

# Open-source CANopen protocol stack extended

*CANopennode is a free and open-source CANopen protocol stack available on Github. Recently, it was extended by a CANopen stack example running on STM32 micro-controllers.*



(Source: Adobe Stock)

**C**ANopennode is written in Ansi-C in an object-oriented way. It runs on different micro-controllers, as standalone application, or with RTOS (real-time operating system). Variables for CANopen network communication, for CANopen device control, or customer-defined functions are collected in the CANopen object dictionary. These variables are accessible from the C-code and from the CANopen network. The object dictionary editor tool is available. Non-volatile storage for object dictionary or other variables is possible. The software stack is suitable for 16-bit micro-controllers and above. The software is multi-threaded and real-time capable. Power saving and bootloader functions are enabled as well. Currently, the version 4 of the protocol stack is available.

## Available features

The provided CANopen object dictionary offers a clear and flexible organization of any variables. Variables can be accessed directly or via read/write functions. The NMT (network management) functionality enables to start, stop, and reset a CANopen device using the simple NMT manager functionality. To monitor the device's availability in the network, the heartbeat producer and consumer error control is included. PDO (process data object) function block can be used for broadcasting process data with high priority and no protocol overhead. Variables from

the object dictionary can be dynamically mapped into the TPDO (transmit PDO), which is then transmitted according to communication rules as specified in CiA 301 CANopen application layer and communication profile (EN 50325-4). This data is received as RPDO (receive PDO) by another device(s).

The SDO (service data object) server functionality, which has to be obligatory implemented in a CANopen device, enables expedited, segmented, and block transfer access to all object dictionary variables inside of this device. An SDO client can access any object dictionary variable on any CANopen device inside the CANopen network. A CANopen device can implement the emergency message producer function. Devices, which have to be informed on emerging failures, implement the emergency consumer function. The provided Sync producer/consumer enables network synchronized transmission, receipt, and processing of exchanged process data, etc. The time-stamp producer/consumer enables date and time synchronization of the networked devices.

In addition to the basic CANopen functionality as covered by CiA 301, the LSS (layer setting service) implementation is given to setup the device's node-ID and bit rate. The LSS server and LSS manager functions as well as the LSS fastscan service are part of the open-source protocol stack. To access CANopen via TCP, a CANopen gateway implementation according to CiA >

309-3 is given. This provides an Ascii command interface for NMT manager, LSS manager, and SDO client. For communication in safety-relevant networks, the CANopen Safety module was developed. The functional safety communication based on CANopen is specified in EN 50325-5 (former CiA 304).

The community providing the protocol stack claims that the implementation passed the CANopen conformance test using the [CANopen conformance test tool](#). The tool is used to verify that CANopen devices are compliant with the CiA 301.

## Protocol stack project modules

The CANopen protocol stack is the base for a CANopen device. It contains no device-specific code (drivers), which must be added separately for each target system. An example (CANopendemo) shows the basic principles, compiles on any system, but does not connect to any CAN hardware. CANopendemo includes a demo program, tutorial, and testing tools. It is based on CANopennode and is included as a git sub-module.

CANopeneditor is an external GUI (graphical user interface) tool for editing the CANopen object dictionary for a custom device. It generates the CANopennode C source-code files, electronic data sheet (EDS), and documentation for the device. The tool imports and exports the CANopen electronic data sheets in EDS or XDD (XML device description) format.

Complete generated documentation for CANopendemo, CANopennode, and other devices, is available online at <https://canopennode.github.io>. All codes are documented in the source header files. Some additional documents are provided in the doc(ument) directory. A tutorial and the possibility to [report issues](#) is given. The [older discussions](#) can be consulted on Sourceforge. The community welcomes contributions of further implementation examples.

## Device support

CANopenNode can run with or without operating system on many different devices (or micro-controllers). It is necessary to implement an own interface to the CANopennode on a specific hardware. Implementations with different development tools are possible. It is not practical to manage all device interfaces in a single project. Thus, interfaces to different micro-controllers are located in separate projects. There are interfaces to Linux SocketCAN, Zephyr RTOS, PIC, Mbed-os RTOS + STM32, NXP, etc. The [known device interfaces](#) are given in a list on the project's website. Most up-to-date implementations of CANopennode are CANopenlinux and CANopenPIC for PIC32 micro-controllers from Microchip.

## Extended by examples for STM32

Recently, the open-source CANopennode library was extended by [CANopenSTM32](#). The latter is a CANopen stack example running on STM32 micro-controllers from ▶

# Rugged CAN Solutions For The Real World

Engineered to meet the expectations of heavy industries, the Kvaser U100 is as well suited to CAN system design as it is to CAN system maintenance and repair.

Learn more: [www.kvaser.com/u100](http://www.kvaser.com/u100)



## STM32 and CANopen protocol stack provider

In July 2020, Emotas' CANopen protocol stack software has been endorsed by ST to become the first MadeForSTM32-approved CANopen software for STM32 micro-controllers. The MadeForSTM32 label is available for ST authorized partners only. It ensures that the software within the STM32 ecosystem had been reviewed and qualified by ST specialists. The protocol stack vendors Microcontrol, Port, and Simma are also taking part in ST's partner program.



*The MadeForSTM32 label is available for ST authorized partners only and ensures that the software had been reviewed and qualified by ST specialists (Source: ST Microelectronics)*

The CANopen software expansion includes the Emotas' CANopen stack basic functionality wrapped into an STM32Cube expansion package. The expansion simplifies the seamless integration into STM32Cube projects. The delivery includes the stack's Ansi-C source code and various ready-to-run examples. Besides a free evaluation package, commercial licenses of the product are available. The product is completed by the CANopen Devicedesigner tool that generates the object dictionary and device descriptions files (EDS). The CANopen expansion is currently available for the STM32G4 series. Support for additional STM32 series is planned.

Three months before, in April 2020, Port extended its CANopen driver portfolio by support of the STM32G4xx family. It can be used together with the company's CANopen protocol stacks covering the CANopen NMT (network management) manager and/ or CANopen NMT server functionality. For integration of the CANopen

library, the Industrial Communication Creator tool is available. It is suited for development of CANopen applications and programming of the CANopen devices. The tool also generates an object dictionary and an initialization function code in Ansi-C. An electronic data sheet and the documentation of the project are created as well. The tool is also used to configure the CANopen library and the CANopen driver packages.

Already in 2018, Emotas joined forces and partnered strategically with HMS Industrial Networks/Ixxat. The cooperation covers protocol stacks for CANopen and SAE J1939, tools, and CAN interface products. The partnership also includes the distribution of the CAN hardware products from HMS in combination with Emotas software tools. This means, at that point HMS/Ixxat has discontinued its own protocol stack developments.

The remaining German off-the-shelf suppliers include Embedded Office, EmSA (formerly Embedded Systems Academy), ESD Electronics, Micrium, Micro-control, Port, and SYS TEC. There are also some CANopen stack suppliers in other countries such as I.s.i.t. (France) and Simma Software (USA). Additionally, open-source projects (for example CANfestival and CANopennode) develop CANopen protocol software. Already a couple of years ago, Vector has had discontinued its CANopen stack business.



*The STM32G4 series is based on a 32-bit Arm Cortex-M4 core (Source: ST Microelectronics)*

ST Microelectronics (ST). It is developed in STM32cube IDE tool, which is the official ST development studio for any STM32 micro-controller. The user can directly open projects and run prepared examples on the board.



*Figure 1: STM32cube IDE tool is the official ST development studio for any STM32 micro-controller (Source: ST Microelectronics)*

The currently used development board is the STM32H735G-DK. It provides various features of the STM32H7xx series and includes three on-board CAN transceivers. Thus, no additional hardware to connect to existing CAN networks is required. It also includes a built-in programmer and a virtual COM (communication) port for evaluation purposes.

The CANopen demo works at the FDCAN1 port and is used for communication at 125 kbit/s. The FDCAN IP block is the same for any STM32H7xx MCU (micro-controller unit) family, hence migration to a custom board should be

straight-forward. CANopen LED control according to CiA 303-3 is integrated. Debug messages are available via the COM port. The examples can be used as a reference code for the end product. The existing projects can be cloned or updated. ◀

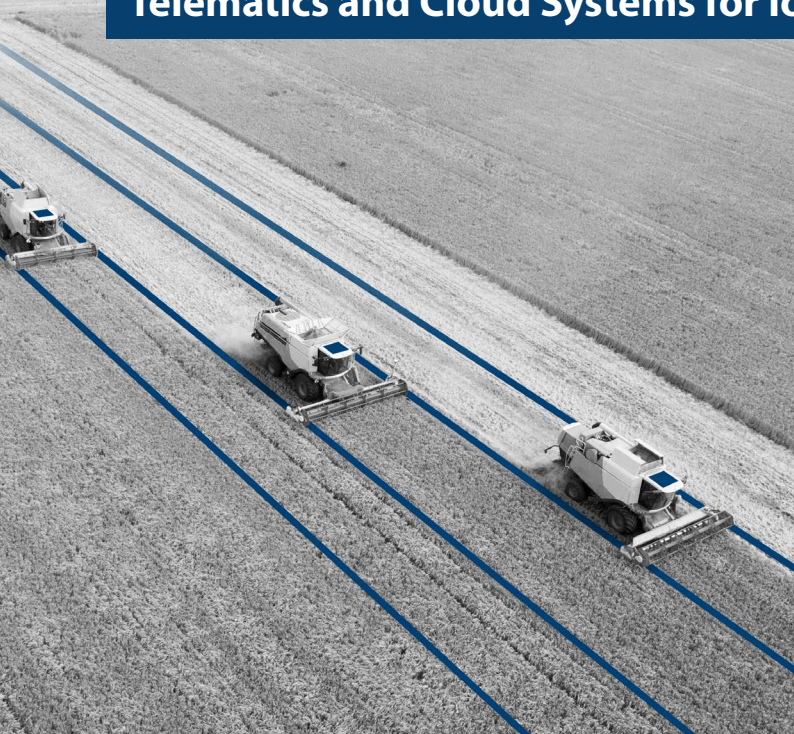
### Author

Olga Fischer  
CAN Newsletter  
[pr@can-cia.org](mailto:pr@can-cia.org)  
[www.can-newsletter.org](http://www.can-newsletter.org)



**High-end Connectivity and Data Management**

**Telematics and Cloud Systems for IoT and Service 4.0**



**Continuous digitization for smart vehicles**

Modular on-board units – from cost-saving entry telematics up to high-end modules. Including updates-over-the-air, embedded diagnostic functionality and up to 4x CAN channels (CAN FD ready).

Sontheim IoT Device Manager and IoT Analytics Manager – for a highly secure, comfortable and individual visualization as well as management of your data and fleet.

**COMhawk® xt – Telematic ECU Series**



Up to 4x CAN acc. to ISO 11898



LTE, WLAN, Bluetooth, LAN



Multi-protocol support (J1939, J2534, UDS, KWP, ...)



Positioning (GNSS)



Embedded diagnostic functionality



Integrated update-over-the-air functionality

