Safety and security related features in AUTOSAR

Dr. Stefan Bunzel – AUTOSAR Spokesperson (Continental)
Co-Authors: S. Fürst, Dr. J. Wagenhuber (BMW), Dr. F. Stappert (Continental)
Automotive - Safety & Security 2010
22 June, 2010, Stuttgart
Overview

Background of safety and security in automotive E/E development

- Overview AUTOSAR software architecture
- Safety related features
- Security related features
Safety and Security in Automotive E/E Development

- **Safety:**
  “With the trend of increasing complexity, software content and mechatronic implementation, there are increasing risks from systematic failures and random hardware failures.”
  (ISO DIS 26262 Road vehicles — Functional safety)

- **Security:**
  ... means protecting a system and its information and data from unauthorized access, use, disclosure, disruption, modification or destruction
ISO 26262:
• provides an **automotive safety lifecycle** (management, development, production, operation, service, decommissioning) and supports tailoring the necessary activities during these lifecycle phases;
• provides an **automotive specific** risk-based approach for determining **risk classes** (Automotive Safety Integrity Levels, ASILs);
• uses ASILs for specifying the item’s **necessary safety requirements** for achieving an acceptable residual risk; and
• provides **requirements for validation and confirmation measures** to ensure a sufficient and acceptable level of safety being achieved.
Why Security in Automotive E/E Development?
Political and Social Issues

Legal regulations requiring additional security measures in vehicles
(Regulation EC 692/2008)

Political and administrative workgroups realize the dependency between safety (“Betriebssicherheit”) and security (“IT Sicherheit”) resulting in new legal requirements regarding security in the automotive domain.
Overview

- Background of safety and security in automotive E/E development

Overview AUTOSAR software architecture

- Safety related features
- Security related features
**AUTOSAR Vision**

AUTOSAR aims to improve complexity management of integrated E/E architectures through increased reuse and exchangeability of SW modules between OEMs and suppliers.
**AUTOSAR Vision**

AUTOSAR aims to standardize the software architecture of ECUs. AUTOSAR paves the way for innovative electronic systems that further improve performance, safety and environmental friendliness.

- Hardware and software will be widely independent of each other.
- Development can be de-coupled by horizontal layers. This reduces development time and costs.
- The reuse of software increases at OEM as well as at suppliers. This enhances quality and efficiency.

![Diagram showing AUTOSAR Vision](image-url)
AUTOSAR – Core Partners and Members
Status: May 6, 2010

9 Core Partner
- BMW Group
- BOSCH
- DAIMLER
- PSA PEUGEOT CITROËN
- VOLKSWAGEN AG

39 Premium Member
- DELPHI
- MAGNA
- ETAS
- intec
- ALTRAN
- dSPACE
- intec
- KIT Cummins Infosystems Limited
- ETRI
- Fraunhofer
- ETRI
- ETRI
- The MathWorks

11 Development Members
- itemis
- SGS
- SYMTA VISION
- KERELA
- OPENSYNERGY
- OFFIS
- TÜV NORD
- VALIDAS

57 Associate Members

5 Attendees

Up-to-date status see: http://www.autosar.org
9 Project Objectives and 3 Main Working Topics

PO1: Implementation and standardization of basic system functions as an OEM wide “Standard Core“ solution

PO2: Scalability to different vehicle and platform variants

PO3: Transferability of functions throughout network

PO4: Integration of functional modules from multiple suppliers

PO5: Maintainability throughout the whole “Product Life Cycle“

PO6: Increased use of “Commercial off the shelf hardware“

PO7: Software updates and upgrades over vehicle lifetime

PO8: Consideration of availability and safety requirements

PO9: Redundancy activation
AUTOSAR
Specifications vs. Products

AUTOSAR compliant products
• SW modules
• Tools
• ...
• ECU
• Cars
• ...

Members

Core Partners,
Premium, and
Development Members

AUTOSAR Standard Specifications
• Architecture
• Methodology
• Appl. Interfaces

AUTOSAR Releases
R4.0, R3.1, R3.0, ...

Cooperate on standards,
compete on implementations.

Build

Apply

Partnership
The AUTOSAR Architecture distinguishes on the highest abstraction level between three software layers running on a microcontroller.

- The Application Layer
- The Run Time Environment (RTE)
- Basic Software (BSW)
The AUTOSAR Basic Software consists of the layers: Services, ECU Abstraction, Microcontroller Abstraction and Complex Drivers.

The BSW layers are further divided into functional groups.
- Examples of Services are
  - System
  - Memory
  - Communication Services
AUTOSAR Architecture

Layered Software Architecture

- Application Layer
- AUTOSAR Runtime Environment (RTE)
- Services Layer
- ECU Abstraction Layer
- Microcontroller Abstraction Layer

Breakdown to / Implementation on ECU

ECU Software Architecture

- Application Software Component
  - AUTOSAR Interface
- Actuator Software Component
  - AUTOSAR Interface
- Sensor Software Component
  - AUTOSAR Interface
- AUTOSAR Software
- Application Software Component
  - AUTOSAR Interface

AUTOSAR Runtime Environment

- Standardized Interface
  - Operating System
  - Standardized Interface
- Standardized Interface
  - Services
  - Standardized Interface
- Standardized Interface
  - Communication
  - Standardized Interface
- AUTOSAR Interface
  - ECU Abstraction
  - Standardized Interface
- AUTOSAR Interface
  - Complex Device Drivers
- AUTOSAR Interface

Basic Software

ECU-Hardware

7 July, 2010

Safety and security related features in AUTOSAR
AUTOSAR Development Methodology Principle

- AUTOSAR description templates:
  - SWC description: application software
  - ECU description: ECU characteristics and configuration
  - System description: network and assignment of SWCs to ECUs

- Descriptions for
  - SWCs
  - ECUs
  - system description

- allow a tool-based deployment of SWCs to ECUs
Overview

- Background of safety and security in automotive E/E development
- Overview AUTOSAR software architecture

Safety related features

- Security related features
AUTOSAR methodology according to ISO26262

- Functional Safety Concept: 3-8
- Specification of Technical Safety Requirements: 4-6
- Specification of SW Safety Requirements: 6-6
- Software architectural design: 6-7

Steps:
1. Design System
2. Design Sub-System
3. Generate ECU Extract
4. Develop an Application Software Component
5. Develop BSW Module
6. Build AUTOSAR ECU software
AUTOSAR methodology according to ISO26262

AUTOSAR Supports safety by offering standard safety mechanisms

Core Tests, Flash tests... E2E protection Memory partitioning...

AUTOSAR SPECIFICATIONS

Software Requirements (SRS)  Software Specifications (SWS)

BSWs  BSWs Config  SW-Cs  Safety related CDDs

Software architectural design  Specification of SW Safety Requirements  6-6

Some safety requirements in ISO26262 part6 are related to SW implementation

Specification of Technical Safety Requirements  4-6

Functional Safety Concept  3-8

SYSTEM

SW

SW implementation
**AUTOSAR Safety Features**

- **Memory partitioning**: separate software applications from each other in order to avoid any data corruption between applications.
- **Defensive behavior**: prevent data corruption and wrong service calls in the AUTOSAR basic software on microcontrollers having no hardware support for memory partitioning.
- **End-to-end communication protection**: protect applications against the effects of faults within the communication link.
- **Program flow monitoring**: control the temporal and logical behavior of applications.
- **Time determinism and timing constraints modeling**: model and implement proper and deterministic timing behavior.
  - Synchronized time bases (i.e. a "global time") across ECU networks,
  - Synchronized execution and deterministic timing of application software components,
  - Controlling the timing behavior and detection of timing violations at runtime,
  - Timing constraints like end-to-end (e.g. sensor-to-actuator or communication) delays, minimum/maximum execution times of runnable entities, or constraints on the triggering rate of events.
- **Hardware testing and checking**: AUTOSAR basic software modules to test hardware (e.g. RAM-Test, Core-Test) and to check the integrity of stored data (e.g. EEPROM Manager).
PARTITIONING

- Partitions are used as fault containment regions.
- Partitions can be terminated or restarted during run-time as a result of a detected error.
- Partitions are configured in the ECU-C.
AUTOSAR Release 4.0 Example for Partitioning

1. A violation (error) has occurred in the system (e.g., memory or timing violation)

2. The partition is terminated by the OS, cleanup possible – communication is stopped

3. The partition is restarting, initial environment for partition set up

4. The partition is restarted and up and running
AUTOSAR Release 4.0
Safety End to End (E2E) Communication Protection

Typical sources of interferences causing errors

Detected by E2E protection

SW-related sources

HW-related sources
AUTOSAR Release 4.0
Safety End to End (E2E) Communication Protection

Typical sources of interferences causing errors
Detected by E2E protection

SW-related sources
HW-related sources
Overview

- Background of safety and security in automotive E/E development
- Overview AUTOSAR software architecture
- Safety related features

Security related features
Security Use Case Examples

- Secure Programming of ECUs
  - Programming only by authorized entities
  - Programming only with original OEM approved software
  - Application (in bootloader) uses standard cryptographic routines and services, e.g. hash, signature verification, and public key encryption (= asymmetric encryption)

- Electronic Immobilizer
  - Protect the vehicle from any unauthorized driving
  - Technical details are totally OEM dependent
  - But: Immobilizer application always uses a specific set of cryptographic routines and services

- Electronic enabling of functions
  - Only a specific subset shall be enabled for regular usage of the car
  - Uses special data structures with cryptographic signature

- Secure diagnosis
  - Only dedicated entities are allowed to use certain diagnostic services
Each main security use case corresponds to a security application.

- Secure Flashing Authentication & Signature
  - MD5, xxx-MAC, RSA

- Function Enabling SWC
  - MD5, RSA, DES, DH

- My Use Case myApp
  - SHA-1, HMAC, RSA, AES

Basic cryptographic routines:

- Each security application uses a different set of cryptographic services.
- Communality of cryptographic routines may lead to slightly different crypto implementations or to duplicated code.
Security and Cryptographic Architecture

Security Use Cases and corresponding security applications

- Separation of security application and cryptographic routines

<table>
<thead>
<tr>
<th>Secure Flashing Authentication &amp; Signature</th>
<th>Function Enabling SWC</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crypto Module</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MD5</td>
<td>SHA-1</td>
<td>SHA-256</td>
</tr>
</tbody>
</table>

- Crypto Module manages requests for cryptographic services from applications and dispatches to a pool of cryptographic basic routines
  - Standard generic interface from above for applications
  - Standard generic interface from below for basis routines (cryptographic services as plugins)
  - Management of internal states
  - Transparent access to crypto hardware devices
Crypto Module exposes an interface for security applications to allow for a generic access to standardized cryptographic routines.

Crypto Module exposes an interface for cryptographic routines to allow for arbitrary implementations to plug-in into crypto module and for use by security applications.

- Cryptographic routines may be offered by different vendors each specified for certain technologies (RSA, ECC, …)
- Security application is not aware of special realization of crypto routine
- Crypto routine may be realized even in hardware without notice of application
Security in AUTOSAR
Embedding of Crypto Module

- Crypto service manager (CSM) in system services of service layer
- Configurable and common access to cryptographic methods

Optional (*):
- Support for cryptographic hardware
Summary

- AUTOSAR has become a global standard for embedded automotive software, providing specifications for
  - Software architecture
  - Development methodology
  - Standardized application interfaces

- Already former releases (R2.1, R3.0, R3.1) can be used for safety related systems. With the R4.0 and further releases safety related systems are more and more supported.

- Security in AUTOSAR enables the use of state-of-the-art cryptography in the automotive domain with standardized interfaces

- AUTOSAR is a key enabler for managing the growing E/E complexity

- First series cars with AUTOSAR technology are on the road
Thank you for your attention!

http://www.autosar.org

request@autosar.org
Became a member and get exploitation rights for the AUTOSAR standard.

Published Releases
For information only, see disclaimer.