Designing a software framework for automated driving

EB robinos
Highly automated driving architecture and software

Embedded Linux

Safety Processor

Performance Processor

Dr.-Ing. Sebastian Ohl, 2017 October 12th
Challenges

- Functional software architecture with open interfaces and a set of well-defined software components for automated driving systems up to SAE level 5
- Seamless cooperation and combination of automotive soft- and hardware
- Highly automated driving software from prototyping to embedded/serial use
EB robinos: EB’s Autonomous Driving Software

Key features:
• Architecture for ADAS up to SAE level 5
• Runs on AUTOSAR
• Central ECU as well as distributed systems
• Incorporates existing or new, customer or third party subsystems
A modular, extensive framework for automated driving

Designing a software framework for automated driving

Interfaces for
- *interoceptive sensors*: wheel ticks, steering, accelerometers/gyros
- "smart" *environment sensors* – point clouds, object lists
- ADASIS for map, SENSORIS for cloud

Extensible function framework
- *rule-based coordination*
- clear *upgrade path* from simple to complex behavior
- guaranteed *testability*

Coordination of vehicle control and HMI
- braking, steering
- safe communication
- *safe hand-over*

Interfaces for
- kinematic vehicle components
- instrument cluster
- infotainment display
A modular framework

Map onto concrete ECU architecture

Ensure differentiation, shorten time to market

Upgrade across models / Upgrade over time
Different Test Vehicles Running the Same Software

Automated Valet Parking & inner-city Driving
- Lidar, ultra sonic, and 360° camera sensors
- High performance computer
- Additional actuators for steering & pedals
- High precision IMU

Highway Driving
- Lidar, camera, and radar sensors
- High performance computer
- Additional actuators for steering & pedals
- High precision GPS

Embedded Valet Parking
- Lidar and radar sensors
- Embedded computing platform
- Using OEM interfaces to actuators
- Low precision IMU
Different Development Stages

EB robindos

EB Assist ADTF Middleware
Embedded Linux Middleware
EB Adaptive AUTOSAR

Rapid prototyping
C, C++, Model based
PC

Rapid embedding
C, C++
Evaluation hardware

Automotive grade software
ECU
A Runtime Environment for EB robinos

System Assembly
- Combine different components
- Describe communication channels

Communication
- Sending Data
- Receiving Data
- Call Interfaces
- Inter- & Intra ECU

Runtime triggering
- Trigger a function cyclically
- Trigger a function on events

High Performance ECU Characteristics
- High computation power with heterogeneous computing
- Hardware virtualization
- Widespread, POSIX-like Operating System (e.g. Linux)
- Extensive update capabilities
- Interfacing with common in-vehicle bus systems
Different Run-time Platforms

**EB Assist ADTF**
- Rapid Prototyping Framework
- Supports Sender/Receiver communication via Pins
- Call interfaces are supported by ADTF services
- Provides MessageBus for Inter-ECU communication
- Configuration Editor for system assembly

**Classic AUTOSAR**
- Mass production grade ECU basic software
- Widely used for 32-Bit controllers
- Provides Sender/Receiver as well as client/server communication
- Various bus systems supported
- System assembly by AUTOSAR system architecture tool

**Embedded Linux**
- Much more suitable for high performance computing systems
- Inter-Application communication via some/IP and MessageBus protocol
- Intra-application communication via C++ calls
- System assembly directly in C++ code

**Adaptive AUTOSAR**
- Upcoming standard for high performance ECUs
- Service orientated communication (via some/IP)
- Dynamic scheduling of applications

---

© Elektrobit (EB) 2017 | Confidential
Designing a software framework for automated driving

**EB Adaptive AUTOSAR**

### Adaptive AUTOSAR Characteristics

- Designed for major market trends like E-Mobility, Automated Driving and Mobility Services
- **Complementing** standard, neither replacing classic AUTOSAR nor replaces proven implementations
- Standardized methodology
- Specifying API’s and functional requirements only

### Key Features & Benefits

- Software Development Kit for construction, qualification and deployment of adaptive applications for high performance controllers
- Offers service-oriented architecture and communication
- Support integration with multiple variants of POSIX-Like Operating Systems (e.g. Linux, QNX, …)
- Enables OEMs and TIER1s to reload software functionality at runtime
EB robinos Grid Fusion Processing Online Data

• Grid: 160x160 cells covering 1x1cm
• Sensor: Hokuyo UST-10LX LIDAR at 40 Hz
• Board: NVIDIA Jetson TX1 (same processor as the Nvidia DrivePX 1)
• Runtime Environment: EB tresos Solution for AD

The demo shows the grid processing online data from the LIDAR sensor. During the demo obstacles are introduced and moved. This results in changes in the grid fusion’s visualization.

© Elektrobit (EB) 2017 | Confidential
EB robinos Path Planning Processing Recorded Data

- Sensor: IBEO Lux Gen4 at 12.5Hz
- Board: Intel based PC
- Runtime Environment: EB Assist ADTF

The demo shows EB robinos Path Planning running on EB Assist ADTF. It is taking local sensors as well as map data into account. Moreover, the path is adapted according to local traffic participants. The motion model is restricted according to traffic rules and vehicle speed.
EB robinos Processing Recorded Data

- Sensor: IBEO Lux Gen4 at 12.5Hz
- Board: NVIDIA Drive PX2
- Runtime Environment: EB robinos Linux Environment

- The demo shows grid processing and inner-city path planning running on a NVIDIA DrivePX 2 hardware. All processing and visualization is running on the embedded platform.
EB robinos for Embedded Linux on NVIDIA DrivePX

- Ready to use HAD software modules
- Different show cases possible (e.g. Valet Parking or people mover)
- Running as pure Linux applications
- Packaged modules (Oct 2017)
  - EB robinos Grid Fusion
  - EB robinos Path Planning
  - EB robinos Cross Product components
Automated Valet Parking with EB robinos

Solution

**EB robinos** software components can be used to automatize parking applications from simple maneuvers up to complete autonomous systems navigating on big parking lots.

Key Message

- Valet Parking as SAE Level 5 function
- Usable on private property (depending on local laws)
- Assisted parking for complex situations
- Garage parking

EB’s added value

**EB robinos** - Software modules

- **for prototyping:** EB Assist ADTF
- **for rapid embedding:** (adaptive) AUTOSAR or embedded Linux
- **for production:** on vehicle ECU

Designing a software framework for automated driving

Trajectory Control

Longitudinal Control

Lateral Control

Positioning

Object Fusion

Grid Fusion

Road and Lane Fusion

Vehicle Database

Function Specific Views

Sensor Data Fusion

Situation Analysis

Path Planning

Vehicle Abstraction

- Sensors

Motion Management

Situative Behavior Arbitration

Sensor Management

Situation Management

Path Planning

Safety Management

Situative Behavior Arbitration

Sensor Management

Situation Management

Path Planning

HMI Management

Situative Behavior Arbitration

Sensor Management

Situation Management

Path Planning

HMI Management
Highway Pilot with EB robinos

Solution

Highway Pilot with EB robinos is providing software modules to bring EB robinos to the highway.

Key Message

- Highly Automated Driving on highways, 0-120 km/h
  - free driving and vehicle following
  - lane keeping
  - automated lane change
- Automatically detect of system failures and limits
- React to failures / limits with appropriate safety maneuvers.

EB’s added value

- Use EB robinos software framework and standard interfaces
- Support a wide variety of sensors, actuators, and ECUs
- Easy integration into customer E/E architecture
Get in touch!

sebastian.ohl@elektrobit.com
www.elektrobit.com
www.eb-robinos.com