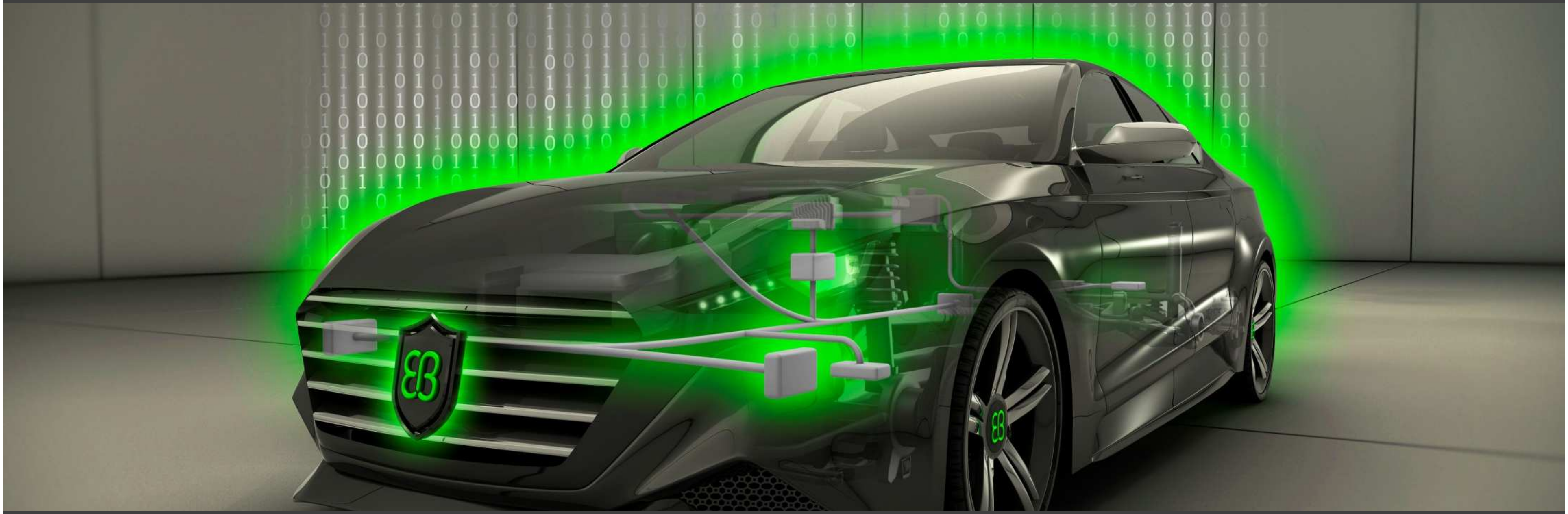
Autonomous Driving – From Fail-Safe to Fail-Operational Systems

Elektrobit

Rudolf Grave
December 3, 2015



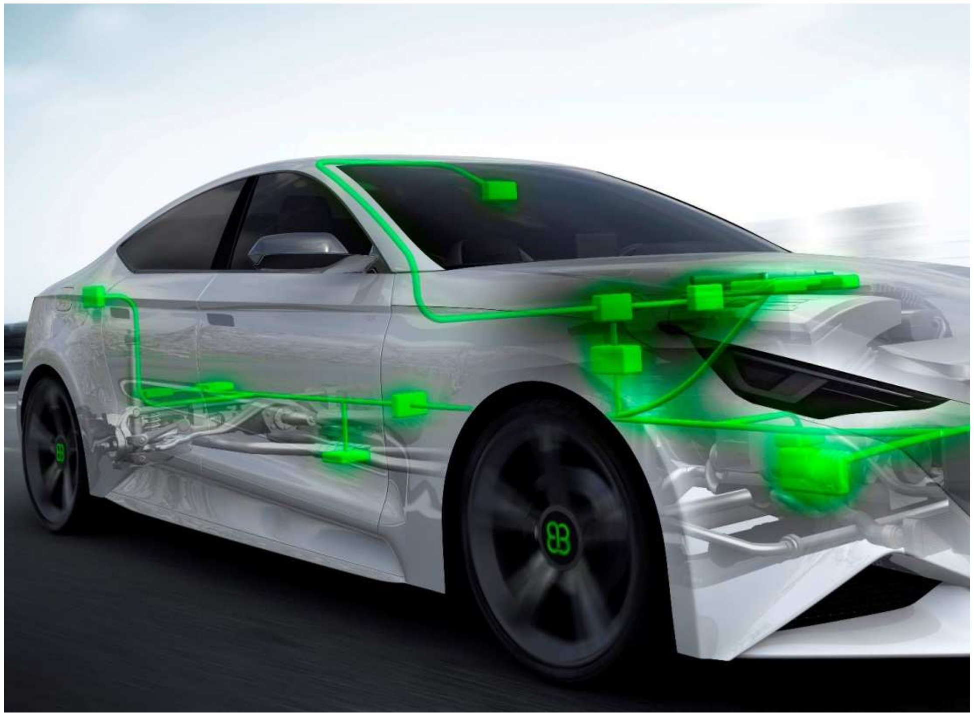
Agenda

- 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



Agenda

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

Autonomous driving

Valet Parking

High automation

Vision of transport

Partial automation

High automation with fun

Pictures taken from http://www.bmwblog.com/

Levels of Autonomous Driving (AD)

degree of automation

Driver → Auto-mation

	Driver only	Assisted	Partial automation	Conditional automation	High automation	Full automation
SAE	0	1	2	3	4	5
NHTSA	0	1	2	3	4	
driver in the loop	yes (required)			not required		
time to take control back	-	~ 1s		several seconds	couple of minutes	
other activities while driving	not allowed			specific	all (even sleeping)	
examples	FCW, LDW	ACC, LKA	Traffic Jam Assistant	Highway Chauffeur	Valet Parking	Robot car

FCW ... Forward Collosion Warning
LDW ... Lane Departure Warning

ACC ... Adaptive Cruise Control
LKA ... Lane Keeping Assistant

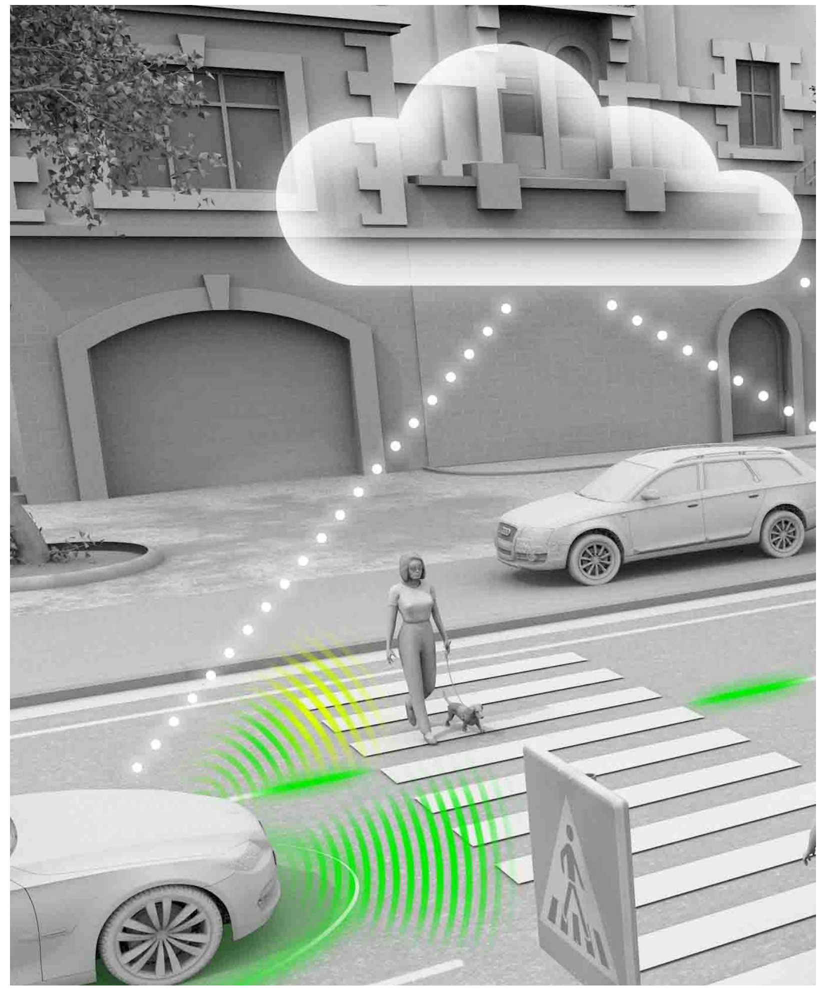
Source: SAE, NHTSA, VDA

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

High Level Requirements	Technical Concepts
High computing power	High Performance Controllers and GPUs
High data rates	Ethernet (1 GigE, 10 GigE)
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

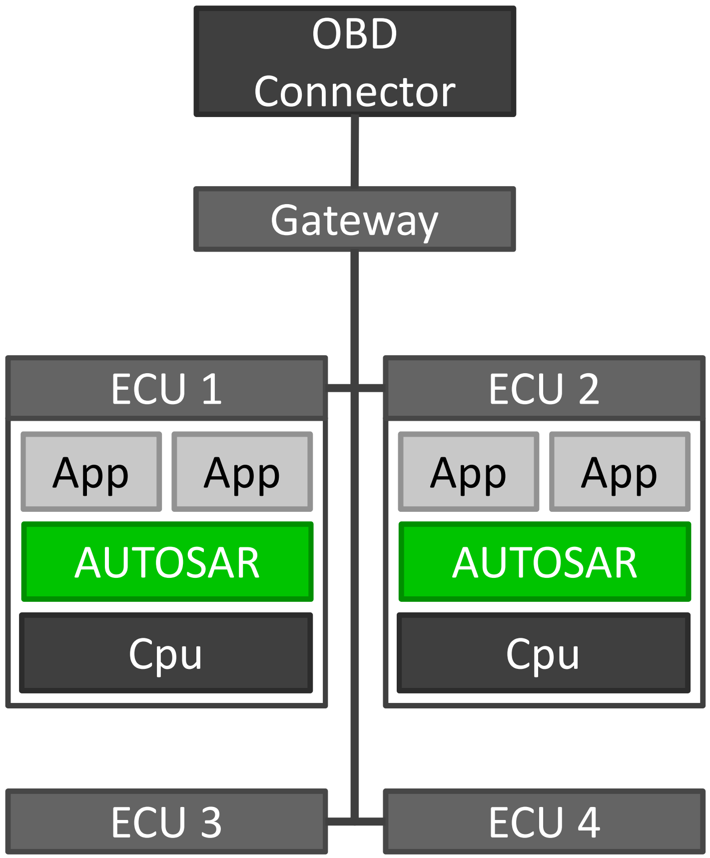
- Basic software mostly based on AUTOSAR or similar proprietary system

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

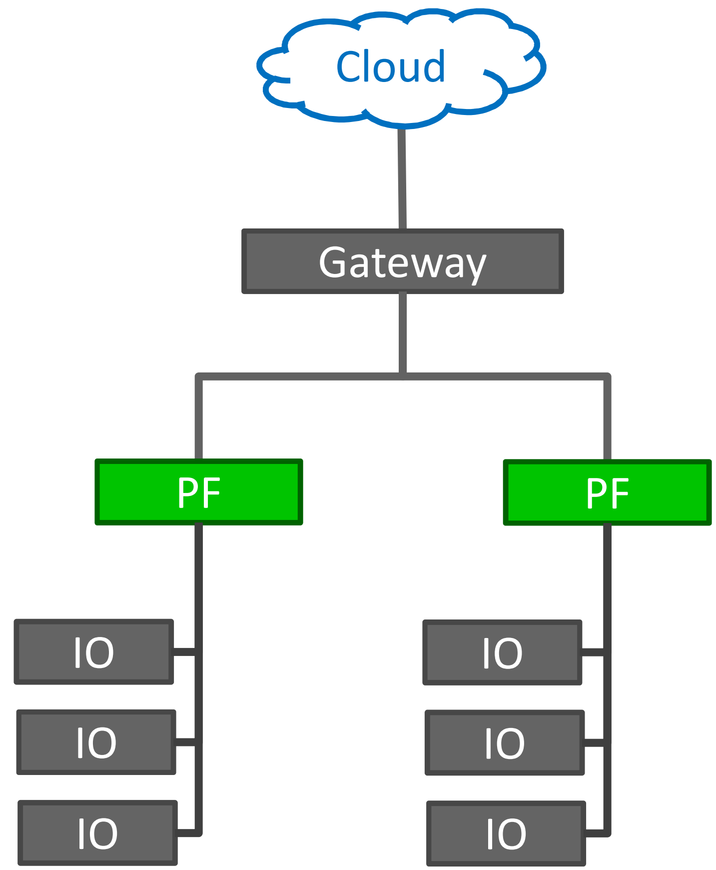


Future architecture of a car infrastructure

- 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

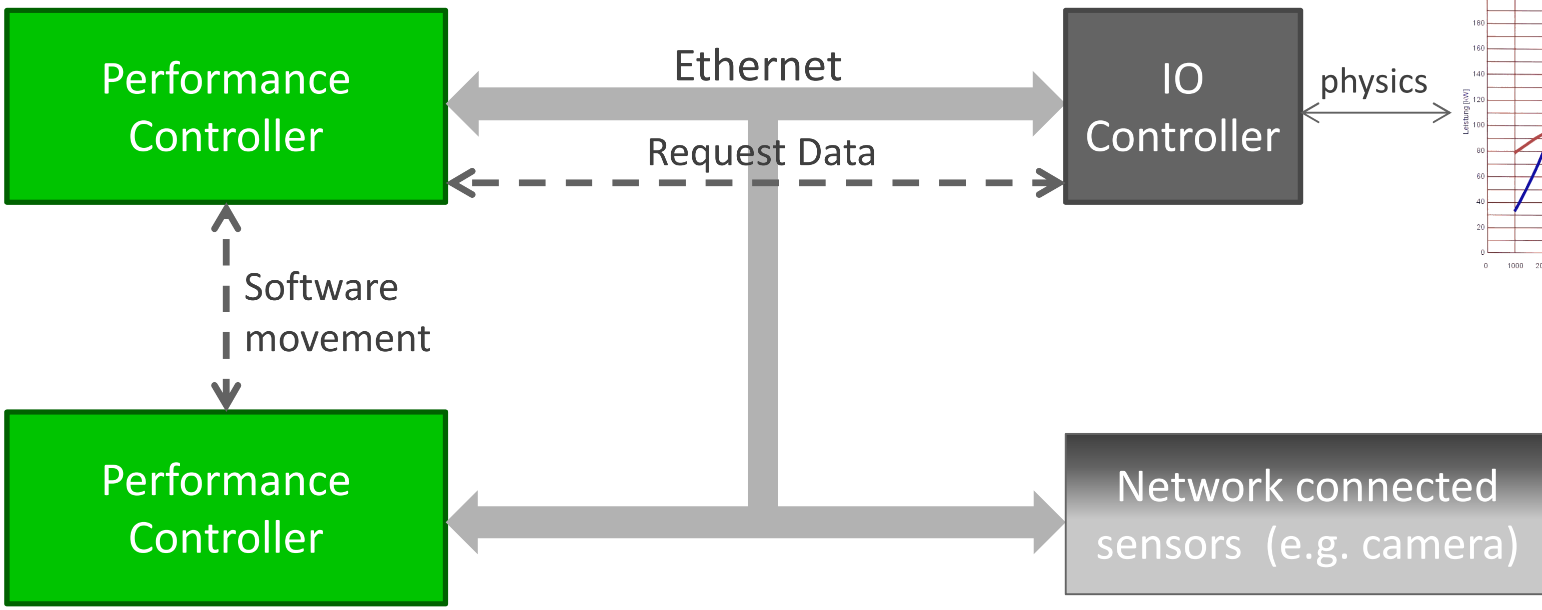
- IO Controller
  - Provide Sensor and Actuator Services
  - Deeply embedded, real-time Operating System (e.g. Classic AUTOSAR)

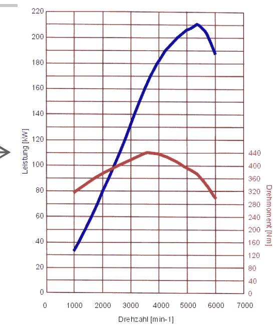


How to divide the functionalities?

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)

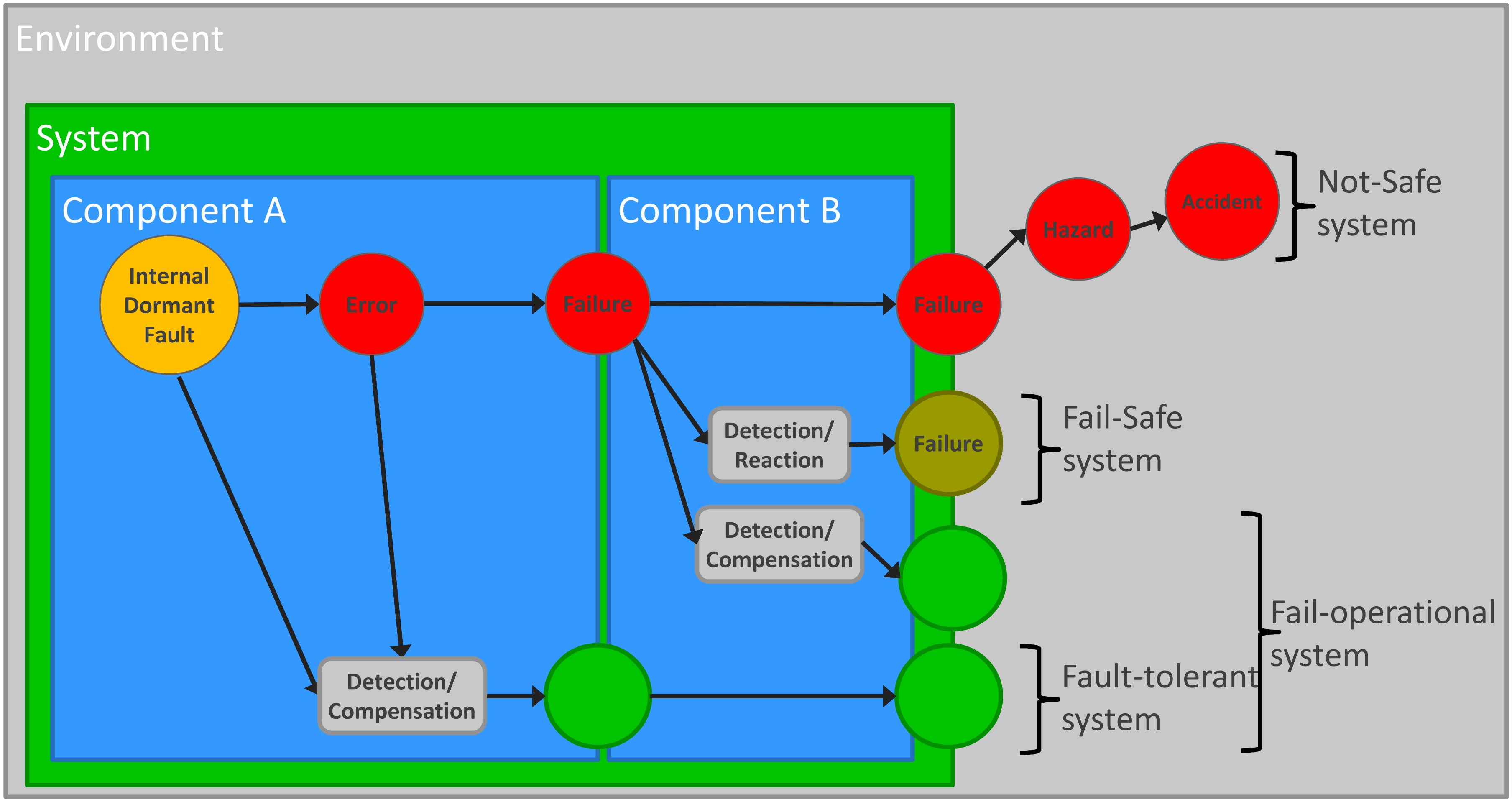
Requirements for a future car infrastructure

High Level Requirements	Technical Concepts	Technologies
High computing power	High Performance Controllers and GPUs	• Autosar Adaptive Platform, Hypervisor
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

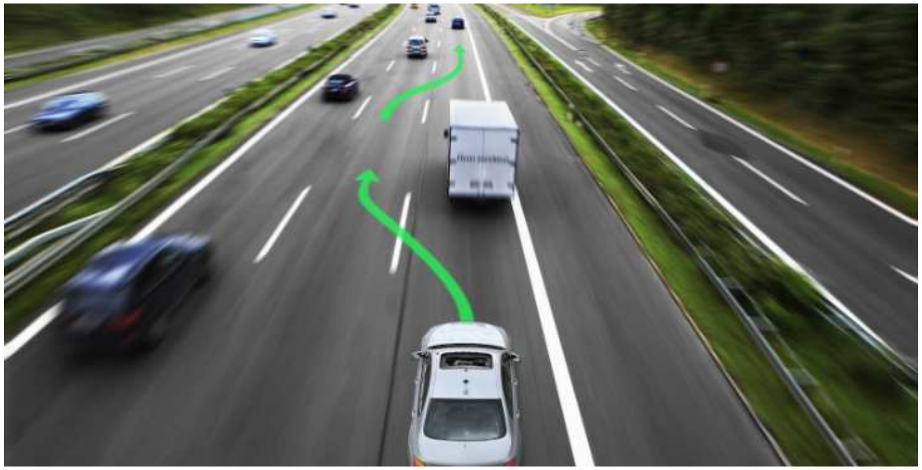


Basic Concepts and Taxonomy of Dependable and Secure Computing, Avizienis et al., 2004

Current Systems (usually fail-safe)

Failure Detected?

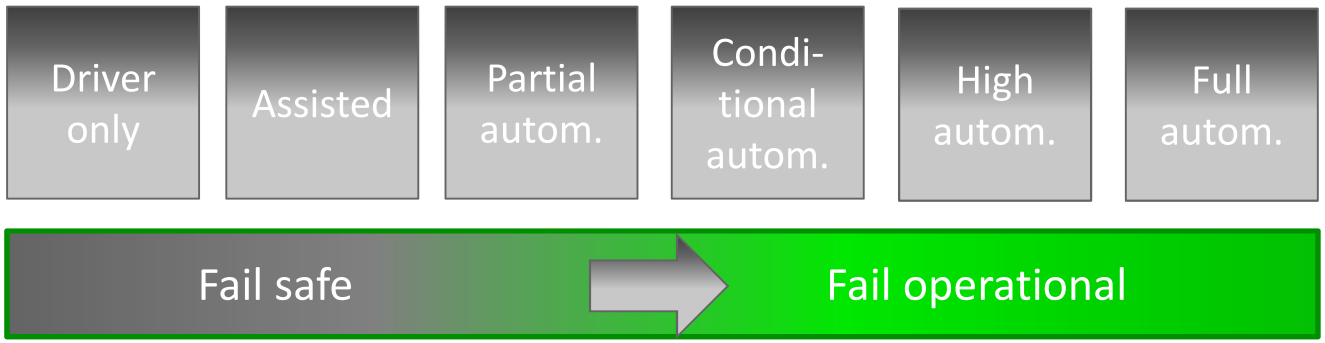
- Deactivate / degrade function
  → Safe State
- Inform the driver
- Report a diagnostic error



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

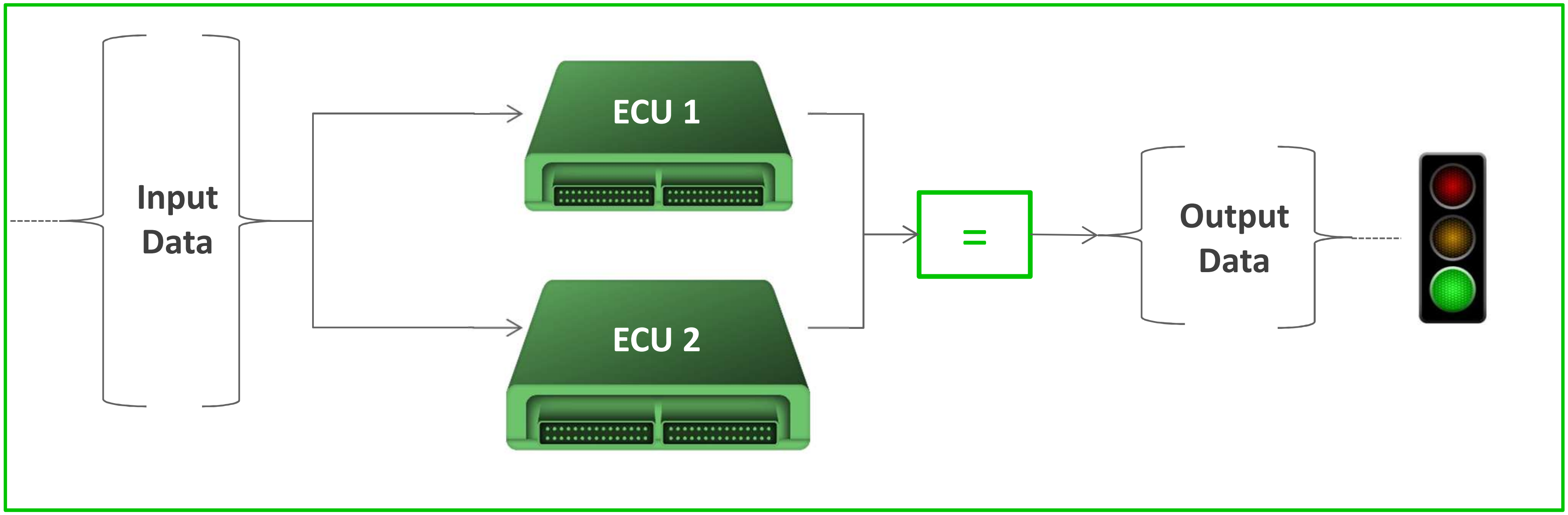
From Fail safe to Fail operational



Safe State means:

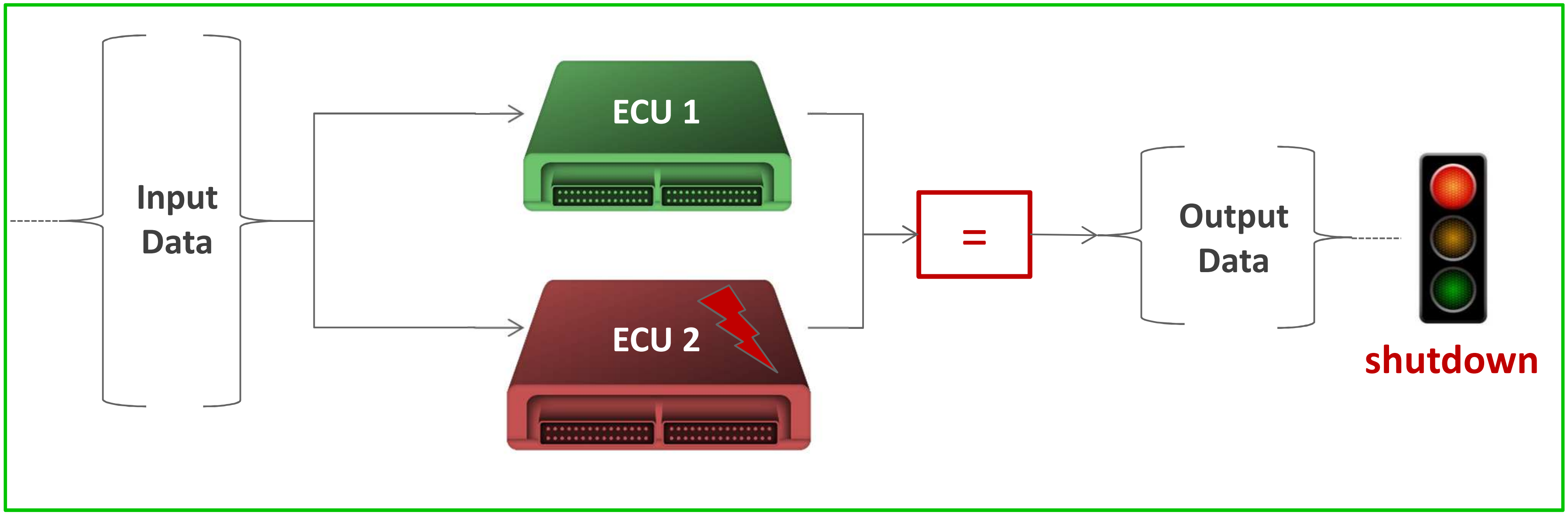
- Continue driving until driver is in the loop
  - approx. 7-15s for conditional autonomous driving
  - Several minutes for high and full autonomous driving
- Perform an autonomous „safe-stop“ (stand-still at a non-hazardous place)
  - Main issue is to get the driver attention focused on the situation
  - Several minutes, depending on the situation

1st approach: 2 channels with comparison



- Two ECUs working on the input data and compare the output data

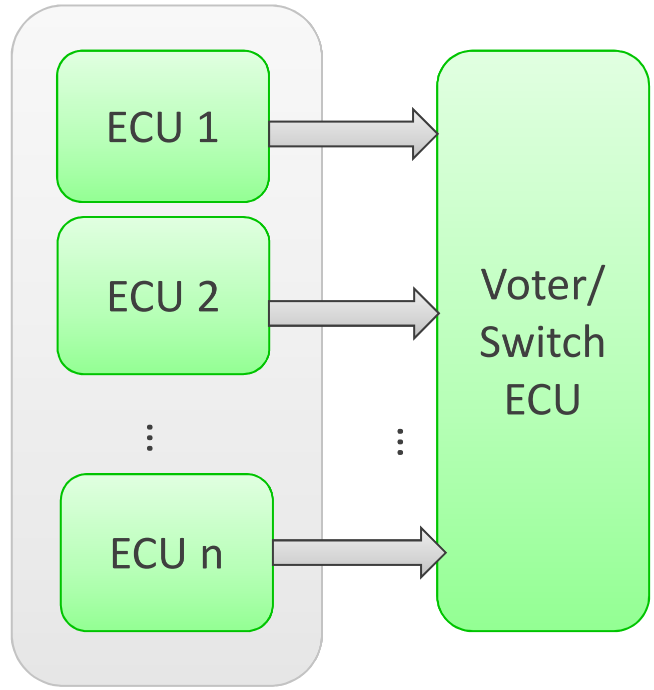
1st approach: 2 channels with comparison



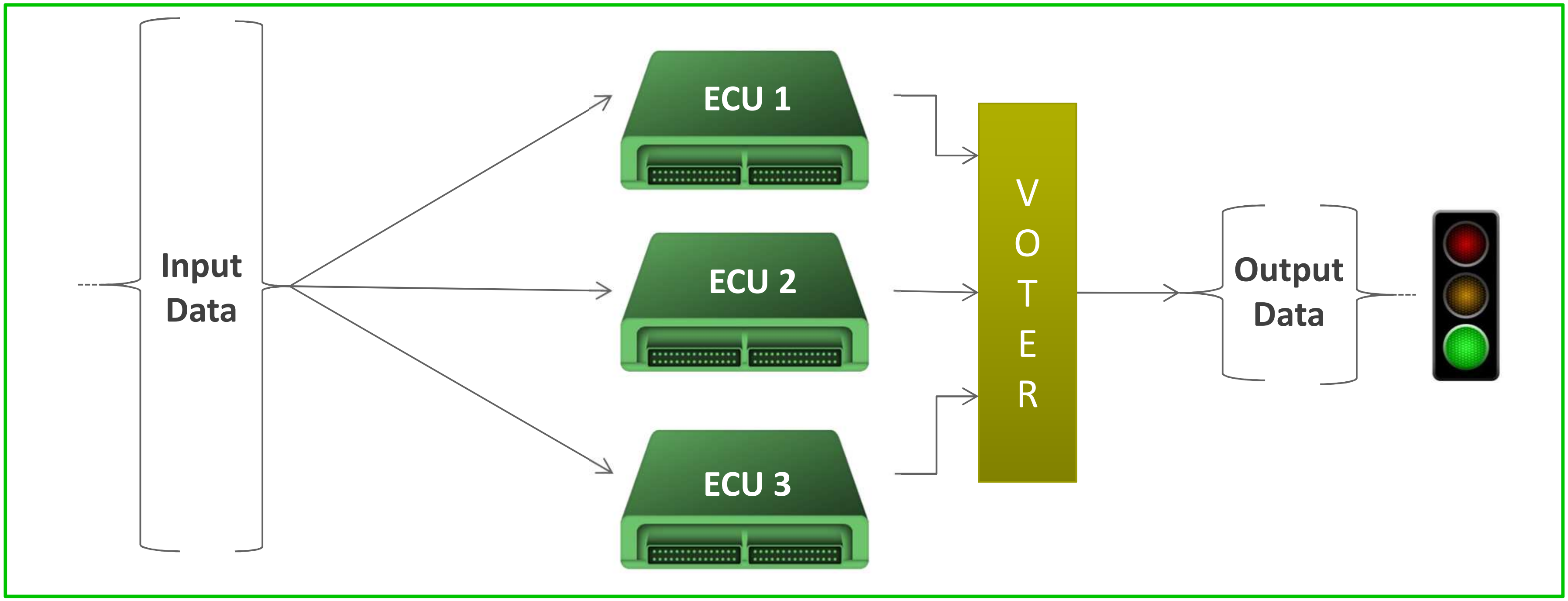
- Two ECUs working on the input data and compare the output data
- **A "2 channels with comparison-system" is simply fail-safe and since it is not possible to distinguish between "ECU1 not ok" and "ECU2 not ok".**
- **The safe state is a complete system shutdown, which is not acceptable for autonomous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

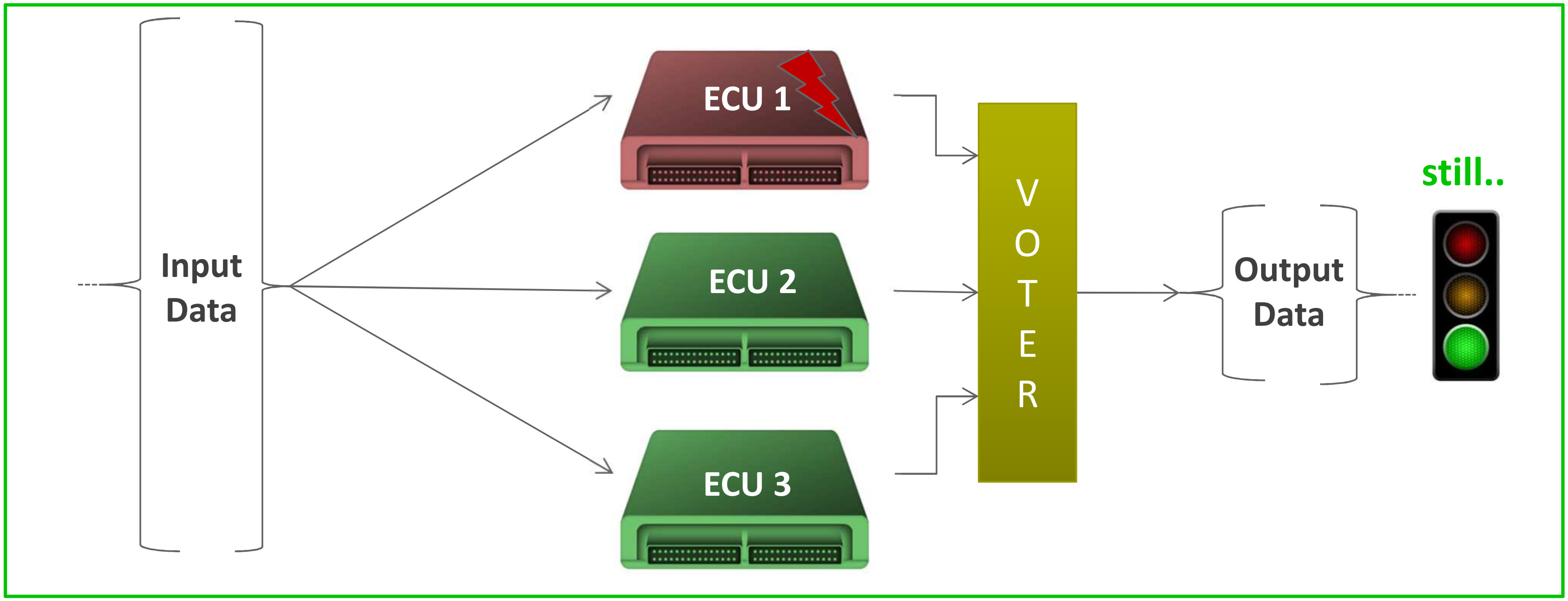


2nd approach: 2oo3 systems



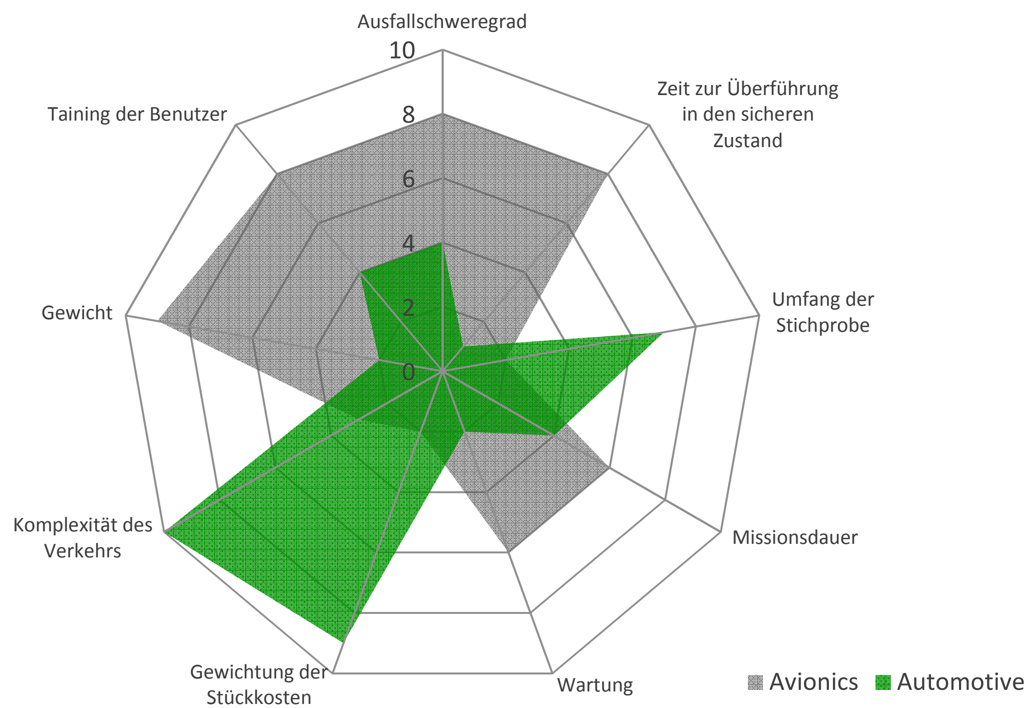
- A well established pattern
- If one of the ECUs fails, the system can continue with the remaining two ECUs.
- Failures in the input data can be detected by an “Input-Voter”.

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”.**

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

2oo3 systems applicable for automotive?

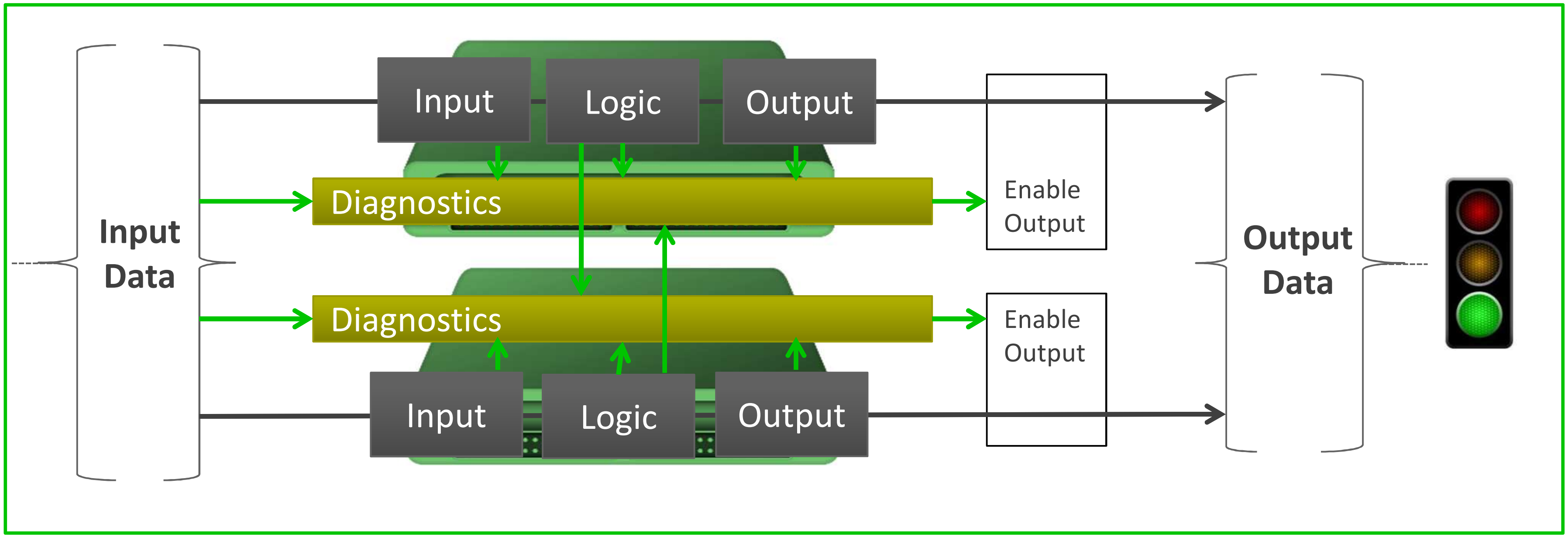
- More ECUs
- More wiring
- More weight
- More power consumption
- More complexity

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)

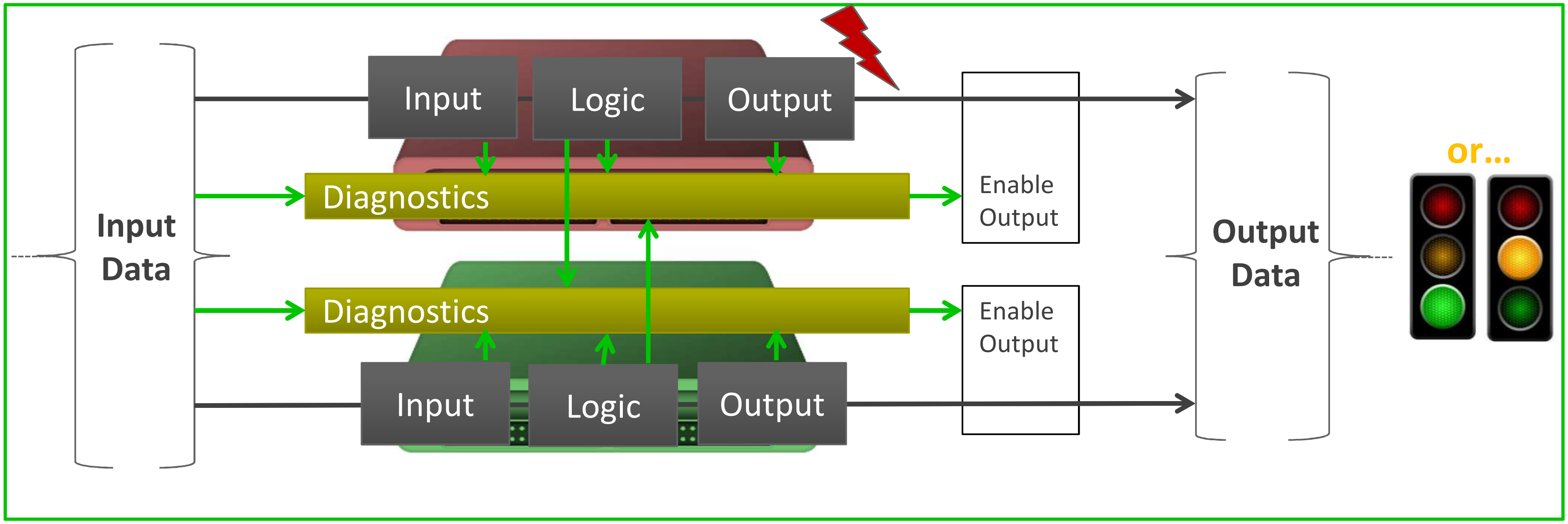
-> 2oo3 can be hardly realized due to costs issues.

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

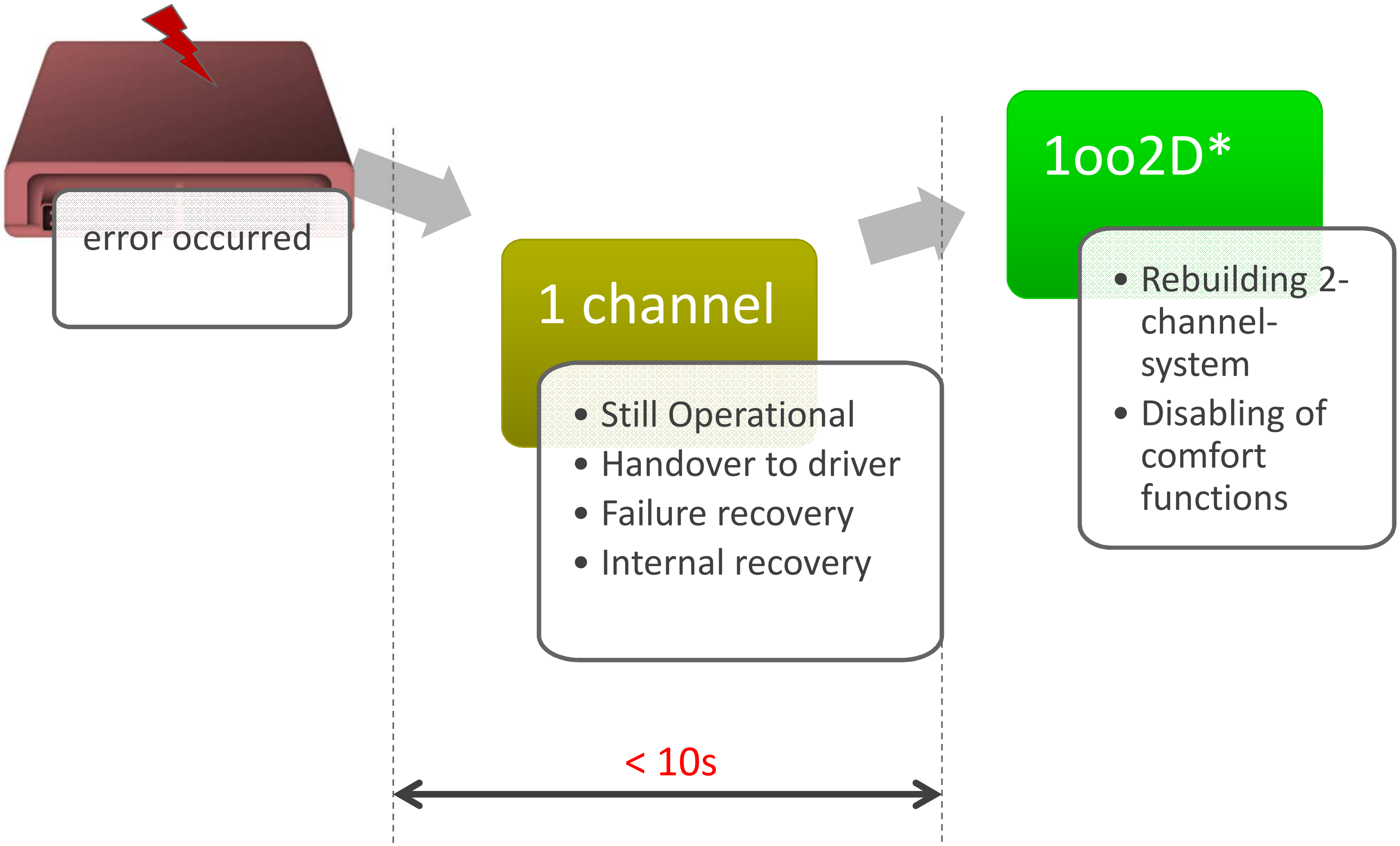


- High diagnostic coverage needed to detect failures in one channel
- **If a component fails in one of the two channels the system does not shut down**
- **The system continues to operate with one channel**

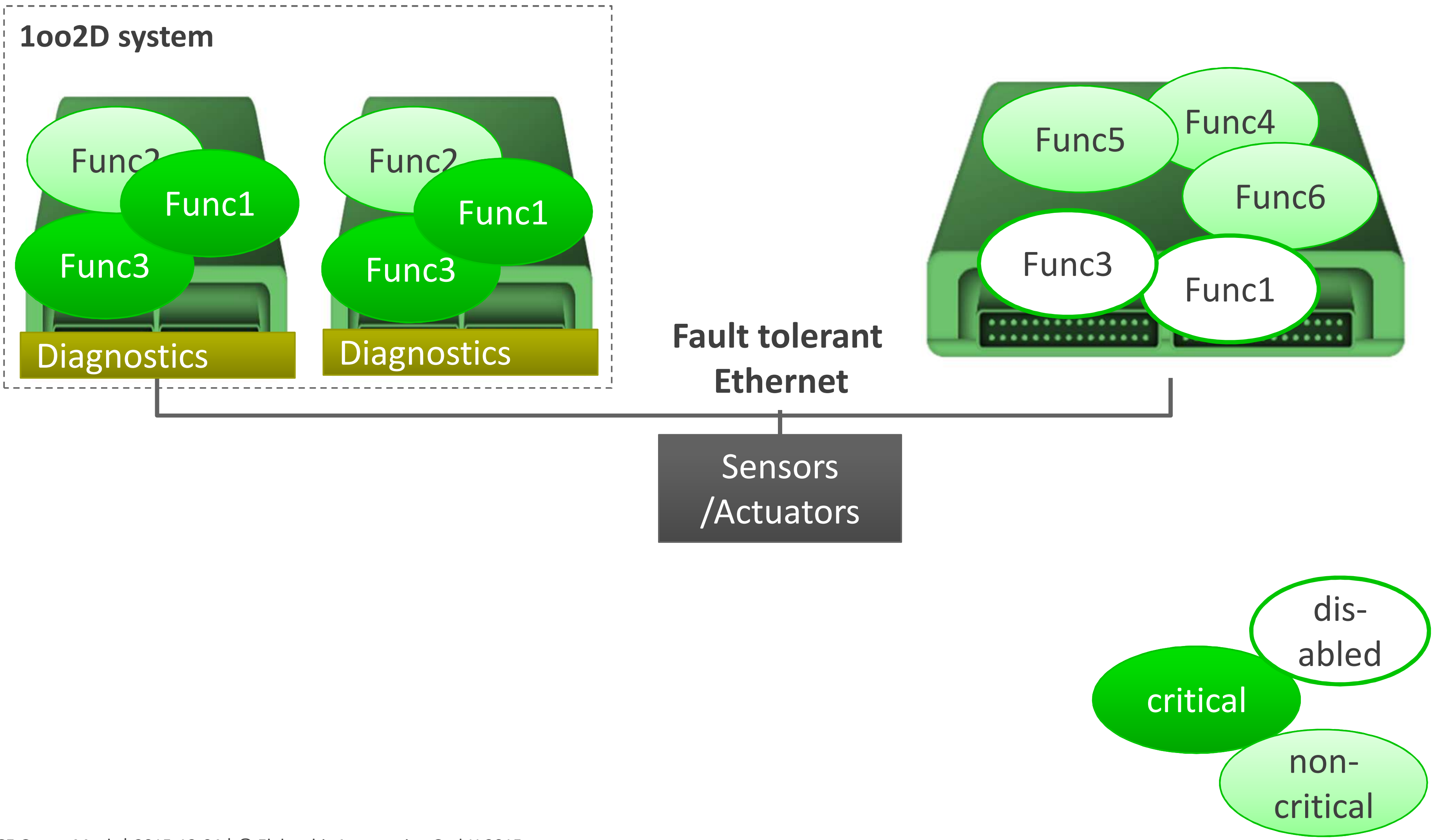
Common sense:

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***

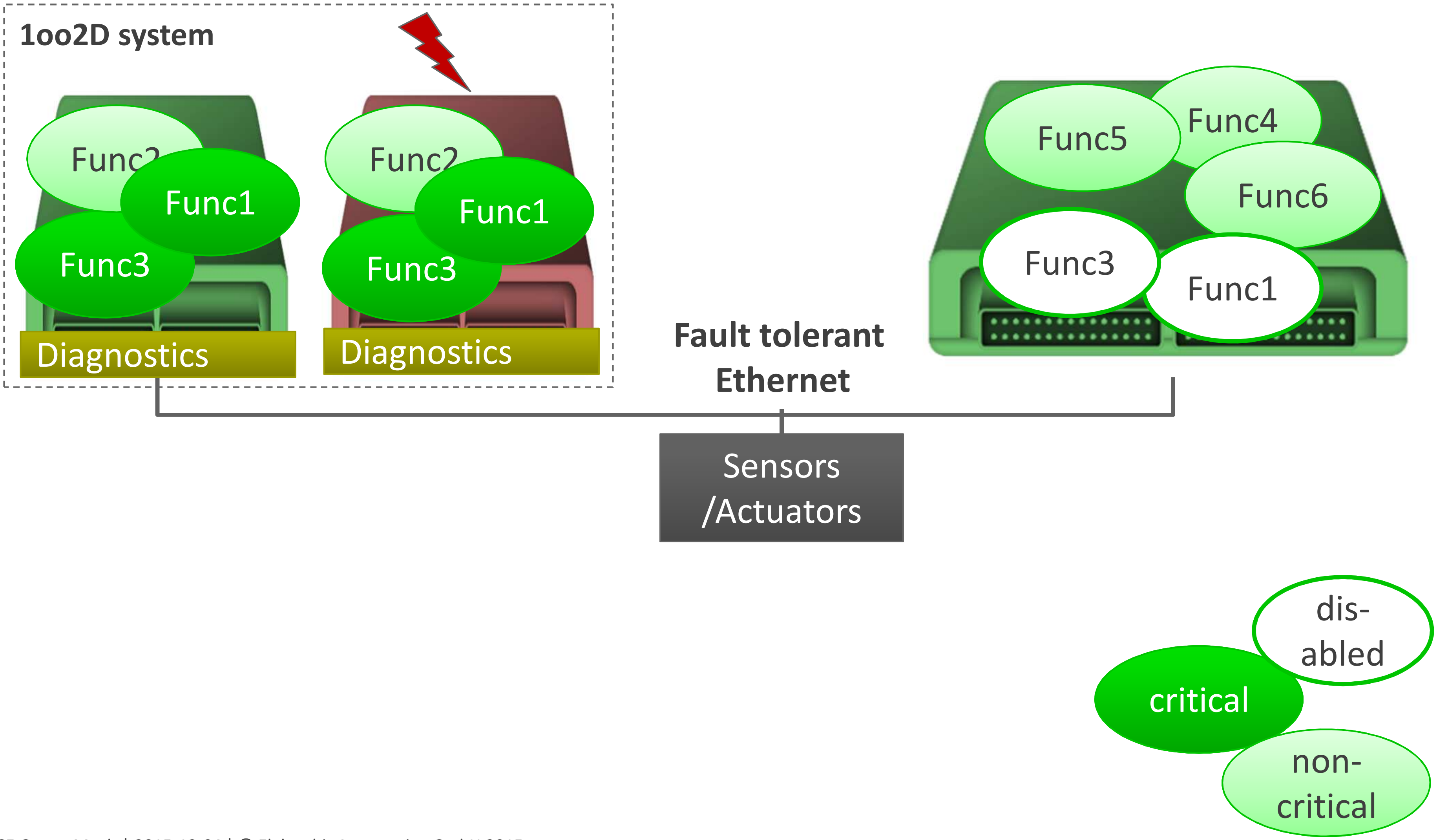
Outlook: Reconfiguration for rebuilding 1oo2D



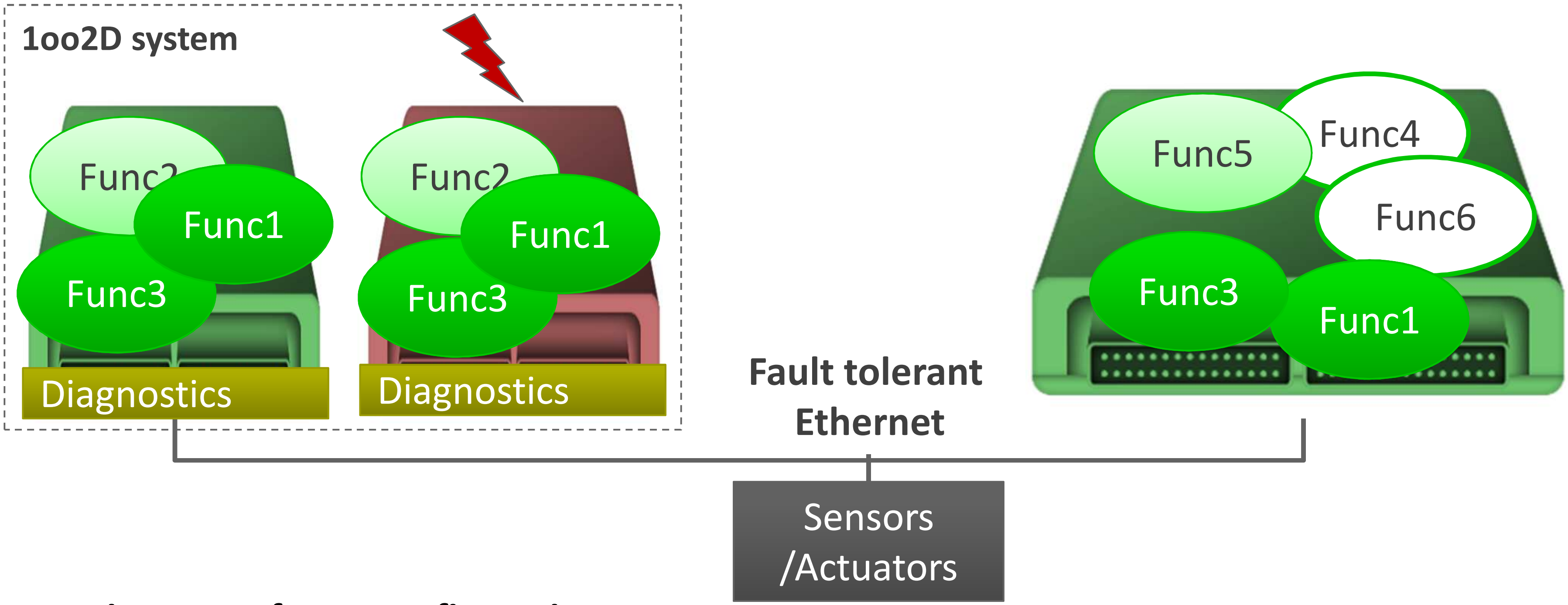
1oo2D - Normal operation



1oo2D – 1 channel

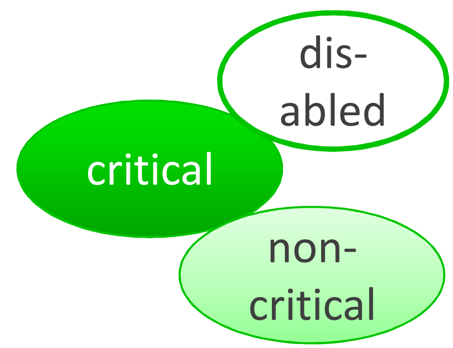


1oo2D*



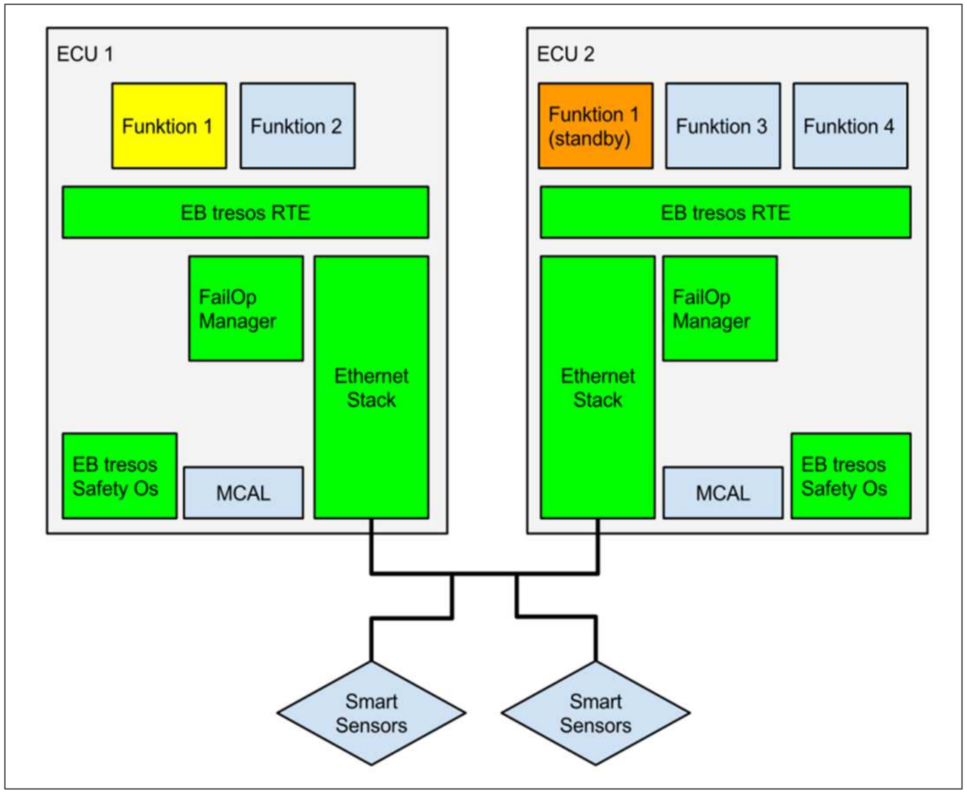
Requirements for Reconfiguration

- 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

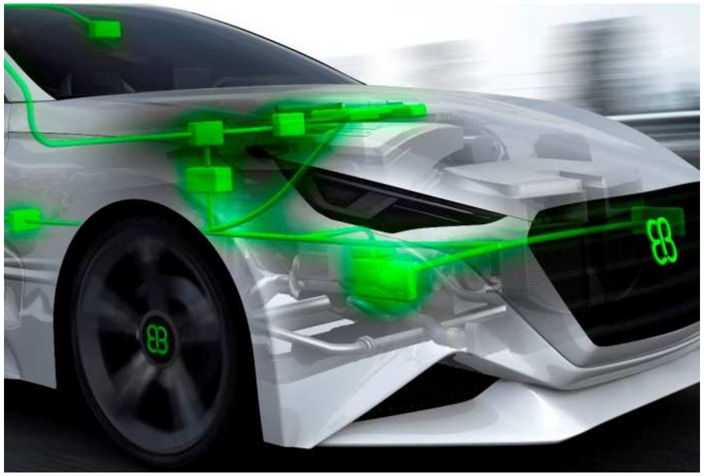
- Service orientated communication (SOME/IP and Service Discovery)
- Multi-cast fault-tolerant Ethernet

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!

Elektrobit

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