Run-Time Measurement and Calibration of ECUs

Vector Solution Overview
# Agenda

<table>
<thead>
<tr>
<th>Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction to Measurement and Calibration</strong></td>
</tr>
<tr>
<td>▶ What is Run-Time Measurement and Calibration?</td>
</tr>
<tr>
<td>▶ Where is Run-Time Measurement and Calibration used?</td>
</tr>
<tr>
<td>▶ What are the Typical Use Cases?</td>
</tr>
<tr>
<td><strong>How does run-time measurement &amp; calibration work?</strong></td>
</tr>
<tr>
<td>▶ Measurement &amp; Calibration Methodology</td>
</tr>
<tr>
<td>▶ ASAM AE MCD Standards (UDS/CCP/XCP &amp; A2L)</td>
</tr>
<tr>
<td><strong>CANape Overview</strong></td>
</tr>
<tr>
<td>▶ Feature Summary</td>
</tr>
<tr>
<td>▶ Technical Details</td>
</tr>
<tr>
<td>▶ Option Packages</td>
</tr>
<tr>
<td><strong>Q &amp; A</strong></td>
</tr>
</tbody>
</table>
What is Run-Time Measurement and Calibration?

- **Run-time** = while the ECU is running (no breakpoints allowed!)
- **Measurement** = reading data from the ECU memory
  - Monitoring a dynamic list of internal software signals
- **Calibration (aka Software Tuning)** = writing data to the ECU memory
  - Changing parameter values without needing to flash program

Many possible applications for development and test
Open standards are key enablers!
Where is Run-Time Measurement and Calibration Used?

Application Areas

- Run-time measurement and calibration strategies are commonly used to support the development of embedded control systems (ECUs)

Examples: Controllers used in mechatronic and electronic systems found in automobiles, commercial vehicles, power generators, etc...

- Diesel / Gas / Electric Powertrain
- Transmission / Steering / Brakes
- Suspension / Chassis / Driveline
- Power / Battery Management
- Passive / Active Safety / ADAS
- HVAC / Body Electronics
- Machines / Implements
- And many others...
What are the Typical Use Cases?

- **Functional Development**
  - Software unit test (on-target and model-based)
    - White-box testing by reading internal variables of control algorithms
  - Base calibration (on-target and model-based)

- **Application Testing**
  - Adding ECU SW access for HIL or Test Benches

- **System / Vehicle Development**
  - System / Vehicle calibration
  - Dyno / hot / cold / altitude test

Common Examples:

- Tracking algorithm states
- Calibrating algorithms to achieve target behavior
What are the Typical Use Cases?

Supporting Multi-Source Measurement Fusion

- **Multi-Source** = Interfacing to multiple source devices, simultaneously
- **Measurement** = Active or passive acquisition of signal data
- **Fusion** = Time synchronously combining data from multiple sources, enabling simultaneous visualization and recording of all data together

  > Providing a more comprehensive, systemic view of the application system

- Measure all data sources with a signal measurement tool!
Run-Time Measurement and Calibration with CANape

Agenda

<table>
<thead>
<tr>
<th>Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Measurement and Calibration</td>
</tr>
<tr>
<td>▶ What is Run-Time Measurement and Calibration?</td>
</tr>
<tr>
<td>▶ Where is Run-Time Measurement and Calibration used?</td>
</tr>
<tr>
<td>▶ What are the Typical Use Cases?</td>
</tr>
</tbody>
</table>

**How does run-time measurement & calibration work?**

- Measurement & Calibration Methodology
- ASAM AE MCD Standards (UDS/CCP/XCP & A2L)

**CANape Overview**

- Feature Summary
- Technical Details
- Option Packages

Q & A
How does Run-time Measurement & Calibration Work?

**Measurement Methodology: Monitoring Software Signals**

Internal ECU software signals and states may be measured on demand.

At **run-time**, select any combination of signals you want to measure:

- Measure any persistent signal variables within the ECU software
- Configure the measurement at run-time, avoiding the need for ECU software changes
  - Tool-side configuration using industry-standard protocols
  - Select any desired combination of measurement signals
  - Adjust the measurement mode and sampling rate for each signal (CCP/XCP)

**INPUT**

- Raw input signals in driver SW
- Parameters

**Application SWC**

- Signals
  - Value 1: 0x03
  - Value 2: 0x22
  - Value 3: 0x55
  - Value X: 0x??
- Algorithm

**OUTPUT**

- Output signals in driver SW
- Raw input signals in driver SW
How does Run-time Measurement & Calibration Work?

Calibration Methodology: Live Parameter Calibration or “Software Tuning”

To optimize the performance of ECU algorithms, a calibrator can “tune” parameters used by the program to influence the output.

Reference data objects, such as:
- Parameters (1x1; scalars)
- Curves (1xN; 1-d arrays)
- Maps (NxN; 2-d arrays)

These reference data objects are not modified by the algorithm, but can be modified through calibration activities to affect the output of the system.
How does Run-time Measurement & Calibration Work?

ASAM AE MCD MC: Industry Standards for Measurement & Calibration

ASAM Vision

▶ “ASAM standards ensure that tools of a development process chain can be freely interconnected and allow a seamless exchange of data”

Advantages of Industry Standard Approaches

▶ Low Cost
  ▶ Off-the-shelf tools are purchased as needed, and do not require development time or upfront commitment

▶ Low Effort
  ▶ Project groups can focus on their product development tasks rather than having to specify and build tools

▶ Faster Ramp-Up
  ▶ Tools are available immediately, with a broad set of features to cover development needs throughout the whole process

▶ Compatibility Between Groups and Organizations
  ▶ Enables exchange of data between teams, suppliers, & OEMs. OEMs don’t have to deal with different tooling from each supplier.
How does Run-time Measurement & Calibration Work?

ASAM AE MCD MC: Industry Standards for Measurement & Calibration

- ASAM MCD 1MC standard ECU interfaces include CCP (CAN Calibration Protocol) and XCP (Universal Calibration protocol)

**Diagram:**
- Upper level automation system
- Measurement and calibration system
- ECU

**ASAM MCD 3MC**
- ECU description file
- *.A2L

**ASAM MCD 2MC**
- ECU memory address oriented mapping hex value to physical value for Speed:
- Example: Address: 0x1357
- Hex Value: 0x1fff
- Phys Value: 95 km/h

**Communication Interfaces:**
- RS232C, Ethernet, Microsoft COM, C-API (DLL)
- CAN, FlexRay, Ethernet, etc.
How does Run-time Measurement & Calibration Work?

ASAM AE MCD MC: Industry Standards for Measurement & Calibration

What is required to do ASAM Standard Run-Time Measurement and Calibration?

ECU Interface (1MC)

How do we send & receive ECU data?

- Via Bus Communication (embedded software)
  - Protocols: CCP/XCP/Diagnostics via SW drivers
- Via Hardware Interfaces
  - e.g. Vector VX1000 (Data Trace / JTAG)

Controller Description Database File (2MC)

- What SW objects are accessible in the ECU?
  - A2L databases describe the controller SW & HW

Measurement & Calibration System

- ASAM compliant MC System: Vector CANape

Vector offers solutions in all of these areas

- We can help you – just contact us!
How does Run-time Measurement & Calibration Work?

ASAM AE MCD 1MC: ECU Interfaces

Software Solutions
- Vector XCP embedded software drivers
  - Compatible with CAN, CAN-FD, J1939, Ethernet, FlexRay, LIN, DLLs, etc.
- Easy to implement
- Scalable features

Hardware Solutions
- Vector VX1000 interfaces
  - Connected via data trace ports
  - XCP on Ethernet to MC tool
  - Performance up to \(~50\times-1000\times\) CAN
- Architecture examples:
  - NXP/Renesas/ST: JTAG/Nexus/Aurora
  - TI (TMS570): RTP/DMM
  - Infineon: DAP/DAP2/HSCT
How does Run-time Measurement & Calibration Work?

ECU Interface Solutions – Choosing the right solution for your needs

- XCP-to-Data Trace
- XCP-to-JTAG
- XCP on Ethernet
- XCP on FlexRay
- CCP/XCP on CAN
- KWP, UDS, OBD

- Hardware-based
- Software-based

MC Bandwidth Requirement

More bandwidth needed due to:
- Faster application loop times
- Increased signal counts

ECU Application Content and Complexity
How does Run-time Measurement & Calibration Work?

ASAM AE MCD 1MC: Understanding XCP

**XCP: Industry-standard interface protocol**

- XCP: Universal (“X”) Calibration Protocol
  - Replacing CCP (CAN Calibration Protocol)
- Master-slave protocol
  - Reading & writing ECU memory

**XCP offers several key advantages:**

- Runs on common automotive / instrumentation channels - variants for CAN(FD), FlexRay, Ethernet, LIN, etc.
- Configure measurements at run-time
- Synchronous & polling data acquisition modes
- Advanced features
  - Rapid-Prototyping / bypassing
  - Flash programming
- Used effectively in a variety of application contexts – P/T, chassis, safety, body, etc.
How does Run-time Measurement & Calibration Work?

Vector VX1000 – High Performance ECU Interface

Hardware for interfacing to microcontroller data trace or debug ports

- **Modular hardware** supporting popular embedded microcontroller architectures
  - NXP/Freescale/ST – JTAG / Nexus / Aurora
  - Infineon – DAP/DAP2/Aurix
  - PCie – for ADAS Fusion ECUs

- **High performance** measurement & calibration
  - Up to **1000x** bandwidth of XCP on CAN
  - Faster loop times (<1ms) & Larger signal counts than XCP on CAN

- **Scalable features**
  - Optional bus interfacing (CAN/FD/FR/BR-R)
  - On-target bypassing
  - Flash programming
  - Cold-Start / First-Loop

- **XCP on Ethernet** tool interface

- **Easy integration** of SW driver and possibility for no-code operation (lower performance)
How does Run-time Measurement & Calibration Work?

VX1000 Modular System Implementation Concept

- **ECU**
- **VX POD**
- **VX Base Unit**
- **FIFO**
- **DPRAM**
- **XCP Driver**

**VX1000** embedded SW driver supports data trace and RAM data copy methods.

“Driver-less” JTAG polling is also possible without a VX1000 SW driver.

DPRAM in VX1000 mirrors RAM in ECU.

**Small FPGA-based POD** connects to ECU debug port and forwards data to base unit.

**ECU data trace transmission on serial cable**

**FIFO queue copies data between ECU data trace and DPRAM**

**XCP driver presents DPRAM to cal tool as if it were the ECU**

**ASAM MC System (e.g. CANape)**
How does Run-time Measurement & Calibration Work?

**A2L database**

**User:** I need to measure Vehicle Speed...

**CANape Measurement & Calibration System**

<table>
<thead>
<tr>
<th>Memory Address</th>
<th>Name</th>
<th>Unit</th>
<th>Min</th>
<th>Max</th>
<th>Conv</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xCF1A35D0</td>
<td>Vehicle Speed</td>
<td>MPH</td>
<td>0</td>
<td>200</td>
<td>( \pi \times \frac{\pi}{2} )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Memory Address</th>
<th>HEX value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xCF1A35D0</td>
<td>0x1FD4</td>
</tr>
</tbody>
</table>

**A2L Database Purpose:**

- Contains all necessary details about the controller software and (optionally) the MC interface
- Allows CANape to abstract the technical details (Address/data type/conversions) for acquiring engineering values for signal data from the ECU
- **Vector’s ASAP2 Toolset** product provides A2L generation, merging, and other automation features
## Agenda

<table>
<thead>
<tr>
<th>Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Introduction to Measurement and Calibration</strong></td>
</tr>
<tr>
<td>▶ What is Run-Time Measurement and Calibration?</td>
</tr>
<tr>
<td>▶ Where is Run-Time Measurement and Calibration used?</td>
</tr>
<tr>
<td>▶ What are the Typical Use Cases?</td>
</tr>
<tr>
<td><strong>How does run-time measurement &amp; calibration work?</strong></td>
</tr>
<tr>
<td>▶ Measurement &amp; Calibration Methodology</td>
</tr>
<tr>
<td>▶ ASAM AE MCD Standards (UDS/CCP/XCP &amp; A2L)</td>
</tr>
<tr>
<td><strong>CANape Overview</strong></td>
</tr>
<tr>
<td>▶ Feature Summary</td>
</tr>
<tr>
<td>▶ Technical Details</td>
</tr>
<tr>
<td>▶ Option Packages</td>
</tr>
<tr>
<td><strong>Q &amp; A</strong></td>
</tr>
</tbody>
</table>
CANape at a Glance

An ASAM Standard MC System

- ASAM MCD compliance
- Vector is a participating & authoring member of ASAM MCD
- Support for the latest versions of the ASAM MCD Standard
- Over 16,000 licenses deployed worldwide across 250+ companies

Vector CANape v17

- Turn-key solution for measurement, calibration, and diagnostic use cases
  - High performance measurement
    - XCP data streams up to 50MB/s!
    - Distributed recording features
  - 64-bit (new for v17!)
  - Drag & drop interface
  - Best-in-class tool support included! (phone & email)

Broad feature set in base package

- Fully featured measurement, calibration, & diagnostics
- Data visualization, analysis, & mining
- Calibration data management
- Function & Scripting language
- MATLAB®/Simulink® integration
CANape Overview

Understanding the Differences Between Vector Tools

CANoe: Network and distributed system simulation and design, diagnostics, automated test, and all CANalyzer network analysis capability

CANalyzer: Measurement and analysis of networks & distributed systems

CANape: Run-time measurement, calibration, and diagnostic interface for ECU application development

vSignalyzer: Measurement data visualization, analysis, & data mining

vMeasure Exp: Run-time measurement of I/O, ECU, and network data

*Some vMeasure Expert components not included in CANape
## CANape Overview

### Feature Component Summary

<table>
<thead>
<tr>
<th>ONLINE Features</th>
<th>OFFLINE Feat.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECU Measurement</td>
<td>Calibration Data Management</td>
</tr>
<tr>
<td>ECU Calibration</td>
<td>Database Handling &amp; Generation</td>
</tr>
<tr>
<td>Bus Monitoring</td>
<td>Offline Evaluation &amp; Data Mining</td>
</tr>
<tr>
<td>Digital/Analog Measurement</td>
<td></td>
</tr>
<tr>
<td>Diagnostics</td>
<td></td>
</tr>
<tr>
<td>Automation &amp; Test Benches</td>
<td></td>
</tr>
<tr>
<td>Audio / Video / Image Processing</td>
<td></td>
</tr>
<tr>
<td>Flash Programming</td>
<td></td>
</tr>
<tr>
<td>Bypassing</td>
<td></td>
</tr>
<tr>
<td>Data Logging</td>
<td></td>
</tr>
</tbody>
</table>

### Model-based Development

### Option Packages:

- Driver Assistance
- Simulink XCP Server
- vCDM Teams
- vMDM
- vCDM Studio
- ASAP2 Toolset*
- vSignalyzer
- vMeasure Expert*

*Some feature components not included within CANape package / sold separately
Device & Data Interfaces

CANape: Device & Data Support

Bus Protocols, Calibration Interfaces, & Diagnostics

- CANape: Device & Data Support
  - Vector XL family
    - USB, PCIe, PXI, Expresscard,…
  - USB
  - Firewire
  - Ethernet
  - RS 232
  - CAN
  - CAN-bus based I/O
  - IOcab
  - CSM MiniModuls
  - CAN-based I/O
  - VN1630/40
  - NI-DAQmx
  - DAIO driver
  - CSM XCP Gateway
  - Video, Audio
  - GPS

Vector VN16xx USB
- Vector XL family
- USB, PCIe, PXI, Expresscard,…

XCP 1.x, 2.0, Cooling

Vector VN16xx USB Network Interfaces

VCX3000
- VN3300
- VN3600

I/O, Multimedia, & GPS

VCX1000 HW XCP Interface

VCX1060 + VX1543 POD

VCX76xx FlexRay

VCX56xx Ethernet

VCX1060 + VX1543 POD

VCX76xx FlexRay

VCX56xx Ethernet

VCX1060 + VX1543 POD
Connect CSM Measurement Devices via CAN or Ethernet

Official Vector Partner for Measurement Technology

MiniModuls®

VN16XX

MiniModuls®

EtherCAT MiniModuls®

XCP-Gateway

XCP V1.4 – Time Synchronization

MC Hardware (e.g. VX1000)

Time Sync. via Broadcast

ECU

I/O Module

I/O Module

XCP on Ethernet

XCP on Ethernet

XCP on Ethernet

XCP on Ethernet

CANape
Easy to Use Interface

Drag and drop

- Drag a signal to the configuration
- Signals are added automatically to the active recorder
- Add signals to pre-existing windows
- Move/Copy signals from one window to another
- Quick and easy configuration design
CANape: Measurement System

CANape Measurement Data Flow

CANape Measurement Configuration

Input Data
- ECU Data
- FlexRay, CAN, LIN messages
- Analog, digital I/O signals
- Video/Audio/GPS signals

Data preparation
- MATLAB®/Simulink®
- User defined script functions
- Trigger/Filter
- Function DLLs

Multiple Recorder

Data Logging
- Data Storage

Data Display
- Signal oriented displays
- Message Traces
- Multimedia windows
## Example Measurement Use Case – Transmission system

### Measuring everything simultaneously, in one tool

<table>
<thead>
<tr>
<th>Transmission System</th>
<th>Signals Measured</th>
<th>CANape (MC System)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clutch actuators, temperatures, ignition/battery/sensor input voltages, pulse counts, etc....</td>
<td>Torque Reduction Request, Engine RPM, Output shaft speed, Selected Gear, any other bus signals to/from ECU, etc....</td>
<td>Example Data devices:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Acquisition Hardware connected via CAN, USB, Ethernet, PCMCIA, etc.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Capture digital, analog, thermocouple, pressure, PWM, sensors, etc...</td>
</tr>
<tr>
<td>OS / DRVs System SW</td>
<td>All software signals: Sensor interpretation, Actuator requests, algorithm states, System/OS, etc..</td>
<td>Data devices:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vector Interface Hardware Measure CAN, LIN, FlexRay, MOST, J1939, etc...</td>
</tr>
<tr>
<td>ECU HW</td>
<td></td>
<td>Data devices:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CCP, XCP, KWP2000, &amp; UDS Measure any software signals w/o halting SW</td>
</tr>
</tbody>
</table>

### Transmission System

- Photo by: A7N8X (wikipedia)

### Signals Measured

- Clutch actuators, temperatures, ignition/battery/sensor input voltages, pulse counts, etc....

### CANape (MC System)

- Example Data devices:
  - Data Acquisition Hardware connected via CAN, USB, Ethernet, PCMCIA, etc.
  - Capture digital, analog, thermocouple, pressure, PWM, sensors, etc...

- Data devices:
  - Vector Interface Hardware Measure CAN, LIN, FlexRay, MOST, J1939, etc...

- Data devices:
  - CCP, XCP, KWP2000, & UDS Measure any software signals w/o halting SW
Visualization & Data Logging Features

**Visualization**
- Powerful display windows
- Display both live & recorded data

**Data Logging Features**
- Acquire data from multiple devices simultaneously
  - All signals can be recorded into a single measurement file (MDF)
- Export of MDF to other formats
- Record data to multiple file recorders simultaneously
  - Each recorder can have a unique trigger and measurement list
- Record raw CAN traffic in BLF/MF4
- Multimedia files (.avi, .mp3, etc...)
- Distributed recording feature (multiple PCs)
Display of Multimedia Signals

**Multimedia Window**

- Record & display video signals
- Example: Rainy weather shown in video

**Graphic Window**

- Plot Sonogram in Graphing window: Visual representation of the frequencies in a sound
Calibration Feature Summary

**Tune Software**
- Visualize software parameters in engineering formats
- Tune at run-time
- Record & document calibrations
- Flash program calibrations
- Support for offline calibration
- Support for AUTOSAR calibration concept

**Manage Parameter Sets**
- CDM Studio Interface
  - Side-by-Side parameter sets comparisons
  - Fully Featured
  - Drag and drop

Scales Up: Enterprise Solution
- vCDM solution
**Example Calibration Use Case – Transmission System**

**Run-Time Calibration: Tuning software during measurement**

- Using a calibration interface, it is possible to manipulate parameter objects used by the algorithm at run-time.
- Run-time calibration saves effort in development iterations compared to older “Flash & Try” methods.

- Easy interface for tuning parameters, maps, & curves.
- Visualize software signals & states.
- Leverage new parameter values to update HEX files or embedded SW.

MC System (e.g. CANape)

**ECU interface**

**Measurements**

| Torque maps, shift-points, timing tables, sensor & actuator tables, gains, offsets, modes, etc. | Calibration |

TCM (ECU)
CANape: Diagnostic System

Diagnostic Feature Summary

UDS ISO 14229 Diagnostic Support

- Seamlessly integrate diagnostic and measurement/calibration functions
- Access to diagnostic data (raw & symbolic access), also OBD data
- Utilize scripts in diagnosing ECUs
- Support of CANdela-based Diagnostics (CDD/ODX)

Vector Security Manager integration (new for v17)

- Implementing security algorithms of OEM-specific Security Sources

Diagnostic Console

- Display of diagnostic response (positive and negative)
CANape: Panels

Custom Panels

- Easily build custom panels for:
  - Presentation of measurement or diagnostic data
  - Calibration of parameters
  - Execution of scripts
Mapping of Vehicle GPS data

- Now included in base CANape software package
- The GPS window in CANape allows users to visualize the current location of the vehicle based on acquired GPS data
- View position on a map live or after the measurement
  - Show data mining hits in GPS window
  - Plot multiple objects & sensor object data
- Several types of mapping software can be used:
  > MapPoint® (U.S.)
  > MappleX® (Japan)
  > OpenStreetMap (World)
  > Satellite images

Data is time synchronized by the global measurement cursor – When activated, all data shown is from the same instant of time.
Features for User Feedback During Testing

In-vehicle user interface features

Text to Speech:
- Audible announcement of values and trigger conditions, allowing the driver to focus on the road

Display Range Violations:
- Quickly identify values of interest among large numbers of signals

Floating Windows:
- Windows can be moved outside the CANape main frame to allow for convenient use of heads-up display monitors
Vertical use case support through the CANape Function & Scripting language

- **Functions**
  - Automatic calibration
  - Triggering
  - Virtual signal calculation
    - Online / Offline

- **Scripts**
  - Search MDF files
  - Parse data
  - Multi-pass capable
  - Output results
    - Global Variables
    - Write window / Text Log
  - Export MDF to other formats

```c
function My_Function (input, output)
{
  double  a = 1;
  long     b = 2;
  output = a + b*input;
  printf("The result is: %d", output);
  return output;
}
```

```c
SPrint(FileMDFName, "%s%s", File, File);
SPrint(FileMATName, "%s%s", File, File);
// remove existing file first
FileRemove(FileMATName);

// call the converter. The different:
// the vector.ini file of the <CANape
if(FileExists(FileMDFName) == 0) {
  do {
    // convert the MDF-File to a Matl.
    // been defined before by the con
    Result = ConvertMeasurementFile(F
    } while (Result != 0);
```
Easy Visualization and Analysis of Data

**Easy Visualization**
- Global Measurement Cursor
- One-click statistics dialog

**Automation through Scripts**
- Parse files programmatically
- Evaluate data, export data, and manage data mining

**Virtual Signals**
- CANape Functions or C/C++ DLLs
- MATLAB/Simulink models

**Data Mining**
- Define event logic
- Batch process entire directories
- Report generation with hits and files
Supporting model-based development

- Vector supports model-based development through a collection of features designed to facilitate tool cooperation between The Mathworks’ MATLAB/Simulink tool suite and CANape
- Provided free via the Vector MATLAB/Simulink MC Add-on package

CANape’s MATLAB/Simulink features:

- Model Explorer window
- Simulink blockset & DLL target
- Algorithm designer
- Export data in MATLAB formats
- .M & .MAT formats
- MATLAB automation of CANape
- xPC target® support
- Option: Simulink XCP Server

Vector is an official partner of The Mathworks®
CANape: MATLAB®/Simulink® Features

Model Explorer - Visualize Simulink® / Stateflow® Models in CANape

Model-based Advantages: Specification and documentation of your algorithm
- During model-based development, the algorithm is created as a model in Simulink
- The model is code-generated to integrate and deploy the software in the ECU target

Model Explorer: Interact with your algorithm, from a model perspective
- After code generation, the model itself still provides the best specification of the system
- To leverage the model when working with code generated software, CANape provides a special model view window that shows signals and parameters integrated in the model
Feature Package for the Validation of Driver Assistance Systems

- **Easily validate object detections** from a variety of driver assistance systems
  - Park Assist, Lane Keep, Adaptive Cruise, Blind Spot Detect, Emergency Brake, etc.
  - Use any object data acquired by CANape

- **Visualize objects live or offline** via bird’s eye views, video overlays, occupancy grids, and ADASIS-enabled GPS mapping
Run-Time Measurement and Calibration with CANape

Contact Information and Q & A

Thank you for your attention.

For detailed information about Vector and our products please visit us at:

www.vector.com

Presented By:

Jeff Rothenberg

Jeff.Rothenberg@vector.com

Vector North America, Inc.

Let us know how we can help you get started!

MATLAB®, Simulink®, Stateflow®, xPC Target®, Real-Time Workshop®, and other related products are copyrights or trademarks of The Mathworks® in the U.S. and other regions. NI-DAQmx® is a copyright of National Instruments™ in the U.S. and other regions. CSM and products are trademark/copyrights CSM GmbH. All other trademarks/copyrights belong to their respective owners in the U.S and elsewhere.
ADAS Logging

CANape V16 DHPR – 2080 MByte/s Write Performance via 2 x BRICK PC

Performance Test

Interfaces:
8 x VX1132 RADAR
4 x Full HD CAM

Recording on 2 x Brick PC

CANape DHPR:
Distributed High-Performance Recording feature
CANape Option “Driver Assistance” for ADAS Validation

Point Cloud “Scene” Visualizations