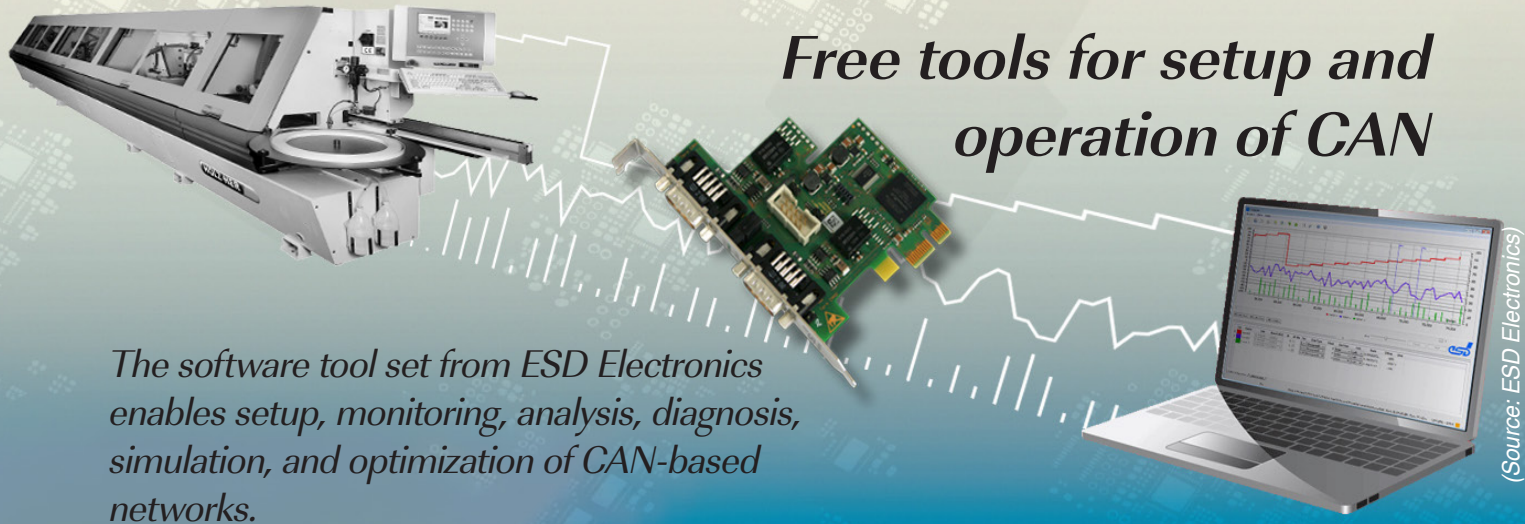
Free tools for setup and operation of CAN



(Source: ESD Electronics)

The software tool set from ESD Electronics enables setup, monitoring, analysis, diagnosis, simulation, and optimization of CAN-based networks.

Since the 1980s, CAN has simplified transmission paths to and from analog and digital devices. Today, CAN networks can be found in many industries: from automotive applications to automation technology, medical engineering, and aircraft technology.

Since the early days of CAN, ESD Electronics has concentrated on the development of CAN-connectable components and devices. The product portfolio ranges from CAN interfaces, gateways and bridges, I/O modules, plug-in card systems as well as CPU (central processing unit) boards. To support users while setup and operation of CAN networks based on these components, the company from Hanover (Germany) offers free software tools.

Software tools included

The CAN Software Development Kit (CAN SDK) for the NTCAN API (application programming interface) includes the CAN diagnostic tools CANreal, CANplot, CANrepro, CANscript, and COBview. The system requirement for using these tools is a current Windows operating system as 32-bit or 64-bit version. In addition to the five CAN tools, the CAN SDK includes header files, libraries, sample applications, and documentation. Another tool is the esdACC Error Injection GUI (graphical user interface) tool, which can be used to simulate CAN errors.

The CAN SDK allows developing, debugging, and testing of applications based on CAN hardware. All tools as well as the programming API share the multi-process NTCAN architecture. It supports CAN FD and time-stamped receiving and transmitting of frames including CAN inter-process communication. A virtual CAN driver for developing and testing applications completes the tool-box.

The libraries and samples included in the CAN SDK are available for many programming languages and environments. These include C/C++ (Visual, Borland, MinGW), Visual Basic 6, Delphi, Purebasic, and Python. In addition, the CAN SDK also includes NTCAN.NET class libraries for the Microsoft.NET framework for implementation of applications in C# or VB.NET. In addition, the kit offers function blocks for API functions as well as the function blocks of the CANopen Tiny Manager for use of CAN and CANopen in Labview.

Third-party software can be used directly with the CAN hardware from ESD through suitable libraries. For example, ESD offers a DLL (dynamic link library) for the CANopen Conformance Test (CCT) from CAN in Automation (CiA) and a corresponding version for the DeviceNet Protocol Conformance Test, a software of the Open DeviceNet Vendors Association (ODVA).

Monitoring and testing of CAN networks

The software tool CANreal is a monitor program for monitoring and analysis. It is also used as a test environment for CAN networks. Thanks to its open plugin interface, both supplied and self-written plugins can be used, such as those for CAN data bases (DBC) or J1939.

With its range of configuration options, the program is also versatile in diagnostics. For example, it is possible to set CAN-Identifier filters for 11-bit and 29-bit CAN-IDs, to log CAN frames and to display parameters decoded with DBC files. Moreover, high-resolution time stamps can be evaluated and CAN error detection as well as a variety of trigger functions can be used. The frames are displayed either as an online list or statically (object mode). The user can configure the columns. The time-stamps of CAN frames can be displayed as absolute values, with or without a date accurate to the microsecond, depending on the settings. An additional column displays the relative value with respect to the previous CAN frame. For CAN hardware supporting IRIG-B time code format, the transmitted time-code can also be set and put out as a time base.

Furthermore, CAN statistics with a calculation of bus loads and with transmit maps for user-defined CAN frames are available for diagnostics. The logging function offers the possibility to split large data sets into several files, or to overwrite them cyclically for data reduction. Files with recorded frames can be converted to CSV files or reloaded into CANreal (offline list) and into such tools as CANplot or CANrepro. The statistics function provides detailed information about the CAN network, the number of CAN data frames, error frames and more.

The single fault diagnosis shows detailed information about faulty CAN frames. A search function with bookmarks allows searching for individual CAN data frames or error frames. An advanced search can be defined individually. Since the tool is based on the ESD CAN driver with the ▷

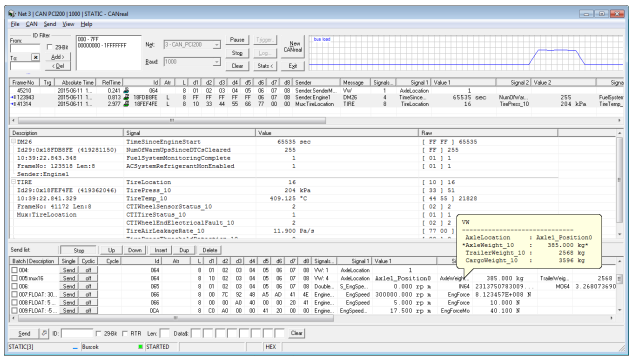


Figure 1: Monitoring program CANreal for control and analysis purposes (Source: ESD Electronics)

multi-process NTCAN architecture, several instances of the CANreal tool can be opened simultaneously to a CAN network. Alternatively, the CAN network used by an application can be directly observed.

Displaying CAN data graphically

Graphics of data not only show values at a glance but also ratios and proportions. CANplot prepares CAN data and displays it graphically on two scalable coordinate axes (online and offline). Thanks to the individual message recognition, criteria such as position in the data field and the data type can be selected and displayed in individual data graphs with color assignment. CAN data can be selected according to the network number, CAN-Identifier, and position in the data field. Several numeric data types are pre-defined for data interpretation. Data formats from Intel and Motorola (big endian/small endian) are supported.

Reproducing CAN frames

For analyzing CAN communication, the CAN frames previously recorded via CANreal can be repeated with CANrepro. This function allows incoming diagnoses or automated test procedures. The original time sequences remain at the reproduced data and the individual CAN-IDs can be selected. Using CANrepro, a realistic simulation of CAN devices is possible and the recorded CAN messages can be reproduced again.

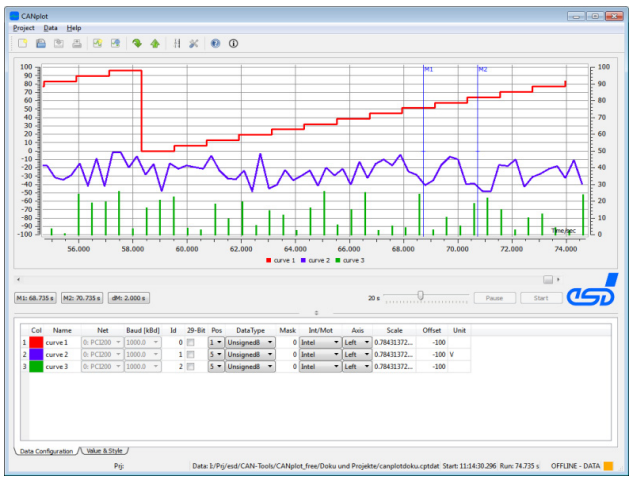


Figure 2: CANplot prepares CAN data and displays them graphically (Source: ESD Electronics)

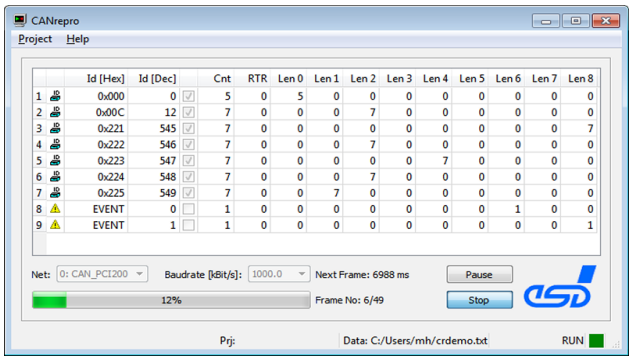


Figure 3: CANrepro for analyzing CAN communication with reproduced messages (Source: ESD Electronics)

Creating Python programs

If executable Python programs (Python scripts) are required, CANscript provides a corresponding GUI frontend allowing execution of PyNTCAN-based scripts for test automation, residual bus simulation, and other applications.

Setting up CANopen nodes

The COBview tool provides the user with a CANopen object overview and enables the modification of device parameters and of the network states for testing purposes as well as for setup of CANopen nodes. In addition, the program helps with the analysis and diagnosis of CANopen nodes and with the search and display of CANopen devices in a CANopen network. The tool offers basic CANopen network management (NMT) functionality (start node, pre-operational, reset, stop) as well as read/write access to the object dictionary. It lists CANopen objects with all sub-indexes and interprets the object data. The objects are read by index and displayed in a list with all sub-indexes. Data interpretation is generic in multiple formats without the need to load EDS (electronic device sheet) files. The network scan function lists all devices on the network.

Simulating CAN errors

The esdACC Error Injection GUI tool provides a free graphical user interface designed for the error injection ▷

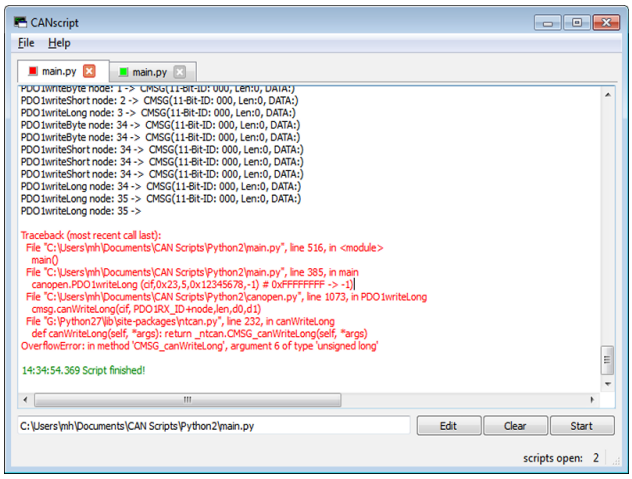


Figure 4: CANscript for test automation of PyNTCAN-based scripts (Source: ESD Electronics)

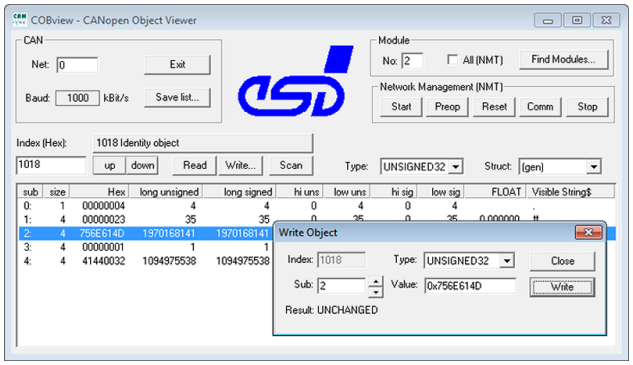
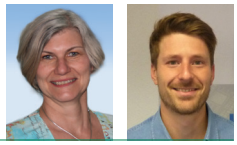


Figure 5: CANview provides CANopen object overview and enables modification of CANopen device parameters (Source: ESD Electronics)

unit integrated into some of manufacturer's CAN interfaces. Conventional CAN controllers available on the market at present, are not able to send faulty CAN frames due to their design. However, the esdACC CAN IP core, supplemented by the error injection unit, can generate or simulate numerous CAN errors. In addition to the GUI tool, error injection can also be configured and used directly via API calls using the NTCAN API. In this way, automated test cases can be realized in complex test scenarios.

Summary

The software support provided in form of different ESD tools enables setup and configuration of CAN networks. Additionally, the tools offer a lot of possibilities for analysis, diagnosis, and optimization of CAN-based communication. Simulations for testing purposes and monitoring, for example in troubleshooting, are given as well.



Authors

Renate Klebe-Klingemann
Hans Kürsten
ESD Electronics
info@esd.eu
www.esd.eu