Towards an Integrated Vehicle Management System in DriveOS

Soham Sinha and Richard West
Department of Computer Science, Boston University and Drako Motors Inc., USA
Modern Automotive Systems
Modern Automotive Systems

- Today’s vehicles have 50 – 150 Electronic Control Units (ECUs)
- 10s to 100s of millions lines of code
- Complex CAN bus network of ECUs
Functional Domains in a Vehicle

- Vehicle software services are divided into a number of functional domains.
New Functions → New Functional Domains

- New vehicle functions like ADAS, high-quality IVI add new functional domains.
- New domains mean more electronics and more software.
New Functions → New Functional Domains

• New vehicle functions like ADAS, high-quality IVI add new functional domains.

• New domains mean more electronics and more software.

Hardware, Wiring and Packaging, Upgradability Cost ↑
Functional Consolidation

WHITE PAPER
Automotive Electronic Control
Unit (ECU) Consolidation

ECU Consolidation Reduces
Vehicle Cost, Weight, and Testing

<table>
<thead>
<tr>
<th>E/E architecture</th>
<th>Generation</th>
<th>High-level architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle centralized</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5th

Infotainment

Central gateway

Sensor

Actuator
Requirements for a Centralized Vehicle OS

- Timing-predictability
- Critical and non-critical tasks in the same platform
  - e.g., ADAS, IVI and IC
- Safety, Security and Reliability
- Preferably Low-cost Computing Hardware
DriveOS
DriveOS

Centralized Vehicle Management System

Timing-predictable

x86 Secure Hardware Virtualization

Runs on PC-class hardware
DriveOS

Target Platform : DX1100
DriveOS Design: Quest-V Separation
Kernel

Sandbox 1

Core 1 ... i

Sandbox 2

Core (i+1) ... m

Hardware
DriveOS Design: RTOS Sandbox

Sandbox 1

- Real-time Control Task, Sensor Data Processing and Actuation
- Quest RTOS Kernel (ring 0)
- VMM (ring -1)
- Core 1 ... i

Sandbox 2

- Core (i+1) ... m

Hardware

ring 3
DriveOS Design: Linux Sandbox

Sandbox 1
- Real-time Control Task, Sensor Data Processing and Actuation
- Quest RTOS Kernel (ring 0)
- VMM (ring -1)
- Core 1 ... i

Sandbox 2
- User Interface and Third-party Apps
- Libraries and Runtime
- Linux Kernel and Drivers (ring 0)
- VMM (ring -1)
- Core (i+1) ... m

Shared Memory

Hardware
DriveOS Inter-sandbox Communication: 

- **Quest SB**
  - Real-time Tasks
  - `libshmcomm`
  - `Kernel`
    - `shmcomm module`
  - `shmcomm manager`
  - `VMM` (VM Exit)
  - `Ring -1`

- **Yocto Linux SB**
  - User Interface and Apps
  - `libshmcomm`
  - `Kernel`
    - `shmcomm module`
  - `shmcomm manager`
  - `VMM` (VM Exit)
  - `Ring -1`

- **Channel Pages**
  - Size = 8 MB
  - `Info Page`

- **EPT**
  - `Data` (blue) - Control (dotted)

- **Drivers**
  - `shmcomm` module

- **Library**
  - `libshmcomm`
DriveOS Applications

- Instrument Cluster (IC)
DriveOS Applications

- Instrument Cluster (IC)
- In-vehicle Infotainment (IVI)
DriveOS Applications

- Instrument Cluster (IC)
- In-vehicle Infotainment (IVI)
- OpenPilot Advanced Driver Assistance System (ADAS)

*Be ready to take over at any time*

Always keep hands on wheel and eyes on road
DriveOS Applications: IC and IVI

1. IC and IVI Sensing

2. HVAC Actuation

CAN Gateway

libusb + canlib

Infotainment CAN Mapper (IMS)

Infotainment CAN Mapper (IML)

FIFO

shmcomm channels

Quest

Yocto Linux

IC

IVI
DriveOS Applications: OpenPilot ADAS

- CAN Gateway
  - libusb + canlib
  - ADAS CAN Mapper (AMS)
  - Longitudinal Feed-forward PI Controller

shmcomm channels

1. IC & IVI Sensing
2. HVAC Actuation
3. ADAS Sensing
4. ADAS Actuation
5. Control Command
6. INIT/RESET Input
7. UPDATE Input
8. UPDATE Output

- CAN Channel 1
- CAN Channel 3

- Infotainment CAN Mapper (IMS)
- ADAS CAN Mapper (AML)
- Infotainment CAN Mapper (IML)
- FIFO
- IC
- IVI
- OpenPilot ADAS
- Quest
- Yocto Linux

DriveOS on DX1100
Hardware-in-the-loop Simulation

1. IC & IVI Sensing
2. HVAC Actuation
3. ADAS Sensing
4. ADAS Actuation
5. Control Cmd
6. INIT/RESET Input
7. UPDATE Input
8. UPDATE Output

Ubuntu 16.04

Kvaser USBcan Light 2x HS
Kvaser USBcan Pro 5x HS

CAN-Hi
1200
CAN-Lo
1200

CAN Channel 1
CAN Channel 3

CAN Gateway
libusb + canlib
ADAS CAN Mapper (AMS)
Infotainment CAN Mapper (IMS)

Quest
DriveOS on DX1100

shmcomm channels

FIFO
IC
IVI

cereal pub-sub
OpenPilot ADAS

Yocto Linux

1. IC & IVI Sensing
2. HVAC Actuation
3. ADAS Sensing
4. ADAS Actuation
5. Control Cmd
6. INIT/RESET Input
7. UPDATE Input
8. UPDATE Output
Evaluation Results
Evaluation: End-to-end Latency

- ADAS Control Loop path is highlighted in pink
Evaluation: End-to-end Latency

- ADAS Control Loop End-to-end Latency in presence of background Linux tasks.

Theoretical Delay = 11ms
Industry Target = 10ms
Evaluation: End-to-end Latency - CDF

- ADAS Control Loop End-to-end Latency in presence of background Linux tasks.

Theoretical Delay = 11ms
Industry Target = 10ms
Evaluation: Throughput

- CAN I/O: Other CAN Channels are being accessed
- Other I/O: Random disk and network I/Os are being issued
Ongoing Work

- More core vehicle functions (e.g., HVAC, Powertrain) in DriveOS
- Implement a MATLAB/Simulink interface for DriveOS
- Intelligent Power Management
  - Fast suspend/resume functionality
Conclusion

- DriveOS is a centralized vehicle-management system.
- It runs on low-cost PC-class machines.
- It is a timing-predictable, extensible and secure system.
- For this paper, DriveOS focuses on IC, IVI and ADAS.
- It meets throughput and latency requirements for the industry.
Acknowledgements

• Big thanks to the whole Celenum and Drako Motors team!
• Thanks to all the past and current Quest(-V) developers including but not limited to Matt, Ying, Tom, Chris, Eric and Sasan!
Thank you!

Questions?
Contact: <soham1, richwest>@bu.edu
References

- Images are from Google Images, Statista
- Few slides and images are taken from our previous presentations at HotMobile 2020, BEVA 2020 and other conferences.
- Our previous work:
  - A Paravirtualized Android for Next Generation Interactive Automotive Systems (HotMobile 2020)
  - Boomerang: Real-Time I/O Meets Legacy Systems (RTAS 2020)