Scalable and Flexible Software Platform for High-Performance ECUs

Dheeraj Sharma, Product Expert
August 23, 2018
Agenda

A  New E/E Architectures and High-Performance ECUs

B  Non-Functional Aspects: Safety | Security | Cloud

C  Handling Software Development Complexity of High-Performance ECUs
Future Value - Created through Software

Value of a car: yesterday vs. tomorrow

Source: Morgan Stanley Research
E/E Architectures with HPC seem Clean and Simple

**80-100** ECUs
- 6 CAN-Bus
- 2 FlexRay
- 1 Ethernet backbone

**4** High-performance ECUs
- 60 Sensor/Actuator ECUs
- 1 Ethernet backbone
- 1 CAN per zone
Possible HPC Architecture for SOP in 2019

Infrastructure software (Operating system and middleware)

- New CPU-intensive (safety-relevant) functions: e.g. sensor fusion
- Novel user functions: e.g. App Store
- Takeover of existing vehicle functions from Classic AUTOSAR (SWCs)
- Secure startup, authentication
- Safety-relevant vehicle functions, monitoring of performance partitions

Virtual machine

Hypervisor

Secure Boot

Performance cores

HSM

Safety cores
Communication in a Service-Oriented Architecture

Public speech

Bulletin board
New Services Require Changes to Gateway

- Open/close door
- Detect crash
- Lock while driving
- Key
- Open via smartphone
- Ego data
- Provider speed
- Authentication

Gateway

Service 1
Service 2
Service 3
Service 4

Service and communication to be adapted.

© Elektrobit (EB) 2018 | Public | All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights
Service Discovery Phase to Find and Match Services

- Open/close door
- Detect crash
- Lock while driving
- Key
- Open via smartphone
- ... (Other services)
- Ego data
- Provider speed
- Authentication
- ... (Other data)
- Service 1
- Service 2
- Service 3
- Service 4

- Services can be distributed flexible on various ECUs
- No dependency or changes in Gateway necessary
Additional Non-Functional Requirements Arise

- Security
- Cloud connectivity
- Functional safety
Reconfiguration of Services

Requirements for reconfiguration

Req. 1: Services can be dynamically relocated

Req. 2: Sensor/actuators are redundant or accessible via network as a service
TEE (Trusted Execution Environment)

Security Stack for Performance Controllers

- Customizable implementations on security cores are the trust anchor in ECUs.
- Root of trust in hardware Trusted Platform Module
- Provides a generic security interface on top of security cores with API for Adaptive Applications
- Security basic software to enable security solutions – Enables secure boot, secure updates, secure debug, secure logging and secure storage.

© Elektrobit (EB) 2018 | Public | All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of applications for industrial property rights
Software Platforms for HPC: what lies ahead of us?

1. Incremental Development
   - Feature-based approach
   - Flexible handling of changes

2. Test and Validation
   - Analysis tools
   - Validation processes

3. Management of Software Variants
   - Reduction of software variants
   - Total cost of ownership view
Agile and Lean Methods to Address Complexity and Change

Sequential development

- Software architecture often created in layers with long turnaround cycles

Incremental development

- Focus on delivering features in short cycles

Agile and lean elements are already used today
- Continuous Integration, Continuous Delivery
- Test automation
- Daily (stand-up) meetings

Scaling agile methods for large projects

<table>
<thead>
<tr>
<th>Scale</th>
<th>Small</th>
<th>Med - Large</th>
<th>Enterprise</th>
<th>Small</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Focal point</strong></td>
<td>Team/structure Inter-team dependencies</td>
<td>Org. descaling, team/structure Agile thinking, PO scale via &quot;areas&quot;</td>
<td>Team/structure Customizable but prescriptive framework</td>
<td>Scrum concepts and mindset at scale</td>
</tr>
</tbody>
</table>

HMI: Human Machine Interface, OS: Operating System, PO: Product Owner
Right Tools to Analyze HPCs?

- Model-based testing
- Communication protocol tests
- Restbus simulation
- Hardware-in-the-loop tests

» Wide range of established tools and processes

Only few hardware-in-the-loop solutions in the market

» Which party in the supply chain must and can test functionalities?
Amount of Hardware Variants Increases Software Costs

Variation of hardware requirements for HPC

**Processor variants**
- 2 Micro processors
- 1 Micro controller, 1 micro processor
- 1 Micro controller, 1 micro processor, 1 GPU
- 1 Micro controller, 2 GPUs

**Performance**
- 10k ... >100k DMIPS

**Network**
- 1-16 CAN buses
- 8/24 LIN buses
- 0-8 FlexRay buses
- 1 ... 7/11/20 Ethernet ports
Successful Introduction of Software Platforms for HPC Needs…

1. **New methods** e.g. incremental development to cope with complexity and changes

2. **Aligned tools** to analyze and validate software and behavior of HPC

3. **Active management** and reduction of software variants and **total cost of ownership view** on software
Thank you.

Questions?

www.elektrobit.com
Dheeraj.Sharma@elektrobit.com