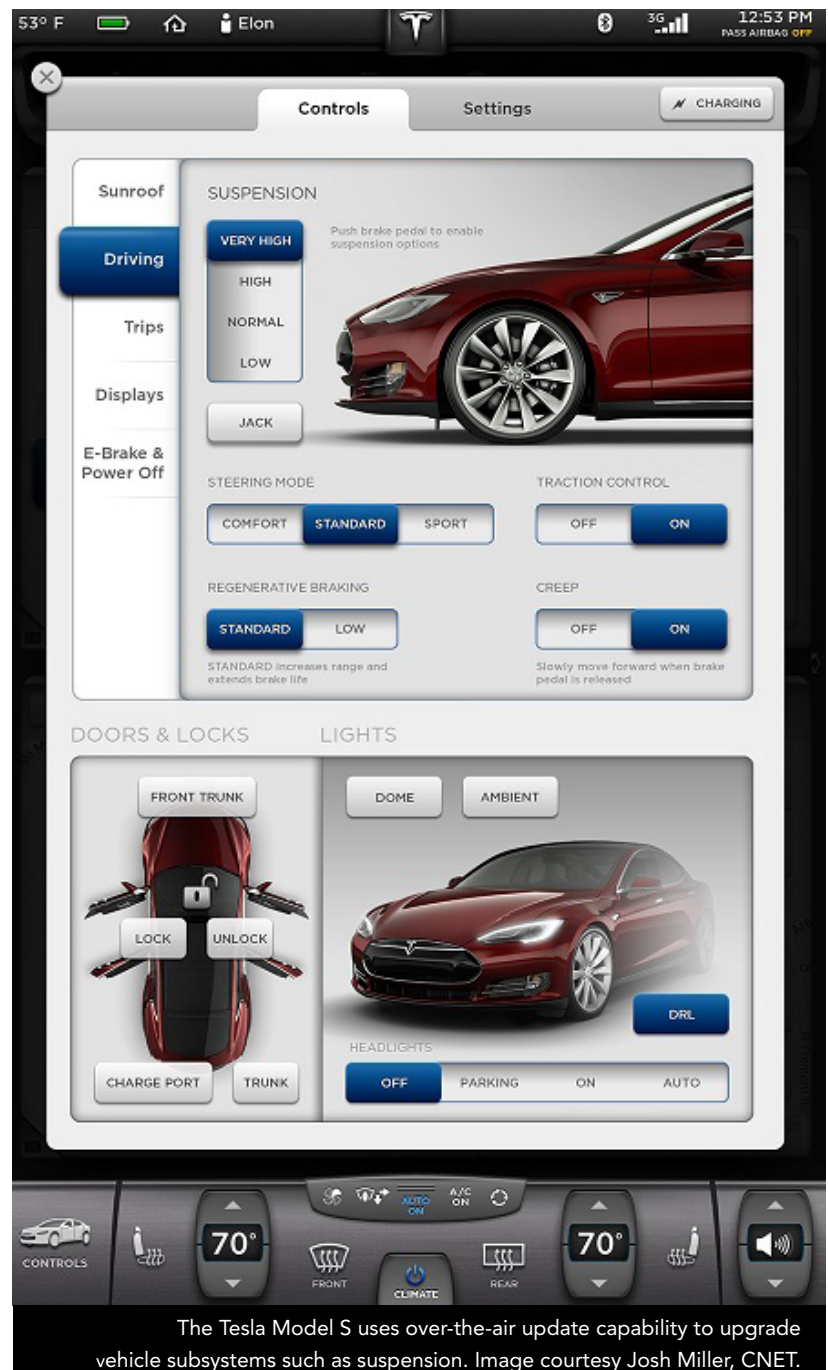


OTA update possibilities put automotive on the road to V2X

By Brandon Lewis, Assistant Managing Editor

From recall remedy to feature enhancement, auto manufacturers and Tier 1 suppliers have been investigating the possibilities of over-the-air (OTA) vehicle software updates as a way to reduce maintenance costs and improve functionality. But while automakers such as Tesla, General Motors, and Audi are moving quickly to improve vehicle connectivity and, in some cases, already deploying OTA software updates to automotive subsystems beyond the infotainment system, safety and security concerns are still being fleshed out before the automotive industry can utilize this technology on the road to vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) implementations.

In the age of connectivity it seems we can't connect anything fast enough. Watches, appliances, door locks, and light bulbs are all recent inductees into the world of the Internet enabled, helping feed our obsession for information and convenience in almost every way imaginable. In recent years, the ubiquitous connectivity movement has entered automobiles as well, with car manufacturers releasing models with the option for built-in Wi-Fi and 4G LTE services as early as this year.



The Tesla Model S uses over-the-air update capability to upgrade vehicle subsystems such as suspension. Image courtesy Josh Miller, CNET.

But beyond the ability to offer revenue-generating services and additional features through in-vehicle infotainment (IVI) consoles and head units, the long-term prospects for the connected car perhaps yield something more significant in the form of over-the-air (OTA) software updates capable of remedying software issues in other automotive subsystems. For example, while Ford, General Motors, Cadillac, and Fiat all experienced recalls related to embedded software bugs in 2014, OTA pioneer Tesla avoided a potential recall related to defective adapter plugs by issuing a remote software update, and also used the center console as a gateway to upgrade the transmission systems of Model S sedans with a creep option – as seamlessly as Apple or Samsung update your smartphone. In an industry where billions of dollars are at stake with every recall and consumer satisfaction can mean the difference between profit and loss, OTA is quickly becoming a necessity rather than a luxury, and the automotive ecosystem is mobilizing now to make the technology refine the technology for next-generation vehicles, says Andreas Dharmawan, Senior Director, Solutions and Services, Electric Cloud, Inc. (www.electric-cloud.com).

“The modern car has over 300 million lines of code (MLOC), and this is going to grow even more once self-piloted cars become common on the road,” Dharmawan says. “With 300 MLOC, you have to deliver patches because there are so many components and subcomponents that make up a system, many of which are actually being made by supply chain partners. So, because of the complexity, it’s bound to have interoperability issues, integration issues, and even bugs inside the modules. Because of this, the need to update or deliver patches increases as the size of the code grows in the car.

“If it becomes frequent that your in-car software needs to be updated, you cannot ask the service engineers to constantly be trained and service cars going into the dealer for software updates,” he continues. “That will create bottlenecks

in car dealerships across the country. And also, it’s going to be very costly, because there is no money in updating software. There is money when you have to fix the transmission, the exhaust system, and the brakes because these are hardware parts that you can sell. So, car manufacturers do not want cars to come back to the dealer for software updates because of economic reasons, and because of convenience.”

“OTA capabilities have the potential to make the development process easier,” says Marques McCammon, Senior Director, Automotive Solutions, Wind River (www.windriver.com). “Tier 1s have hundreds of systems to develop in the last cycle of car production. Often, software is the last thing that gets updated before a car is released. In the development process, imagine having hundreds of vehicles in a fleet test and the team is faced with several software revision cycles. This could be a versioning nightmare, especially as OEMs are under pressure to increasingly compress their development time. If you have ability to perform OTA updates and avoid time-consuming individual car updates, this is a huge cost and time advantage.

“One specific use case is around software-related recalls; there have been a number of safety campaigns and recalls around vehicle software in recent years,” he says. “At some OEMs the costs of managing warranty and recalls can reach into the billions of dollars. These events can impact literally millions of vehicles at a time, and several global markets. By conventional means the cost per vehicle to implement these recalls can be in the hundreds of dollars per vehicle and that says nothing about the factor of time.

“For example, if there are software issues after the car is sold, the ability to use OTA to update software or fix issues is more convenient, more efficient, and less costly than to physically bring cars into service shops,” McCammon continues. “With a robust and dependable OTA strategy, OEMs can update the automotive systems in near real-time over the lifetime of the vehicle from development through production.

Additionally, as the useful life of vehicles continues to grow, there is further opportunity to increase the value of the deployed vehicle base by continuously freshening the product experience. This may open new revenue streams to the industry that were not practical before.”

sECUring safety-critical systems

The term “connected car” typically refers to using some means of wireless communication to provide vehicles with Internet access, but rarely is it interpreted as an expansion of connectivity between the various subsystems of the vehicle itself. While IVI systems and telematics control units (TCUs) are quickly becoming the de facto methods of connecting the car in general, intra-vehicle networks linking the electronic control units (ECUs) of internal systems are required in order to enable updates such as those implemented on Tesla’s Model S. However, vicariously connecting safety-critical vehicle subsystems to the Internet also presents significant security challenges for automotive engineers, as cars are essentially transformed into clients in a vast IT ecosystem. This has resulted in a rethink of automotive security to protect both private information as well as non-IVI systems that could present safety risks if compromised, McCammon says.

“Besides IVI, we see a huge opportunity for OTA updates on more critical vehicle systems,” he says. “Many vehicle functions today are electronically actuated where they were once all mechanical. Steering, accelerator function, and braking are all systems that have moved, or are moving to electrical actuation where they were once exclusively mechanical. As these functions are critical to vehicle operation it is essential that they operate with the latest and most complete software. In these cases the ability to affect OTA updates may have real impact on a vehicle’s ability to protect the life of its occupants.

“More automotive systems are increasingly integrating and relying on software, and quite frankly any point where the data in or function of the car can be accessed remotely is a potential point of risk,” McCammon continues. “There

are several places along the data connection that must be secured or follow appropriate protocol, not only from the embedded device, but also the website where the software was launched and even to the data center. There is work currently being done in the industry to investigate the need for automotive-only data centers. Companies are looking at different modes of isolation to ensure the discrete pass of communication from device and over the air."

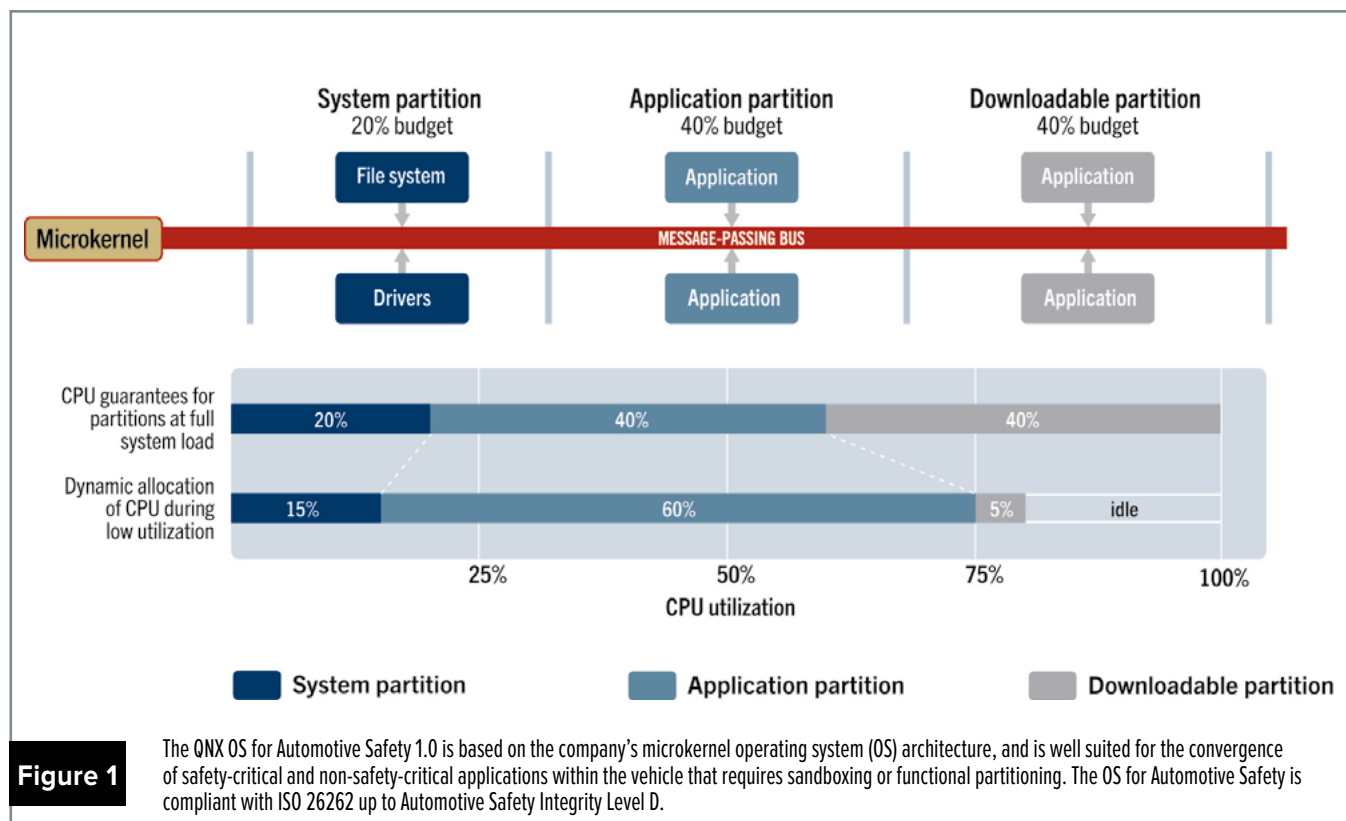
"The vehicle is becoming more connected within itself, or networked, if you will," says Grant Courville, Director, Product Management, Automotive and General Embedded, QNX Software Systems (www.qnx.com). "Within vehicles what we're seeing is an ECU, which could be the infotainment system or it could be the telematics system, that will be used as your wireless gateway to connect to the cloud. That device will connect to the cloud through a wireless connection, and then there's the interconnectivity within the vehicle where you'll perhaps see the infotainment system as your gateway for OTA software updates. Then what you're looking to do is provide software updates to the various other ECUs within the vehicle that are capable of receiving them."

"The car security architecture must be considered more with connectivity and security risks in mind. This is where machine-to-machine (M2M) technology and mobile network operators can be of significant value, especially M2M Connectivity with a car having its own M2M SIM card that can be limited to trusted parties."

— Andrew Morawski, Head of M2M, Americas, Vodafone.

"So from a security perspective, there's security of the connectivity itself, whether that's through SSL or TLS or some of the other security mechanisms you'll have for the connectivity. There's the authentication – am I talking to the server I should be talking to and is that server talking to the device it believes it's talking to? Then there's the payload itself – am I receiving the payload I should be receiving, has it been tampered with? And then there's the installation of that payload," Courville continues while explaining the company's OS for Automotive Safety (Figure 1).

"From an embedded perspective, in terms of being able to provide a software update you have to go through all of those scenarios. Let's assume that the software payload was delivered securely, properly, and now I have an image sitting there. Depending on what you're updating, you have to worry about physical space – do I have the space on my solid-state storage (SSD) to install it? When I'm installing it, are any of my active or passive systems disabled, or can I do this software update while I'm driving? And, if so, are there any systems that become disabled as a result?"



"The main fear is that the update package comes from a non-authorized back end, or a hacker, and it convinces the car that the IP address of the hacker's server is the IP address that the car should use in order to perform the software update, and an unwanted firmware will go to the car," says Yoram Berholtz, Business Line Director, Automotive, Red Bend Software (www.redbend.com). "Our solution has several mechanisms to address that (Figure 2). First is the standard security functionality that exists in the OMA-DM protocol. Second, we're partnering with leading security companies like ESCRYPT and Cisco to provide a client-server security architecture, which means that in the back end there is also a key management system so that every package we are sending to the car must first go through this key management system in order to get a signature and possibly be encrypted. Then in the ECU in the car there is a security client that analyzes if this key is correct and performs encryption. By doing so we are guaranteed that the update channel is secure."

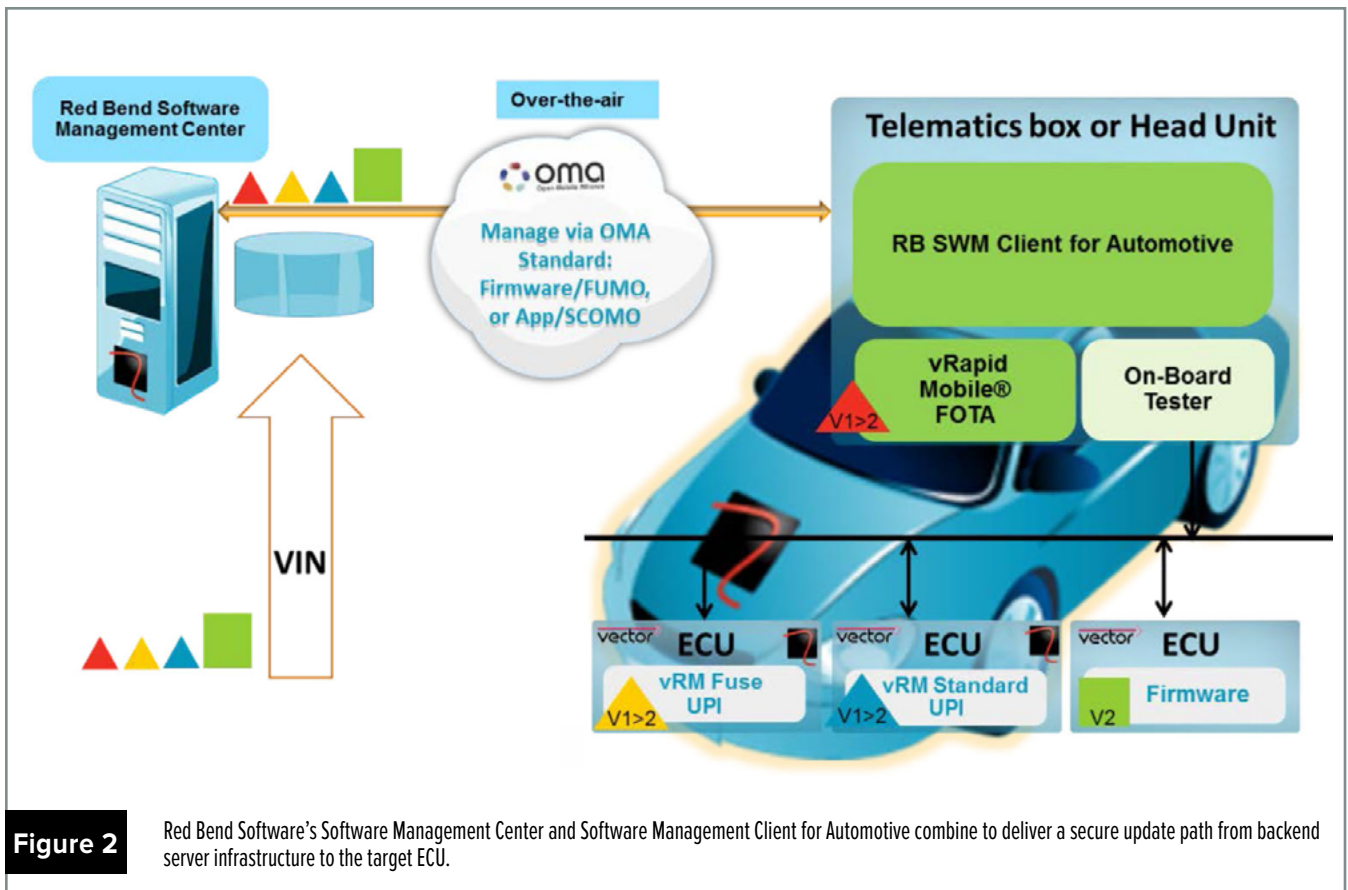
Additional automotive update challenges

Beyond the IT-style security concerns being introduced into connected cars, OTA updates must also contend with other challenges inherent to autos, for example, the reality that vehicles will inevitably travel to or through areas with insufficient cellular coverage or the possibility of a catastrophic power failure. In addition, as vehicles age there is the chance that massive updates may be required that potentially harm hardware systems, Dharmawan says.

"In the worst case, there is a scenario that I can potentially see in the future that there is so much that needs to be updated that if the upgrade fails it may damage some of the components," says Dharmawan. "In the off chance that this happens, there should be a backup software that allows the car to operate at some minimum capacity so the car is still drivable but your radio doesn't work, for example. This type of practice has been implemented in the airline industry, as well as in the aerospace and defense industry.

"So, if for some reason your new firmware is unable to install, there's always a failsafe strategy, which is basically that the system will fall back to the original firmware from when the car comes out of the factory, and then it cannot be erased because it's part of the protected memory space," Dharmawan says. "That's a backup so you can actually fall back into that mode. These types of best practices in developing high-availability, reliable software and disaster recovery processes have been around for a long time, but this discipline needs to be applied in the automotive industry."

"What happens if power goes out?" asks Courville. "I need to be able to roll back and roll back safely. Vehicles are so interconnected today that if all of a sudden you had an inoperable infotainment system, I guarantee that it would not only be the infotainment system that had been affected. Chances are you'd see other ECUs or systems within the vehicle that could have been affected to the point where potentially your vehicle



could be rendered non operational. So you start to have discussions about essentially having an image you can always fall back on, and having a secure way but also a very reliable way to make those software updates. (See sidebar, "SSDs store added security for OTA")

"There are some things that are very intuitive," Courville continues. "If I can update software incrementally instead of one big blob of software, that's obviously a lot safer, and depending on what's disabled in the vehicle, that's also much more convenient for the user."

"The solution should be designed with assumptions that connectivity will be lost, and power will be lost as well," Says Walter Buga, CEO, Arynga (www.arynga.com). "Arynga's CarSync is designed to accommodate that. It includes queuing capabilities for data delivery to vehicles, so if connection is lost the data stays in the queue, without the loss of data. The

update process in vehicles includes multiple steps and recovery procedures in case of power or other failures, and the idea is the same – we will resume for a 'lost' point and provide recovery procedures as well."

OTA to V2X

Over-the-air update technology is key to realizing a truly connected car and paves the way for more advanced automotive architectures, including autonomous vehicles that will rely heavily on vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications to navigate tomorrow's roads. With pilot V2X implementations already underway (see www.densodynamics.com), industry and government are now innovating to accelerate the future of transportation.

"Now that vehicles are very fast becoming connected, all of a sudden it opens a window of possibilities,"

Courville says. "It opens the door to having more connected vehicles, convenient vehicles, safety comes into play, and V2V/V2I – or more generically V2X – and obviously those require vehicle connectivity, either to infrastructure or other vehicles. So there are a number of initiatives under way there.

"I was at a conference in July that had to do with vehicle connectivity, and a heavy focus on V2X, and it was everything from cellular modules to the frequency spectrum that's going to be allocated to V2X technology," he continues. "So there's a lot of discussion about that, but then a lot of discussion also about best practices. For all of this to come together there's a need for private industry as well as government as regulatory bodies to all work together and we're really starting to see that. You're starting to see trials out there and pilots. That's exactly what needs to happen." **ECD**

SSDs store added security for OTA

Beyond software and networking, in-vehicle storage is fundamental to the success or failure of OTA updates. In recent years embedded storage devices for safety-critical automotive have transitioned to solid-state memory to provide the high-performance and capacity needed for connected car applications, as well as for their robust designs that help minimize mean time to failure (MTTF) in harsh vehicle environments. By also incorporating hardware-based security and power loss protection, solid-state drives (SSDs) provide another critical level of security in the event of a system attack or failure.

"To enable OTA updates, storage solutions need to defend against data breach and power loss through such features as hardware-based encryption for data security and data path protection," says Alex Schiller, Marketing Director, Automotive Business Unit, Micron (www.micron.com). "Micron has a dedicated automotive lab that incorporates power interrupt profiles from automotive systems to design and optimize storage with robust power loss protection. In the event of loss of power or connectivity during an OTA update, both static data and existing firmware are protected. Micron can provide software driver support and system validation for OTA updates, according to the actual enablement, secure authorization, and logistics the OEM wishes to implement.

"Micron's firmware is adapted to the given automotive file and operating systems and can be further optimized for the end application through system-level validation with the customer," he continues. "Micron's SSD firmware supports such features as host-initiated power management (HIPM) and device-initiated power management (DIPM) for low power modes, in addition to improved time-to-ready (TTR) through innovative power-up schemes enabling drives to be ready within 200 milliseconds as vehicles become more autonomous."



Figure 1

Micron's M500IT Automotive SSD is equipped with AEC-Q100-compliant NAND flash memory components, and includes features such as hardware-based encryption, data-at-rest protection, and adaptive thermal monitoring for connected vehicle applications.



Smart Vision Systems Poised for Takeoff

By Imagination Technologies

Next generation computer vision technologies are paving the way for new applications across a wide range of industries, from automotive to consumer electronics and retail. GPUs in application processors that are often developed primarily for the mobile market are accelerating applications in platforms ranging from cars, to mobile phones and tablets, to drones and robots.

In particular, we will begin to see a major shift in many cameras becoming vision systems, providing such capabilities as enhanced face detection, smile shutter trigger and even analytics. Moving into 2015, the rapid evolution of these "smart" cameras will drive advanced features in video and stills, including face beautification for video conferencing.

Such capabilities are also providing the basis for products addressing surveillance, home security and driver safety. One example is Mobileye, whose proprietary software algorithms and chip are able to "interpret" a scene in real-time and provide drivers with an immediate evaluation based on its analysis. Automakers have already begun to adopt this camera-based technology into their rapidly expanding safety feature applications known as Advanced Driver Assistance Systems (ADAS).

In fact, Mobileye has now shipped more than 2.5 million EyeQ2 and EyeQ3 SoCs (Systems-on-Chips) that are based on Imagination's MIPS processor architecture. These vision processors, together with Mobileye's broad

range of algorithms for mono-camera driver assistance systems, target vehicle active safety applications such as lane departure warning, vehicle detection, pedestrian detection, intelligent headlight control and traffic sign recognition. Mobileye has announced that it is leveraging Imagination's MIPS Aptiv and Series5 Warrior CPU cores in its next-generation design.

There's no doubt the classical role of the image sensor is changing rapidly, and will continue to do so in 2015 and beyond. In today's vision applications, the job of the image signal processor (ISP) is evolving away from its traditional role as a separate chip responsible only for producing the best possible picture and moving onto an SoC, where it's able to take advantage of other system resources such as the graphics processor (GPU).

The emergence of computational photography in mobile applications, as well as advanced computer vision algorithms using multi-camera arrays and higher pixel depths in every application area, have also led to the creation of a new class of smarter image signal processor cores. And emerging standards like OpenVX will help to drive the standard functions into hardware on the ISP, reducing power consumption and freeing up the GPU for even more advanced tasks.

Collaborative, heterogeneous computing is needed to address these changes and growing requirements. It's all about doing the most in the lowest power profile and leveraging all the computing resources



on the SoC. Imagination Technologies' PowerVR Raptor imaging processor, for instance, does just that by providing scalable and highly-configurable solutions which join other PowerVR multimedia cores to form a complete, integrated vision platform that saves power and bandwidth for today's camera applications and other smart sensors.

PowerVR Raptor cores are low-power, highly-configurable ISP cores designed for SoC integration. The cores can be configured to meet the needs of a wide variety of markets, including mobile, automotive, surveillance, and industrial.

By implementing optimized data paths between the ISP and other on-chip multimedia processors (graphics, video encoders and decoder, and display), system designers can minimize memory bandwidth and latency when implementing algorithms such as those used for computational photography.

Highly intelligent and integrated smart camera technologies present a potentially enormous opportunity both for businesses and consumers. In fact, BCC Research reports that the global machine vision systems market is expected to reach \$17.1 billion in 2015 and about \$26.9 billion in 2020 (opsy.st/BCCResearchReport). And, according to Yole Développement, the CMOS image sensor market will reach \$13 billion by 2018 (opsy.st/YoleDeveloppementReport). Companies that adopt and fuel these innovations early will become the leaders in their sector.

Imagination Technologies

➤ www.imgtec.com



Real-Time Frameworks, Agile Modeling, and Code Generation

By Quantum Leaps, LLC



In 2015, just as in every year before, we will need to develop more embedded code with less people in less time. The only way to achieve this is to increase programmers' productivity. And the only known way to boost productivity is to increase the level of abstraction and automate the coding process as much as possible. We also need to start reusing entire software architectures, not just individual pieces of code.

Real-time frameworks

Embedded software will increasingly be based on generic, real-time, event-driven frameworks, which will augment and eventually displace the traditional Real-Time Operating Systems (RTOS).

The software developers for the desktop, the web, or mobile devices have already moved to frameworks (.NET, Cocoa, Android, Qt, etc.), far

beyond the raw APIs of the underlying operating systems. In contrast, the dominating approach in the embedded space is still based on the venerable RTOS (or the "super-loop" at the lowest end). The main difference is that when you use an RTOS, you write the main body of the application (such as the thread routines for all your tasks) and you call the RTOS services (e.g., a semaphore or a time delay). When you use a framework, you reuse the main body and write the code that it calls. In other words, the control resides in the framework, not in your code, so the control is inverted compared to an application based on an RTOS.

The advantage is that a framework offers a much higher level of architectural reuse and provides architectural integrity (often based on proven design patterns) to the application. Typically also, a framework encapsulates the difficult or dangerous aspects of your application domain. For example, an event-driven framework based on active objects (actors) inherently supports and automatically enforces the best practices of concurrent programming such as: keeping the thread's data local and bound to the thread itself, asynchronous inter-thread communication without blocking, and using state machines instead

of the customary "spaghetti code." In contrast, raw RTOS-based threading lets you do anything and offers no help or automation for the best practices.

Many embedded software engineers might not realize that event-driven, frameworks based on active objects and state machines can be actually smaller than a conventional RTOS. For example, the open source QP active object frameworks from Quantum Leaps require only 3-4KB of ROM and significantly less RAM than a bare-bones RTOS kernel.

Agile modeling and code generation

Real-time frameworks based on hierarchical state machines will also be the key enabler for a wider adoption of modeling and automatic code generation. Of course, big and "high-ceremony" modeling tools have been tried and failed to penetrate the embedded market beyond a few percentage points. The main reason is the poor ROI (return on investment) of such tools.

However, a new class of agile tools will simplify modeling by skipping indirection layers of platform-independent models (PIMs), model transformations, action languages, etc. and will allow you to work directly with files, state machines, events, and actions expressed directly in C or C++. Obviously, such tools will be far less ambitious and lower level than "high-ceremony" tools of the past. But "lower-level" is not necessarily pejorative. In fact, the "lower-level" nature of the C programming language is the main reason for its popularity in embedded programming. We will see more innovation in the area of "low-level" and low-cost modeling and code generating tools (such as the free QM modeling tool from Quantum Leaps). This innovation will eventually bring us "modeling and code generation for the masses."

Quantum Leaps, LLC

➔ www.state-machine.com



2015: When security concerns meet safety concerns, Formal Methods become increasingly attractive

By Cyrille Comar, Co-Founder/Managing Director of AdaCore Europe

An important trend we're anticipating for 2015 is a significant move toward using "Formal Methods" for verifying safety- and security-critical software.

Because they are founded on rigorous mathematics-based techniques and often require help from mathematicians, Formal Methods have long been considered too complex for developers of embedded software. Another drawback was that they also required very extensive changes in the standard software verification strategies.

But the ever-increasing size and complexity of the software needed to control things like trains, aircraft or cars have made standard verification techniques (those mainly based on testing) increasingly more expensive and less effective. Moreover, transportation industries – whether civil or military – are facing rising pressures to deal as effectively with security threats as they do with safety concerns. The truth is, security issues can no longer be ignored when ensuring the safety of transportation software.

To this end, industries like automotive, aerospace and railway are organizing working groups with goals around both safety and security concerns in the system and software lifecycles objectives required by their industry certification standards. A first step already has been accomplished in recent years through the large-scale adoption of static analysis tools such as CodePeer from AdaCore and PolySpace from MathWorks. These static analysis tools detect the presence of typical programming errors. More

daring users may attempt to use static analysis tools to prove the absence of such errors, but static analysis tools guaranteeing the detection of all possible errors in specified categories usually come with a fair number of "false positives." In cases where they detect more errors than really exist, the developer must conduct additional analysis to verify that those cases detected by the tools are not truly errors.

In contrast, Formal Methods based on deductive verification (proofs) are not yet used widely in industry, with some notable exceptions. For instance, aviation company Airbus has used Formal Methods for significant parts of the software controlling the A380 and the A350, and the French railway industry has used them for many years.

Deductive-based Formal Methods allow developers to go one step further than static analysis techniques. They guarantee the absence of runtime errors while also allowing verification that a program completely implements its specification. They also provide a better verification than traditional unit testing since the latter verify only a limited number of test cases while the former verify all possible cases. In fact, it is possible to replace some significant parts of the testing by using such Formal Methods while increasing the quality of the overall verification. One issue was still making it impractical: For proofs to be consistent required the formal verification to be achieved on a complete system, which is challenging when some parts of the system are not amenable to such techniques.



AdaCore's Spark 2014 is a good example of a modern language that brings formal proof capabilities and a way to solve this last barrier of entry. Spark 2014 supports contracts that can be either executed or mathematically proved because they can be interpreted as regular SPARK expressions or as first-order logic formulas. With SPARK, users can combine verification techniques based on testing with those based on proofs without having to worry about the frontier between the code that is tested and the code that is formally proved. In other words, if the proof engine is intelligent enough to automatically and completely offer formal verification, some testing is made unnecessary. For areas of code that would require manual proofs and the intervention of mathematicians, standard testing techniques can continue to be used instead, making it possible to introduce such techniques gradually and benefit from them very quickly without having to completely change verification strategies. Spark 2014 and solutions like it are viable for a variety of different usages and scenarios, such as the formal proof of safety properties, complying with standards in regulated industries, performing flow analysis for security, enhancing existing Ada programming language code, combining evidence from proof and test, determining dependency relations, and more.

2015 will be the year of expanding the boundaries of safe and secure programming, and deductive-based Formal Methods will be at the heart of that expansion.



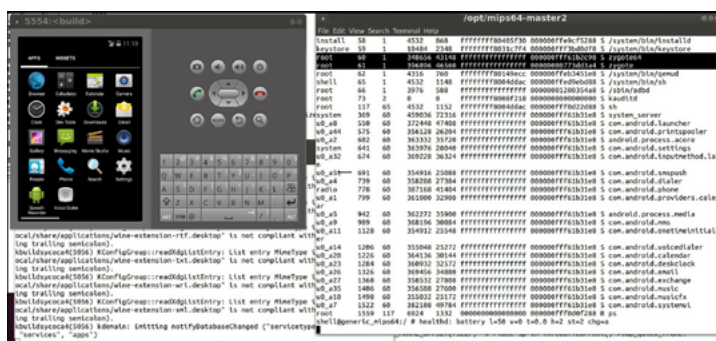
Xively | www.xively.com
embedded-computing.com/p372436

IoT platform as a service (PaaS)

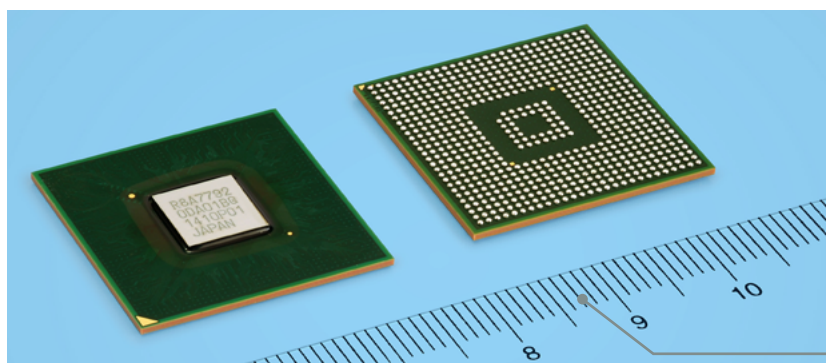
The Xively Connected Object Cloud enables the interconnection of applications, devices, data, places, and users for a variety of IoT applications. The services include a searchable directory of objects, time-series archiving data services, and provisioning and management for business applications that ride on a real-time message bus. An API using REST, sockets, or MQTT is provided for access by IoT applications, business support systems, and other objects. The platform also provides a developer workbench and management console for providing security and proper access permissions for your IoT application.

Open-source hosted emulator

Quick EMULATOR (QEMU) is an open source emulator that performs hardware virtualization. It is a hosted virtual machine monitor that emulates central processing units (CPUs) through dynamic binary translation and provides a set of device models, enabling it to run a variety of unmodified guest operating systems. QEMU can also be used purely for CPU emulation for user-level processes, allowing applications compiled for one architecture to be run on another. prpl Foundation is making QEMU available for the new MIPS release 6 architecture to enable developers to perform rapid prototyping of code and speed development around the new 64-bit MIPS Warrior I6400 CPU.



prpl Foundation | www.prplfoundation.org
embedded-computing.com/p372437



Renesas Electronics Corporation | www.renesas.com
embedded-computing.com/p372438

High-res image recognition SoC for driver assistance

The R-Car V2H system-on-a-chip (SoC) from Renesas delivers high-resolution surround-view monitoring systems with multiple cameras for advanced point-of-view switching for advanced driver assistance systems (ADAS). The R-Car V2H SoC series integrates ADAS functions with low power consumption, enhanced open source image recognition library, Ethernet AVB, and optimized video codecs for automotive networks and camera systems.



MIPS, THE CPU ARCHITECTURE FOR THE FUTURE

By Imagination Technologies

MIPS offers the industry's broadest array of low power, high-performance embedded microprocessor cores that power hundreds of millions of products around the globe.

➤ opsy.st/MIPSImgTecVideo



BUILDING INFRASTRUCTURE FOR A SMARTER WORLD AT THE INDUSTRIAL AUTOMATION CONFERENCE

By Rory Dear, Technical Contributor

The IHS Industrial Automation Conference held in London in October 2014 demonstrated how the Internet of Things (IoT), smart factories, and the Industrial IoT (IIoT) are creating smarter systems, factories, and cities.

➤ opsy.st/2014IndAutoCon



WITH ISO 26262 IN HALF THE TIME

By Rich Nass, Embedded Computing Brand Director

Functional safety verification is a must have for automotive applications. Or is it? First, let's define what the term actually means. In most cases, the term refers to ISO standard 26262, which is an adaptation of the IEC standard 61508, representing the functional safety of automotive electrical systems. Have I lost you yet?

➤ opsy.st/ISO26262Blog



THREE QUESTIONS TO ASK WHEN CHOOSING A PROCESSOR FOR MULTIMEDIA DISPLAY AND IOT APPLICATIONS

By Freescale Semiconductor

When choosing a processor for multimedia, display and IoT applications, the three primary considerations are: processing power, multimedia requirements, and interfaces with other systems. Designers must choose processors carefully, considering how their systems interface with human beings comfortable with a data-rich, multimedia world, especially as rising end-user expectations now directly influence embedded designs.

➤ opsy.st/FreescaleMultimediaWP



ACTIVE STEERING TECHNIQUES IN WIRELESS DEVICES CAN ALLEVIATE THE SPECTRUM CRUNCH

By Jeff Shamblin, Ethertronics

It's an exciting time for the wireless industry. Carriers are on a roll launching 4G networks. OEMs are debuting some of the coolest wireless devices yet, from sexy smartphones and tablets to a number of new wearables and IoT products. And consumers can't seem to get enough – they want the latest feature-packed devices, fastest data speeds, and more.

➤ opsy.st/SpectrumCrunchEthertronics



SSD E-MAG

The SSD E-mag explores how SSD storage technology is taking over from HDDs, with features examining application classes, advanced encryption techniques, and its role in automotive applications, and more.

➤ opsy.st/SSDmag2014



IOT E-MAG

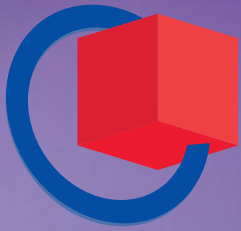
The Internet of Things E-mag deconstructs the IoT with features that investigate device/network infrastructure, comprehensive cyber security, reengineering business models, and much, much more.

➤ opsy.st/IoTEmag



Register now and
make sure of your tickets!
embedded-world.de

Nuremberg, Germany
24 – 26.2.2015



embedded world 2015

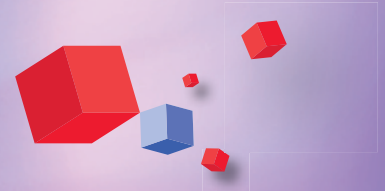
Exhibition & Conference

... it's a smarter world

THE gathering of the embedded community!

The world's biggest event for embedded technologies gets players in the embedded sector talking to each other.

Be there too when the priority is on cultivating contacts and networking at international level and setting trends.



Media partners

elektroniknet.de

computer-automation.de

energie-und-technik.de

MEDIZIN-UND-elektronik.DE

Markt & Technik
DIE UNABHÄNGIGE WOCHENZEITUNG FÜR ELEKTRONIK

**DESIGN &
ELEKTRONIK**
KNOW-HOW FÜR ENTWICKLER

elektroniknet.de
Elektronik
Fachmedium für industrielle Anwender und Entwickler

**Elektronik
automotive**
Fachmedium für professionelle Automobilelektronik

**ENERGIE
& TECHNIK**
Fachmedium für Energieeffizienz

**Computer &
AUTOMATION**
Fachmedium der Automatisierungstechnik

MEDIZIN  **elektronik**
Fachmedium für Elektronik in der Medizintechnik

Trade fair organizer

NürnbergMesse GmbH

Tel +49 (0) 9 11.86 06-49 12

visitorservice@nuernbergmesse.de

Conference organizer

WEKA FACHMEDIEN GmbH

Tel +49 (0) 89.2 55 56-13 49

info@embedded-world.eu

NÜRNBERG MESSE



Small Form Factor Computers
Intel® Atom™ E3800 and i.MX6 CPUs
Fanless -40° to +85°C Operation
Mini PCIe and IO60 Expansion



PC/104 Single Board Computers
Rugged, Stackable Form Factor
I/O Modules and Power Supplies



Industrial Computer Systems
Off-the-shelf and Custom Solutions
Fanless -40° to +85°C Operation

Single Board Computers
COM Express Solutions
Power Supplies
I/O Modules
Panel PCs



Accelerate Your Product Development Cycle

Speed up time-to-market with embedded solutions from WinSystems. Our industrial computers include expansion options, so customers can expedite prototyping and integration without the burden of CPU or carrier board design. These proven hardware platforms also provide the building blocks to create customized, application-specific designs. Products are in stock and available for immediate shipment from our Arlington, Texas facility.

Let our factory Application Engineers accelerate your capabilities.

715 Stadium Drive | Arlington, Texas 76011
Phone: 817-274-7553 | Fax: 817-548-1358
info@winsystems.com

Call 817-274-7553 or visit www.winsystems.com.
Ask about our product evaluation!

