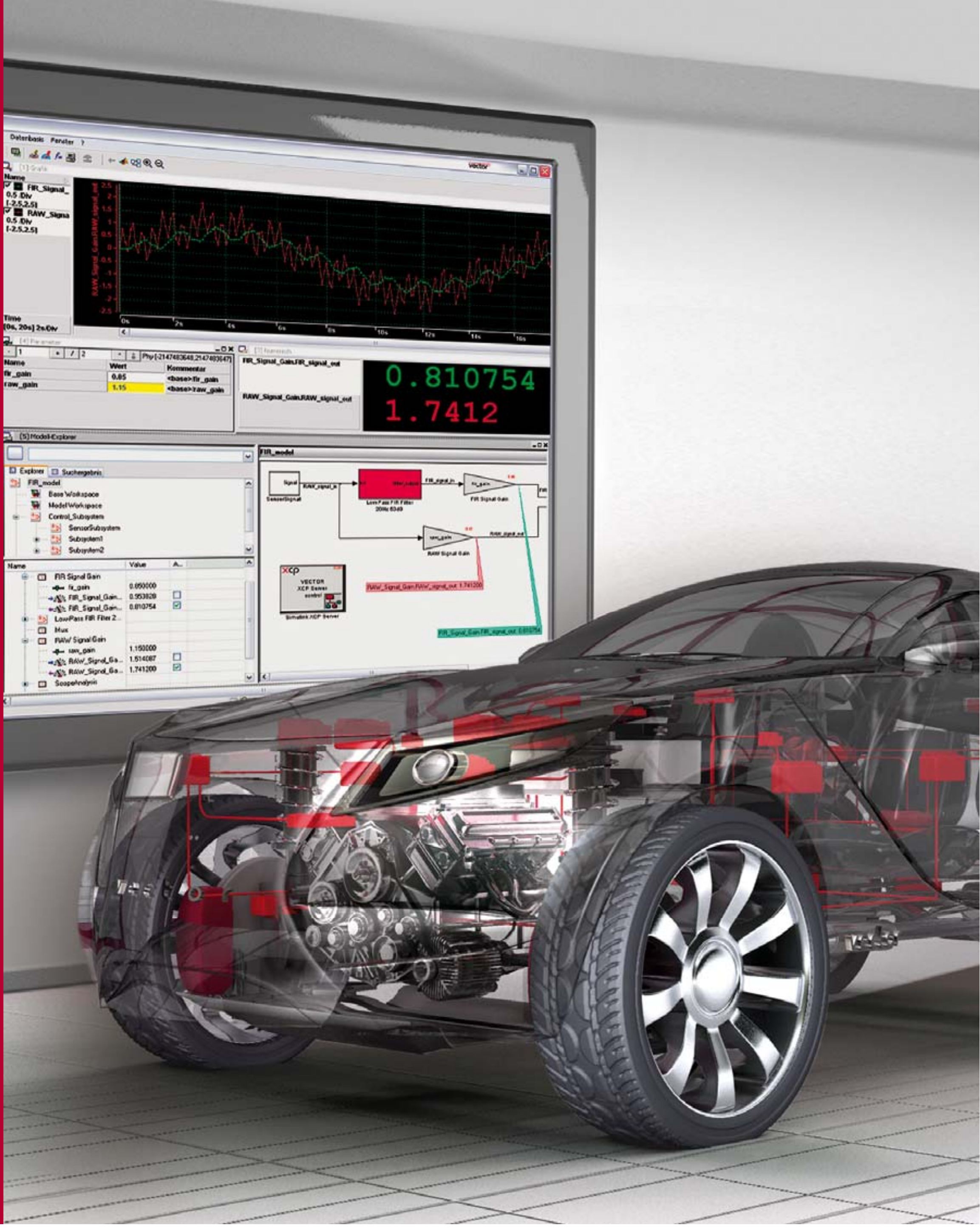


Rapid Prototyping with CANape
Version 1.0
2010-11-22

vector

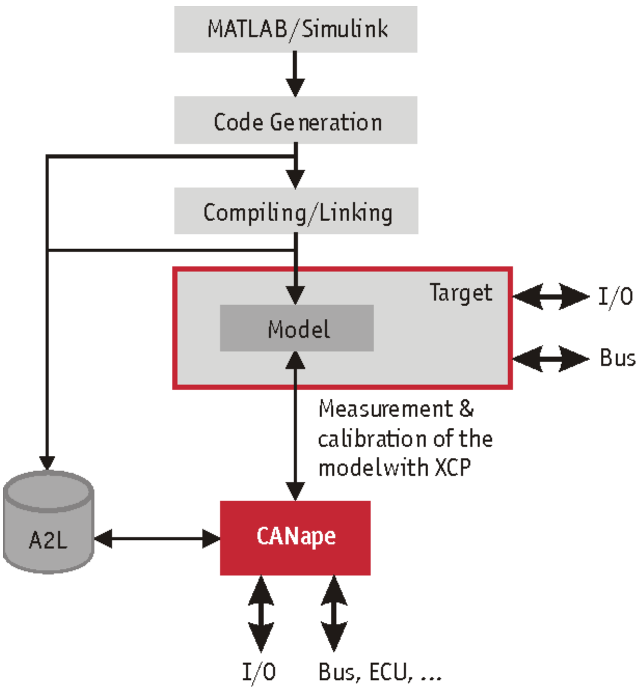
Table of Contents

1.0 Overview

Development tasks can be solved significantly more efficiently and cost-effectively when standards are applied. The XCP ASAM standard is available for model-based software development or rapid prototyping. You can use XCP to measure and calibrate code regardless of the specific runtime environment that is used. The specific environment in which the code was developed does not matter either. Usage of XCP and CANape does not depend on the use of specific development tools; they can handle traditionally developed code just as well.

From a Simulink model, the model-based development code and a description file (A2L) are generated for a specific runtime environment (target). After compiling and linking, the model runs on this target. CANape updates the address in the A2L from the linker MAP file and lets you measure and calibrate the model. Various solutions are available to you as targets:
CANape, a PC with Win32 operating system or real-time operating system, dedicated rapid prototyping hardware, evaluation boards or ECUs.

CANape and XCP offer support to function and software developers and calibrators – from the start of a development to its production maturity.



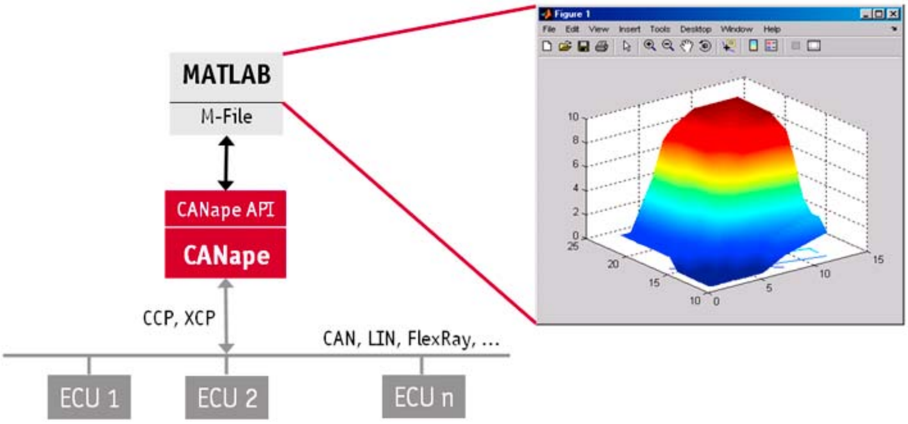
The Simulink models are visualized via the Model Explorer in CANape, which is independent of the runtime environment of the code (for more information, please refer to Chapter 10.0 of the same name).

In the following, we give you an overview of capabilities and options in model-based software development with CANape.

2.0 Functional development with MATLAB

To ensure that other programs can access internal ECU parameters or bus signals, CANape provides them with an application programming interface (CANape API).

Controlled by a MATLAB M-file, measurement and calibration tasks are implemented via CANape with the help of script language. CANape takes on all handling of the drivers and interfaces here. This approach is used to implement tasks such as automated optimization of algorithms in the ECU.

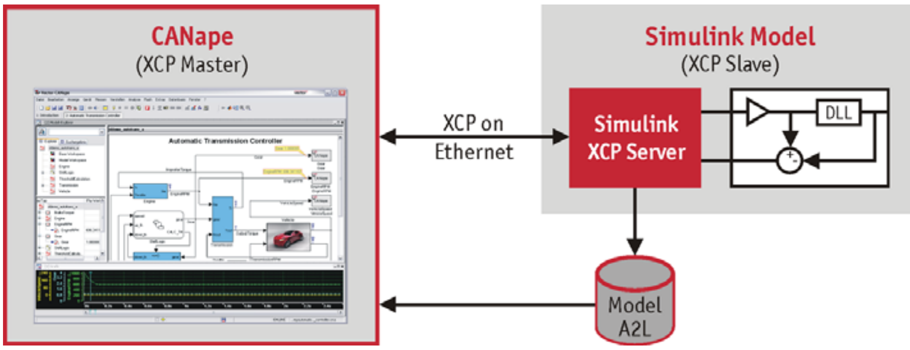


Note:	You will find additional information in the Application Note "AN-IMC-1-004 Interface Programming between CANape and MATLAB"

3.0 Simulink as runtime environment – CANape as convenient user interface

In the context of model-based software development, functions of the application under development are checked in an iterative process. This involves repeatedly re-running the model in Simulink from The MathWorks. The CANape Option Simulink XCP Server gives all functional and software developers a convenient way to manage parameters and measure signals from the Simulink model, efficiently and without instrumentation. If the model contains binary components, such as MEXW32 files or DLLs, they can also be measured and calibrated by XCP.

The CANape configuration used here can be reused in a later development phase.



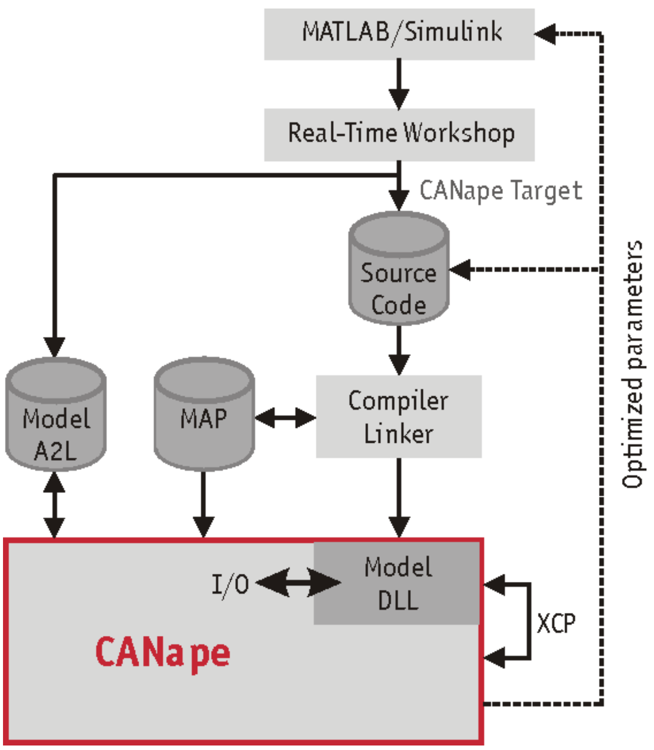
4.0 CANape as runtime environment

CANape is a runtime environment for models developed in Simulink. Model inputs and outputs are linked to parameters from the CANape project via I/O blocks. Code is generated via the Real-Time Workshop. After compiling, a DLL that works together with CANape and has an XCP interface is available to you. So you can measure and parameterize the model without having to change parameters in Simulink and generate new code first. The final parameterization can then be exported from CANape and read back into MATLAB/Simulink.

The input blocks make it possible to pass scalar values, matrices and time-based input vectors into the DLL generated from the model. The model, expanded by the CANape IO blocks, can be run in Simulink as before.

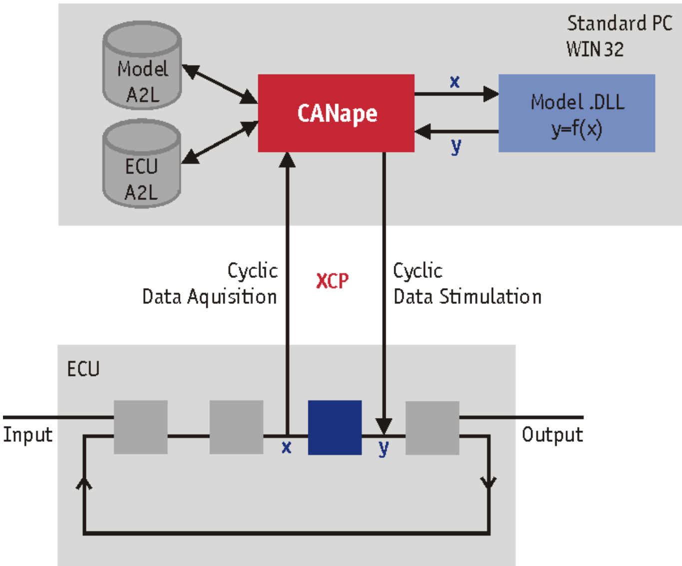
Besides being used to develop a control algorithm that is fed with specific data from ECUs, bus data and analog data, etc., CANape covers other uses as well, such as:

- Online computations during a measurement
- Offline computations for evaluation purposes

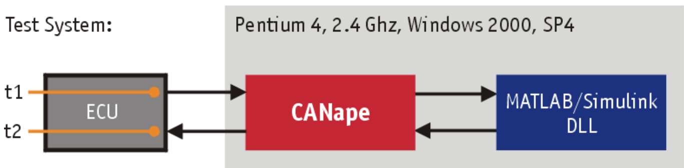


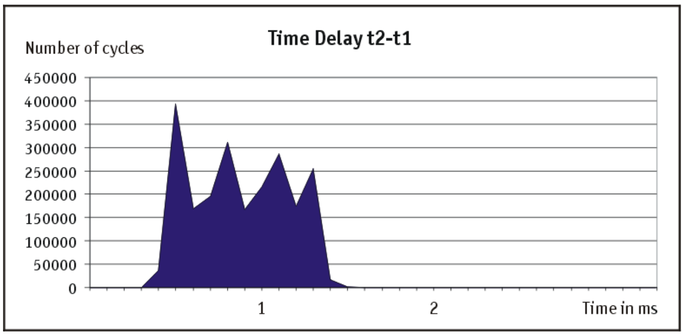
5.0 Bypassing with CANape

A new or improved algorithm is to be integrated in an ECU. This involves creating a model in Simulink and generating the CANape target code via the Real-Time Workshop. After compiling and linking, a DLL is available to you that runs in CANape.
Parameters from the ECU are sent to CANape in an event-driven way via standard mechanisms (DAQ) of the XCP calibration protocol. The signals serve as input parameters to the model. The output data of the computation are synchronously written back to the ECU via CANape and the XCP mechanism (STIM).



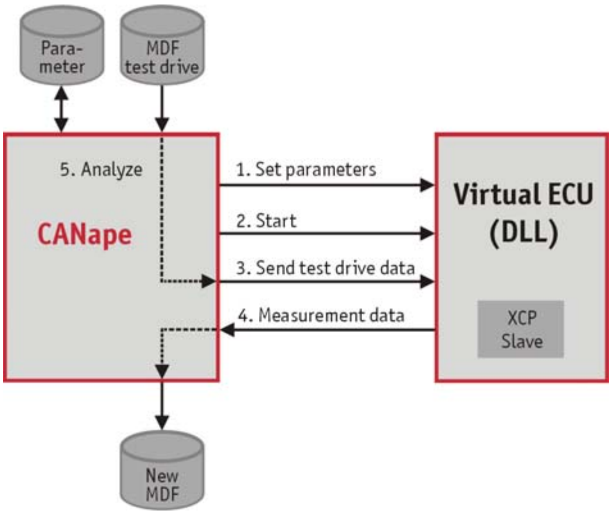
Bypassing does not incur any additional hardware costs, since the necessary standard PC is available to every user. Although a Windows operating system does not represent a classic real-time environment, it can be used to achieve very short round-trip times of under 2ms:



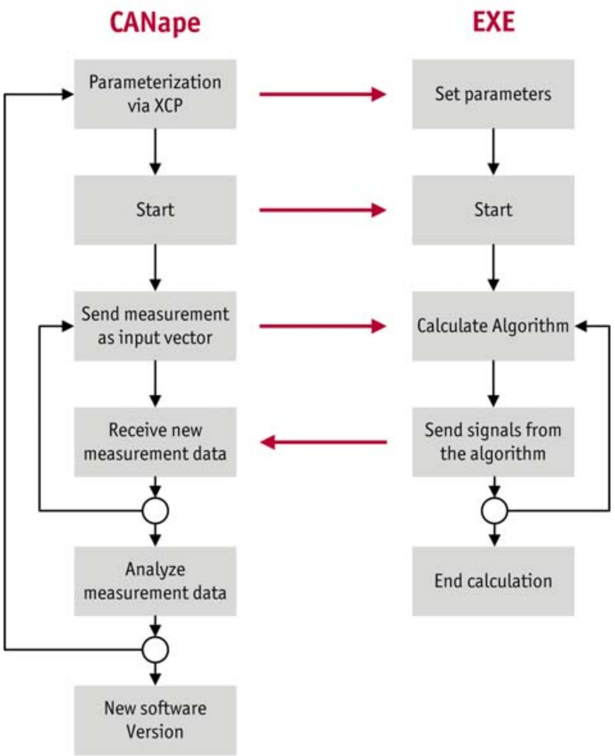


6.0 Prototyping on PC platform – The PC as ECU

CANape offers the possibility to optimize your application running as a virtual ECU (DLL) on the PC via XCP on Ethernet. The test drive data are the input vector for the application (represents e.g. 1h of a test drive) and the application is running much faster on the PC than on the ECU (e.g. some seconds or minutes).
The new MDF file represents the same length than the input test drive.



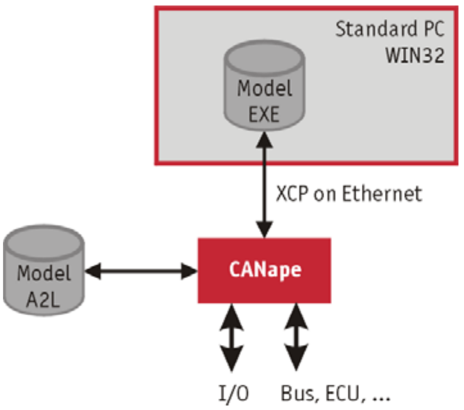
The user can manually evaluate results and set parameters here, or this can be done in an automated process.



7.0 SiL solution with CANape

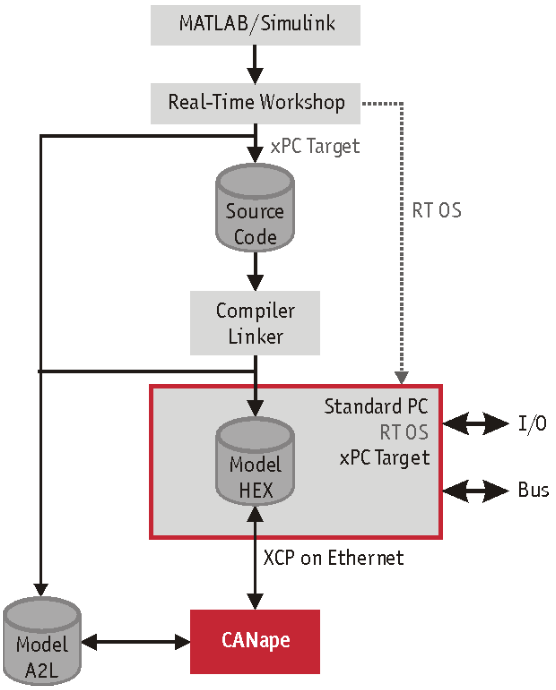
ECU applications generally run much quicker on a PC platform than on the controller platform.
CANape lets you execute parameterization of the application running on the development computer (standard PC with Windows operating system), start the computation and acquire the newly generated measurement data from the application. The application runs as quickly as possible, and it might use logged measurement data as input parameters during a vehicle trial, for example. Analysis of the new results in CANape can be performed either individually or automated – in turn, this serves as a basis for re-parameterization of the application. After parameterization, the new measurement results are available very quickly, which significantly shortens development time because of the very short iteration cycles. The results of different parameterizations can be compared by utilizing identical data as an input vector.

XCP on Ethernet is implemented for communication between CANape and the application. Vector would be glad to support you in integrating the XCP driver in your application.



8.0 Rapid Prototyping with xPC Target

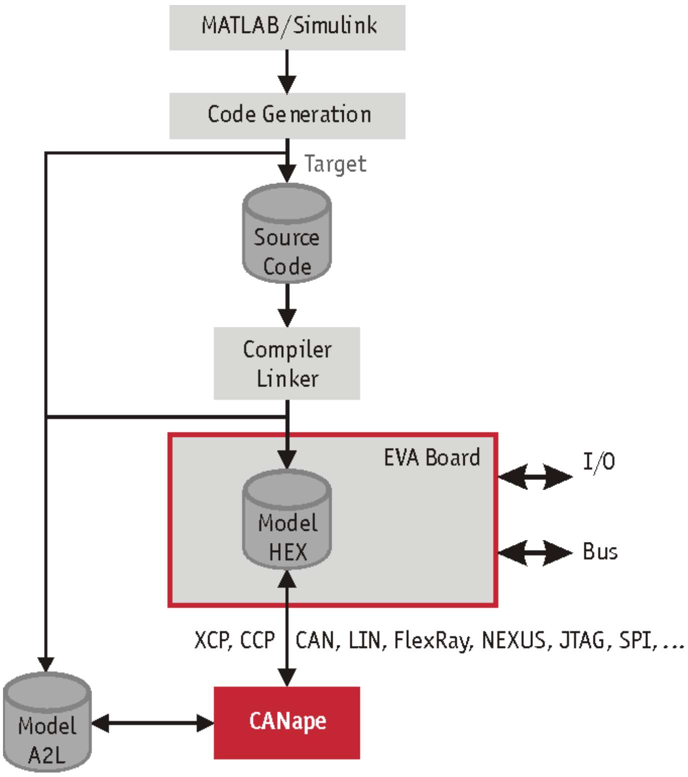
The xPC Target from The Mathworks offers the option of generating a real-time operating system (RT OS) for a PC hardware platform. In generating the code for the model, an XCP driver is automatically integrated in the model. You can use standard PCs or industrial PCs as hardware platforms.



9.0 Rapid prototyping with existing ECUs or EVA boards

The existing pre-production ECU is very often available as a very economical rapid prototyping hardware. The advantage – besides cost savings – is that existing I/O may be used, and protocol stack elements are available, e.g. diagnostics, XCP/CCP and/or network management.

On projects with new controller platforms, evaluation boards are the natural choice. The board is easily accessed by XCP. Furthermore, the cost-free Vector XCP driver may be used. Then the Real-Time Workshop Embedded Coder can be used to generate the code with XCP driver for the evaluation board.

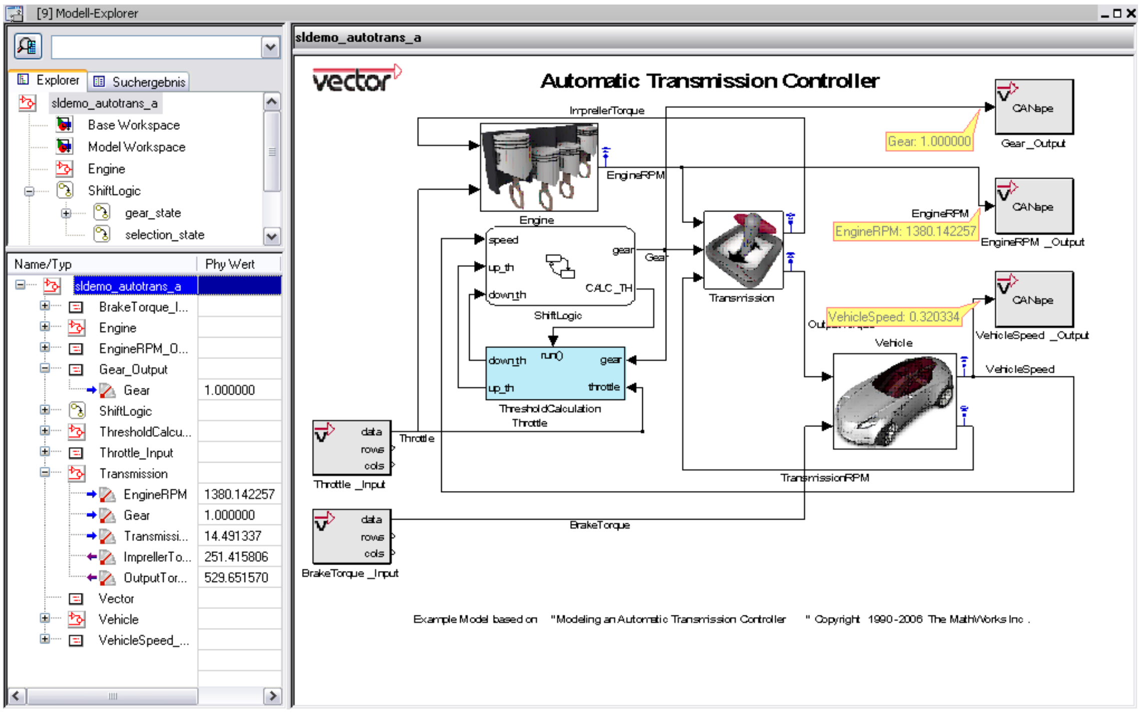


10.0 Visualization of Simulink/Stateflow models – CANape Model Explorer

In model-based software development, as a general rule there are two different perspectives on the same contents: Software developers know their applications from the Simulink perspective, while application engineers have their A2L perspective that is control module centered.
The Model Explorer in CANape provides you with a unified representation of these two perspectives.

After installing the "MATLAB Integration Package" supplied with CANape, a graphic-based view of the model is generated from the Simulink model. These graphics are integrated in CANape and let you navigate through the model. You can view every layer of a hierarchically organized model, and calibration parameters can be selected and parameterized directly. Search functions in CANape facilitate the efficient location of objects in the model.

Graphics may be utilized in every CANape project, even without a MATLAB/Simulink license.
If an A2L file is available the Model Explorer gives you direct access to the model's objects and offers navigation capabilities.


[9] Modell-Explorer
sldemo_autotrans_a
Explorer
Suchergebnis
Base Workspace
Model Workspace
Engine
ShiftLogic
gear_state
selection_state
Name/Typ
Phy Wert
BrakeTorque_I...
EngineRPM_O...
Gear_Output
Gear
1.000000
ThresholdCalcu...
Throttle_Input
Transmission
EngineRPM
1380.142257
Transmissi...
14.491337
ImprellerTo...
251.415806
OutputTor...
529.651570
Vector
Vehicle
VehicleSpeed_...
Automatic Transmission Controller
ImpellerTorque
EngineRPM
CANape
Gear: 1.000000
Gear_Output
EngineRPM: 1380.142257
EngineRPM _Output
VehicleSpeed: 0.320334
VehicleSpeed _Output
VehicleSpeed
speed
up_th
down_th
gear
CALC_TH
ShiftLogic
Transmission
Vehicle
ThresholdCalculation
throttle
run()
Throttle
data
rows
cols
Throttle _Input
TransmissionRPM
BrakeTorque
BrakeTorque _Input
Example Model based on "Modeling an Automatic Transmission Controller " Copyright 1990-2006 The MathWorks Inc .

Get more Information!

Visit our Website for:

> News

> Products

> Demo Software

> Support

> Training Classes

> Addresses

www.vector.com