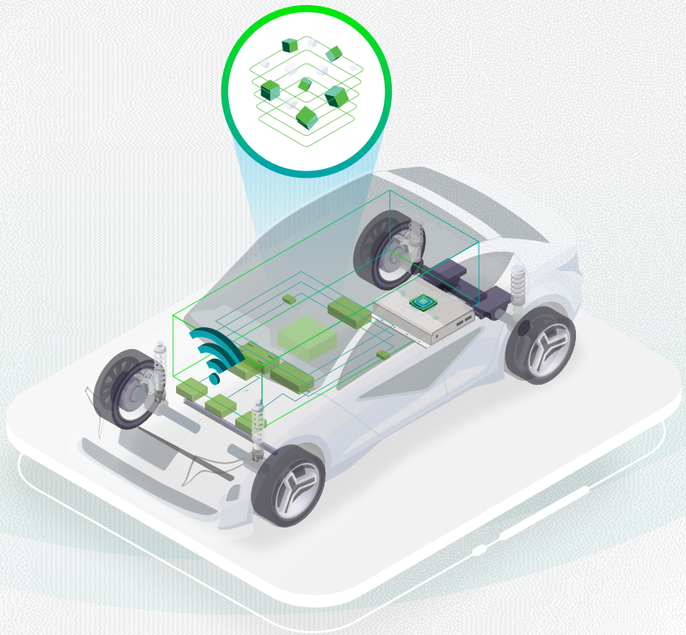
Real-time ECU software development: Challenges, methods and tools



Short development cycles

Challenges

- Fast time to market
- Evolution after start of production (SOP)
- High-quality standards
- Verification of “OEM“ requirements

Solutions

- EB tresos, real-time basic software (BSW) as an off-the-shelf product
- Continuous feature enchantments on EB tresos major versions
- Reliable basic software with state-of-the-art processes
- Requirement compliance of EB tresos AutoCore OEM Extension

Development efficiency

Challenges

- Highly automated software development process
- AUTOSAR integration knowledge

Solutions

- Support for virtualization workflows
- Worldwide individual support

Software complexity

Challenges

- Integration of complex real-time applications
- Variant management
- Integration of third-party software
- Micro controller unit performance enhancement through hardware acceleration

Solutions

- Multicore capable BSW
- AUTOSAR post-build
- Open tooling environment with EB tresos Studio
- Support for gateway communication acceleration

Safety and security

Challenges

- Integration of fail-safe applications
- Integration of safety fail-operational application
- Protection of ECU against cyberattacks
- Support for onboard security features
- Utilization of hardware acceleration

Solutions

- Safety solutions up to highest ASIL levels
- Solutions for fail-operational communication and execution environment
- Cybersecurity management system (CSMS) and reporting of software vulnerabilities
- Implementation of cryptography and security features
- Support for HSM firmware on various microcontrollers

inquiries@elektrobit.com · elektrobit.com/eb-tresos