Development of an Embedded Architecture for Autonomous Driving

Dr. Johannes S. Mueller-Roemer
h_da — December 21, 2023

Outline

Learning Goals

Requirement Analysis

System Architecture

Outlook

Learning Goals

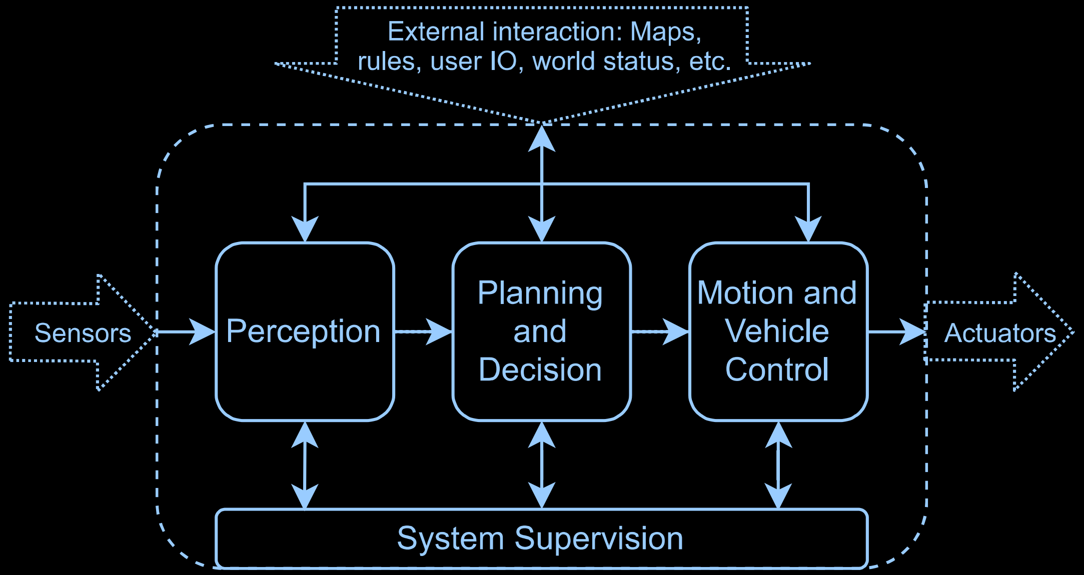
Learning Goals

- Know important terms in automotive E/E architectures
- Understand how requirements of autonomous vehicles differ from existing architectures
- Understand how these requirements affect E/E architecture and software

Requirement Analysis

Requirement Analysis

- What does an Autonomous Vehicle need to do?
- Everything a normal vehicle needs to do…
  - Powertrain management
  - Stability control (ESC/TCS/ABS)
- …as well as perceive, plan, and control
  - Based on user input



VELASCO-HERNANDEZ, G. ET AL. Autonomous Driving Architectures, Perception and Data Fusion: A Review. *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)*, 2020

Perception

- Perception is one of the most difficult tasks in autonomous driving
  - Identify and interpret traffic signs, traffic lights and lane markings
  - Identify and estimate motion of pedestrians, bicycles, and other vehicles
- Many types of sensors available, each with their own benefits and issues
  - Use multiple with *sensor fusion*

	Camera	LiDAR	RADAR	Fusion
Range	—	—	✓	✓
Resolution	✓	—	✗	✓
Distance Accuracy	—	✓	✓	✓
Velocity	—	✗	✓	✓
Color Perception	✓	✗	✗	✓
Lane Detection	✓	✗	✗	✓
Object Classification	✓	✗	✗	✓
Illumination and Weather Conditions	✗	—	✓	✓

IGNATIOUS, H. A., EL SAYED, H., and KHAN, M. An Overview of sensors in Autonomous Vehicles. *Procedia Computer Science 198*, 2022

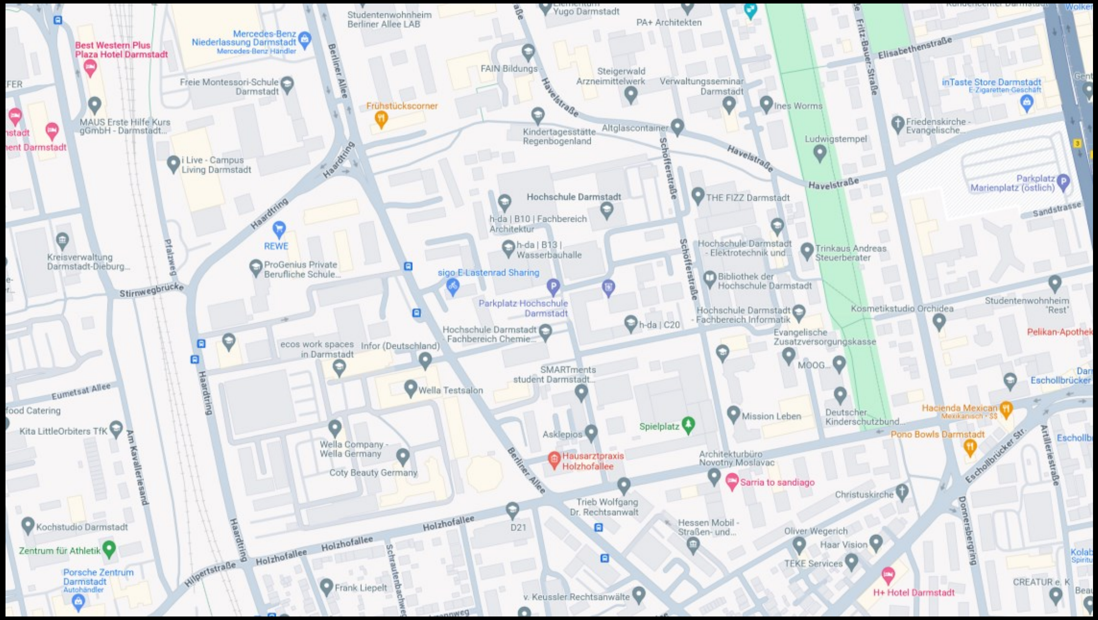
Bandwidth and Processing Power

- Many of these sensor require high bandwidth
  - A single camera can produce more data than a typical CAN (1 Mbit/s) or FlexRay (10 Mbit/s) bus can handle
- Deep learning models require massive processing power
- Resulting in high power demand
  - Estimate: hundreds of watts to 1 kW
- Massively parallel processing (e.g., GPUs) can help offset the compute and energy requirements

MALAWADE A. ET AL. SAGE: A Split-Architecture Methodology for Efficient End-to-End Autonomous Vehicle Control. *ACM Trans. Embedd. Comput. Syst. 20 (5s)*, 2021.

Planning

- High-level planning / decisions
  - What goals need to be achieved?
  - With which priorities?
- World trajectory planning
  - Route planning based on up-to-date maps and information (→ car as an IoT device)
- Local trajectory planning
  - Navigation around obstacles, rerouting based on local map (→ SLAM)



© Google, GeoBasis-DE/BKG

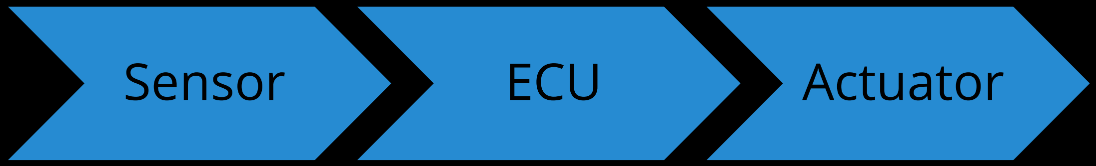
Control

- Control—or actuation—can be achieved via existing drive-by-wire systems
  - Throttle-by-wire
  - Brake-by-wire
  - Steer-by-wire
- Requires close interaction with local trajectory planning and SLAM
  - Only plan physically feasible trajectories (e.g., via simplified physical modeling)
  - Correct position and motion with feedback from SLAM, IMU, and GPS-RTC

System Architecture

Signal-oriented Architecture and ECUs

- Existing vehicle E/E architectures are typically signal-oriented
- Sensors produce signals, i.e., low-dimensional, scalar data
- ECUs perform one function and create output signals
- Signal routing and data rates are static



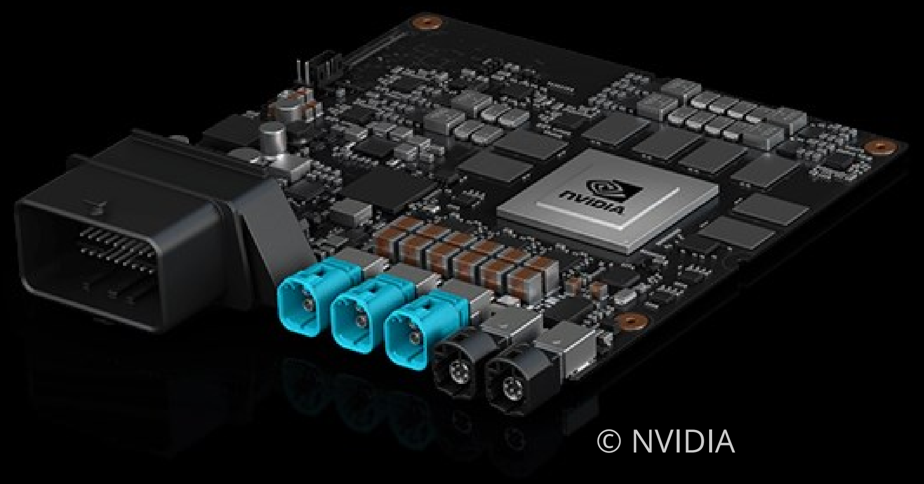
Show of Hands —Number of ECUs

- How many ECUs would you estimate are in a modern, high-end vehicle?
  - < 50
  - 50–100
  - 101–200
  - > 200

HAMMERSCHMIDT, C. Number of automotive ECUs continues to rise. *eeNews Automotive*, 2019
VETTER, A. ET AL. Development Processes in Automotive Service-oriented Architectures. *2020 9th Mediterranean Conference on Embedded Computing (MECO)*, 2020

Service-oriented Architecture and HCPs/HPCs

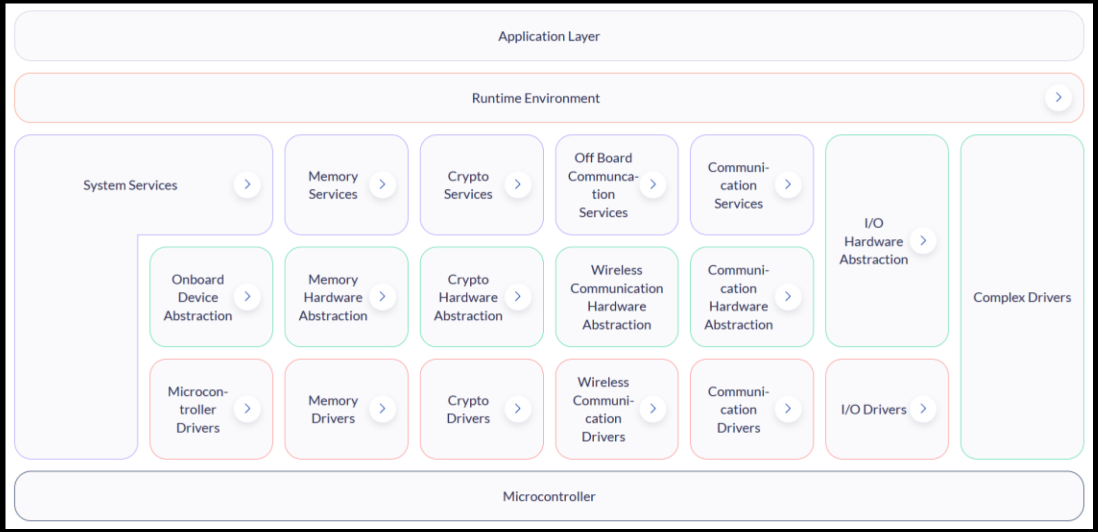
- Autonomous vehicles require…
  - Higher bandwidth → Automotive Ethernet
  - More compute power → High-performance computers
  - Adaptivity and over the air updates → Wireless communication, IoT
- Data rates are dynamic
- Data are structured and higher-dimensional



© NVIDIA

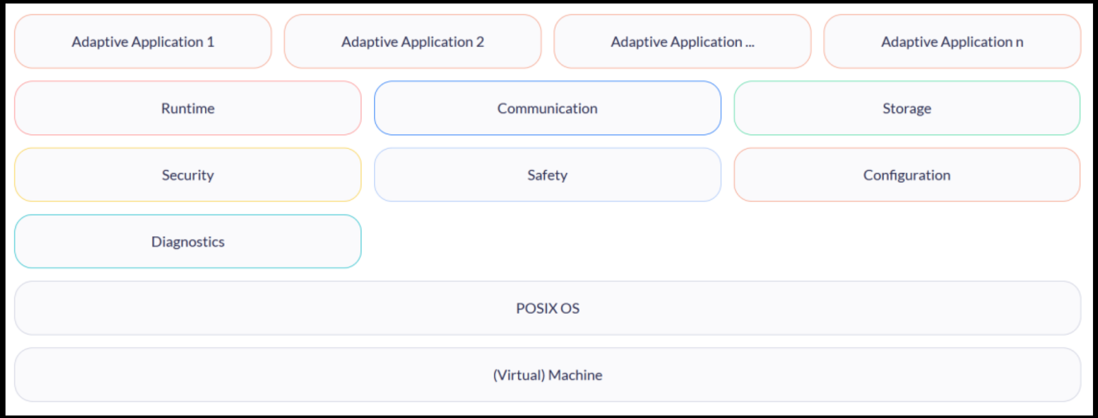
AUTOSAR Classic Platform

- To improve compatibility, AUTOSAR was introduced
- Standardized operating system (OS) and runtime environment (drivers, APIs)
- Single function / application per ECU / µC
- Supports hard real-time requirements
- C APIs



AUTOSAR Adaptive Platform

- For dynamic, connected vehicles
- Use of existing POSIX OSs (Linux, Android, iOS, …)
- Multiple applications in separate processes or OS instances (hypervisor/VM)
- C++ APIs
- Compatible with ACP
  - Real-world vehicles are hybrid and include ECUs and HPCs



Outlook

Outlook

- Real-time systems
- Embedded GPU computing
- Ferrocene – Rust for critical systems (ASIL D, SIL 4 qualified)
- Distributed IoT architecture
- Time-sensitive networking
- Deeper dive into automotive E/E architecture including developments such as
  - Zonal architectures
  - Software-defined vehicles

Further Reading

Further Reading (and Viewing)

- E/E systems and AUTOSAR
  - Vector – EnginEEring the Jigsaw
    https://www.youtube.com/@vectorinformatik
  - AUTOSAR
    https://www.autosar.org/
- GPUs for autonomous driving
  - NVIDIA Solutions for Self-Driving Cars
    https://www.nvidia.com/en-us/self-driving-cars/
- Ferrocene
  https://ferrous-systems.com/ferrocene/

Further Reading

- Review and survey papers
  - YURTSEVER, E. ET AL. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access 8*, 2020
  - VDOVIC, H., BABIC, J., and PODOBNIK, V. Automotive Software in Connected and Autonomous Electric Vehicles: A Review. *IEEE Access 7*, 2019
- Ethical and legal aspects
  - MAURER, M. ET AL. Autonomous Driving – Technical, Legal and Social Aspects. Springer Berlin, Heidelberg, 2016 (CC BY)
  - BARTNECK, C. ET AL. An Introduction to Ethics in Robotics and AI. Springer Cham, 2021(CC BY)

Dr. Johannes S. Mueller-Roemer

h_da — 21. Dezember 2023

Slides