Autonomous Driving –
From Fail-Safe to Fail-Operational Systems

Rudolf Grave
December 3, 2015
Autonomous Driving – From Fail-Safe to Fail-Operational Systems

Agenda

• About EB Automotive
• Autonomous Driving
• Requirements for a future car infrastructure
• Concepts for fail-operational systems

• Summary
In-car infrastructure solutions

We provide products and engineering services for in-car infrastructures to address your project-specific electronic control unit (ECU) requirements

- Architecture development and software integration for ECUs
- Full AUTOSAR support with one basic software stack and one tool environment
- Tailor-made products, services, and support for all leading OEMs
- Meeting latest automotive technologies like functional safety, security, Ethernet
- Extensive partner ecosystem: car manufacturers, 3rd party tool vendors, and microcontroller manufacturers
Autonomous Driving – From Fail-Safe to Fail-Operational Systems

Agenda

- About EB Automotive
- Autonomous Driving
- Requirements for a future car infrastructure
- Concepts for fail-operational systems
- Summary
Autonomous driving

Valet Parking

High automation

Partial automation

High automation with fun

Vision of transport
## Levels of Autonomous Driving (AD)

<table>
<thead>
<tr>
<th>degree of automation</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driver</td>
<td>Driver only</td>
<td>Assisted</td>
<td>Partial automation</td>
<td>Conditional automation</td>
<td>High automation</td>
<td>Full automation</td>
</tr>
</tbody>
</table>

### Driver in the Loop
- **Driver only**: Yes (required)
- **Assisted**: Not required

### Time to Take Control Back
- **Driver only**: Not applicable
- **Assisted**: ~ 1s
- **Partial automation**: Several seconds
- **Conditional automation**: Couple of minutes

### Other Activities while Driving
- **Driver only**: Not allowed
- **Assisted**: Specific
- **Full automation**: All (even sleeping)

### Examples
- **FCW, LDW**: Forward Collision Warning, Lane Departure Warning
- **ACC, LKA**: Adaptive Cruise Control, Lane Keeping Assistant
- **Traffic Jam Assistant**: Highway Chauffeur, Valet Parking, Robot car

Source: SAE, NHTSA, VDA
Autonomous Driving – From Fail-Safe to Fail-Operational Systems

Agenda

• About EB Automotive
• Autonomous Driving
• Requirements for a future car infrastructure
• Concepts for fail-operational systems
• Summary
Requirements for a future car infrastructure

• Main drivers
  – Automated Driving
  – Car-2-X applications

• Requirements
  – High computing power
  – High data rates
  – High availability, fail-operational systems
  – Update over the air
## Requirements for a future car infrastructure

<table>
<thead>
<tr>
<th>High Level Requirements</th>
<th>Technical Concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td>High computing power</td>
<td>High <strong>Performance Controllers</strong> and GPUs</td>
</tr>
<tr>
<td>High data rates</td>
<td><strong>Ethernet</strong> (1 GigE, 10 GigE)</td>
</tr>
</tbody>
</table>
| High availability, fail-operational systems | **Redundancy Concept**  
**Service oriented architecture** (SOA)  
**Dependable Communication Software System Engineering** |
| Car-2-X communication, update over the air | Reliable **Security** mechanisms, concepts and infrastructure |
Contemporary car infrastructure

- Basic software mostly based on AUTOSAR or similar proprietary system

Pro:
- Efficient on small microcontrollers
- Well suited for time-critical, safe and secure applications

Contra:
- Only proprietary solutions for fail-over and redundant functionality
- Fixed, inflexible communication mechanisms
Future architecture of a car infrastructure

- Split up ECUs in low performance IO Controller and high performance controller
- Establish a service oriented architecture (SOA)

- **Performance Controller**
  - High computation power
  - Widespread, POSIX-like Operating System (e.g. Linux), Adaptive AUTOSAR

- **IO Controller**
  - Provide Sensor and Actuator Services
  - Deeply embedded, real-time Operating System (e.g. Classic AUTOSAR)
How to divide the functionalities?

### Performance Controller
- calculations
  - driver assistant functions
  - non-high speed functions
  - Software varies on available functions

### IO Controller
- physics
  - high speed functions
  - filter processing
  - Calculation with strong timing constraints (e.g. no jitter)
  - Software varies on attached sensors/actuators
Benefit of performance controller

Performance Controller
- request IOs/data on demand (SOME/IP)
- can be updated over the air (new functions, bug fixing, function on demand)
- substitute each other (fail-operational)
## Requirements for a future car infrastructure

<table>
<thead>
<tr>
<th>High Level Requirements</th>
<th>Technical Concepts</th>
<th>Technologies</th>
</tr>
</thead>
<tbody>
<tr>
<td>High computing power</td>
<td>High <strong>Performance Controllers</strong> and GPUs</td>
<td>• Autosar Adaptive Platform, Hypervisor</td>
</tr>
</tbody>
</table>
| High data rates         | **Ethernet** (1 GigE, 10 GigE) **Dependable Communication** | • Fault-tolerant Communication  
• QoS and Timesync  
• Safe & Secure Communication |
| High availability, fail-operational systems | **Redundancy Concept**  
**Service oriented architecture**  
**Software System Engineering** | • 2oo3, 1oo2D,...  
• *(Semi-)* dynamic reconfiguration |
| Car-2-X communication, update over the air | Reliable **Security** mechanisms, concepts and infrastructure | • Secure Onboard Communication & Key management  
• Crypto Algorithms, Security HW  
• Secure Separation |
Agenda

• About EB Automotive
• Autonomous Driving
• Requirements for a future car infrastructure
• Concepts for fail-operational systems

• Summary
Fault Propagation in Systems

- **Component A**
  - Internal Dormant Fault
  - Error
  - Failure
  - Detection/Compensation

- **Component B**
  - Error
  - Failure
  - Detection/Compensation

- Detection/Reaction

- **Environment**
  - Hazard
  - Accident

- **System**
  - Fault-tolerant system
  - Fail-safe system
  - Fail-operational system
  - Not-Safe system

Basic Concepts and Taxonomy of Dependable and Secure Computing, Avizienis et al., 2004
Current Systems (usually fail-safe)

Failure Detected?

- Deactivate / degrade function → Safe State
- Inform the driver
- Report a diagnostic error

Standard approach in many safety relevant systems:
- Airbag, ESP, air conditioning, battery charging, ...
- Driver assistant functions such as adaptive cruise control, lane assist, ...

Some functions provide a degraded mode, sometimes limited in time:
- Electronic Power Steering
- Braking
From Fail safe to Fail operational

Safe State means:

- **Continue driving until driver is in the loop**
  - approx. 7-15s for conditional autonomous driving
  - Several minutes for high and full autonomous driving
- **Perform an autonomous „safe-stop“ (stand-still at a non-hazardous place)**
  - Main issue is to get the driver attention focused on the situation
  - Several minutes, depending on the situation
**1\textsuperscript{st} approach: 2 channels with comparison**

- Two ECUs working on the input data and compare the output data
1st approach: 2 channels with comparison

- Two ECUs working on the input data and compare the output data
- A “2 channels with comparison-system” is simply fail-safe and since it is not possible to distinguish between “ECU1 not ok” and “ECU2 not ok”.
- The safe state is a complete system shutdown, which is not acceptable for autonomous driving
Improving Availability by Redundancy

• Aerospace domain
  – Space Shuttle: 5 identical general purpose digital computers
  – Saturn V: triple redundancy

• Avionics
  – Boing 777: triple triplex
  – Airbus: Triple redundancy plus software diversity
2nd approach: 2oo3 systems

- A well established pattern
- If one of the ECUs fails, the system can continue with the remaining two ECUs.
- Failures in the input data can be detected by an “Input-Voter”. 
2nd approach: 2oo3 systems

- The “2 out of 3 system approach” is a well established pattern.
- If one of the ECUs fails, the system can continue with the remaining two ECUs.
- Failures in the input data can be detected by an “Input-Voter”.

![Diagram showing the 2 out of 3 system approach with ECUs and a voter.](image-url)
Avionics vs Automotive Domain

Automotive:
- Time to reach safe state < 5min
- It is assumed unlikely that a further independent failure occurs, whereas in avionics time to reach safe state several hours
2003 systems applicable for automotive?

- More ECUs
- More wiring
- More weight
- More power consumption
- More complexity

**Key question: What does it mean to the car driver?**

According to 2 independent studies by KPMG 2013 and autelligence2015, customers would pay 1500 – 3000$ more for an autonomous driving car (mid-size)

-> **2003 can be hardly realized due to costs issues.**
3rd approach: 1oo2D System

- High diagnostic coverage needed to detect failures in one channel
- If a component fails in one of the two channels, the system does not shut down
- The system continues to operate with one channel
3rd approach: 1oo2D System

- High diagnostic coverage needed to detect failures in one channel
- If a component fails in one of the two channels the system does not shut down
- The system continues to operate with one channel

**Common sense:**

*It’s not best policy to operate a highly safety critical system on a single channel – but it’s sufficient for a certain period of time, the so called hand-over-time to the car driver*
Outlook: Reconfiguration for rebuilding 1oo2D

- Still Operational
- Handover to driver
- Failure recovery
- Internal recovery

< 10s

1oo2D*
- Rebuilding 2-channel-system
- Disabling of comfort functions
1oo2D - Normal operation

1oo2D system

Fault tolerant
Ethernet

Sensors
/Actuators

Diagnostics

Critical

Disabled

Non-critical
1oo2D – 1 channel

Fault tolerant
Ethernet

Sensors /Actuators

Diagnostics

1oo2D system

Func1

Func2

Func3

Diagnostics

critical

disabled

non-critical
**Requirements for Reconfiguration**

- **Req. 1:** Functions can be dynamically relocated
- **Req. 2:** Sensor/Actuators are redundant or accessible via network as a service
Req. 1: Reconfiguration in classic AUTOSAR systems

- Application information based on AUTOSAR xml description available
- Runtime environment (RTE) supporting starting and stopping of software components
- Threads can started/stopped in EB tresos Safety OS via partitions
- FailOpManager
  - Monitoring of own health status
  - Monitoring of foreign health status
  - Triggering of reconfiguration
 Req. 2: Sensor/Actuators are redundant or accessible via network

Redundant Sensor/Actuators
• Duplication and higher costs
• Only limited reconfiguration of vehicle lifetime due to hardwired sensors

Sensor/Actuators are accessible via network
• Service orientated communication (SOME/IP and Service Discovery)
• Multi-cast fault-tolerant Ethernet
Agenda

- About EB Automotive
- Autonomous Driving
- Requirements for a future car infrastructure
- Concepts for fail-operational systems

- Summary
Summary

• Re-use of available integrity mechanisms from fail-safe systems is the basis for building fail-operational systems.
• Software systems that are designed to achieve a high diagnostic coverage are available today.
• Fault tolerant Automotive Ethernet is available today.
• Established concepts for fail-operational system are available and can be reused in automotive systems with cost constraints.

Let’s build the next generation software systems for autonomous driving!
Contact us!

automotive.elektrobit.com
Rudolf.Grave@elektrobit.com