# Networked Embedded Systems
# NES for Automotive Applications

Dr.-Ing. Vlado Handziski

Summer Semester 2015

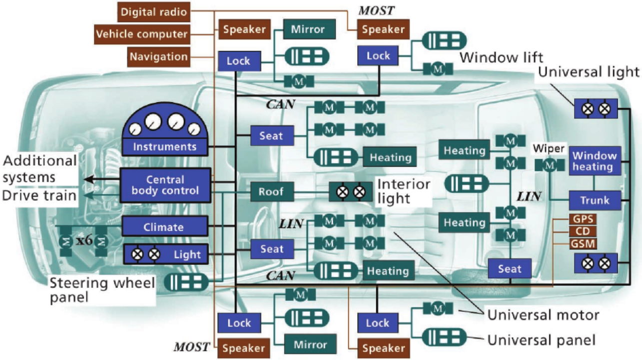
# From mechanical and hydraulic to electronic

- Exponential rise in the number of complex electronic systems
- Driver assistance
  - Anti-lock Braking System (ABS)
  - Electronic Stability Program (ESP)
  - Electric Power Steering (EPS)
  - active suspension
  - engine control
- Device control
  - lights
  - wipers
  - doors
  - windows
- Entertainment and communication
  - radio
  - DVD
  - hand-free phones
  - navigation systems

# From point-to-point to multiplexed communication

- Originally, each new function was implemented as a stand-alone electronic control unit (ECU)
- Hard to implement functions distributed over several ECUs
- In modern cars more than 70 ECUs exchange up to 2500 signals
- Point-to-point connections between the ECUs lead to inefficient information exchange $O(n^2)$ communication channels are needed
- Weight, cost, complexity and reliability issues
- Use of networks with multiplexed communication over a shared bus as an alternative

# Typical network architecture in a modern car




| | |
|---|---|
| CAN | Controller area network |
| GPS | Global Positioning System |
| GSM | Global System for Mobile Communications |
| LIN | Local interconnect network |
| MOST | Media-oriented systems transport |

# Diversity in QoS requirements

- Functional domains of automotive electronics
  - powertrain (engine and transmission)
  - chassis (suspension, steering and braking)
  - body (comfort functions)
  - telematics (wireless communication, vehicle monitoring, GPS)
  - multimedia (entertainment systems, mobile communication)
  - safety (air-bags, belt pretension, active headrest)
- Diversity in performance and safety needs
  - response time
  - jitter
  - bandwidth
  - error control
  - redundancy

# Automotive communication protocols

- Society for Automotive Engineers (SAE) classification of automotive car protocols
- Four classes based on data transmission speed and traffic needs
- Class A
  - data rate lower than 10 kbit/s
  - simple control data with low-cost technology
  - mainly used in *body* domain
  - Examples: LIN, TTP/A
- Class B!
  - data rate between 10 and 125 kbit/s
  - support of data exchange between ECUs to reduce number of sensors
  - Examples: J1850, low-speed CAN

# Automotive communication protocols (cont'd)

- Class C
  - data rate between 125 kbit/s and 1Mbit/s
  - high-speed real-time communication
  - mainly used in *powertrain* and *chassis* domain
  - Examples: high-speed CAN
- Class D
  - data rate higher than 1Mbit/s
  - safety-critical applications with predictability and fault-tolerance needs
  - multimedia data
  - gateways between subsystems
  - Examples: TTP/C, FlexRay

# Component Interoperability

- Performance, quality and safety depend on tight cooperation of

many subsystems

- Complex cooperative multi-partner development process with suppliers and OEMs
- Problem of interoperability of components
  - portability of components from one ECU to another
  - reuse of components between different platforms
- Need for an abstraction layer that decouples the application development from the evolution of the underlying hardware

# Requirements for automotive system software

- Hide the heterogeneity
  - APIs that are independent from the underlying protocols or CPU architecture
- Hide the distribution
  - APIs that are the same for intra-ECU, inter-ECU, inter-domain communication
- Provide high-level services
  - APIs that raise the level of abstraction, simplify the development and increase quality through reuse of validated services
- Provide QoS guarantees
  - APIs that boost the QoS provided by lower-level protocols through additional error control mechanisms

# OSEK/VDX

- Defined by the OSEK/VDX consortium (www.osek-vdx.org)
- OSEK/VDX communication layer
  - Common software interfaces and behavior for internal and external communication between application components
  - Statically defined signal packing
  - Automatic serialization and queuing
- OSEK/VDX transport layer
  - Segmentation and reassembly of frames
- OSEK/VDX OS
  - Services for task, event and interrupt management
- Does not provide full transparency for inta-ECU and inter-ECU application process communication
- Does not obey TT approach and can not be used directly on top of a TT protocol like TTP/C or FlexRay
- Some of these deficiencies are addressed in OSEK/VDX FTCom allowing operation on top of a TT OS like OSEK/VDX Time

# AUTOSAR

- Defined by the consortium (www.autosar.org)
- Main goal is to improve the quality and reliability of the automotive embedded systems
- Focuses on the architecture of the software embedded in an ECU
  - Application Layer
  - Basic software
  - Run-time environment
- Hides the characteristics of the hardware platform *as well as the distribution* of the application software components
- Two communication models
  - Sender/Receiver (data exchange)
    - Explicit mode
    - Implicit mode
  - Client/Server (function invocation)