Introduction of AUTOSAR R23-11

Qian Jiamin
14 Mar 2024

AUTOSAR China day
Introduction of AUTOSAR Release R23-11

- Charging Interface
- Service Discovery Control for Application Software
- DDS Support on CP
- Secure SOME/IP-ACL
- Deterministic Communication with TSN
- Time Validation
- MACsec
- Extended AP & CP
- Extended AP
- Firewall
- Safe API for Hardware Accelerators
- Service Oriented Vehicle Diagnostics
- Tracing For Adaptive Platform
Safe API for HW Accelerator

Architecture

The code is supplied by the vendor of ECU knowing its HWA features best.
Safe API for HW Accelerator

Functionality

► Data storage and management:
  ▪ We have big sets of data
  ▪ That must be stored safe AND efficiently for computing

► Task execution:
  ▪ Queueing
  ▪ Event and error handling

► Device management and monitoring:
  ▪ Abstraction of HWA
  ▪ Provide status (healthy, unhealthy, high load …)

► Runtime configuration:
  ▪ Configuration of HWA resources according to machine manifest
Many sensor values are only valid with a valid time stamp.

If one value depends on several values from different ECU then these values should be taken at the same time.

If this value is safety relevant then it shall be computed on the base of safe inputs.

Also a safe time value.

AUTOSAR offers a time value. How to make it a safe time value?
Time Validation

An ECU shall have a local instance of a global time in ASIL B quality.

The ASIL B step is done with a HW timer, independent from StbM.
MACsec

- MACsec is a secure communication protocol on Data Link Layer
- All higher-level protocols (IP, TCP, …) are protected with MACsec
- MACsec operates on a low layer, utilizing HW acceleration feature for high throughput
- MACsec operates too close to the HW for a detailed specification in Adaptive AUTOSAR
- New document AP_EXP_MACsec provides guidelines how to use MACsec with Adaptive AUTOSAR
Firewall

► Protects ECU by filtering unwanted Ethernet communication

► Supports stateless filtering, stateful filtering and deep packet inspection (SOME/IP, DoIP, DDS, generic)

► Firewall for CP (R23-11) has the same goal/use-cases as Firewall for AP (R22-11) + Firewall on switches

► Configuration of firewall rules via ARXML
Introduction of EXP_IAM

► AUTOSAR Identity & Access Management protects services and resources from unauthorized use in Adaptive

► The IAM specification is distributed among many documents
  ▪ Policy enforcement in functional cluster specification
  ▪ IAM configuration in AP_TPS_ManifestSpecification
  ▪ SWS_IdentityAndAccessManagement: Mostly explanatory, not much specification

► IAM specification can be confusing due to its distributed specification

► With R23-11, SWS_IAM has been replaced with EXP_IAM to give a clear picture of the feature
Secure SOME/IP-ACL

This concept introduces the possibility of limiting the Ethernet communication “SOME/IP” from known permitted listed authenticated communication partners to specific service instance, so secured service instance can only be accessed (Offered, Subscribed, or Consumed) by defined partners.
Secure SOME/IP-ACL

Client/Server Subscription Sequence “Classic Platform” with ACL check injected
Charging Interface

Motivation

To support the e-mobility, the worldwide established charging infrastructure shall be supported by AUTOSAR according to the ISO 15118-2:2014.

▪ AC & DC charging.
▪ EIM (external identification means) & PnC (plug and charge).
▪ Digital communication between electric vehicle EV and EVSE (electric vehicle supply equipment) as per ISO-15118-2/20.
▪ Supported by the AUTOSAR Classic Platform.
▪ Charging over Ethernet protocol.
Charging Interface

Charging Manager (ChrgM)

► ChrgM will be a part of Communication Services and will communicate with other AUTOSAR modules to support the charging process.


► The EXI (efficient XML interchange) is responsible for compression of the XML data for faster processing.

► The V2GTP (vehicle to grid transport protocol) defines the structure of the PDU, which is the header and payload definitions.
Service Discovery Control for Application Software

Motivation

Service Oriented Communication on classic platform is today typically used with “auto offer / auto subscribe” where SD is performed during startup phase of all ECUs.

If Application Software does not need Service Oriented Communication from the start, it shall be possible to trigger Service Discovery by the Application at any point in time – both for Server and Clients.

User Benefit

Control Service Discovery functionality by Well defined Interface and Behavior of the BswM for Service Oriented Communication.

Figure 3.3: Service Discovery Control Flow Overview
Service Discovery Control for Application Software

Use Cases

▪ A Software Component in Service Server Offer / Stop-Offer the Service.

▪ Service Server checks whether there are any subscribers of his offered service.

▪ Service Client subscribes to a Service.

▪ Service Client checks whether the subscription was successful.
Deterministic Communication with TSN

Extensions to Enhance Precision for Global Time Synchronization

- Support neighbour rate ratio calculation specified by IEEE802.1AS-2022
- Support configuration of up to two hardware clocks
  - PTP (Precision Time Protocol) hardware clock, with adjustable offset and rate
  - free running hardware clock
- Support configuration of PPS (Pulse Per Second) signal generation to validate quality of synchronized global time
Support transmission and reception of IEEE1722 streams
- Audio and video streams: AAF (AVTP Audio Format), 61883_IIDC, RVF (Raw Video Format)
- Control streams: CRF (Clock Reference Format), TSCF (Time Sensitive Control Format), NTSCF

Extend Ethernet Switch Driver to support the following features:
- passive stream identification specified by IEEE802.1CB
- Per-Stream Filtering and Policing specified by IEEE802.1Q-2022
- extension of Credit Based Shaper, introduction of Asynchronous Traffic Shaper

Introduce HW supported data transfer (e.g. DMA) for efficient handling of large data transfer
What is the Adaptive Platform all about?

**Mixed-critical automotive extension to POSIX**
- Standard automotive connectivity
- Automotive-specific functional add-ons
- Functional Safety and Cyber Security
- E/E Architecture development
  - Top-Down Workflow for distributed development
  - Formal exchange formats
Logical Architecture View - overview

AUTOSAR Runtime for Adaptive applications - ARA

**Platform Foundation Functional Clusters (FCs)**

- FCs provide platform local functionality as C++ libraries

**Standardized App/Interface FCs**

- FCs provide standardized interfaces or applications

**Vehicle Service FCs**

- FCs provide vehicle-wide functionality possibly covering multiple AP machines

**POSIX PSE51 / C++ STL Operating System Interface**

- PSE51: single-process profile of POSIX standard
- STL: C++ 14 Standard Template Library

**Platform Service FCs**

- FCs provide platform-specific functionality covering a single AP machine

Base Software

- POSIX OS

New AP Architecture 14 Mar 2024
FCs provide standardized interfaces or applications
Logical Architecture View - base

AUTOSAR Runtime for Adaptive applications - ARA

POSIX PSE51 / C++ STL Operating System Interface

Platform Foundation Functional Clusters (FCs)
- Communication Management
  ara::com
- Execution Management
  ara::exec
- Diagnostic Management
  ara::diag
- Intrusion Detection System Manager
  ara::dsm
- Network Management
  ara::nm
- Persistence
  ara::per
- Time Synchronization
  ara::tsync
- Cryptography
  ara::crypto
- Raw Data Stream
  ara::rds
- Platform Health Management
  ara::phm
- Log and Trace
  ara::log
- Firewall
  ara::fw
- Core
  ara::core

Platform Service FCs
- Update and Config Management
  ara::ucm service
- Sensor Interfaces
  ara::sensorif service
- State Management
  ara::sm service
- Platform Health Management
  ara::phm service
- Log and Trace
  ara::log service
- Communication Management
  ara::com service
- Execution Management
  ara::exec service
- Diagnostic Management
  ara::diag service
- Intrusion Detection System Manager
  ara::dsm service
- Network Management
  ara::nm service
- Persistence
  ara::per service
- Time Synchronization
  ara::tsync service
- Cryptography
  ara::crypto service
- Raw Data Stream
  ara::rds service
- Platform Health Management
  ara::phm service
- Log and Trace
  ara::log service
- Firewall
  ara::fw service
- Core
  ara::core service

Standardized App/Interface FCs

Vehicle Service FCs

Vehicle Update and Config Management
ara::vucm service

Base Software

POSIX OS
Logical Architecture View - protocols

User Application

Application

AUTOSAR Runtime for Adaptive applications - ARA

POSIX
PSE51 / C++ STL Operating System Interface

Communication Management
- SOME/IP, S2S, DDS, zero-copy IPC
- UdpNM

Network Management
- ara:nm

Raw Data Stream
- ara:rds

Firewall
- ara:fw

Execution Management
- ara:exec

Persistency
- ara:per

Platform Health Management
- ara:phm

Core
- ara:core

Diagnostic Management
- ara:diag

Time Synchronization
- ara:tsync

Log and Trace
- ara:log

Intrusion Detection System Manager
- ara:idsm

Cryptography
- ara:crypto

Platform Foundation Functional Clusters (FCs)

Platform Service FCs

Update and Config Management
- ara:ucm service

Sensor Interfaces
- ara:idsm service

Vehicle Service FCs

Vehicle Update and Config Management
- ara:vucm service

Standardized App/Interface FCs

ISO 23150

User Application Protocols

- gPTP
- DLT
- SOME/IP; S2S; DDS; zero-copy IPC
- TCP, UDP
- ISO 23150

Base Software

POSIX OS

New AP Architecture 14 Mar 2024
## Logical Architecture View - security

### AUTOSAR Runtime for Adaptive applications - ARA

<table>
<thead>
<tr>
<th>Platform Foundation Functional Clusters (FCs)</th>
<th>Platform Service FCs</th>
<th>Standardized App/Interface FCs</th>
<th>Vehicle Service FCs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication Management</td>
<td>Update and Config Management</td>
<td>Sensor Interfaces</td>
<td>Vehicle Update and Config Management</td>
</tr>
<tr>
<td>SecOC, TLS, Ipsec, MACsec</td>
<td>IDS</td>
<td>ara::idsm</td>
<td>ara::sensoritf service</td>
</tr>
<tr>
<td>Execution Management</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::exec</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trusted Platform</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network Management</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::rm</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Persistence</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::per</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Secure storage</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(confidentiality, authenticity)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time Synchronization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::tsync</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Secure time</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cryptography</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::crypto</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Certificate and key management, access to</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>crypto primitives</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Platform Health Management</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::phm</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log and Trace</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::log</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Authenticated logging</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diagnostics</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::diag</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Raw Data Stream</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::rds</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Firewall</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::fw</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Core</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ara::core</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Firewall</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Base Software

<table>
<thead>
<tr>
<th>POSIX OS</th>
</tr>
</thead>
</table>

## New AP Architecture

- **POSIX**
  - PSE51 / C++ STL
  - Operating System Interface

## Logical Architecture View - security

- User Application

## Logical Architecture View - security

- Application

## Logical Architecture View - security

- Base Software
Logical Architecture View – safety

AUTOSAR Runtime for Adaptive applications - ARA

platform Foundation Functional Clusters (FCs)

- Communication Management
  ara::com
- Execution Management
  ara::exec
- Diagnostic Management
  ara::diag
- Intrusion Detection
  System Manager
  ara::dsm
- Platform Health Management
  ara::phm
- Log and Trace
  ara::log
- Platform Health Management
  ara::status

Platform Service FCs

- Update and Config Management
  ara::ucm service
- Sensor Interfaces
  ara::sensorif service
- Vehicle Update and Config Management
  ara::vucm service

Vehicle Service FCs

- Safe E2E Communication
  ara::com
- "Authentic" boot,
  Resource Groups
- Persistence
  ara::per
- Time Synchronization
  ara::tsync
- Cryptography
  ara::crypto
- State Management
  ara::sm service
- Coordination of Degradation
  ara::coord
- Exception-less
  API design

Base Software

- POSIX
- PSE51 / C++ STL
- Operating System Interface

User Application
AUTOSAR Release R23-11 in numbers

1726  Incorporation Tasks in all 3 AUTOSAR Standards
1245  Discussed Change Requests within the AUTOSAR community
198   AUTOSAR Partner collaborating in 20 Working Groups
328,757 Adaptive Platform code lines to demonstrate and validate the Adaptive Platform standard
11    New Concepts for Adaptive and Classic Platform to tackle future goals
2     Standardized AUTOSAR Software Platforms to
1     Define one holistic E/E System Architecture for Future Intelligent Mobility
Thanks a lot for your attention!
Enjoy the coming deep dives.