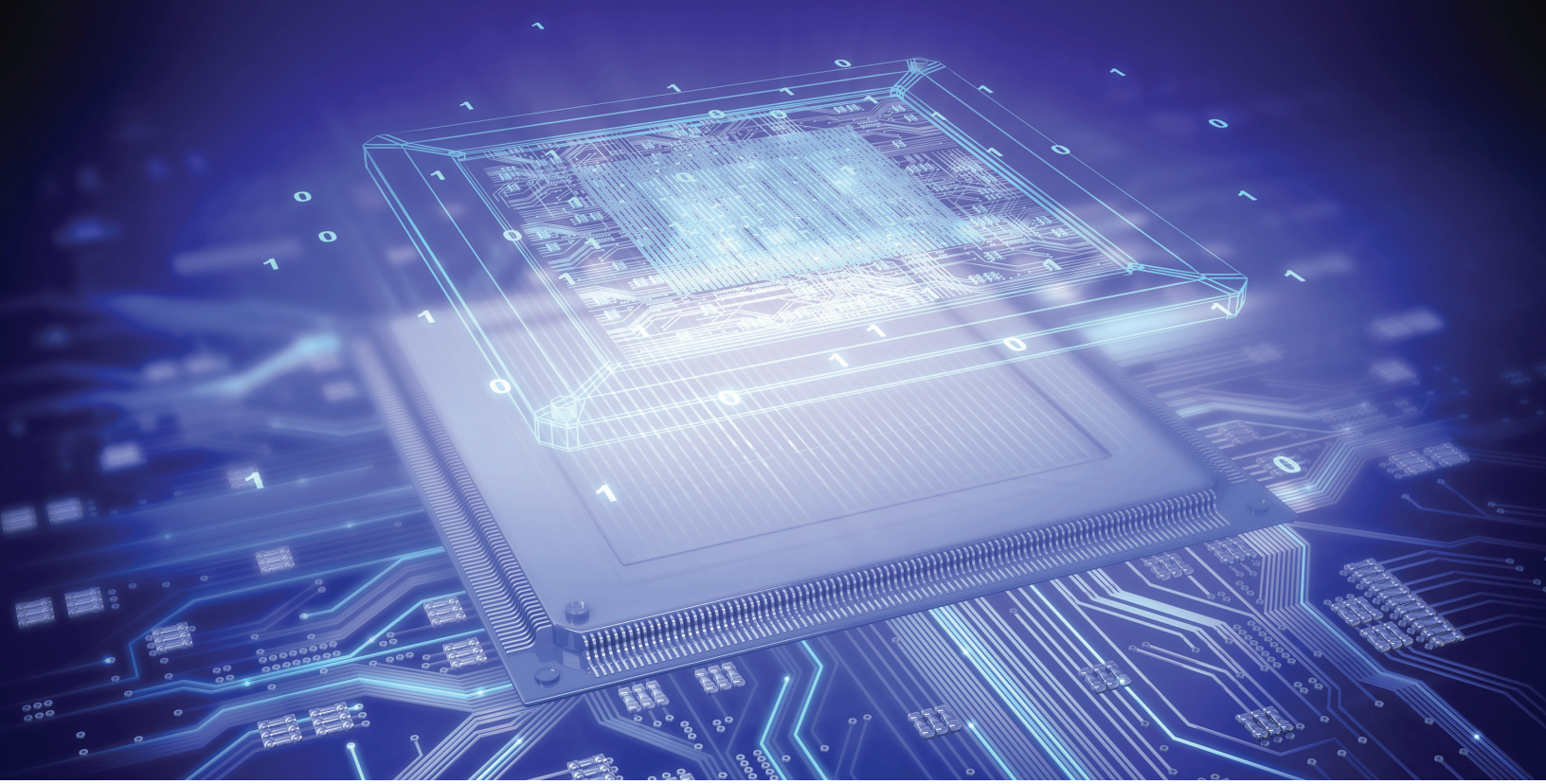

SIEMENS DIGITAL INDUSTRIES SOFTWARE

How to Reduce the Impact of the Global Microcontroller Shortage on ECU Software Development

Executive summary

The Covid-19 pandemic had a massive impact on all facets of business and commerce, as widespread supply chain disruptions rippled through every industry. Multiple factors collided to create a global microcontroller shortage that is now impacting the automotive industry, and is forcing developers to redesign Electronic Control Units (ECUs) using alternative Microcontrollers (MCUs) and to otherwise find alternative solutions for designing and testing their software.

Contents

Introduction

Using Siemens Capital® VSTAR™, developers leverage a complete set of tools and software that includes a general-purpose MCU simulation model that can be used to verify AUTOSAR-compliant software in a virtual environment using only a PC. In this way, teams can proceed with ECU development without needing actual hardware, and can arrive more easily and effectively at software that is less dependent on specific hard-to-get MCU hardware.

This whitepaper explores:

- What caused the global MCU shortage?
- What can be done to mitigate the impact?
- How can you easily transition to different MCUs?
- How can you verify your software in the absence of development hardware?
- How can you reduce MCU hardware dependencies within your software?
- How does Siemens support your ECU development efforts?

Why is the MCU Shortage a Big Problem?

The trend toward the software-defined vehicle is completely dependent on having available systems-on-chips and microcontrollers. Autonomous driving, electrification, and smart mobility services all require data-driven networks and service-oriented architectures. According to VW's CEO, Herbert Deiss, innovation in the auto industry is 90% based on software, and Roland Berger asserts that the silicon a vehicle runs on will be 30% of its content by the year 2030.

As ECUs are critical components of the modern automobile — there are up to 150 ECUs in a single vehicle — many automotive manufacturing have had to stop or significantly cut back production. For example, Ford predicts it could suffer a $2.5 reduction in operating profit as production drops by 1.1 million vehicles in 2021.

What Caused the Global MCU Shortage?

At the beginning of 2020, mobile chips were already in high demand, due to the rollout of 5G devices from Apple and other manufacturers. When Covid-19 hit and billions of people began to quarantine, demand for mobile devices increased exponentially, as people relied on them to work from home, connect with friends and family, support distance learning for their children, and access digital forms of entertainment. Manufacturers ramped up production for these types of MCUs and scaled back production for automotive chips. Demand for new vehicles plunged as shelter-in-place orders reduced the need for and the ability to travel. Additionally, when thousands of businesses closed their doors — either temporarily or permanently — many people lost their jobs, which limited their ability to make large purchases.

These factors created the perfect storm for an MCU shortage for automakers. As soon as restrictions eased, people emerged from their homes, went back to work and demand for new vehicles returned. Having scaled back production during the pandemic, most MCU manufacturers were still working to fill demand in other industries and were ill-prepared to ramp up to accommodate a surge in vehicle manufacturing. Severe weather in Texas — home to one of Samsung's largest foundries — reduced production by 28%, compounding the problem.

Unfortunately, the shortage is not likely to end soon. The Delta variant of the Covid-19 virus is continuing to impact global supply chains, particularly in India and China.

The Impact on ECU Software Design and Verification

While it is relatively straightforward to understand how the MCU shortage affects vehicle hardware production, it is not so obvious that the shortage further exacerbates production in a software-driven vehicle because ECU software development is also affected. Software updates are a continuous part of the vehicle feature delivery process, and stringent safety standards means there's no cutting corners — all of the software components in an ECU must be verified before they can be released to production.

ECU developers are thus in a quandary: they need MCUs to design new ECUs and verify their software designs and updates, but cannot get enough of them. The impact is especially problematic if ECU software is dependent on specific MCU hardware. In fact, this could shut down production altogether if updating or even switching to an alternative MCU requires a lengthy software redesign.

What can be Done to Mitigate the Impact?

Companies need real and effective solutions and methods to deal with this shortage. This paper presents several practical means to reduce the impact:

- Digital verification. Test software without hardware using virtual ECUs.
- Switching. Accelerate MCU replacement using reference platforms and expert engineering services.
- Prevention. Reduce MCU hardware dependencies within vehicle software.

Digital verification: Test software without hardware using virtual ECUs

An effective first step towards shortage mitigation is to put strategic test strategies and methods in place that lessen the impact. If you can reduce the number of hardware test benches required, you are simply less prone to delays caused by hardware availability problems. An easy way to do this is to "go digital" and test your software using virtual ECUs — which are simulation models of your ECU, including the MCU and other hardware, that can execute your actual vehicle software while exposing the same test interfaces you use to verify your software running on real hardware.

Switching: Accelerate MCU replacement using reference platforms and expert services

Mitigating the problem by completely replacing the MCU on your ECU with an available alternative may become unavoidable. When you are faced with this choice, you can reduce the impact by accelerating redesign using available, ready-to-integrate ECU reference platforms. To move even more quickly with high quality, you can take advantage of expert engineering services provided by the vendors of these rapid ECU deployment solutions.

Prevention: Reduce MCU hardware dependencies within vehicle software

The oft-learned lesson to the toughest of problems is prevention. How can we develop ECU software such that MCU shortages have less impact? One viable solution is to leverage hardware abstraction standards, such as those found within the proven AUTOSAR standard. Ensuring that your software is dependent only on hardware concepts supported by the standard — and not on specialized hardware concepts that require hardware-dependent software — means you can more easily migrate to different MCUs that are supported with standard hardware drivers. Also, and perhaps even more beneficial, is that you can use any compliant hardware platform to verify most aspects of your software, and substantially reduce the number of verification suites that do require MCU-specific testing.

AUTOSAR: Designed to Manage MCU Hardware Dependencies

As all ECUs on a network within the vehicle depend on each other to operate properly, a late hardware change of highly hardware-dependent software can cause catastrophic delays throughout the manufacturing process. Figure 1 shows a diagram of the anatomy of an ECU.

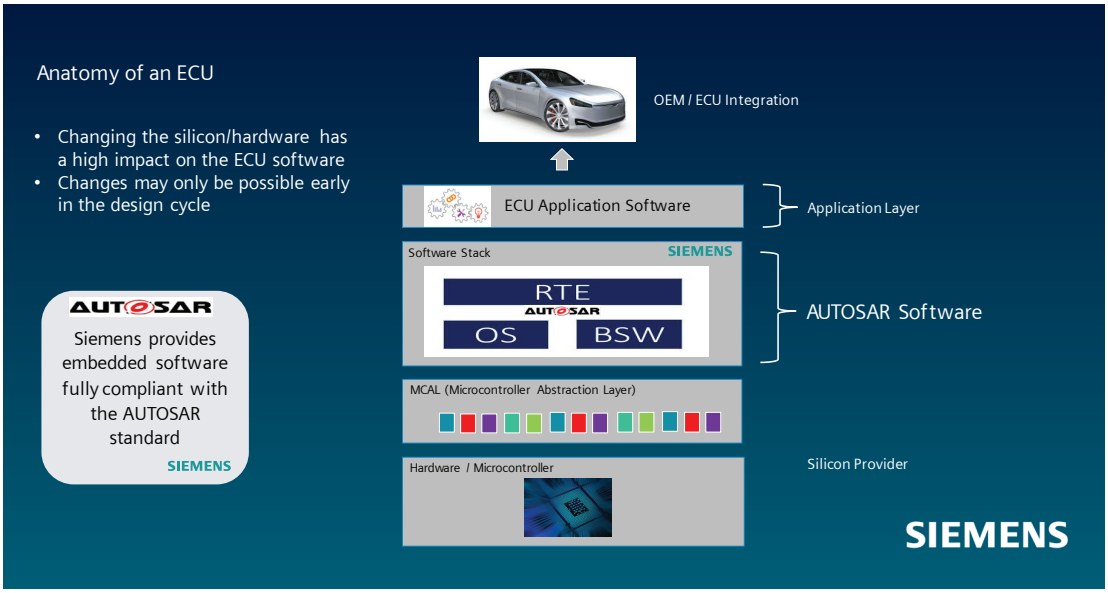

Figure 1: Diagram of an ECU.

Software developed to the AUTOSAR standard is highly portable and interfaces to the silicon through layered software that culminates within the so-called Microcontroller Abstraction Layer (MCAL). The application software that defines the behavior of a specific ECU sits on top of this layered architecture, well-isolated from hardware details. AUTOSAR emphasises the need to isolate MCU hardware dependencies behind the MCAL to manage hardware changes and dependencies. A significant advantage is the isolation of the application, sensor/actuator, ECU I/O, and all of the basic software from hardware-dependent software.

A Best Practice for Verification without Hardware

Effectively, AUTOSAR abstracts software dependencies to a concept of generalized MCU hardware — where software executes on an abstract compute resource as a so-called "generic MCU." Used strategically, this enables developers to verify and validate a substantial amount of their software independently from underlying MCU hardware details and differences.

A best practice thus naturally emerges: a development methodology where a substantial amount of software is pre-verified against the generalized AUTOSAR MCU hardware concept. To do this, developers simply use a generic MCU-based verification apparatus that exposes standardized test interfaces. Of course, this will not eliminate entirely the need for final verification on real ECU hardware, but it does allow teams to move software to the target MCU more easily, with the added confidence that the software is proven to be logically and architecturally correct.

Capital VSTAR: Delivering the AUTOSAR Promise

Capital® VSTAR™ is Siemens' professional implementation of the AUTOSAR vehicle software standard (see Figure 2). It is a comprehensive embedded software solution that includes productivity tools for efficiently configuring basic software modules and otherwise manages the overall ECU development process.

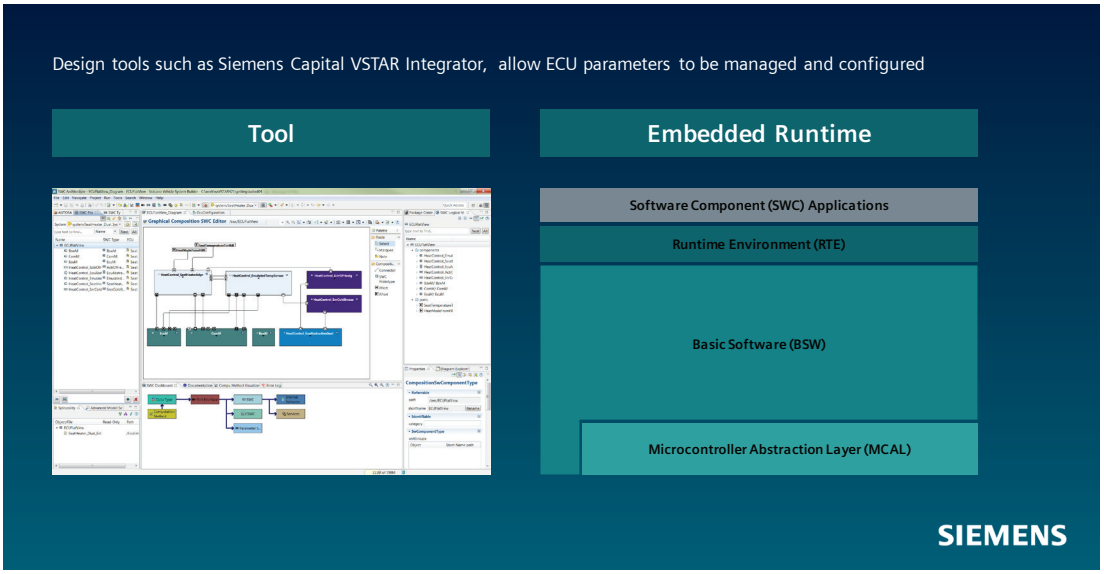

Figure 2: Siemens' Capital VSTAR platform.

Capital® VSTAR™ Integrator™: Simplifying AUTOSAR-based Development

The Capital VSTAR software configuration and integration tool is called Capital VSTAR Integrator, the front end to the highly automated Capital VSTAR engineering workflow. It includes an ECU Configuration Generator (ECG) that processes data from up-stream system-level development tools and automatically configures the Capital VSTAR software platform. It also provides design support for application software components.

Capital® VSTAR™ Virtualizer™: Verifying Software without Hardware

The Capital VSTAR tool for software verification (debug, analysis, and visualization) is called Capital VSTAR Virtualizer. It includes a generic MCU and I/O hardware simulation model — a virtual ECU or "digital twin" — that is delivered with Capital VSTAR's ready-to-integrate AUTOSAR RTE, OS, basic software, and MCAL.

When you integrate your application software into the Capital VSTAR Virtualizer AUTOSAR platform, the result is a digital representation of your ECU — its digital twin — that can be executed using only a PC against the same verification suites you use to test your real ECUs. No real hardware is required. You can test using:

- Virtual Hardware-In-the-Loop (vHIL). Connect your virtual ECU to virtual sensors, actuators, and plants.
- Networks. Use tools to inject signals over standard automotive buses.
- Diagnostics and Calibration. Connect standard tools to validate important subsystems.
- AUTOSAR-Aware Analysis. Create scripts to stimulate the system, measure its response, and visualize it.
- AUTOSAR-Aware Debugging. Use the supplied C/C++ symbolic debugger to troubleshoot behavioral problems.

Shift-Left

Using digital twins, Capital VSTAR Virtualizer delivers shift-left software development and enables testing at the earliest possible time. This helps teams identify and fix problems early in the design cycle, when it is easier and less expensive. Early access to virtual hardware enables teams to develop, share, and reuse tests throughout the development process, and collaborate more effectively.

Capital VSTAR Virtualizer: Encouraging Hardware-Independent Software

A very effective way to reduce the MCU dependencies within your software is to use Capital VSTAR Virtualizer strategically within your software development process. Confining the bulk of your software to execute within Capital VSTAR Virtualizer almost guarantees that it will execute on any AUTOSAR-compliant platform since the AUTOSAR-compliant MCAL within Capital VSTAR Virtualizer does not include any specific concepts.

Capital VSTAR Virtualizer: "Scalable-Fidelity" Eases the Transition to Hardware

Rest assured, you are not limited to verifying only hardware-independent software within Capital VSTAR Virtualizer. The simulation model is extensible through scripting and replaceable through a plug-in architecture. This means you can continue to leverage the advantages of digital twins, and step from the base generic MCU hardware to a more specific virtual MCU model (where the simulation model is based on a specific MCU variant). You can also extend the model to include ECU hardware specifics.

This concept is called scalable-fidelity. This Capital VSTAR Virtualizer feature allows developers to plug in specific MCU simulation models and use the corresponding Capital VSTAR platform software packages for their target hardware. The developer experience is forward-consistent; identical to how Capital VSTAR tools are used when targeting real MCU hardware. Also, the experience is backward-consistent; identical to the use of Capital VSTAR tools while targeting the generic MCU hardware.

Siemens Software and Integration Services Mitigates Change Impact

When the MCU shortage forces you to quickly change your ECU hardware, it will impact the performance, safety and security of your ECU. Anything that goes wrong could also impact other ECUs within the vehicle's electronic network.

Expert Engineering Services

To help you deal with these impacts, Siemens provides software and integration services to help you adapt to change and integrate your software and hardware quickly — with high quality. We help you with:

- Performance. Capital VSTAR is optimized for performance and reduced memory consumption when using multi-core MCUs. When porting an existing application to a new MCU with multiple cores, our core partitioning concept enables the CPU load to be efficiently and effectively distributed across those multiple cores.
- Safety. Capital VSTAR has been implemented with SPICE level 3 development processes and safety as a top concern, adhering to the ISO 26262 standard. A key concept in ISO 26262 is the Safety Element out of Context (SEooC); OS, E2E protection and watchdog are some of the components/packages available as SEooC with ASIL D classification.
- Security. Capital VSTAR offers multi-layered security with Secure Onboard Communication (SecOC) and standardized crypto drivers that leverage custom hardware features such as crypto algorithm acceleration and key storage. It also offers an extension for a state-of-the-art IP firewall.
- Communications. Capital VSTAR supports multiple versions of AUTOSAR (4.3.1 with support for more recent versions added with each release) as well as numerous automotive communication buses such as LIN, CAN/CAN-FD, Ethernet and FlexRay.
- Diagnostics. Capital VSTAR provides a diagnostics implementation supporting the UDS standard as well as the legislative requirements covered by the OBD standard. The implementation supports a variety of OEMs with their specific requirements.

- Calibration. Capital VSTAR provides a measurement and calibration implementation supporting the XCP standard.

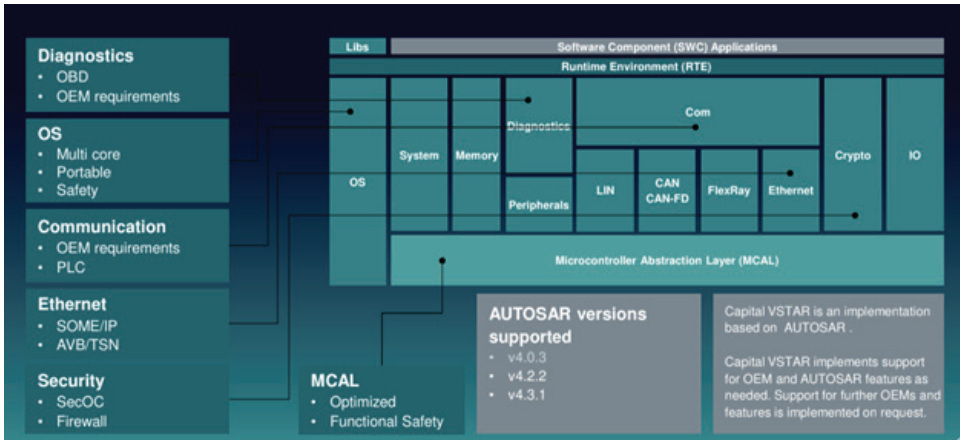

Figure 3: Capital VSTAR supports a variety of industry-standard protocols ECU system components.

Siemens has dedicated AUTOSAR experts who can engage with you to optimize your ECU performance and accelerate your deployment. For example, if your software must be ported to a new MCU with less memory than the current MCU, our Siemens experts can review the configuration, make suggestions for optimization, and help define the needed architecture to fit onto the smaller MCU. Additionally, we can develop AUTOSAR-compliant transceiver drivers for external peripherals, assist with MCAL integration, and help prepare the AUTOSAR-compliant package for final production.

Pre-Ported MCUs Reduce Development Time

We at Siemens have analyzed the market, and determined the most commonly used MCUs for automotive applications. We then pre-ported Capital VSTAR to these MCUs. Using pre-ported MCUs can reduce the ECU development lifecycle by up to several months by enabling verification and validation to begin sooner. Pre-ported reference platforms are available for popular MCU families from Infineon, NXP, Renesas, ST, and others.

Conclusion

Siemens understands the global MCU shortage, and is dedicated to helping companies mitigate the crisis by shifting left with techniques for verifying software without hardware using virtual ECUs, accelerating MCU replacement using reference platforms and offering expert engineering services, and helping them reduce MCU hardware dependencies within their software to effectively reduce future shortage impact.

Of course, making a change in any design process is always easier if the tools used are integrated. Siemens promotes a full design flow, from requirements through system-level configuration, that includes embedded software development and electrical engineering for electrical systems design. Our overall solution adds continuous verification and validation, integration with project and application lifecycle management tools, and mechanical design tools to improve design quality, reduce costs, and accelerate time to market for new, innovative ECUs.

These embedded solutions are part of Siemens's Capital portfolio for the automotive marketplace, designed to support the development of autonomous driving solutions, electrification, and mobility services. Learn more about our solutions for Automotive and Transportation at https://www.plm.automation.siemens.com/global/en/industries/automotive-transportation/

Learn more about our AUTOSAR embedded software solutions here: www.siemens.com/autosar

Siemens Digital Industries Software

Americas: 1 800 498 5351

EMEA: 00 800 70002222

Asia-Pacific: 001 800 03061910

For additional numbers, click here.

About Siemens Digital Industries Software

Siemens Digital Industries Software is driving transformation to enable a digital enterprise where engineering, manufacturing and electronics design meet tomorrow. Xcelerator, the comprehensive and integrated portfolio of software and services from Siemens Digital Industries Software, helps companies of all sizes create and leverage a comprehensive digital twin that provides organizations with new insights, opportunities and levels of automation to drive innovation. For more information on Siemens Digital Industries Software products and services, visit siemens.com/software or follow us on LinkedIn, Twitter, Facebook and Instagram. Siemens Digital Industries Software – Where today meets tomorrow.

About the author

Andrew Patterson leads the Automotive Embedded Software business at Siemens Digital Industries. The business specializes in embedded software solutions addressing the latest trends in automotive electronics, covering ADAS, Digital Cockpit, Electric Vehicle and Autonomous Driving applications. Prior to his current position, Andrew spent over 20 years in the design automation business, specializing in automotive technologies including wire harness design, automotive simulation model development, virtual prototyping, and mechatronics. He is heavily involved in AUTOSAR, vehicle architectures, and the industry standardization of automotive enabling technologies. Andrew holds a master's degree in Engineering and Electrical Sciences from Cambridge University, UK.

Chintu Parikh, Application Engineer (Technical Sales), has a demonstrated history of working in the Automotive industry and has been working on AUTOSAR and Audio solutions for Siemens for five years. He is a strong engineering professional with a Master of Science (M.Sc.) focused in Embedded Systems from University of Stuttgart.

Lance Brooks, Principal Engineer at Siemens, has more than 30 years of experience creating embedded systems and related development tools, and over 15 years dedicated to automotive software development, with a specific focus on hardware/software modeling and simulation.

siemens.com/software

© 2021 Siemens. A list of relevant Siemens trademarks can be found here. Other trademarks belong to their respective owners.

84141-D1 9/21 K