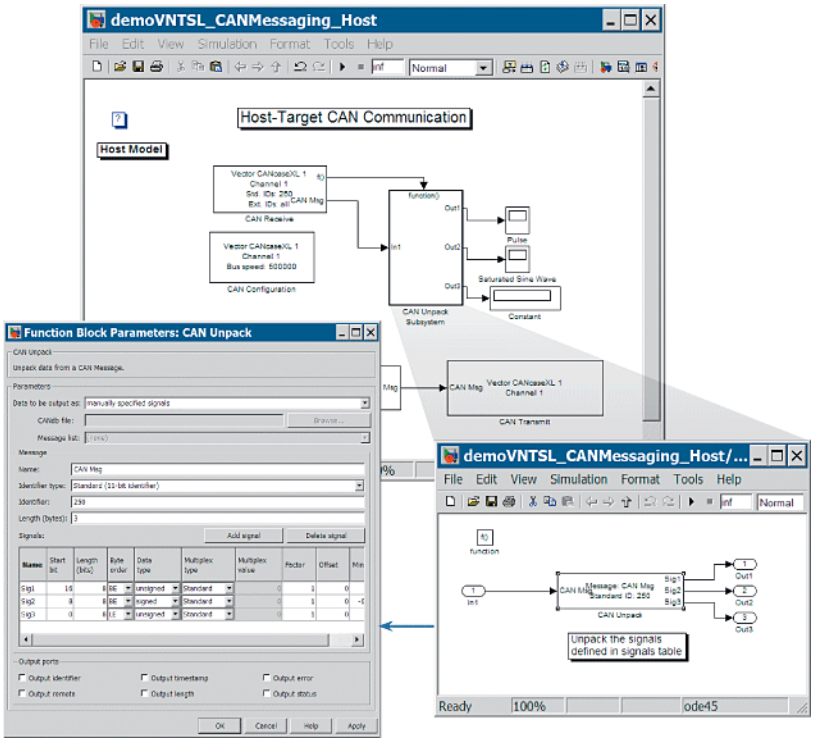
Vehicle Network Toolbox 1.4

Communicate with in-vehicle networks using CAN protocol

Vehicle Network Toolbox™ provides connectivity to CAN devices from MATLAB® and Simulink®. The toolbox provides functions for encoding, decoding, and filtering CAN messages, enabling you to transfer messages between the MATLAB workspace and a CAN bus. You can use the toolbox to run test sequences that log and record CAN messages for later analysis. You can also use it to connect with industry-standard CAN database files and replay recorded sequences of messages. The toolbox includes Simulink blocks that enable you to simulate message traffic on a CAN bus or connect models to a live network. Vehicle Network Toolbox supports CAN interface devices from Vector, Kvaser, and National Instruments.

Key Features

- MATLAB functions for transmitting and receiving CAN packets
- CAN communication blocks for connecting a CAN bus to a Simulink model
- Bit packing and unpacking functions and blocks for simplified encoding and decoding of CAN messages
- Message traffic GUI for visualizing live CAN network traffic
- Message filtering and logging
- Support for Vector, Kvaser, and National Instruments interface hardware and Vector CAN database (.dbc) files



Simulink model illustrating a host model for host-target CAN communication.

Working with Vehicle Network Toolbox

Vehicle Network Toolbox lets you interact directly with the CAN bus from MATLAB and Simulink. You can execute toolbox functions from the MATLAB command line and through MATLAB programs. The toolbox also contains Simulink blocks that enable you to connect a Simulink model to the CAN bus.

Configuring CAN Channels

CAN channel functions in MATLAB and CAN configuration blocks in Simulink enable you to define a connection to Vector CAN interface hardware that establishes a physical connection with the CAN bus. The toolbox provides CAN channel functions to query and configure CAN interface hardware settings, such as bus speed and transceiver settings. You can also verify other CAN channel properties, such as the number of messages available and the number of messages received or transmitted on the channel. By attaching Vector CAN database files to CAN channels, incoming messages are automatically presented using information stored in the database. After defining a CAN channel, you can send and receive CAN messages on the channel.

Sending and Receiving CAN Packets

CAN messages contain properties for storing the CAN message identifier (standard 11-bit or extended 29-bit), the time stamp, and up to 8 bytes of CAN data. Transmit and receive functions and blocks in the toolbox enable the sending and receiving of CAN messages over CAN channels. For large data sets, you can log CAN messages for offline analysis.

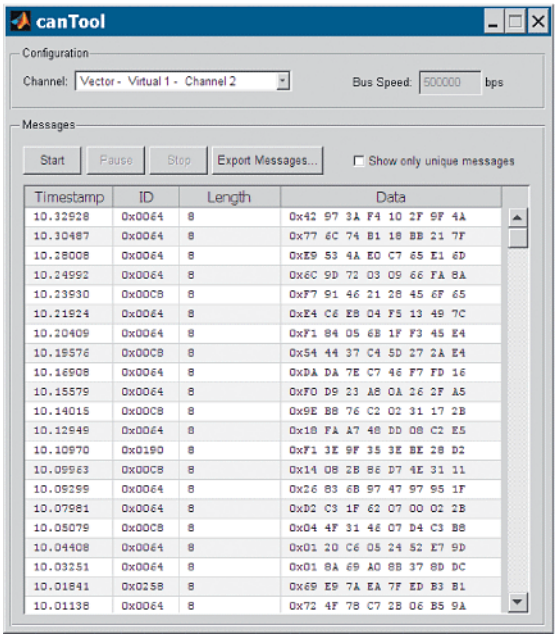
Building and Extracting Signals from CAN Messages

Vehicle Network Toolbox provides functions and blocks for encoding and decoding CAN messages. CAN message data may contain data representing multiple signals. Unpack functions and blocks let the user specify start bit, signal length, data type, and byte ordering. Pack functions and blocks provide the same options for building up data for CAN message transmission.

Visualizing CAN Traffic and Handling CAN Messages

The toolbox provides a graphical user interface (GUI) for visualizing active traffic on a particular CAN channel. You can run the canTool GUI while performing other tasks in MATLAB or Simulink.

When traffic on the network contains more information than needed for your application, you can limit the number of CAN messages received by a CAN channel to a defined range of CAN message identifiers. Using filter functions and mask settings in the toolbox, you receive only the messages needed for your application.



Live CAN bus traffic on the network displayed by canTool, a graphical user interface for visualizing data in Vehicle Network Toolbox.

Working with Vector CAN Database Files

In MATLAB or Simulink, you can associate a Vector CAN database file with a CAN channel or message, enabling you to encode and decode CAN messages using application-specific message and signal names, such as EngineMsg and EngineRPM, as well as scaled engineering units. The ability to work with industry-standard database files simplifies the interaction with the CAN bus because the database not only specifies the message list and component signals, but also provides the bit packing and unpacking rules for the associated signals. Because signal data type, start bit, length, and byte order are all predefined for the messages in the database, you can focus on analyzing your signals rather than on defining them.

Resources

Product Details, Demos, and System Requirements
www.mathworks.com/products/vehicle-network

Trial Software
www.mathworks.com/trialrequest

Sales
www.mathworks.com/contactsales

Technical Support
www.mathworks.com/support

Online User Community
www.mathworks.com/matlabcentral

Training Services
www.mathworks.com/training

Third-Party Products and Services
www.mathworks.com/connections

Worldwide Contacts
www.mathworks.com/contact