September 2024

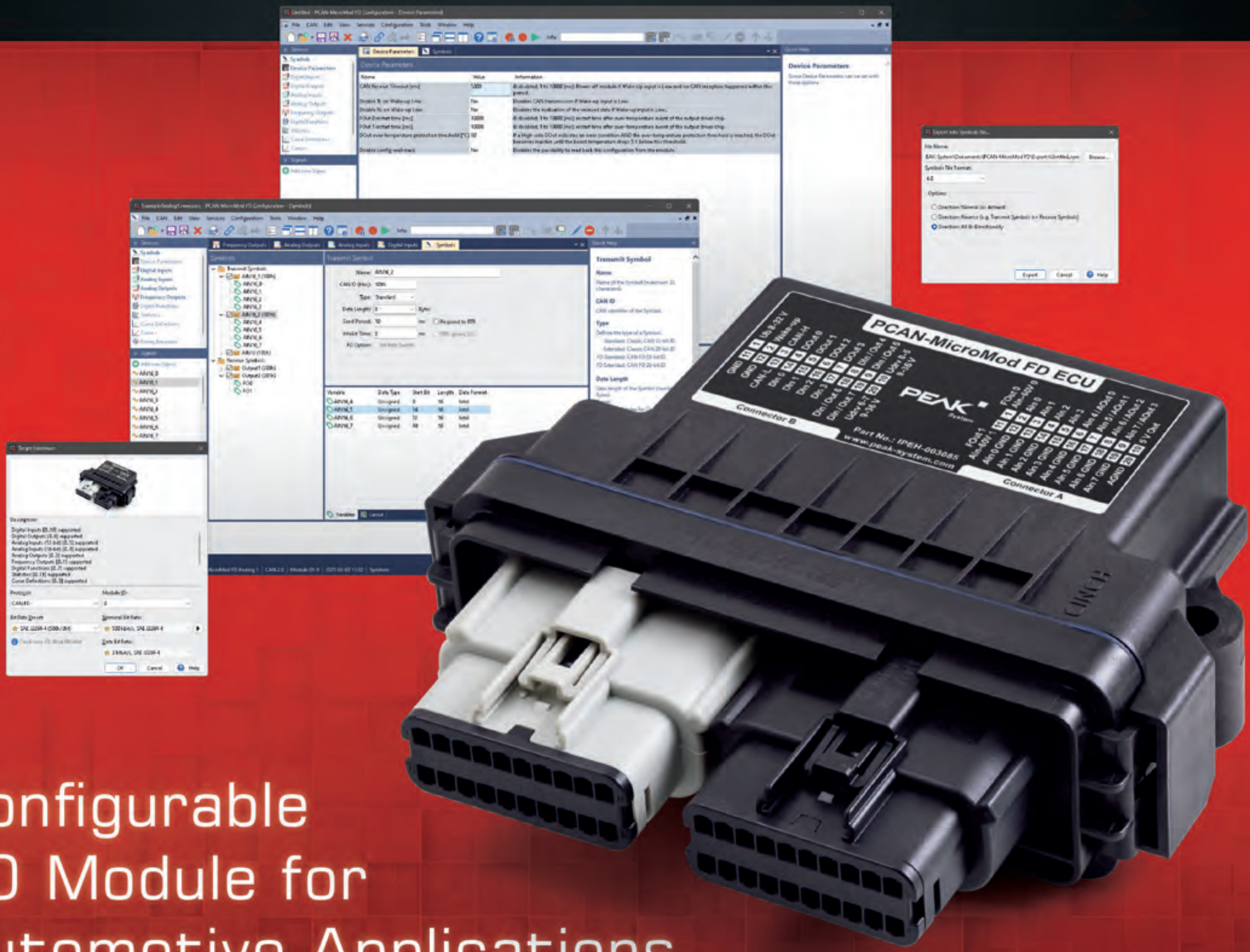
# CAN Newsletter

Hardware + Software + Tools + Engineering

Cover story

## *CANopen pressure sensors*

Pressure is one of the most frequently measured values. Pressure sensors are used in diesel engines powering vessels, heavy-duty vehicles, and many other industries. Compared to sensors with analog outputs, pressure transmitters with CANopen connectivity offer different advantages. The cover story provides an overview about such devices.




# Configurable I/O Module for Automotive Applications

## ■ PCAN-MicroMod FD ECU

With CAN FD, a mix of digital and analog I/Os, and its tough case, the PCAN-MicroMod FD ECU can be your solution for integrating custom accessories in utility and heavy duty vehicles operating under harsh conditions.

The PCAN-MicroMod FD ECU can be configured with a Windows software via CAN. Besides simply mapping its I/Os to CAN messages, several function blocks for processing the data are available as well.

**Specifications**

- High-speed CAN connection (ISO 11898-2)
  - Complies with CAN specifications 2.0 A/B and FD
  - CAN FD bit rates for the data field (64 bytes max.) from 40 kbit/s up to 10 Mbit/s
  - CAN bit rates from 40 kbit/s up to 1 Mbit/s
- Wake-up by CAN bus or by separate input
- 4 digital inputs
  - Pull-up or pull-down configurable
- 8 digital outputs with High-side switches
  - 2 outputs with 5 A and 6 outputs with 2 A
  - 4 alternatively usable as a digital input or additionally for reading back the output level
- 8 analog inputs
  - Resolution 16 bit
  - Measuring range adjustable: ±2.5 V, ±5 V, ±10 V, ±20 V
- 4 of the analog inputs alternatively usable as analog output
  - Resolution 12 bit
  - Voltage range adjustable: 0 to 5 V or 0 to 10 V
- 2 frequency outputs
  - Low-side switches (3 A)
  - Adjustable frequency range from 0 to 20 kHz
  - Alternatively usable as analog inputs with voltage range from 0 to 60 V
- Connections for CAN, I/O, and power supply via two 20-pole automotive connectors (Molex MX150)
- Plastic casing with increased Ingress Protection IP67 and flange
- Operating voltage 8 to 32 V; suitable for use in 12 and 24 V vehicle electrical systems
- Extended operating temperature range from -40 to +85 °C (-40 to +185 °F)
- E1 type approval

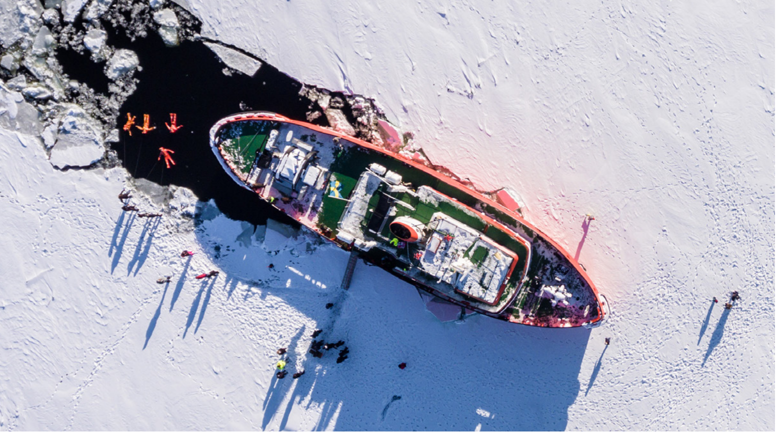
www.peak-system.com

Take a look at our website for the international sales partners. Scan the QR code on the left to open that page.

PEAK-System Technik GmbH

Phone: +49 6151 8173-20
Fax: +49 6151 8173-29
E-mail: info@peak-system.com

PEAK System



# Cover story

CANopen pressure sensors 6

Pressure is one of the most frequently measured values. Pressure sensors are used in diesel engines powering vessels, heavy-duty vehicles, and many other industries. Compared to sensors with analog outputs, pressure transmitters with CANopen connectivity offer different advantages. This article provides an overview about such devices.

# Devices

PLd-compliant inclinometer 10
HMS Networks acquired Red Lion Controls 12
CANopen sensor systems for urban gardening 15
Redundant train control and monitoring system (TCMS) 18

# Engineering

Supporting IEC 61131-3 for 30 years 22
Harmonizing CANopen bootloaders 24
Real-time communication: Part IV - CANopen source code configuration 26
Physical layer options in CAN XL networks 30
Visualization of terabytes of CAN data at low cost 41

# Semiconductors

64-bit microprocessor for the outer space 34
Evolution of (D)CAN 36
Single-chip powertrain controller supports CAN XL 40

# Brief news

Standards and specifications 4

# Imprint

**Publishing house**
CAN in Automation GmbH
Kontumazgarten 3
DE-90429 Nuremberg
marketing@can-cia.org
www.can-cia.org
Tel.: +49-911-928819-0
Fax: +49-911-928819-79

Reiner Zitzmann (CEO)
VAT-ID: DE812852184
HRB: AG Nürnberg 24338

**Publisher**
CAN in Automation e. V.
Kontumazgarten 3
DE-90429 Nuremberg
VAT-ID: DE169332292
VR: AG Nürnberg 200497

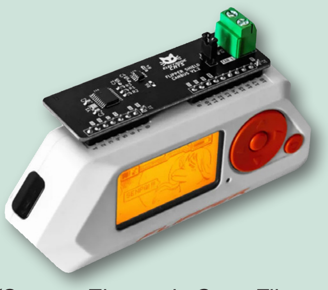
**Editors**
Olga Fischer (of)
Holger Zeltwanger (hz)
(responsible according to the press law)
pr@can-cia.org

**Layout**
Nickel Plankermann

**Media consultants**
Julia Dallhammer
Tobias Kammerer
Birgit Ruedel (responsible according to the press law)
marketing@can-cia.org



## CAN add-on board for a contactless USB dongle



(Source: Electronic Cats, Flipper Devices)

Electronic Cats, a Mexican company, has launched a Flipper Zero add-on module, which features an MCP2515 stand-alone CAN CC (classic) controller. Flipper Zero by Flipper Devices, a U.S. company, provides a contactless attachment to electrical cables. In combination with the CAN add-on modules, the Flipper Zero can read CAN frames. They are forwarded to the micro-controller in the Flipper Zero device by means of a 10-MHz SPI (serial peripheral interface) link. The CAN shield module implements a MAX3051 3,3-V transceiver by Analog Devices. This CAN HS (high-speed) transceiver supports bit rates up to 1 Mbit/s. The pluggable shield measures 67 mm x 21,3 mm. The supplier provides a tutorial for installing and using the MCP2515 CAN app. Additionally, the company made the hardware design files open-sourced, meaning they can be downloaded from a GitHub repository. *hz*

Table of contents

# Standards and specifications

*This section provides news from standardization bodies and nonprofit associations regarding CAN-related documents. Included are also recommended practices, application notes, implementation guidelines, and technical reports.*

## CAN transceiver terminology

The recently updated ISO 11898-2:2024 standard specifies CAN HS (high-speed) transceivers as well as CAN FD (flexible data rate) transceivers as in the ISO 11898-2:2016 standard. In general, there is no difference in these standards regarding these transceiver types. An exception is the definition for the dynamic symmetry parameters. Now these parameters are guaranteed for the complete bit-rate range, which is specified in the transceiver product datasheet. For example, a lot of so-called 5-Mbit/s CAN FD transceivers are used at 2 Mbit/s. The intention behind, that the so-called 5-Mbit/s transceiver behaves at 2 Mbit/s as specified at 5 Mbit/s, is not covered in the ISO 11898-2:2016 standard. This modification will be explained in the updated CiA 601-1 technical report about CAN FD node and system design recommendations (revision is in preparation and comes soon).

Additionally, ISO 11898-2:2024 specifies the CAN SIC (signal improvement capability) transceiver, originally described in the CiA 601-4 specification (document has been withdrawn after releasing the ISO standard), and CAN SIC XL (extended data-field length) transceivers, deriving from the CiA 610-3 specification (has been withdrawn, too).

In order to avoid confusions, the following terms should be used to typify CAN transceivers:

- CAN HS: legacy transceiver supporting bit rates up to 1 Mbit/s, with or without lower-power mode and/or selective wake-up capability.
- CAN FD: legacy transceiver supporting bit rates of 2 Mbit/s and more (e.g., when using a point-to-point topology), with or without lower-power mode and/or selective wake-up capability (only when qualified for 5 Mbit/s).
- CAN SIC: new transceiver providing signal improvement capability and supporting bit rates up to 8 Mbit/s, with or without lower-power mode and/or selective wake-up capability (only when qualified for 5 Mbit/s; ISO 11898-2:2024 limits the bit rate to 5 Mbit/s).
- CAN SIC XL: new transceiver providing signal improvement capability and the FAST mode with error signaling disabled (up to 20 Mbit/s), with or without lower-power mode (selective wake-up capability is not possible).

There are also other CAN transceiver types specified in other documents, such as CAN LS (low-speed) transceivers with fault-tolerant functionality and low-power capability supporting bit rates up to 125 kbit/s (ISO 11898-3). There is an additional fault-tolerant transceiver standardized in ISO 11992-1 supporting a bit rate of 125 kbit/s for point-to-point links; it is used for connecting towing and towed heavy-duty road vehicles. The SWC (single-wire CAN) transceiver specified in the SAE J2411 document supports a nominal bit rate of 33,3 kbit/s. It is not recommended for new designs.

NOTE: Unfortunately, the published ISO 11898-2:2024 standard contains some misleading and partly wrong information. The CiA 140 document provides a corrigendum proposal; it can be requested free-of-charge from CiA office. By the way, the next edition of ISO 11898-2 (overcoming these issues) is already in preparation. *hz*

## New CAN protocol terms

The ISO 11898-1:2024 standard uses for the three CAN data link layer (DLL) and physical coding sublayer (PCS) approaches the following terms:

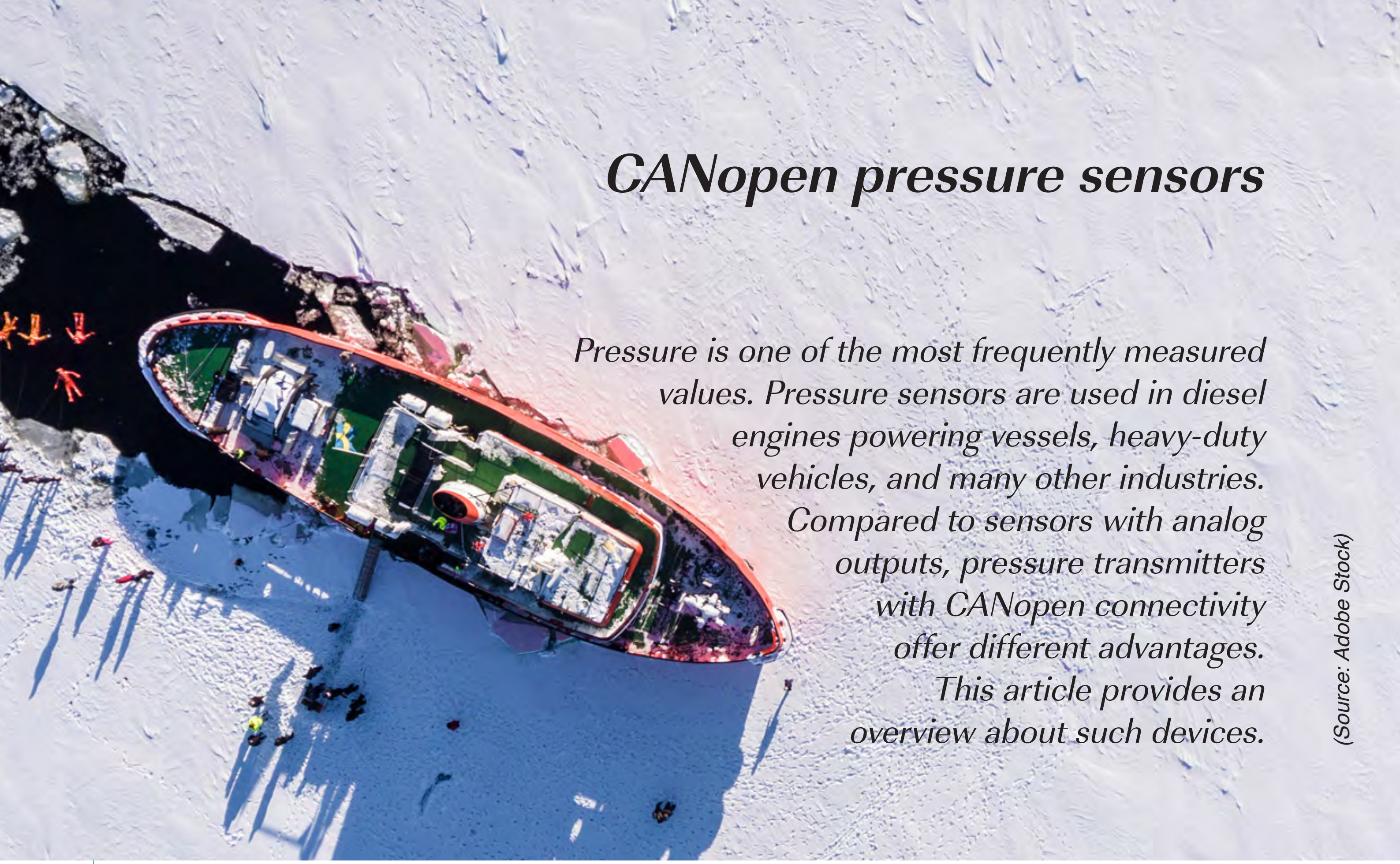
- CAN CC (classic);
- CAN FD (flexible data rate);
- CAN XL (extended data-field length).

The term CAN should be used, when you like to talk about all three CAN protocol generations. Similarly, the term CANopen indicates both higher-layer protocols: CANopen CC based on CAN CC (specified in CiA 301) as well as CANopen FD based on CAN FD (specified in CiA 1301). In SAE J1939, there are two specification names applied to distinguish between the legacy CAN CC based (SAE J1939-21) and the new CAN FD based (SAE J1939-22) application layer approaches. *hz*

## Brief news

- ***ISO 25200:*** The submission of DIN 4630 (CAN-based commercial vehicle body application network) for international standardization has been accepted. It will be standardized in ISO 25200. This work item has been assigned to ISO/TC 22/SC 31/WG 4.
- ***ISO 11992 series:*** The ISO 11992-2 (parameters and parameter groups for the CAN-based link between towing and towed commercial vehicles) document is under revision. To involve as many stakeholders as possible, an ISO 11992 workshop is scheduled in Berlin (Germany) on September 16, 2024.
- ***CiA 611-1 and CiA 611-2:*** An updated version of CiA 611-1 (SDT specification for CAN XL) has been released as well as a first version of CiA 611-2 (Multi-PDU specification for CAN XL).
- ***CiA 406 series:*** The profile for encoders has been revised. The version 2.0.0 of Part B specifies the functional behavior and parameters including those for functional safety. Part C specifies the mapping to CANopen CC and CANopen Safety (EN 50325-5). Part F describes the mapping to CANopen FD and Part J to the J1939 higher-layer protocols.
- ***SAE J1939DA:*** The digital annex of the J1939 specification series is released quarterly. It contains among other items (suspect) parameter and Parameter Group (PG) specifications. The latest version has been published in June.
- ***FMS specification:*** ACEA, the European road-vehicle manufacturer association, has released version 5 of its fleet management specification (FMS) for commercial vehicles. It is based on J1939 and references parameters and Parameter Groups (PGs) specified in SAE J1939DA.

*hz*



(Source: Adobe Stock)

# CANopen pressure sensors

*Pressure is one of the most frequently measured values. Pressure sensors are used in diesel engines powering vessels, heavy-duty vehicles, and many other industries. Compared to sensors with analog outputs, pressure transmitters with CANopen connectivity offer different advantages. This article provides an overview about such devices.*

According to a report from MarketsandMarkets, the global pressure sensor market is expected to have a CAGR (compound annual growth rate) of 6,5 % from 2024 to 2029. Piezoresistive pressure sensors are poised to dominate the market. Regarding connectivity, the wired pressure sensors hold the largest share. These are directly connected to PLCs (programmable logic controllers), monitoring systems, or data acquisition devices allowing uninterrupted data exchange as well as real-time monitoring and response.

## Pressure measuring devices

Pressure is defined as the exertion of force per unit area. Pressure sensing solutions measure the pressure of gases or liquids. Such devices include pressure gauges, pressure sensors, pressure transducers, as well as pressure transmitters. Pressure gauges are simple mechanical devices that are read visually. They do not require electrical power, and do not provide a feedback signal.

Pressure sensors operate by sensing pressure and converting it into an electrical quantity. Piezoresistive and capacitive are the two most commonly seen types in industrial applications with piezoresistive being the most used. A piezoresistive pressure sensor consists of a silicon diaphragm with a piezoresistive strain gauge diffused into it. The electrical resistance of piezoresistive material changes when it is strained or compressed. As pressure increases, the piezoresistive material is more resistant to the electrical current passing through it. This results in the output voltage, being directly proportional to the pressure. Capacitive pressure sensors use a thin diaphragm (e.g. from metal) that serves as a capacitor plate. The diaphragm is exposed to a reference pressure on one side and to the process pressure on the other. Pressure changes cause a deformation of the sensing plate causing changes in the electrical capacitance. The latter are directly proportional to the applied process pressure. Thus, such pressure sensors detect pressure variations by sensing changes in the resistance or capacitance.

Pressure transducers consist of a pressure sensor mounted to an industrial process connection. The transducer converts pressure into an analog electrical signal, typically measured in millivolts. These signals are relatively weak and require amplification to carry these values over a distance. The signals are also not linearized or temperature compensated.

Pressure transmitters include the elements of the transducers and have additional circuitry that linearizes, compensates, and amplifies the signal from a transducer. Thus, such devices can transmit the signal over large distances to a remote receiver and provide calibration features to improve the accuracy and measuring range of the device. Typical outputs of pressure transmitters are analog signals (e.g. 0 $V_{DC}$ to 10 $V_{DC}$ or 4 mA to 20 mA) or digital interfaces such as, for example, CANopen.

In principle, there are absolute pressure transmitters, gauge pressure transmitters, and differential pressure transmitters. The absolute pressure transmitters measure the absolute pressure referenced to a full vacuum. Fluid pressure is compared against the reference pressure of an absolute vacuum in a sealed reference chamber. Thus, the sensor measures pressures that are not influenced by ▷

atmospheric pressures. Such transmitters are often used for high-accuracy low-pressure measurements. Gauge pressure transmitters are referenced to atmospheric conditions and are widely used in the process industries to measure the pressures of liquids, gas, and steam. They can measure low and high pressures. Differential pressure transmitters measure the difference between two pressures and are used for a long time to measure the flow rate. Today, absolute and gauge pressure transmitters are replacing pressure gauges, switches, and transducers as they are more stable and reliable and can transmit the pressure values to control devices.

## CiA 404 CANopen device profile for measuring devices

The CiA 404 device profile is dedicated to analog transducers (e.g. pressure transducers) that may also provide digital I/O capability. The specified analog input function distinguishes between a field value and a process value. Field values are non-scaled readings from the analog/digital converter. This value is converted to the physical dimension or is given in percent of the measuring range, and the result is called process value. An example for a process value is a value in Pa or bar for pressure measurement. The process values are given with selectable resolutions e.g. 8-bit, 16-bit, and 32-bit integer, or Real32. Thus, the pressure values are output on the CANopen interface with the offset, resolution, and physical unit required by the respective process application. The electronics implementing the CiA 404 CANopen profile in pressure measuring devices enables them to provide pressure values in a standardized way. This allows for a simplified integration into CANopen networks and enables the use of off-the-shelf CANopen tools e.g. for device setup and diagnostics.

## CANopen-capable pressure transmitters

Following, an (incomplete) overview on pressure transmitters with CANopen connectivity is given. Most of them support the CiA 404 CANopen profile or/and the CAN-based J1939 higher-layer protocol. The transmitters with CANopen/J1939 interfaces reduce installation costs due to the simplified wiring compared with analog sensors. Via the CAN interface users can get access to additional device data e.g. for diagnostics and maintenance purposes. The digitally available data can be made available worldwide using IoT (Internet of Things) applications. Calibration and control of the sensors is also possible via CAN.

The KD41 gauge pressure transmitter from Ashcroft (U.S.A.) offers CANopen and J1939 interfaces. The stainless-steel device with stem mounting and ingress protections of IP65, IP67, or IP69K is dedicated for pressure measurements from 1 bar to 1400 bar. Markets and applications include machine automation, hydraulics, pneumatics, off-road equipment, pumps and compressors, etc. Bit rates up to 1 Mbit/s (with 125 kbit/s as default) are supported. The user can choose between variants with different start-up behavior (automatic and non-automatic), bus termination, galvanic isolation, and EMC requirements.

The pressure transmitter enables conversion of field values (calibrated electrical value at input terminal of the transducer) to process values (with a real physical dimension) according to the principles as specified in the CiA 404. This process is also called linear transformation or linearization. The device is working within the temperature limits from -40 °C to +105 °C.

On request, the long-term CiA member Baumer (Germany) provides the PBM4 gauge pressure transmitter with a CANopen interface. It enables relative pressure measurements from 0 bar to 2000 bar in industrial and mobile hydraulic applications. The IP67-rated device with stainless-steel housing is fully welded and provides the E1 approval for mobile hydraulics. The possible operation temperature ranges from -40 °C to +125 °C. The CANopen interface works at bit rates of up to 1 Mbit/s (125 kbit/s as default) and supports the CANopen specifications CiA 301 and CiA 404. To set the bit rate and the node-ID via CAN, the layer setting services according to CiA 305 are implemented. The device deploys the 5-pin M12 connector with pin assignment according to the CiA 106 document.

Buerkert (Germany), also a member from CiA, offers the 8312 pressure transmitter with CANopen. It is used for measuring either relative or absolute pressures (on request) in liquids or gases and is available with ceramic thick film or metallic thin film strain gauge measuring principles. Different linearized and temperature-compensated measuring ranges from -1 bar to 5 bar, 0 bar to 0,25 bar up to 0 bar to 16 bar are selectable. The possible media temperature range is from -20 °C to +85 °C (ceramic) and -40 °C to +125 °C (metallic). The IP67-protected transmitter uses the M12 socket connector with CiA-106 pin assignment. The device digitizes the pressure value and makes it available for use through the CANopen interface. Therefore, it supports the CiA 404 CANopen profile for measuring devices and closed-loop controllers. A driver used for data exchange and settings of the 8312 is integrated in the Buerkert PC tool Communicator and is available on company's website. LSS (see CiA 305) services are implemented as well.

The DST 10B OEM (original equipment manufacturer) gauge pressure transmitter is produced from Danfoss (Denmark). The manufacturer is CiA member with different subsidiaries. The device implementing the CiA 404 profile is dedicated for water distribution (e.g. water pumps) and air handling (e.g. industrial air compressors) applications. It is available in pressure ranges from 0 bar to 4 bar up to 0 bar to 40 bar. The device is able to work with media temperatures from -40 °C to +100 °C. The CAN interface is accessible at the 5-pin M12 connector integrated in the IP67-rated stainless-steel housing. The connector pin assignment accords to the CiA 106. Beginning of 2024, the DST 10B has passed the CANopen conformance test ▷


*Figure 1: The DST-P10B pressure transmitter (Source: Danfoss)*

Cover story

by CiA, which ensures the device compliance with the CANopen application layer and communication profile (CiA 301).

All industrial pressure sensor models from Dynisco (U.S.A.) are available in CANopen versions, states the manufacturer. For instance, the strain gauge pressure transmitter IDA3XCAN series for pressure ranges from 0 bar to 20 bar up to 0 bar to 2000 bar. The device's functions and communication software are implemented according to the CiA 404 CANopen profile. The digital pressure output value has a resolution of 12 bit. The bidirectional communication combined with internal functions such as transducer setup and calibration, pressure operating alarms, sensor watchdog control, etc. enable more versatile operating conditions and security functions than analog transmitters. The device's node-ID can be adjusted by hardware jumpers for simplified field replacement. The stainless-steel device is protected according to IP65 or IP67 and can be deployed at medium temperatures from -35 °C to +120 °C. It uses the 7-pin round socket connector.

KMC pressure transmitters from CiA-member Gefran (Italy) are based on a film sensing element deposited on a stainless-steel diaphragm. Due to the stainless-steel construction, resistance to harsh environmental conditions, as well as support of CANopen and J1939, these products are especially suited for mobile hydraulic applications. In addition to pressure measurement, measured data contains the temperature of the device. The devices measure fluid pressures from 4 bar to 1000 bar and are dedicated for operating temperatures from -40 °C to +125 °C. Ingress protection is IP67/IP69K with the socket homologated connector mounted. The device supports bit rates up to 1 Mbit/s with a default bit rate of 250 kbit/s. The CANopen variants implement the CiA 404 profile and CiA 305 specification for layer setting services (LSS).

HDA 4700 by Hydac (U.S.A.) with a CAN interface is a gauge pressure transmitter used to measure relative pressures in hydraulics and pneumatics. Depending on the version, measuring ranges from 0 bar to 40 bar up to 0 bar to 2000 bar are possible. The measured pressure value is digitized and made available via the CANopen or J1939 interface supporting bit rates up to 1 Mbit/s. The device's parameters can be viewed and configured using off-the-shelf CANopen software. The IP67-protected device works at temperatures from -25 °C to +85 °C with wider ranges on request. In addition to the CiA 404 profile for sensors, the layer setting services (CiA 305) and automatic bit-rate detection (CiA 801) are implemented in the CANopen variant. The company is also member of CAN in Automation.

Jumo (Germany) provides the CANtrans p transmitter, which acquires either the relative (-1 bar to 600 bar) or absolute pressures (0 bar to 25 bar) of liquids and gases. It digitizes the pressure value and makes it available for further use via CANopen. Due to the implemented CiA 404 profile, the pressure value can be scaled to any dimensional unit or in percent of the measuring range. Fine calibration features, an auto-zeroing function, and an adjustable shift of the characteristic are possible. Undesirable signal fluctuations can be suppressed through the (adjustable) filter constant. An emergency telegram is triggered in the event of a sensor fault. The bit rate and node-ID can be set via CAN using the LSS services. The used M12 plug connector provides pinning according to CiA 106. The stainless-steel unit withstands media temperatures from -40 °C to +125 °C and is typically used in such environments as wind power plants.


*Figure 2: 23SXc pressure transmitter (Source: Keller)*

Keller Druckmesstechnik (Switzerland) produces the 23SXc absolute and relative pressure transmitters for measuring ranges from 0 bar to 0,16 bar respectively from 0 bar to 1000 bar. The devices capturing pressures and temperatures support the CiA 404 profile. In the transmitters, temperature dependencies and non-linearity are compensated by means of a mathematical model in the micro-controller. Insulated and encapsulated piezoresistive pressure sensor is housed in a fully welded stainless-steel construction without internal seals. The transmitter deploys the 5-pin M12 connector with the CiA-106 pin assignment. The IP67/IP68-rated devices can work at media temperatures from -40 °C to +125 °C. The CANopen connectivity offers a simplified integration into a wide range of automation solutions, informs the CiA member. Typical applications include engine test benches, industrial applications, automation technology, and mobile hydraulics.

The CiA-member company Sensata Technologies (U.S.A.) provides the PTE7500 series of CANopen gauge pressure transmitters for measuring ranges from 0 bar to 16 bar up to 0 bar to 600 bar. The manufacturer uses a modified version of the CANopen stack created by Embedded Office. Beside the basic CiA 301 CANopen profile, the devices support the CiA 404 and CiA 305 (LSS) specifications. Bit rates up to 1 Mbit/s are possible. The IP69K-rated, stainless-steel design with a hermetic port is built to last in high vibration and shock environments. Operating media temperatures can vary from -40 °C to +125 °C. Applications include machine tools, injection molding equipment, industrial hydraulics and pneumatics, mobile hydraulics, off-highway vehicles, pumps, compressors, etc.

The M01-CAN pressure sensor series from Sensor-Technik Wiedemann (STW), a long-years CiA member, has been developed for use in commercial vehicles and mobile machines. Variants for absolute (0,1 bar to 7 bar) and relative pressure measurements (5 bar to 2000 bar) are provided. The respective media temperatures can lie at -40 °C to +85 °C and -40 °C to +150 °C. The sensors are available with CANopen, J1939, or STW-proprietary connectivity. The modular principle of the series permits a large number of combinations of pressure connection, pressure range, electrical output signal, and electrical connection. The M01 series supports PL b (performance level) according to ISO 13849 and is therefore suitable for machine safety applications. For electrical connection, the M12 connector (plastic or stainless steel) is used. The protection class of IP67 and IPX9K depends on the ▷

electrical connection. A software package for CAN setting parameters is available. The variant M01-CAN2 provides a state-of-the-art CAN protocol stack and support of the STW open-source software platform Opensyde. M01-CAN is no longer recommended for new designs.

The 0630 digital pressure transmitter series from Suco (Germany) is supporting the CiA 404 CANopen profile as well as J1939 (version 0631). Variants are available for pressures from 0 bar to 1 bar up to 0 bar to 600 bar. The robust completely welded stainless-steel sensor housing enables deployment of the device in harsh operating environments. Bit rates up to 1 Mbit/s (with 250kbit/s as default) are supported. The transmitter also supports the layer setting services (LSS) according to CiA 305. The possible media temperature may range between -40 °C to +125 °C. For CANopen and J1939 connection, the IP67-protected transmitter is using the M12 connector with CiA-106-compatible pinning.

*Figure 3: CMP 8271 pressure transmitter (Source: Trafag)*

Trafag (Switzerland) offers the CMP 8270 and CMP 8271 pressure transmitters. Both devices from the CiA-member company have passed the CANopen conformance test performed by CiA. Sensor variants are available for relative and absolute pressure measurements from 0,25 bar up to 700 bar (600 bar for CMP 8270). The transmitters are dedicated for heavy-duty applications as they are completely-welded, provide high shock and vibration resistance, and are designed for operating temperatures from -40 °C to +125 °C. The devices support bit rates up to 1 Mbit/s, the self-starting function, the CANopen device profile for measuring devices (CiA 404), as well as the layer setting services (LSS, CiA 305). The manufacturer also offers a CANopen configuration tool for the transmitter, which allows configuration of all CANopen parameters as well as the access to the complete CANopen object dictionary.


*Figure 4: MH-4-CAN OEM pressure transmitter (Source: Wika)*

The MH-4-CAN OEM pressure transmitter from Wika (Germany) supports CANopen and J1939 interfaces and is dedicated for use in mobile working machines. It enables working and control pressure measurement up to 600 bar in construction equipment, agricultural and forestry machines, mobile cranes and mobile elevating work platforms as well as material handling equipment and municipal vehicles. The IP67-rated transmitter implements the CiA 404, CiA 305, and CiA 801 (automatic bit-rate detection) specifications. It works at bit rates up to 1 Mbit/s (250 kbit/s as default) and temperatures between -40 °C and +125 °C. The deployed 5-pin M12 connector provides pin-assignment according to CiA 106. In 2023, the MH-4-CAN has passed the CANopen conformance test by CiA. In 2017, the CiA-member Wika has acquired the German measuring technology specialist Tecsis, which has formerly produced the CANopen-/J1939-capable gauge pressure transmitters 3327 and 3328 still available on the market.

Zila (Germany) offers the piezoresistive CANopen pressure transmitter DS-CAN-01 for relative measurements up to 4000 bar in hydraulic and pneumatic applications. The device with operating temperatures spanning from -10 °C to +80 °C supports the CiA 404 profile. For bit-rate and node-ID setting via CAN, it implements the layer setting services (LSS, CiA 305). For CANopen connection, the 5-pin M12 plug connector with CiA-106-compatibe pinning is used. The company also provides the CANopen-connectable DTS-CAN-01 pressure-temperature transmitter allowing additional temperature measurements from -10 °C to +80 °C. The device with pressure ranges from 2 bar to 4000 bar is medium-compatible with hydraulic oil, brake fluid, gasoline, diesel, compressed air, etc. ◀

**Autor**

Olga Fischer
CAN in Automation (CiA)
pr@can-cia.org
www.can-cia.org

# PLd-compliant inclinometer

*The Swiss company Baumer has introduced the GIM600R functional-safe tilt sensor featuring CANopen or J1939 connectivity. It has been developed for rough conditions in mobile machines and off-highway vehicles.*


(Source: Baumer)

Inclinometers are sensors for measuring angles of slope (or tilt), elevation, or depression of an object with respect to gravity. The launched GIM600R sensor complies with the ISO 13849-1:2023 safety standard, ensuring robustness and measurement accuracy. It is intended for applications in aerial work platforms, loader cranes, and telehandlers. The temperature-stable precision of the product simplifies machine design and allows for increased working range. The PLd-compliant (performance level d) sensor also enables reliable measurements for static stability control in construction vehicles.

The sensor improves the machine uptime providing a high environmental resistance. For the proof of resistance, the product went through numerous, demanding endurance tests. It meets the ingress protection requirements up to IP69K rating and is corrosion-resistant up to CX level. It is further immune to shocks, vibrations, dust, dirt, and salt spray.

The CANopen respectively the J1939 interface enables an easy integration into the control system. The supplier provides a system configuration suite to parameterize the sensor as known for other sensors, featuring an intuitive dashboard. OEMs (original equipment manufacturers), who have previously relied on GIM140R or GIM500R inclination sensors can simply replace them with the GIM600R.

The GIM600R sensor comes in different versions:

- 1-dimensional inclination sensor to be installed vertically;
- 2-dimensional inclination sensor to be installed vertically;
- 2-dimensional inclination sensor to be installed horizontally.

▷

## *Brief news: Sensors*

- ***Cable extension transducer:*** TSM (Top Sensors Manufacture) located in Italy offers draw-wire encoders measuring lengths of up to 5 m (CET5) and up to 12,5 m (CET12). The products comply with CANopen and feature CANopen Safety (EN 50325-5) connectivity.
- ***Draw-wire sensor:*** Gefran situated in Italy offers the GSH-A sensor device measuring lengths of up to 12,5 m. The sensor embedding an inclinometer provides a CANopen interface. The housing is IP67-rated.
- ***Encoder:*** The EN580C sensor by Baumer (Switzerland) comes with a CANopen Interface. The rotary absolute encoder with housing diameter of 58 mm features a 21-bit single-turn resolution and a 16-bit resolution in multi-turn mode.
- ***Cable extension transducer:*** The WST61 CANopen sensor by ASM (Germany) has a measurement range of up to 20 m. It integrates an inclination sensor for ±180° angles. The device is available in a redundant version to be used as functional-safe sensor. Other variants support shorter length measurements (6 m respectively 3 m).
- ***Inclinometer:*** STW (Sensor-Technik Wiedemann) (Germany) has developed the SMX.igs-a family of sensors. The redundant inclinometers can transmit the measured values via CANopen Safety (EN 50325-5), CANopen, or J1939. A version with only the CAN hardware is available, too.
- ***Safety encoder:*** ifm (Germany) offers the 58-mm sensor device, coming with CANopen Safety and CANopen CC (classic) ports. They comply with PLd (performance level d) and SIL 2 (safety integrity level 2). The certified absolute encoders are available with a 10-mm solid shaft or a 12-mm hollow shaft.

*hz*

These two-channel inclinometers provide two CANopen or J1939 interfaces available on two M12 connectors. The CANopen versions feature two node-IDs. They comply with the CiA 301 (version 4.2.0) CANopen base specification, CiA 305 (version 3.0.0) for layer setting services, and CiA 410 (version 2.0.0) CANopen profile for inclinometers, but not with EN 50325-5 (CANopen Safety). The redundant inclination measurements need to be cross-checked in the functional-safe host controller. Optionally, the 3-axis acceleration parameters are mappable into PDO (process data object) messages. Another option is an integrated termination resistor. The J1939 version maps the inclination values into two parameter groups with the number 65363, containing additionally the module-ID, the error-ID, and an 8-bit device temperature value. CAN in Automation (CiA) has also released a CiA 410 compatible profile specification for J1939 (CiA 410-J, version 1.1.0), which is not supported by the GIM600R sensor. *hz*

# HMS Networks acquired Red Lion Controls

*End of last year, the CiA member HMS Networks (Sweden) entered into a binding agreement to acquire Red Lion Controls (U.S.A.), also a CiA member. Both companies offer products with CAN-based interfaces.*


*Figure 1: "The companies are a great match both when it comes to products, geographic presence, and cultural aspects," stated Staffan Dahlstroem (Source: HMS)*

The acquisition of Red Lion Controls represents a key milestone in HMS's growth strategy, bringing together two innovators in the industrial automation sector. This collaboration leverages the complementary strengths of both companies, providing customers with a comprehensive suite of cutting-edge solutions designed to meet the evolving demands of the Industrial Internet of Things (IIoT) landscape.

"The combination of Red Lion's product portfolio, which is complementary to HMS' offering and has limited overlap, and the geographic match with Red Lion having a strong position in the attractive North American market, and HMS with its core markets in Europe, will generate good cross-selling opportunities," said Staffan Dahlstroem, CEO (chief executive officer) of HMS. The Swedish company sells generic CAN-connectable interfaces and tools under the Ixxat brand.

Red Lion Controls, known for its advanced automation and networking solutions, contributes its expertise to further enhance HMS's portfolio. Marcel van Helten, President of Red Lion Controls stated: "This is a great step for Red Lion. I would like to thank Spectris and Andrew Heath for their support in our growth over the past several years. I am looking forward to being able to work with the HMS team to make this a great success."

In the first glance, there will be no changes for the customers: Both companies produce and ship their products as before. In the future, the merger of the two companies will provide synergies, especially as it widens the American market for HMS and the European market for Red Lion. The greatest proportion of Red Lion's sales comes from North America, through its distributor network that HMS will be able to utilize to drive sales of HMS' gateway and remote access offering. With about 60 percent of its sales in the large automation markets in Europe and well-developed and targeted market channels, HMS will be able to cross-sell selected Red Lion's products in Europe.

HMS Networks offers several CAN, CANopen, and DeviceNet interface modules for its Anybus series. Additionally, the company provides the ET 200S CAN interface module for Simatic host controllers from Siemens. The Swedish company uses the Ixxat brand to promote a broad range of CAN-related products. Ixxat has been acquired in by HMS in 2013.

## J1939 and CAN sleds

Recently, Red Lion Controls has expanded its Flexedge modular automation device platform by means of J1939 and CAN sleds. Additionally, the company has introduced configurable strain-gauge modules. They come in both SSR (solid state relay) output and relay output options. Both offer single loop PID capabilities to monitor, to measure, and to control equipment. Designed to thrive in harsh environments, the modules accept signals from load cell, pressure, and torque bridge transducers. With a software-selectable 5-$V_{DC}$ or 10-$V_{DC}$ bridge excitation voltage, each strain-gauge module can drive up to four 350-W bridges. The inputs are also software selectable for low-level inputs at ±20 mV, ±33 mV, and ±200 mV (full scale).

The J1939 and the CAN sleds provide galvanic-isolated CAN ports. The on-board termination resistor is switch selectable. It is rated for 100 Ω at 1 W. The CAN sled provides two channels, supporting bit rates up to 1 Mbit/s. The J1939 sled is specified for a bit rate of 250 kbit/s. These sleds can be powered and configured from the Flexedge DA50 or DA70 controller using Red Lion's Crimson software.

## CAN driver software on Github

For the PC-to-CAN interface modules under the Ixxat brand, HMS offers on Github regularly updated software drivers, APIs (application programming interfaces), and programming examples. This includes .NET, VCI-interface, .NET-wrapper, and Python support for various CAN interfaces. These software packages offered free of charge are continuously maintained and updated.


*Figure 2: Several free-of-charge CAN-related software products are available on Github (Source: HMS/Ixxat)*

In addition to HMS' own .NET package, a Python interface is available on the open-source platform that supports CAN interfaces from various manufacturers. In combination with the Ixxat VCI-V4 driver, the Python package enables access to the Ixxat interface modules for sending and receiving CAN data frames.

## CAN in battery energy systems

HMS Networks offers communication solutions for the growing battery market, focusing on battery energy storage systems (BESS). They support CAN connectivity, remote monitoring, and integrate various devices in large battery parks.

Since "green" energy sources, such as wind and sun light, are not always available, large battery parks can make it possible to store the energy to use when there is less sun or wind. A BESS is made up of battery cells, which are combined into battery packs, which can then be combined into containers. The latter in turn, can make up ▷

battery parks with the size of several football fields. It can use new or recycled batteries and combine many different components. However, these large battery parks need a lot of communication capabilities in order to work. HMS now presents a comprehensive package of communication solutions catered for this application area.


*Figure 3: Battery energy storage systems (BESS) often use CAN-based networks applying gateways to integrate other communication technologies (Source: HMS/Ixxat)*

Since the battery market has been spearheaded by the car industry, CAN is often the communication system of choice. Here, HMS offers a portfolio of communication solutions via the Ixxat brand – for PC connection, networking, line extension, galvanic isolation, and more. In networking, the protection of devices is crucial, both from external sources (like lightning strikes) and from system-related issues (AC/DC noise). Device protection is a must to avoid damages. Therefore, a BESS often combines a variety of different equipment from different industries within a single application – batteries, BMS (battery management system), PLCs (programmable logic controller), fire-detection units, or air-conditioning systems. To combine different equipment, it is necessary to interconnect various communication systems. CAN/CANopen, Modbus, BACnet, KNX, and other protocols can be linked via gateways from HMS, which handle the data exchange while also enabling centralized control. In addition, these gateways can selectively filter, map and, if necessary, manipulate the data in order to realize the best possible data exchange.

One of the customers is Super-B in the Netherlands. The Dutch company offers lithium iron phosphate batteries. Its battery management system (BMS) is based on CAN/CANopen products by Ixxat. It enables an increased power outtake as the batteries can be connected to a chain. Another important feature is connecting batteries to external systems by means of a CAN-to-USB dongle. The BMS also comprises the protocol software by Ixxat as the basis for the operating system of the CANopen-based monitoring system.

The BMS can adapt the balancing according to the battery charge and discharge current, as well as balancing individual batteries, if they are connected to the CAN network. The system is also used to monitor the batteries' state, to report secondary data, to protect the battery, to control the environment, and to balance it. This is done by transmitting battery alarms and information to the CAN interface so it can be displayed on a PC. The system can manage over-voltage, under-voltage, deep-discharge as well as over-charge and can control the temperature of the batteries. Furthermore, the Ixxat bootloader enables the remote update of the multi-processor firmware inside the battery.

"The Ixxat protocol software stacks were especially well-suited for this project as they enable very quick re-connect times", said Kurt van Buul from Super-B. "The batteries are always on, even if they are not discharging energy, but the CAN bus can go up and down, and therefore it is important that the re-connecting does not take a long time." The project was accompanied by the HMS partner Twincomm located in Veldhoven, Netherlands.

"We have very skilled programmers in-house for our own electronics systems, but we needed help to implement the CANopen parts," explained Super-B engineers. "Twincomm did a really good job implementing the system and the Ixxat USB-to-CAN V2 product works great with very few issues. We have had to work a bit with the stacks to get them working the way we want, but we've found that the support from Twincomm and HMS has been quick and efficient, which has been important for us."

*hz*

## PCI Express mini with four CAN FD ports

ESD Electronics released CAN-PCIeMiniHS/402 interface board in PCI Express mini half-size format. The card enables expansion with up to four CAN FD ports in tiny computers and SBC (single-board computer) applications. The interface card is also available as a variant with extended temperature range. The four ports are connected to the CAN network via separate compact D-Sub 9-pin adapters with integrated galvanic isolation and on-board CAN terminating resistors that can be switched via jumpers.

*CAN-PCIeMiniHS/402 interface board (Source: ESD Electronics)*

For the exchange of CAN data, the card has an integrated bus-mastering unit (first-party DMA). At high data rates, this reduces the CPU load on the host system and the overall system latency.

Thanks to the support of MSI (message signaled interrupts), the mini card is suited for use in hypervisor environments. In addition, the module supports high-resolution 64-bit hardware time-stamps to enable high-precision reception and sending of CAN frames.

Software drivers for Windows and Linux are included in the delivery. On request, the company also provides drivers for various real-time operating systems such as QNX, RTX64, VxWorks, etc. Individual adjustments to hardware or software can be implemented according to the customer's requirements.

*of*

# CANopen sensor systems for urban gardening

*Pironex (Germany) develops electronic, software, and hardware technologies. Based in Mecklenburg-Western Pomerania, the company is committed to creating smart solutions that enhance urban living and address the challenges of modern cities. Its latest development, a smart sensor system for plant pots, epitomises the manufacturer's mission to integrate technology with everyday life, promoting sustainability and efficiency.*

(Source: Pironex)



As urbanisation continues to rise, the need for green spaces in cities becomes increasingly vital. Plants in urban gardens and green spaces not only improve air quality and provide aesthetic value, but they also contribute to the well-being of city dwellers as green spaces lower city temperatures and improve thermal comfort. However, maintaining these spaces in large cities poses significant challenges. Ensuring that plants receive the right amount of water and nutrients is crucial for their health, but it can be difficult to monitor their condition consistently.

The management of community gardens presents several challenges. Traditional methods of plant care are often inefficient, relying on manual inspection and watering schedules that may not meet the specific needs of each plant. This can lead to overwatering, underwatering, and overall poor plant health, diminishing the benefits of urban gardens. This is where smart sensor systems for plant pots, utilizing CANopen, offer a solution that simplifies plant care and enhances urban greenery.

## The sensor system

The smart soil analysis sensor system has been designed for real-time plant condition monitoring and optimal care for plants in urban gardens through a robust and efficient CANopen network. The system is composed of sensors embedded in plant pots and a cable connected to a gateway, which transmits data to the Pironex IoT (Internet of Things) portal. The key features of the system are:

- The system supports cable lengths of up to 25 meters, connecting the sensors to the gateway and providing flexibility for various configurations in different garden setups.
- The sensors are supplied with power via the gateway. If a larger number of sensors is required, an additional power supply module with a larger solar panel is provided to ensure the necessary power is supplied to the sensors.
- The system utilizes the CANopen protocol, known for its reliability and efficiency in industrial and automation applications, ensuring robust data transmission and seamless integration with other systems.

The sensor system includes various types of sensors, such as plant sensors that monitor the health and growth conditions of the plant, and soil quality sensors that measure critical soil parameters such as moisture and conductivity. The gateway is housed in a durable plastic enclosure with protection against dust and waterjets, making it suitable for outdoor environments. The measurement ranges include temperatures from -10 °C to +70 °C ensuring functionality in various climates. The sensors continuously monitor soil moisture levels to prevent overwatering or drought conditions and measure soil electrical conductivity to assess nutrient levels and soil health. The system supports the CiA 401 CANopen profile for I/O modules, ensuring compatibility with standard automation and control systems.

## Data collection and transmission

The parameters measured by the CANopen sensor system include:

- Temperature: recorded in degrees Celsius or Fahrenheit, providing crucial data on environmental conditions.

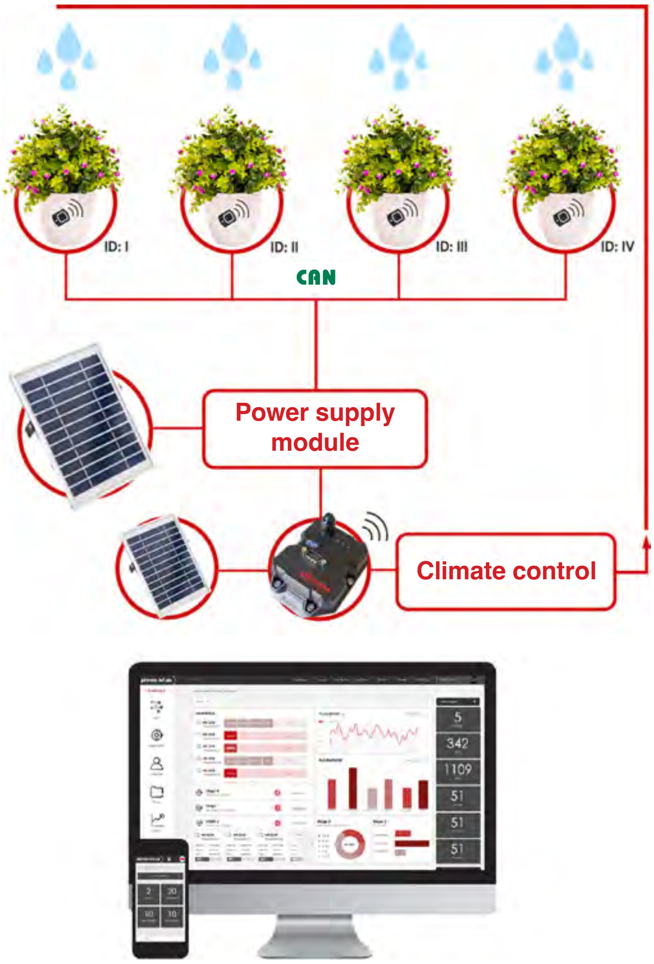



*Figure 1: The sensor system operation principle on example of watering (Source: Pironex)*

- Soil moisture: monitored in real-time to maintain optimal hydration levels for plants.
- Volumetric water content (VWC): measures the volume of water in the soil, offering precise information on soil moisture.
- Soil conductivity: indicates the soil's ability to conduct electricity, reflecting its nutrient content and overall health.

The gateway collects the sensor data and sends it to the Pironex IoT portal. Users can access the portal to view real-time data, analyse trends, and receive alerts when plants need attention.

## Advantages of CANopen networking

Utilizing CANopen for the smart sensor system offers several key advantages. CANopen is well-known for its robustness, making it ideal for applications (such as urban gardening systems) that require high reliability, as it ensures that data is transmitted accurately even in environments with high electromagnetic interference. CANopen networks are easily scalable, allowing for the addition of more sensors if needed, which is beneficial for expanding urban gardens or integrating additional monitoring capabilities. Compliance with the CiA 401 profile for I/O modules ensures that the system can integrate seamlessly with other CANopen automation and control systems, providing a unified solution for urban gardening management.

## Real-world applications and deployment

The sensor system is designed for a variety of smart city applications, with a focus on improving urban gardening and plant care. Key applications include smart city initiatives that demonstrate the system's capabilities in real-time plant monitoring and automated irrigation, serving as a proof of concept and demonstrating the benefits and potential of smart plant care systems. Installation in urban gardens ensures that plants receive optimal care, enhancing the aesthetic and environmental value of urban green spaces and contributing to the overall well-being of city residents. Equipping school gardens with sensors educates students about plant care and environmental stewardship, supports plant health, and fosters environmental awareness and responsibility in young learners.

The sensor system's versatility extends to climate control applications, making it an ideal solution for automated environmental management in both outdoor and indoor settings, such as greenhouses. It can automatically regulate irrigation based on soil moisture levels, ensuring optimal water usage and plant health. Additionally, the system can monitor and adjust temperature, humidity, and $CO_2$ levels within greenhouses, creating the perfect growing conditions for a wide variety of plants. By maintaining these critical parameters within optimal ranges, the system not only enhances plant growth and productivity but also reduces resource wastage.

## Challenges and solutions

Implementing a smart sensor system in urban environments presents several challenges, particularly regarding the power supply. One potential solution is integrating a solar power system to ensure continuous operation of the gateway and power supply module. This approach not only promotes sustainability but also reduces dependency on traditional power sources. The integration of solar power ensures that the system can operate autonomously, even in remote or hard-to-reach locations.

To enhance energy efficiency, the sensor system utilizes a standby mode that markedly reduces the power usage. This approach to using a standby mode conserves ▷


*Figure 2: Sensors are supplied with power via the gateway. If a larger number of sensors is required, additional power supply module with a larger solar panel is provided (Source: Pironex)*


*Figure 3: The gateway transmits collected data to the Pironex IoT portal (Source: Pironex)*

*Figure 4: The sensor system enables real-time monitoring and optimal plant care (Source: Pironex)*

energy, making the company's sensor systems well-suited for sustainable urban gardening and smart city projects.

## Conclusion

The smart sensor system offers a new approach to urban gardening, providing real-time monitoring and optimal plant care. Utilising the robustness and scalability of CANopen networking, the system addresses the challenges of maintaining green spaces in cities, improving the quality of urban life and supporting the growing trend towards smart cities.

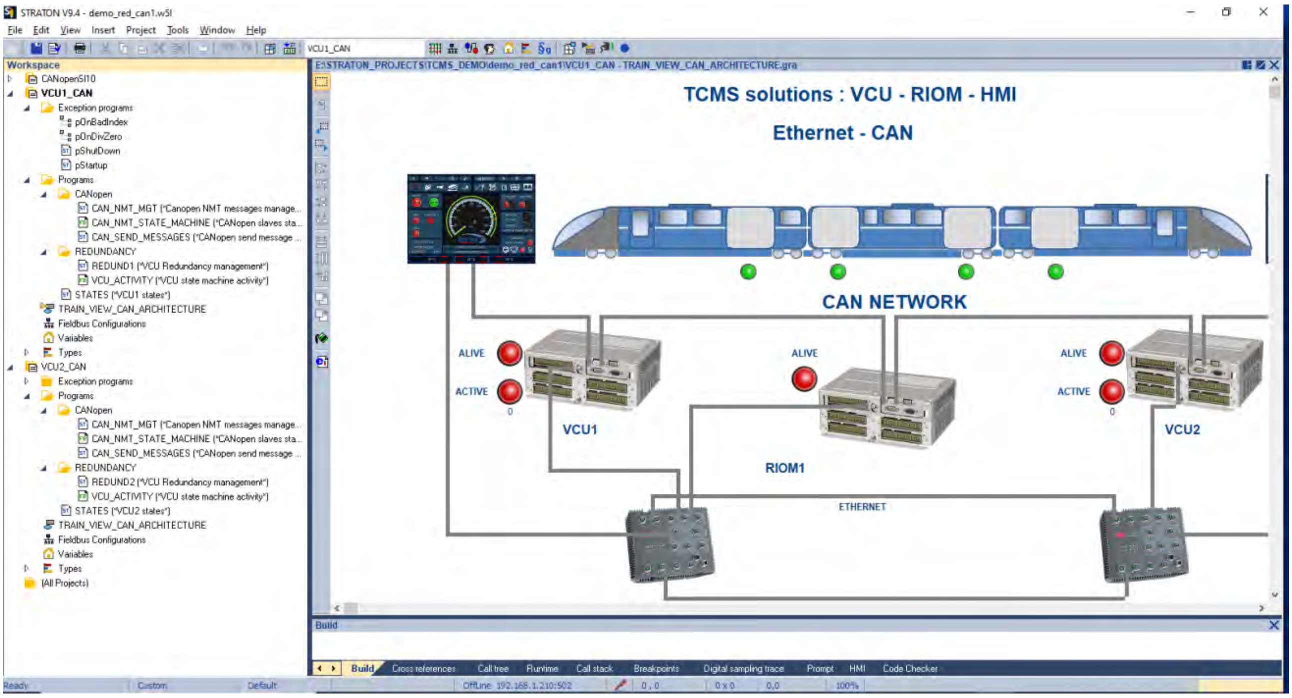
The Pironex's commitment to innovation and sustainability drives the company to improve and expand its solutions. The employees are dedicated to explore new technologies and partnerships to provide more advanced and effective smart IoT solutions for urban gardening and beyond. ◀

**Author**

Aileen Raddatz
Pironex
info@pironex.de
www.pironex.de

# Redundant train control and monitoring system (TCMS)

*Leroy Automation, part of Agon Electronics, has developed a redundant TCMS host controller with CANopen NMT (network management) manager functionality. The product is suitable for mainlines and metro cars as well as power sub-station automation with dedicated remote terminal units (RTU).*




(Source: Leroy)

The advantages of CAN lower layers and CANopen higher-layer protocols are key elements in the success of this project. This includes the robust and reliable communication regarding the lower layers and the off-the-shelf interoperability of the higher layers. The availability of CAN hardware and CANopen software as well as tools for reasonable prices was another selection criteria.

Unlike, historical buses and protocols based on EIA-232 or EIA-422/-485 such as Modbus-RTU and Profibus, or even MVB (Multifunction Vehicle Bus), CAN-based networks allow the transmission of prioritized data frames without altering or destroying the data frame currently sent. Indeed, EIA-232 and EIA-422/-485 serial buses allow commander/responder polling-type protocols, where the commander "speaks" and the responders "listen" by responding each one at a time. This method, called "polling", requires a cycle whose duration depends on the number of responders and the quantity of data to be transmitted.

On the contrary, CAN-based networks allow communication according to the producer/consumer model, where each node, whether commander or responder, can speak whenever it wants, without worrying about the state of the network. This method achieves a much higher performance, because frame collisions are taken care of by the CAN controller, which facilitates the developer's work.

The CAN CC (classic) protocol covers the data link layer and the physical coding sublayer of the 7-layer OSI (open system interconnection) model. EIA-232 and EIA-422/-485 specifications relate to the physical layer. They only provide basic rules and the physical link for byte exchange to enable transmission of serial data using a multi-drop bus topology. The content of the frames is user-defined. This is an advantage in terms of simplicity, but when it comes to compatibility, it becomes a disadvantage given the multitude of existing protocols.

CANopen is a high-level protocol (layer 3 to layer 7 of the OSI model), which implements the mechanisms necessary for monitoring the network, transmitting not-confirmed PDO (process data object) messages and confirmed SDO (service data object) messages. CANopen thus supports commander/responder-type communication (NMT messages), producer/consumer-type communication (PDO), and client/server-type communication (SDO messages).

With asynchronous EIA-232/-422/-485-type links, the designer needs to develop a proprietary protocol, which ▷

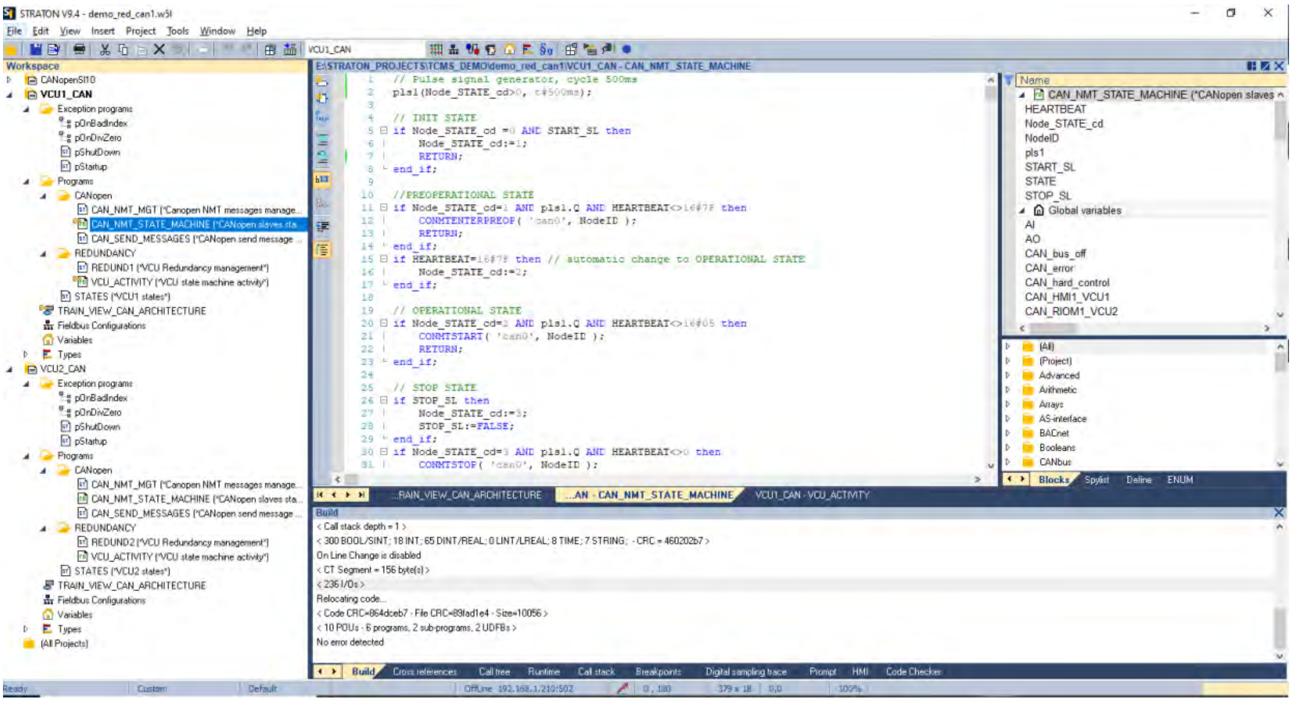



*Figure 1: Screen of the Straton programming and configuration tool (Source: Leroy Automation)*

requires significant time for definition and implementation, with support for each node of the network. The development may take a significant time and the reliability and robustness of a proprietary protocol could be uncertain!

## The CANopen object dictionary

A fundamental notion of CANopen is the object dictionary. It is used to inform other CANopen devices (normally the unit with NMT manager functionality) about the properties of a device, in a standardized manner by means of SDO communication. This mechanism thus allows this device to be modeled, in order to make the hardware independent of the control software. The benefit for the end user is to be able to choose their equipment independently of the manufacturer.

In order to facilitate this approach, significant work has been carried out by the non-profit organization CiA (CAN in Automation) through specifying CANopen profiles: generic I/O modules (CiA 401), drives and motion control (CiA 402), medical devices (CiA 412), control systems for elevators (CiA 417), and profiles for rolling stock specific equipment:

- CiA 421: CANopen application profile for train vehicle control networks;
- CiA 423: CANopen application profile for rail vehicle power drive systems;
- CiA 424: CANopen application profile for rail vehicle door control systems;
- CiA 426: CANopen application profile for rail vehicle exterior lighting control;
- CiA 430: CANopen application profile for rail vehicle auxiliary operating systems;
- CiA 433: CANopen application profile for rail vehicle interior lighting control.

The CANopen connectable TCMS controller and the RIOM product line by Leroy is an embedded control system with built-in input/output features designed to be on-board integrated in rolling stock vehicles. The products are compliant with the EN 50155 standard. The VCU (vehicle control unit) TCMS devices feature CANopen NMT manager redundancy. They are on the same CANopen network negotiating who is the active NMT manager and taking over the NMT responsibility in case of a malfunction of the active NMT manager. For maintenance purposes an additional Ethernet link is applied.

## VCU programming and redundancy

The host PC (personal computer) workbench used for programming the VCU is the Straton tool from Copa-Data. It comprises:

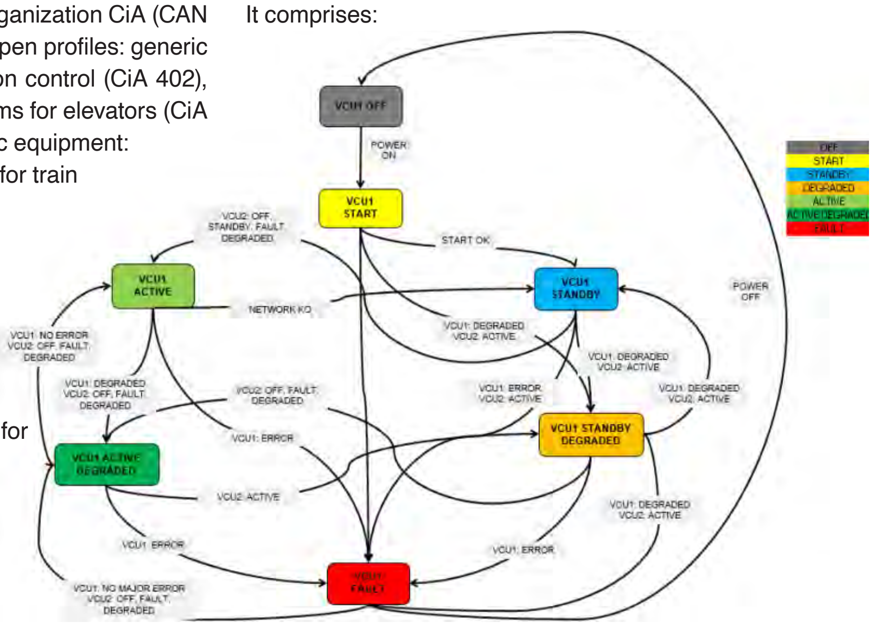



*Figure 2: VCU "hot" redundancy state diagram (Source: Leroy Automation)*

Devices

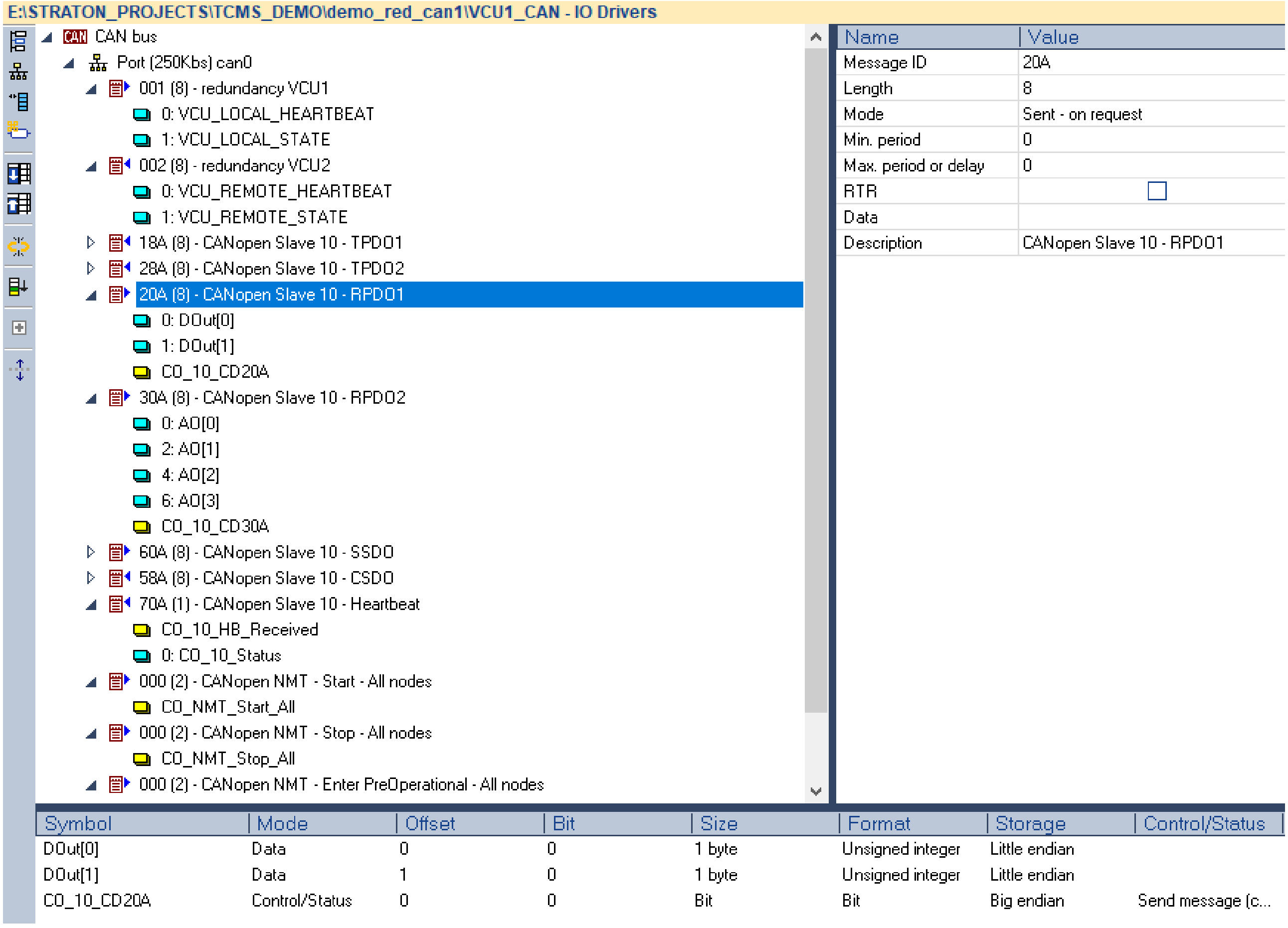



*Figure 3: Configuring PDOs with the Straton tool (Source: Leroy Automation)*

- an IEC 61131-3 languages programming tool;
- a CANopen editor for network configuration;
- real-time monitoring tools for debugging purposes.

The tool shows in Figure 1 a list of projects in the upper workspace area. The three available projects can be designed and debugged. The project called "VCU1_CAN" (see Figure 1) is divided into several programs and tools to manage different features: startup, VCU redundancy as well as CAN and CANopen communication.

The VCU redundancy is managed to enable a "hot" redundancy, if one VCU fails. In each VCU, two programs manage the redundancy process:

- Redundancy management with the second VCU ("REDUND" program): This program calls the function block "VCU_ACTIVITY".
- VCU activity state machine ("VCU_ACTIVITY" function block): This function block allows to calculate the VCU state: start, standby (nominal), standby (degraded), active (nominal), and active (degraded).

## CANopen communication

The CAN communication is determined by means of the configuration editor and structured programs. The CANopen messages are configured by means of the Straton tool. Data exchange and diagnostic/control variables can be defined for each CANopen message. CANopen messages mapped to CAN data frames using CAN-ID 1 and CAN-ID 2 are used for the VCU redundancy between VCU 1 and VCU 2. Each VCU sends a heartbeat message and its VCU activity state: those values are used in the "REDUND" program to manage the VCU activity state. PDO messages mapped to CAN data frames with CAN-ID $18A_h$ and CAN-ID $28A_h$ are transmitted by the CANopen device with the node-ID 10. The PDOs mapped to CAN data frames using CAN-ID $20A_h$ and CAN-ID $30A_h$ are transmitted by the active VCU to the CANopen device with the node-ID 10. These PDOs set the message control parameter respectively the "CO_10_CD10A" parameter.

Several programs have been developed to manage the messages defined in the fieldbus configuration editor:

- PDO transmitting program CAN_SEND_MESSAGES: This program is used to trigger the sending of PDO messages from the active VCU host controller to other CANopen devices with NMT server functionality.
- NMT program "REDUND": Only the active VCU sends NMT messages and manages other CANopen devices. It calls the CANopen state machine function block instance for each managed CANopen device.
- CANopen state machine function block CAN_NMT_STATE_MACHINE: This function block allows to manage the CANopen states for a CANopen device.

For debugging the system, several Straton tools are available: list of variables, graphic monitoring views, soft oscilloscope, test sequences, output-window view, and digital-sampling traces. In a graphic-monitoring view, all parameters can be animated by the project variables' values, when in simulation or online mode. ▷

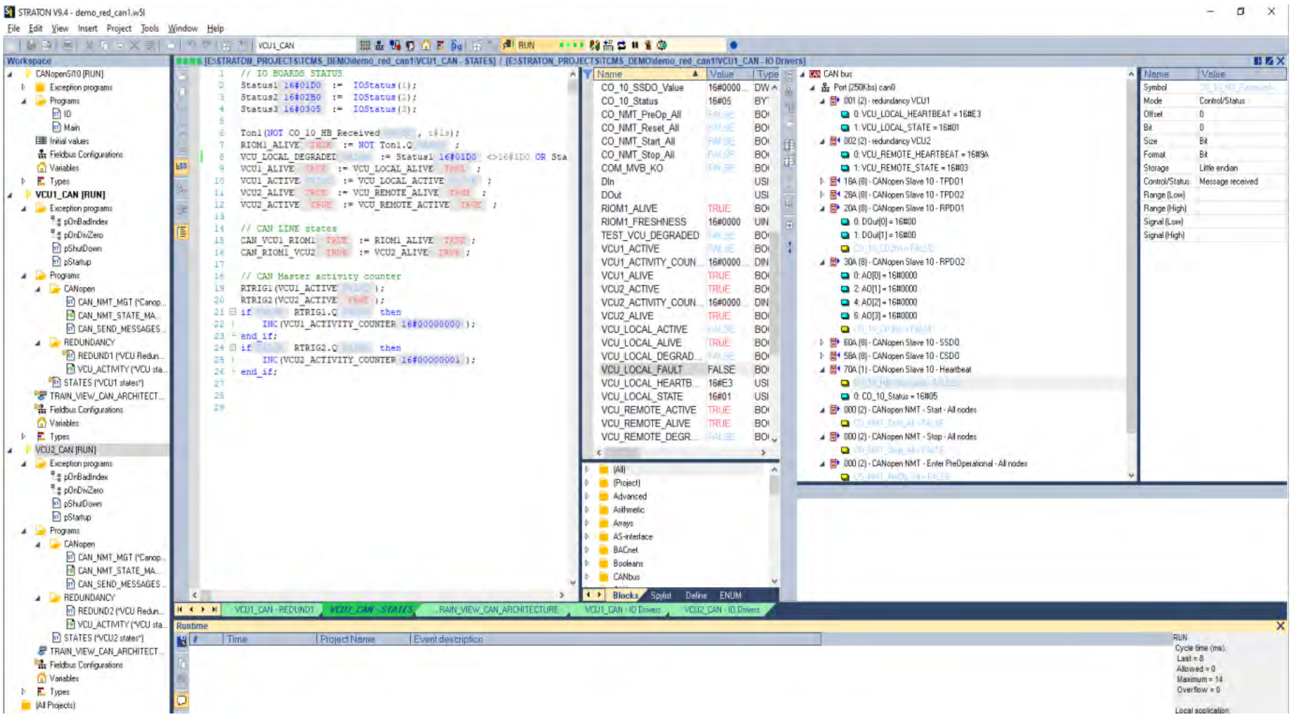


*Figure 4: Busmaster software tool (Source: Leroy Automation)*

The three devices in the system were downloaded with their Straton project from the project list. All of those devices can be debugged at the same time through the workbench or in several workbench instances. The low-level CAN communication can be monitored via a PC with a specific CAN network software tool Busmaster. An USB/CAN dongle is applied to connect the PC to the CAN network. The tool can monitor all CAN frames and the bus load.

*hz*

# Supporting IEC 61131-3 for 30 years

*In July 1994, Dieter Hess and Manfred Werner founded 3S-Smart Software Solutions. Since June 2020, the company is known as Codesys, which is part of the Codesys Group. Today, the entire group employs more than 230 people at its headquarters in Kempten (Germany) and in its subsidiaries in China, Italy, and the United States.*

*"Our continuity is one of our great strengths," said Manfred Werner. "It is essential in an industry where product lifetimes are measured in decades." In fact, all employees from the very beginning are still working for the Codesys Group. Even the customers who placed their first order by fax in 1994 are loyal to Codesys to this day." Dieter Hess stated: "While our industry used to be blamed for being a job killer, it now offers the opportunity to at least partially compensate for a lack of personnel by increasing productivity." Hilmar Panzer added: "In the coming decades, we want to continue our role as the market leading supplier of hardware-independent control software and expand it with further useful products and services." (Source: Codesys)*

In the 30 years since its market launch, Codesys has become more than an IEC 61131-3 tool. It features a complete ecosystem integrating visualization and connection to serial and Ethernet communication technologies through to motion and safety. The Codesys Automation Server is an Industry 4.0 solution.

New in the portfolio is Codesys Virtual Control SL. This virtual controller can be installed on any architecture with a container or hypervisor/VM – as often as required and with scalable performance. In this way, virtual controllers can be realized that are equipped with all the familiar features of Codesys and programmed in IEC 61131-3 languages using the Codesys Development System. Another innovation is the web-based development environment "Codesys go!". It will make development independent of Windows desktops, as it will also run on the server and on the controller.

Hilmar Panzer, Manfred Werner, and Dieter Hess (see photo, from left to right) answered jointly questions from the CAN Newsletter editors.

**Q** When did Codesys start to support CAN communication? Which companies were your first customers for this technology?

**A** The first customers who needed CAN/CANopen support approached us back in the late 1990s – for example, the company ifm. At that time, we did not yet have our own CAN product range. Customers had to implement the low-level driver for the CAN chip they were using themselves and provide us with one of the commercially available protocol stacks. We then embedded this stack in our Codesys runtime system, which required major effort. After a short time, we had our own low-level drivers for the most common CAN chips. At the same time, we developed product solutions to supply a CANopen stack for our customers more conveniently than before. The first step was a simple stack for the then very popular Infineon C166/C167 micro-controller family at the beginning of the 2000s, which we implemented in C ourselves. With the further development of Codesys to the current V3 architecture, we switched to implementing the CANopen protocol stack in the IEC 61131-3 languages, specifically in Structured Text. This has the huge advantage that the stack is now completely hardware-independent and is compiled along with the control application onto the respective target hardware. And, it also allows users to change the control platform with just a few mouse clicks – at least as far as the connection of the inputs and outputs via CAN/CANopen is concerned.

**Q** How many Codesys runtime licenses (or other licenses) with CAN communication have been sold? What is the market share of Codesys with CAN support?

**A** In the past, our IEC 61131-3 product range around Codesys was almost exclusively tailored to device manufacturers. More than half of our customers purchased the add-on buyout for CAN/CANopen for our runtime system toolkit and thus implemented the integrated support in their devices. In this way, device manufacturers were able to equip hundreds or even many thousands of individual devices with Codesys CAN support. Even at this stage, it was difficult to estimate how many devices there were in detail. Since our business model has changed a great deal, we are now even less able to say how large the proportion of customers using CAN/CANopen is. That is why the number of licenses does not say much. After all, we now also serve many machine ▷

and plant manufacturers directly with our SoftPLCs, device manufacturers being no longer involved. CAN support is already included in many SoftPLC licenses and is not counted separately. In any case, it is safe to say that direct turnover of CAN products is still on the rise.

*Q* Which are the most important application fields using Codesys with CAN connectivity? We would also like to know some key customers using Codesys with CAN support.

*A* Most of our customers require CAN support for use in mobile machines, such as machines in agriculture and forestry, in the construction industry, or in transportation and logistics. Accordingly, our key customers are control system manufacturers with a specific range of ECUs/mobile control systems, such as STW (Sensor-Technik Wiedemann), Moba, or Epec. At https://www.codesys.com/industries/mobile-automation.html we have listed a number of relevant applications, almost all of which use CAN or CANopen in Codesys. For example, aircraft tractors from Goldhofer or Trepel tow all types of aircraft onto the tarmac – right up to the A380. Then there are AGVs (automated guided vehicles) and, of course, applications in classic, stationary mechanical engineering.

*Q* Which CAN communication options are currently supported by Codesys?

*A* As mentioned above, we now offer everything that is needed in order to use common CAN chips and interface cards instantly. For example, we supply reference implementations for SJA stand-alone CAN controllers by NXP and SocketCAN, as well as available drivers for the most common cards and interfaces, including PeakCAN, CANfox, and the EL6751 Ethercat-to-CAN terminal by Beckhoff. Communication between the IDE (integrated development environment), i.e., the Codesys Development System, and the target system can be established via a CAN block driver. To communicate from the target system with connected CAN nodes, Codesys users can employ "raw" CAN data frames in their control application. With our stacks for CANopen and J1939, however, it is much easier. And not only for functional communication via NMT manager and NMT device stack, but also via CANopen Safety and J1939-76 Safety. To make the application as convenient as possible, the Codesys IDE also includes configurators for the systems. This means that you do not have to leave the Codesys platform to integrate and to configure I/O modules. Of course, Codesys supports the import of device descriptions in EDS format as well as a DBC import for J1939 databases. The usual parameterization of the modules can also be carried out in the integrated configurator – no need for external configuration tools. At the same time, users have a good overview of the bus structure - including comprehensive diagnostic information.

*Q* Which additional CAN communication options are in the pipeline (e.g. CAN FD, CANopen FD, J1939-22, etc.)?

*A* A block driver is being developed for CAN FD. This will significantly speed up communication between the IDE, the Codesys Development System, and the target device. Further developments in this area are highly customer-driven. As soon as more and more requirements are brought to our attention, such requirements will be added to our agenda. *hz*

# Harmonizing CANopen bootloaders

*CAN in Automation (CiA) has almost finalized the CiA 710 document specifying a generic CANopen bootloader function. The bootloader enables secure managing of the device's firmware updates via CAN. It is suitable for CANopen CC (classic) and CANopen FD devices.*

(Source: Adobe Stock)

The CAN-based higher-layer protocol CANopen is used in embedded control networks in many application fields. An essential aspect of modern embedded networking is the bootloader, which facilitates firmware updates for embedded devices, e.g. within a CANopen network. CAN in Automation has almost finalized the development of a harmonized handling strategy for bootloader applications in a CAN/CANopen environment, known as CiA 710. The intent of the specification is to allow software tools from any manufacturer to update any CAN/CANopen device's firmware by means of the same basic principles. Of course, among others, this approach considers cybersecurity aspects.

## What is a bootloader?

A bootloader is a small program that is, by using a minimum of resources, responsible for initializing the hardware and loading the main firmware of a device during its startup. In the context of CANopen, the CANopen bootloader is responsible for managing firmware updates via CAN. CANopen bootloaders ensure that devices can receive new firmware versions via CAN, and can validate and install them. Supporting a CANopen bootloader enhances flexibility for system maintainers. A new version of device firmware may remove identified security weaknesses or may add new application-related functions. Thus, the bootloader extends the longevity of devices.

Prior to the start-up of the system, tools or CANopen host controllers double-check the configuration as well as the software version of the CANopen devices. During that task they may identify that an update of the configuration or the entire CANopen device's firmware is required. With the aim to simplify and generalize the way of running a firmware update via CAN, the CiA TF (task force) Generic bootloader has specified a harmonized procedure.

## Key features of CANopen bootloader

To allow a harmonized control of the firmware update via CAN, the CiA 710 introduces a finite state automation (FSA) for embedded devices. The FSA defines a predictable device behavior for all FSA states. Thus, configuration tools or host controllers switching the FSA states of CANopen devices can expect a given device behavior. The FSA specified in CiA 710 (see Figure 1) differentiates between the basic modes of operation, a bootloader mode (BM) and an application mode (AM). When an embedded device is switched on, at first, it transits to the "bootloader initializing" state. The existence of a valid application program to be started is checked by means of the autostart information, referred to by data object $1F59_h$. If no valid program is found or starting a program is not intended, the device switches to the bootloader mode by default. As usual for CANopen devices, then the device enters the "Initialization" state, where the setup operations, including CAN controller initialization and bit rate configuration are performed, in accordance to CiA 1301. In bootloader mode (BM), the device expects and verifies the user authentication. Legal tools or CANopen host controllers have therefore the possibility to identify themselves to the CANopen device in bootloader mode. The successful identification initiates the CANopen device being ready to accept new application programs or configuration data. Therefore, the CANopen device enters the "BM allow application download" state, where it waits to receive a new application program. Prior to the data transfer of the firmware via CAN, tools or CANopen host controllers can learn the CANopen device's attributes such as flash status and flash operation times. This enables them to adapt their behavior – in particular their internal time-outs – accordingly. After a successful program transfer, tools or CANopen host controllers force the CANopen device to leave the bootloader mode, and (typically) to start the new application program. When the new application program has been started, the device runs in application mode (AM).

If a transition back to the bootloader mode is required (e.g., to modify the configuration or the entire application program) tools or host controllers have to pass the security check again, prior to initiate switching back to the bootloader mode. In addition, the status of the currently running device's application has to be taken into account. Of course, switching to BM is only possible, in case the device's application ▷

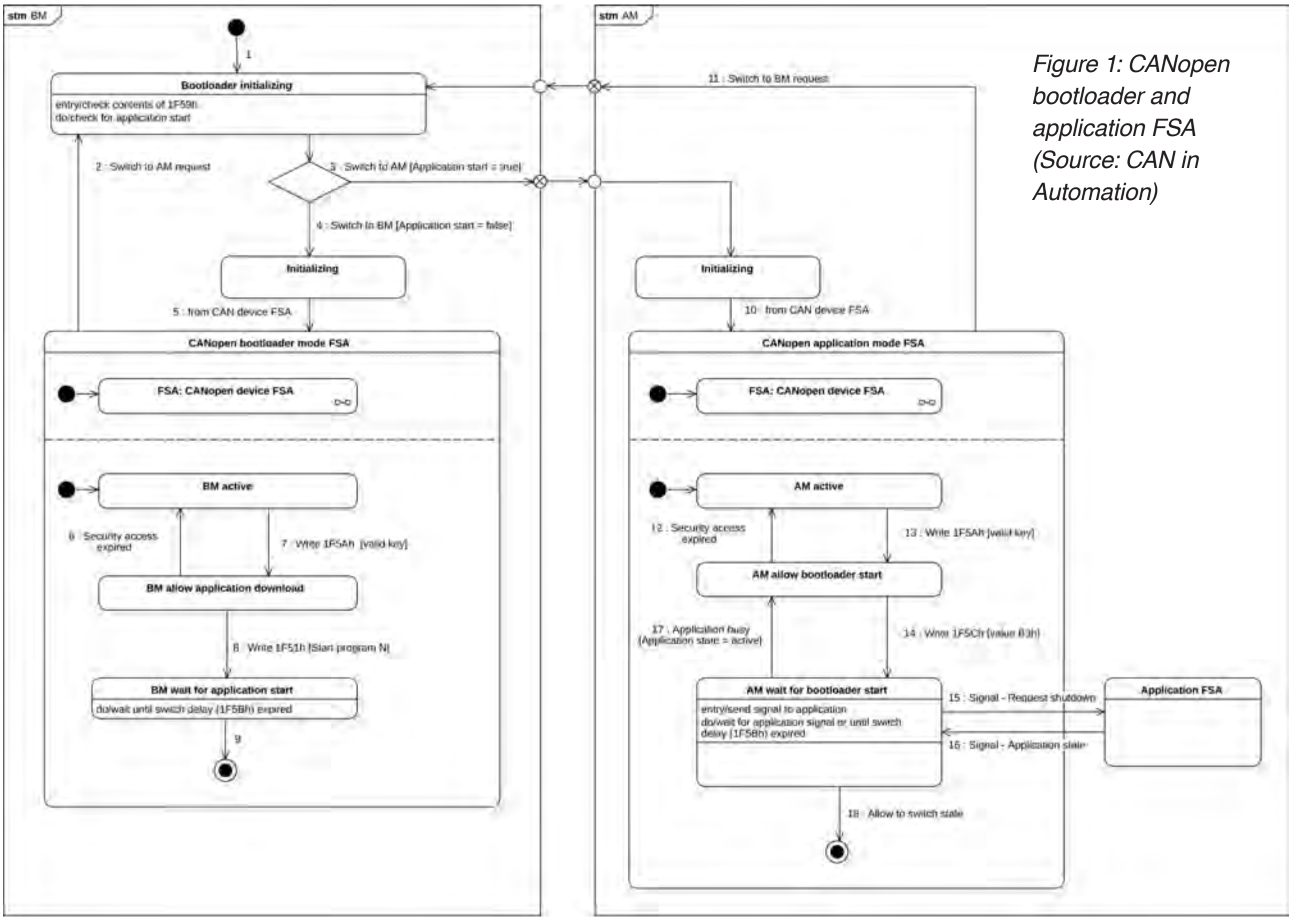



*Figure 1: CANopen bootloader and application FSA (Source: CAN in Automation)*

state allows a safe mode switching from AM to BM. To avoid "loosing of device" due to a non-successful firmware transfer, a roll-back respectively recovery function has been considered in CiA 710. In case of an error scenario during the firmware update, the device is not lost but will start with a pre-configured default application program. Overall, CiA 710 enables tools and CANopen host controllers to orchestrate the operating states of the device, to ensure secure firmware updates, and to maintain the integrity and reliability of the system operation.

## CiA 710 pre-implementation

Microcontrol (Germany) has pre-implemented the generic CANopen bootloader specification (CiA 710) in its CANopen bootloader protocol stack. It is used to securely update software on devices in a CANopen (FD) network without having to remove them from their environment. Versatile configuration options enable individual customization to a target product. The bootloader has been implemented to meet low storage requirements. It comprises a reduced object dictionary and supports NMT (network management), SDO (service data object), emergency, heartbeat as well as LSS (layer setting services) functionality. Up to four separate sections for storage of programs and data can be defined. A defined API (application programming interface) facilitates adjustment to a flash memory of the target hardware. Flash drivers for various controllers (e.g. STM32 series) are part of the CANpie driver. Ready-to-run examples for different demo boards are in the scope of delivery. The solution has been introduced at the Embedded World 2024 trade show in Nuremberg (Germany). *of*

## Conclusion

A bootloader is a fundamental function in modern embedded control systems, offering a flexible method to react on modified system/device requirements, by means of a device's firmware update over CAN. The CiA 710 specification represents a significant step forward in this field, providing a harmonized handling that supports interoperability, security, and devices' operational efficiency. CiA 710 considers various use cases such as firmware update at end-of-line production, diagnostic scenarios in laboratories, or updates over the air in the field, via a CANopen gateway that is not necessarily embedded in the network's CANopen host controller. As the industry evolves, adhering to CiA 710 maintains the reliability and performance of embedded systems across many applications. CiA 710 has almost been finalized. A CiA-internal release of CiA 710 as DSP (draft standard proposal) is expected in the second half of 2024. ◀

**Author**

Reiner Zitzmann
CAN in Automation
headquarters@can-cia.org
www.can-cia.org

# Real-time communication: Part IV - CANopen source code configuration

(Source: Embedded Systems Academy)

*Within this article series, Olaf Pfeiffer from Embedded Systems Academy (EmSA) is setting in perspective the timing requirements for different real-time capable communication systems, such as CAN, CANopen, and real-time Ethernet.*

In this fourth and last article in the series, the author shows which optimization options are typically available when working with CANopen source code, here, on example of Micro CANopen Plus from EmSA. The Part I, published in the December 2023 issue of the CAN Newsletter magazine, describes how to select the right real-time timeframe for certain applications. The Part II, published in March 2024, shows the different timeframes required by different applications and reviews what this means for the communication system used. In Part III (June 2024), the author examines the real-time capabilities and limitations of CAN and CANopen in greater detail.

Throughout this article, the author will examine specific examples illuminating the process of configuring, optimizing, and fine-tuning a CANopen stack to cater to advanced temporal requirements and possible system constraints. While the focus is on Micro CANopen Plus from EmSA, the explored principles and methods are likely to work with other CANopen source code implementations in a similar way.

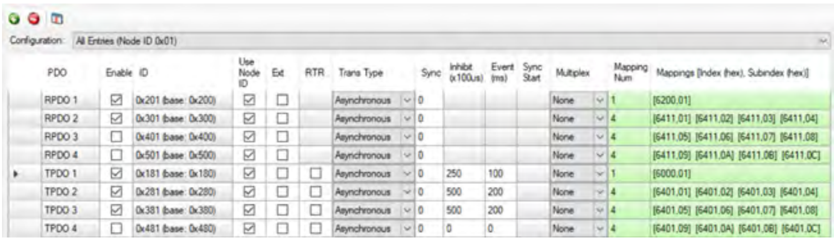
This part, translating the theoretical knowledge into practical implementation, aims to provide comprehensive insights and guidance for experienced CAN system designers and newcomers.

## CANopen PDO configurations

The configuration of CANopen PDOs (process data objects) plays a critical role in determining the response time in various applications. Depending on the required response time and the necessity to synchronize signals across multiple nodes, different PDO triggering mechanisms can be applied.

***Response time of 100 ms:*** For applications where the required response time is 100 ms or longer, there are typically two configuration methods that work well (and can also be combined):

- PDO triggering by event time (cyclic transmission): Here, the PDOs are transmitted cyclically at specified time intervals, such as every 50 ms. This periodic transmission ensures consistent response times.
- Change of state (COS) detection with inhibit time: This configuration transmits PDOs based on changes in state, with a minimum time (inhibit time) between transmissions. This inhibit time ensures that a toggling input does not produce back-to-back messages.



Configuration: All Entries (Node ID 0x01)

| PDO | Enable | ID | Use Node ID | Ext | RTR | Trans Type | Sync | Inhibit (x100us) | Event (ms) | Sync Start | Multiplex | Mapping Num | Mappings [Index (hex), Subindex (hex)] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RPDO 1 | ☑ | 0x201 (base: 0x200) | ☑ | ☐ | | Asynchronous | 0 | | | | None | 1 | [6200,01] |
| RPDO 2 | ☑ | 0x301 (base: 0x300) | ☑ | ☐ | | Asynchronous | 0 | | | | None | 4 | [6411,01] [6411,02] [6411,03] [6411,04] |
| RPDO 3 | ☐ | 0x401 (base: 0x400) | ☑ | ☐ | | Asynchronous | 0 | | | | None | 4 | [6411,05] [6411,06] [6411,07] [6411,08] |
| RPDO 4 | ☐ | 0x501 (base: 0x500) | ☑ | ☐ | | Asynchronous | 0 | | | | None | 4 | [6411,09] [6411,0A] [6411,0B] [6411,0C] |
| TPDO 1 | ☑ | 0x181 (base: 0x180) | ☑ | ☐ | ☐ | Asynchronous | 0 | 250 | 100 | | None | 1 | [6000,01] |
| TPDO 2 | ☑ | 0x281 (base: 0x280) | ☑ | ☐ | ☐ | Asynchronous | 0 | 500 | 200 | | None | 4 | [6401,01] [6401,02] [6401,03] [6401,04] |
| TPDO 3 | ☑ | 0x381 (base: 0x380) | ☑ | ☐ | ☐ | Asynchronous | 0 | 500 | 200 | | None | 4 | [6401,05] [6401,06] [6401,07] [6401,08] |
| TPDO 4 | ☐ | 0x481 (base: 0x480) | ☑ | ☐ | ☐ | Asynchronous | 0 | 0 | 0 | | None | 4 | [6401,09] [6401,0A] [6401,0B] [6401,0C] |

*Figure 1: PDO configuration with CANopen Architect (Source: EmSA)*

***Smaller response times or synchronized signals across multiple nodes:*** For applications requiring smaller response times or where there is a need to synchronize signals across multiple nodes, the Sync mode becomes the preferred method:

- Sync mode: In this configuration, one Sync producer generates a Sync CANopen message at stable, repeating intervals, e.g., every 10 ms. This Sync message serves as a triggering mechanism, used by all devices in the network to apply data synchronously, at the same time.

***Advanced Sync usage:*** When using the CANopen Sync mode, there are two advanced features to take advantage of. First, most Sync consumers allow the configuration of the Sync CANopen message identifier to be used. So, one could configure a system to use multiple Sync trigger messages and select which devices react to which trigger. Secondly, the latest version of CANopen supports the use of Sync with an integrated configurable counter. As an example, this could be configured to count until 4. On the Sync consumer side, one can configure the count value that each consumer listens to, again providing the option of grouping devices to react on specific Syncs on the system.

Choosing the right PDO configuration is vital for achieving the lowest response times. While cyclic transmission and COS (change of state) detection with inhibit time are suitable for more relaxed response time requirements, the Sync mode becomes essential when handling tighter time constraints or needing to synchronize multiple devices. More details on these different trigger mechanisms and how they can be combined is shown in a video by EmSA.

## Generic data flow in a CANopen protocol stack

On the lowest hardware level, a CAN controller will be receiving CAN frames. Depending on filters, these might be placed into pre-selected buffers or queues, and an interrupt signal will be generated. The processor handling the CAN controller starts processing the "CAN receive interrupt service" – typically part of a processor-specific driver. A generic driver will now simply pass on the CAN data to another software queue for later processing; Advanced options will be discussed later.

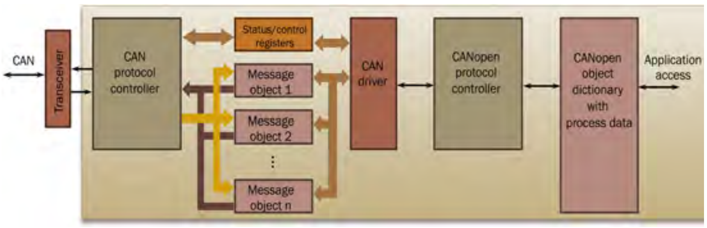



*Figure 2: CANopen data flow (Source: EmSA)*

At any time, the application program might update some of the process data to be transmitted via CANopen. To keep the CANopen stack alive, there will be a function, such as "ProcessStack()", that needs to be called frequently (for example, simply in a "main while(1) background loop"). When called, this function typically first checks if CANopen messages were received; if so, they are processed. If the data involves updating process data, then there is typically a callback to inform the application that new process data has arrived.

When all received CANopen messages are processed, the function checks if there is anything to transmit. It may detect that outgoing process data was modified by the application, and depending on the configuration of timers and transmission mode, initiate the transmission of a corresponding CANopen message. Such transmissions are typically passed to the driver level, possibly into a transmit queue, and it depends on the driver configuration when exactly this CANopen message will be passed to the CAN controller for transmission.

## Basic configuration and control options

Unless the required processing and response time is smaller than 100 ms such a data flow works good enough for most applications. If required response times get smaller, you should start looking into possible optimizations. When reviewing the generic data flow above, possible optimizations include the CAN driver optimization for receive, CAN driver optimization for transmit, and optimization of "ProcessStack()".

***CAN driver optimization for receiving:*** Many default drivers supplied by chip manufacturers (or even CANopen stack providers) might not take full advantage of the specific features of a CAN controller. One of the first possible optimization checks is to ensure that, where possible, hardware receive filtering and hardware receive buffers or queues are utilized, thus eliminating the need for a long (delaying) software receive queue.

***CAN driver optimization for transmission:*** Before reviewing this, consider what is more critical to your application – whether this device can transmit as many CAN frames back-to-back as possible, or whether the transmission should be somewhat throttled to ensure that no single device can produce too long a period of high-priority back-to-back traffic. If it should be throttled, consider implementing a transmit trigger based on a timer, such as being able to transmit one CAN frame every millisecond (EmSA's default driver uses the 1-ms timer interrupt for this).

***Considerations for "ProcessStack()":*** A typical question regarding "ProcessStack()" is how often it should be called and what the worst-case execution time is. Some prefer to call it from a fixed timer interrupt instead of the background loop. There is no generic answer to these questions. In the Micro CANopen Plus implementation, we try to keep the execution time short by not executing all pending CANopen tasks but only the most vital ones. How often it should be called depends heavily on the local device's communication. The Micro CANopen Plus implementation, however, has a slick feature here: with every call, only the most critical pending tasks of the CANopen stack are performed. Producing the heartbeat message is always the least important task. Therefore, by monitoring the device's heartbeat signal's accuracy, you can determine if calls to "ProcessStack()" occur often enough. If there are not enough calls to "ProcessStack()", the heartbeat becomes slower than specified, or it may not be transmitted at all. ▷

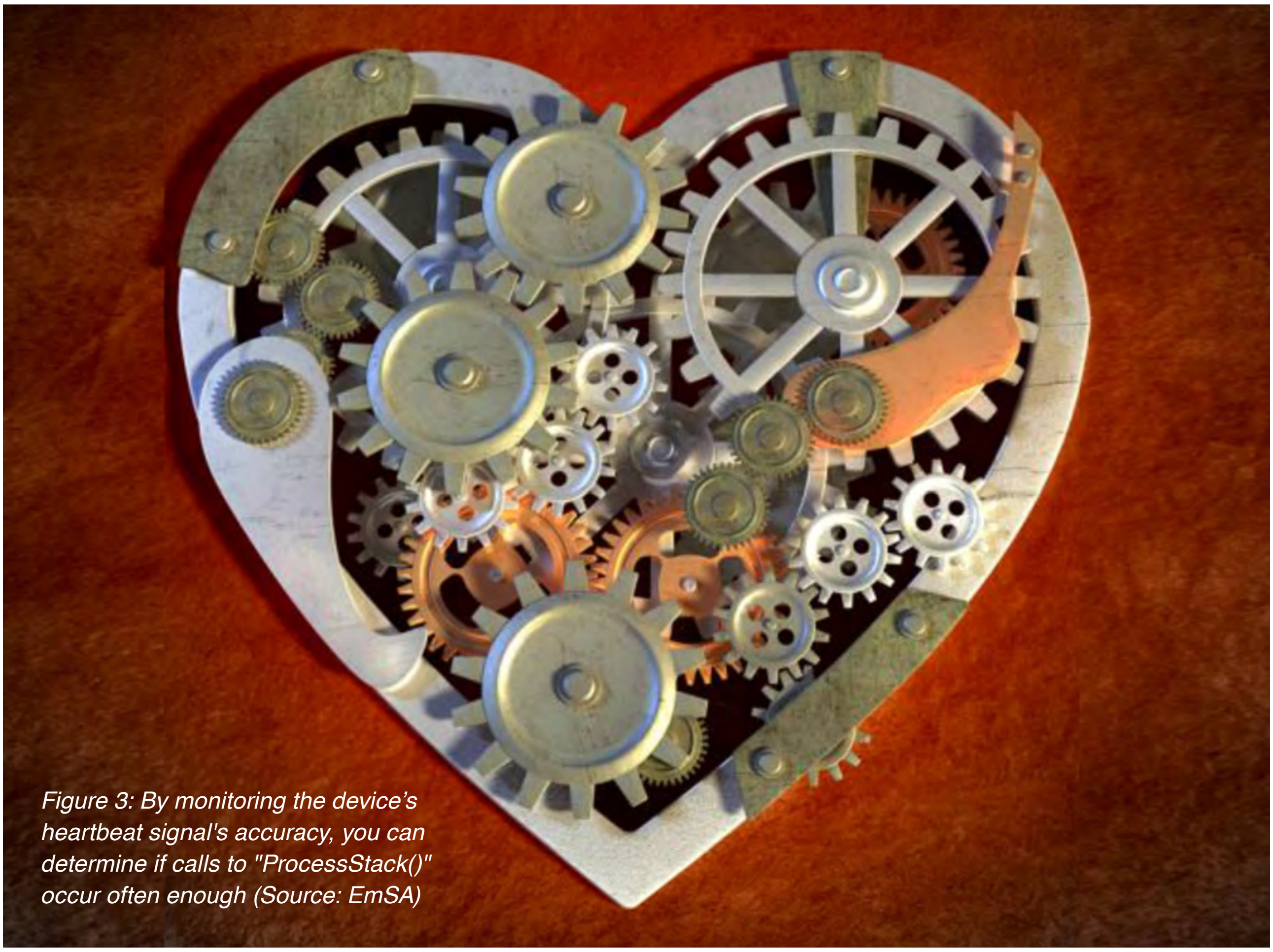


*Figure 3: By monitoring the device's heartbeat signal's accuracy, you can determine if calls to "ProcessStack()" occur often enough (Source: EmSA)*

***Note on return value for "ProcessStack()":*** Another important factor is the "ProcessStack()" function's return value. It returns TRUE when a pending CANopen task was executed, and FALSE when there is no CANopen task pending. If you want to ensure that all pending CANopen tasks are executed in your code, use:

```
while(ProcessStack())
{
}
```

This will keep re-calling the function until all pending CANopen tasks have been executed.

***Direct task trigger:*** The function "ProcessStack()" serves those who prefer not to go into the details of all the CANopen tasks executed from within. For further optimization, an application can bypass calling this function and directly invoke the dedicated CANopen tasks: "ProcessStackRx()" and "ProcessStackTick()".

- ***Sub-task "ProcessStackRx()":*** This task handles processing a received CANopen message. For an optimized call, it would ideally be initiated directly from the CAN receive interrupt or triggered by some signal set in the CAN receive interrupt.
- ***Sub-task "ProcessStackTick()":*** This task checks if the process data to be transmitted has changed (or was triggered for transmission) and if any actions based on the millisecond-timer need to be taken. The most efficient way to call this is only after process data has changed or the millisecond-timer has incremented.

This approach provides a more refined control over the execution of specific tasks within the CANopen stack, allowing for more precise tuning of performance and responsiveness.

## Bringing together CAN driver, CANopen stack, and application

On most 32-bit-based micro-controllers, the enhancements discussed so far are suitable for bringing the total response time down to a range of 10 ms to "a few milliseconds." This can be achieved without requiring optimizations that lead to a fully custom implementation that might be challenging to maintain. These optimizations were confined to leveraging individually triggered CANopen stack processes when needed. In general, this can be taken further. However, making changes at such an intrinsic level in a system can make it much more challenging to maintain or port to a different architecture when necessary. Therefore, the following is more to illustrate "what is theoretically possible," pushing optimizations beyond the point where a system remains easy to test, maintain, and port.

On the lowest hardware level, review if your CAN controller is configured to directly create a CAN receive interrupt with the reception of the Sync message and if you can easily detect the difference to any other CANopen message (e.g. own filter/receive buffer).

The only reasonable CANopen PDO communication mode for further real-time improvements would be the CANopen Sync mode. If that is used and we concentrate on ▷

optimizing it, then the previous other optimizations might become redundant.

Focusing on Sync optimization would require to modify the CAN receive interrupt service routine to directly call the CANopen stack function(s) responsible for Sync handling when a Sync signal is received. In the case of Micro CANopen Plus, this would be the function “HandleSync()”. When executing this from within the interrupt service routine, please keep in mind:

- Not to store this Sync in the regular receive queue (we already process it).
- That for both Sync-related transmit and receive data, call back functions to the application will be called – still executing at the interrupt service level.
- When using an RTOS (real-time operating system), a better solution would be to set a trigger signal in the interrupt that Sync was received, subsequently triggering the execution task immediately after the interrupt has completed.

With such a modification, a response time within a millisecond is achievable. If all devices participating in the Sync communication implement the Sync handling with equal optimization, the variation among the devices (e.g. when they each apply their outputs synchronously) can be as low as a few microseconds. Nevertheless, these are extreme values that have been observed to work in test and lab environments. For real-world applications demanding such short response or sync times, careful testing would be required to ensure that these targets can be reached under all realistic circumstances.

## Final conclusion

The journey through this four-part series has provided an in-depth exploration of the components that shape embedded real-time communication. From hardware selection to advanced optimization techniques, the underlying theme that resonates is the pivotal role of response times in determining every aspect of system design and configuration.

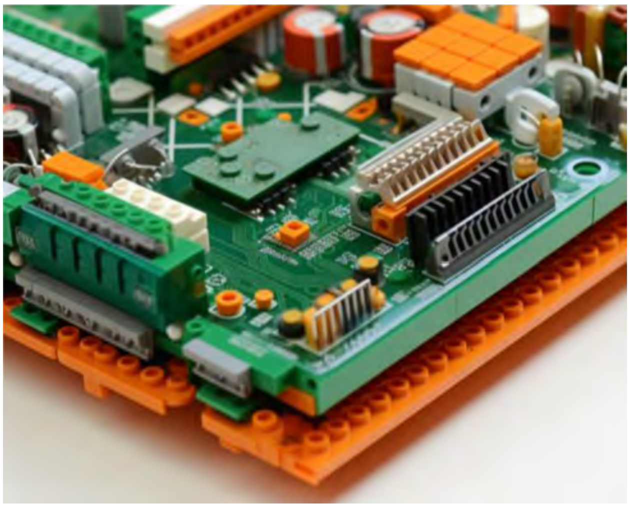


*Figure 4: From hardware selection to optimization techniques, the underlying theme that resonates is the pivotal role of response times in determining every aspect of system design and configuration (Source: EmSA)*

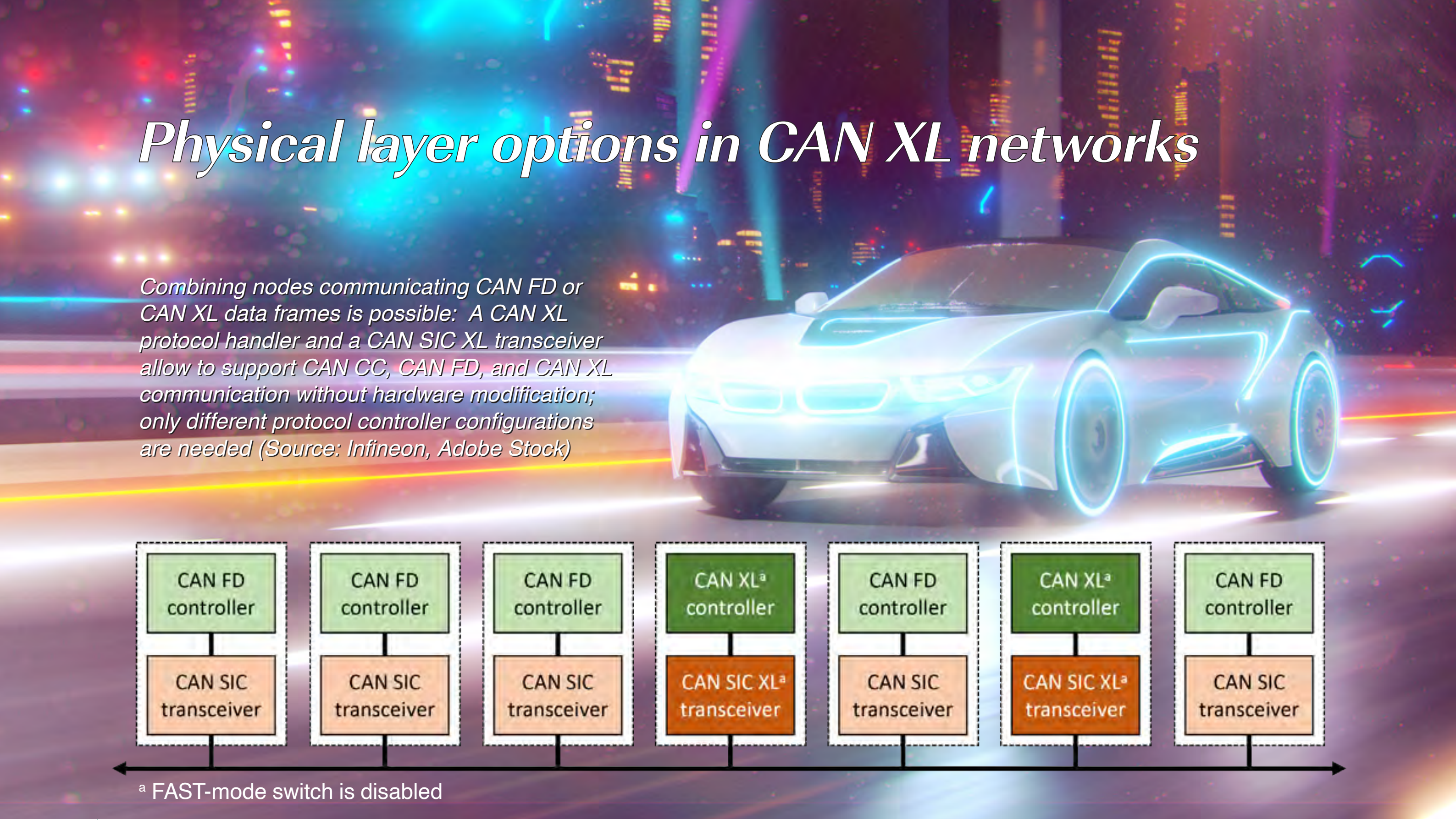
1. *Hardware selection:* The required response times dictate the hardware capabilities needed, influencing decisions on modules, possibly micro-controllers and other essential components.
2. *Operating system considerations:* Whether working with an RTOS or implementing a more specific, bespoke system, the response times heavily influence how the operating system needs to be configured.
3. *Network technologies:* Depending on the required throughput and speed, different network protocols and technologies must be taken into consideration. As an example, this series looked at the specifics of CANopen and its configurations, illustrating the nuanced choices required to meet different application demands.
4. *Optimization choices:* Perhaps one of the most profound insights is the realization that optimization is not a one-size-fits-all approach. Depending on the required response times, certain optimizations become essential, while others can be bypassed. It's a matter of fine-tuning, understanding what needs to be harnessed, and what can be left untouched without affecting performance.
5. *Strategic ignorance:* Contrary to the instinct to utilize every possible advantage, there are instances where the time frame allows for the deliberate ignoring of certain optimizations. Not every register provided by a network controller needs to be exploited; it's a balance between performance and the demands of the particular application.

Through this series, the author has illuminated the complex interplay of hardware, operating systems, and network technologies, all governed by the essential factor of response times. The insights offered serve as a guide for making strategic choices in system design, highlighting the importance of tailored optimization and thoughtful decision-making. These principles enable you to craft robust and efficient real-time communication systems, suited to your application demands. ◀


**Author**

Olaf Pfeiffer
EmSA (Embedded Systems Academy)
info@esacademy.com
www.esacademy.de

# Physical layer options in CAN XL networks




*Combining nodes communicating CAN FD or CAN XL data frames is possible: A CAN XL protocol handler and a CAN SIC XL transceiver allow to support CAN CC, CAN FD, and CAN XL communication without hardware modification; only different protocol controller configurations are needed (Source: Infineon, Adobe Stock)*

*At the beginning of the CAN story, 1 Mbit/s was the highest bit rate. Nowadays, you can achieve up to 20 Mbit/s in the data phase of a CAN XL data frame, when using CAN SIC XL transceivers and enabling the FAST mode. With one CAN node using a CAN XL protocol controller and a CAN SIC XL transceiver, you can realize CAN CC, CAN FD, and CAN XL communication by means of different configurations.*

An important advantage of CAN communication is that the CAN physical layer supports multi-drop networks. All nodes on the network receive the CAN frames at the same time. No switches are needed and there is no propagation delay between the different nodes. To organize such kind of communication, especially at the beginning, an arbitration phase is needed. In the arbitration phase all nodes transmit a logical “1” or a logical “0” on the network. To make this possible without damaging nodes or transmit undefined signal levels on the network, only the “0” is actively transmitted on the network as a dominant signal, while a “1” is passively generated by the termination resistors and is called recessive level. In Figure 1, the transmitter-output behavior is shown. The transmitter is switching between high impedance to allow recessive level on the network and low impedance to generate a dominant level. The recessive-to-dominant transition is controlled by the transmitter, while during the dominant-to-recessive transition, the maximum possible slew rate is limited only by the transmitter. This transition is mostly controlled by the wiring harness and the termination resistors. The permanently changing transmitter output impedance during the transmission of dominant and recessive signals causes a ringing on the network. This ringing limits the maximum possible bit rate in CAN networks.

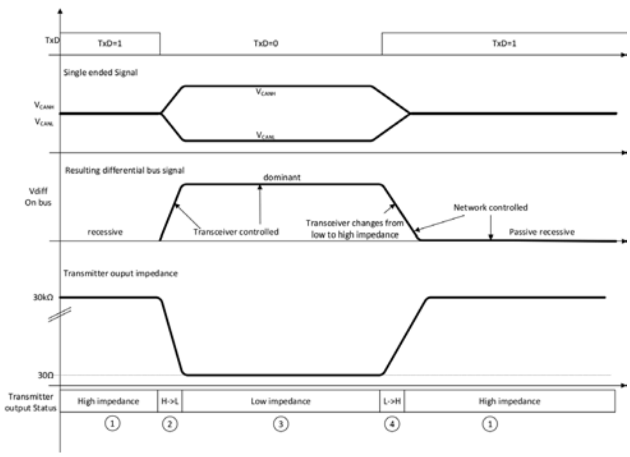



*Figure 1: CAN FD transceiver impedance (Source: Infineon)*

To achieve higher bit rates, a modification of the transmitter concept has been needed. With the integration of the SIC (signal improvement capability) transmitter, a first step has been done. In Figure 2 the SIC transmitter-output impedance behavior is shown. The dominant-to-recessive phase is now controlled by the transmitter, too. The output impedance changes now from low impedance in dominant phase to medium impedance of 100 Ohm for ▷

maximum 500 ns. This phase is called active-recessive phase. After the active-recessive phase, the transmitter-output impedance changes from medium to high impedance, in order to allow collisions on the network. This phase is called passive-recessive phase. The 100-Ohm impedance in the SIC phase has been chosen to match the transmitter impedance with the typical CAN twisted-pair wire impedance, which is 100 Ohm.

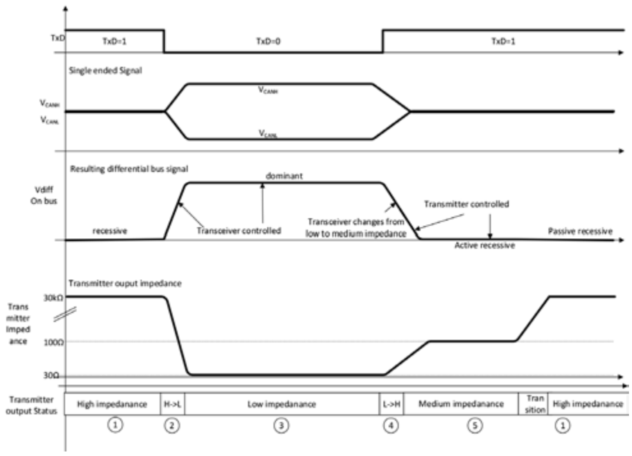



*Figure 2: CAN SIC transceiver output impedance characteristic (Source: Infineon)*

This modification improves the reliability of existing CAN FD networks and allows bit rates up to 8 Mbit/s. To achieve bit rates above 8 Mbit/s with CAN XL, a new transmitter concept has been needed. In the arbitration phase, the CAN SIC transmitter concept has been chosen and is called SIC mode and in the data phase the push-pull transmitter concept has been chosen, to achieve bit rates up to 20 Mbit/s. This mode of the transceiver is called FAST mode. During the ADS (arbitration to data sequence) phase of the CAN XL protocol, the transceiver changes from the SIC mode to the FAST mode. The mode change is controlled by the CAN XL protocol controller via the TxD pin. The impedance in the FAST mode is, as well as in active-recessive SIC mode, 100 Ohm. But due to the alternating waveforms, the symmetry of the transmitted bits is better than in the SIC mode and allows bit rates up to 20 Mbit/s.

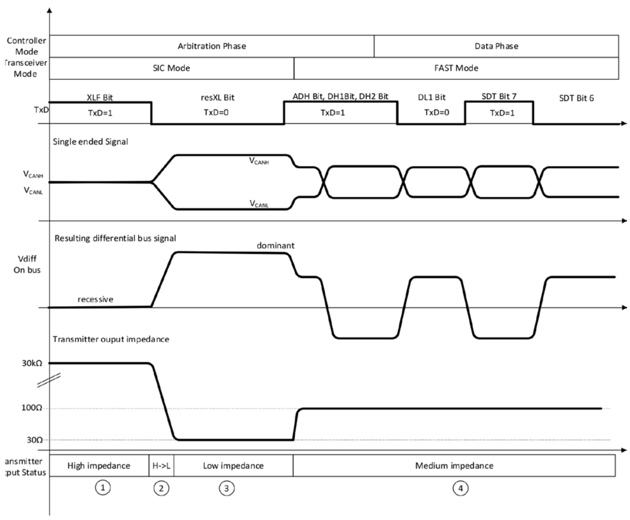



*Figure 3: Transmitter impedance characteristic during the SIC to FAST mode transition (Source: Infineon)*

In Figure 3, the SIC XL transmitter impedance characteristic during the SIC mode to FAST mode transition and in the FAST mode is shown. The transition starts with a change from dominant level to level_0 and afterwards to level_1. The change from dominant to level_0 is done to get the same voltage swing like in the FAST mode. Otherwise, the ringing due to the higher voltage swing must be analyzed in the transition phase separately. In Figure 4, the mode transition from the FAST mode to SIC mode at the end of the data phase is shown. During the complete FAST mode phase, which is identical with the data phase, the impedance is constant at 100 Ohm and matches with the wire impedance.

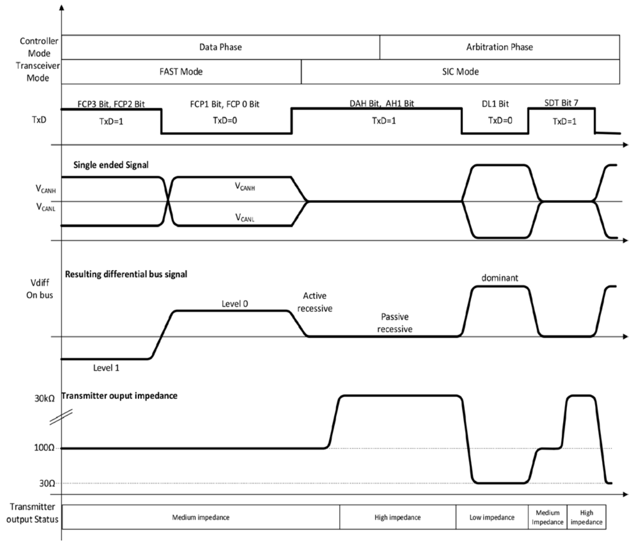



*Figure 4: Transmitter impedance characteristic during the FAST to SIC mode transition (Source: Infineon)*

The mode change from the SIC mode to FAST mode is controlled by the CAN XL controller via the TxD pin. During the arbitration phase the TxD signals are the same like for all other kinds of transceivers. TxD-high controls recessive level on the network and TxD-low controls the dominant level on the network. In the FAST mode, the controller transmits PWM-coded (pulse-width modulation) symbols to the transceiver. The length of the PWM symbols can vary between 50 ns and 200 ns. If a transceiver detects this PWM symbol, the transceiver changes the mode from SIC to FAST and if no symbols are detected anymore the transceiver switches back to the SIC mode. The duty cycle of the PWM symbol represents the level, which will be transmitted to the network. If the duty cycle is less than 50 %, this represents a logical 0 and level_0 (positive differential signal) is transmitted to the network lines. If the duty cycle is above 50 %, this represents a logical 1 and level_1 (negative differential signal) is transmitted to the network.

Not only the transmitting transceiver is controlled during the data phase with PWM signals, also the receiving transceiver uses the PWM signal to switch into the FAST mode. In the FAST mode, the receiver threshold is set to 0 V instead of 700 mV in the SIC mode. This guarantees that the CAN XL controller and the CAN SIC XL transceiver are always in the same mode. There is no mismatch due to errors, for example. ▷

## How to verify CAN XL networks

The most critical scenarios in the CAN XL data frame are:
- the transition from SIC mode to FAST mode,
- a burst of short bits, and
- a short bit after a long level_0 or level_1 phase (in maximum eleven bits according to the stuff-bit rules) with the opposite level.

During the ADH (arbitration to data high) bit, the transmitter switches from dominant to level_0 and afterwards to level_1 and, in parallel, all receiving nodes change the receiver thresholds. This happens by PWM-coded symbols sent from the CAN XL controller to the TxD pin of the transceiver. Before the PWM-coded symbol is detected on the TxD pin, the receiving nodes transmit a short dominant pulse followed by a shortened SIC phase. The requirement is that level_1 is stable before the SDT (service data unit type) field starts. Also, the length of the DL1 (data low) bit is of interest. The transition from the DH2 (data high) bit to the DL1 bit is used for resynchronization of the CAN XL protocol controller after the transition into the data phase. Also, level_0 should be achieved. In the SDT field a “0101” bit pattern has been chosen to analyze the impact of short bits in case of high bit rates.

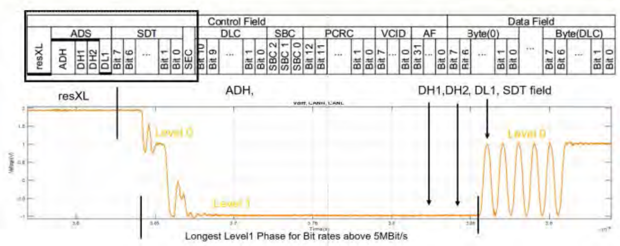



*Figure 5: SIC-mode-to-FAST-mode transition during the ADS field (Source: Infineon)*

Also, short bits after a high number of bits with the same level is a critical situation. One bit after eleven consecutive level_1 or level_0 bits (highest possible number of consecutive bits) should be used for the investigation. The target is to find out, how the bit length and the level of a short bit behaves after the longest possible phase of the same level in the CAN XL data frame.

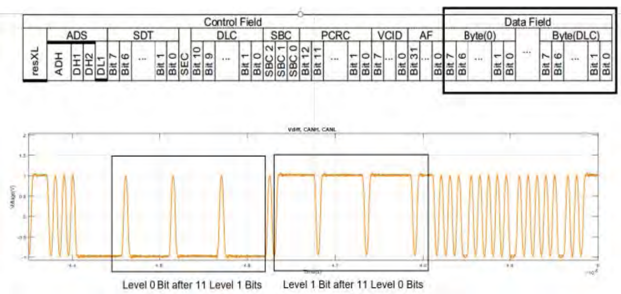



*Figure 6: Level_1 or level_0 bit after a long phase (Source: Infineon)*

For the verification, the bit-time lengths measured at the +100-mV and -100-mV thresholds should be used. The bit time should be close to the nominal bit time or a multiple of it. For high bit rates, the 0-V threshold can be used, too. Glitches with a length of 20 ns can be ignored.

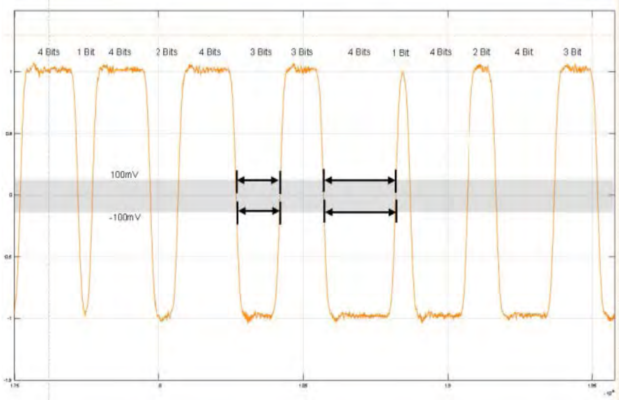



*Figure 7: Test criteria timing (Source: Infineon)*

Also, the level of short bits should be analyzed and should achieve in minimum 80 % of the nominal level. For an easier verification of CAN XL networks, CAN in Automation (CiA) members develop recommendations for an eye diagram in CAN XL data frames.

## Combinations of CAN protocol controllers and CAN SIC XL transceivers

CAN SIC XL transceivers can be used in combination with all variants of the CAN protocol. This allows a lot of opportunities in applications.

The CAN XL protocol handler supports all variants of the CAN protocol:
- CAN CC (classic) with 11-bit and 29-bit identifiers,
- CAN FD protocol with 11-bit and 29-bit identifiers,
- CAN FD light with 11-bit identifiers,
- CAN XL with 11-bit identifiers.

The CAN SIC XL transceiver features:
- the SIC mode (like a SIC transceiver according to ISO 11898-2:2024), and
- the FAST mode (for high bit rates in the CAN XL data phase).

This flexibility allows the combinations of CAN SIC XL transceivers and CAN protocol controllers as shown in Table 1.

*Table 1: Combination possibilities of CAN SIC XL transceivers (Source: Infineon)*

| CAN protocol version | Supported CAN SIC XL transceiver mode | | Maximal possible bit rate |
|---|---|---|---|
| | SIC mode | SIC mode and FAST mode | |
| CAN CC | + | - | 500 kbit/s |
| CAN FD | + | - | up to 8 Mbit/s |
| CAN XL (FAST mode disabled) | + | - | up to 8 Mbit/s |
| CAN XL (FAST mode enabled) | - | + | up to 20 Mbit/s |

In Table 2, all possible combinations of CAN transceivers, CAN FD transceivers, and CAN SIC (XL) transceivers are given.

The maximum bit rate, given in Table 2, depends on the network topology and might be lower. The maximum possible bit rate can be achieved in a point-to-point network with termination resistors on both sides as well as in many network topologies with very short stubs.

## MCU for digital car cockpits

Mediatek has based on Infineon's Traveo CYT4DN microcontroller family a digital cockpit system. The cockpit also comprises an entry-level Dimensity Auto SoC (system-on-chip) solution by Mediatek.


*The Traveo MCU dedicated for cockpit designs features up to eight CAN FD interfaces (Source: Infineon)*

The trend in the automotive industry is towards digital cockpits, with buttons and controls disappearing from the dashboard and being replaced by advanced displays. This implies also functional safety aspects. Traditionally, high-performance SoCs with hypervisors are used. However, the up-front investment required to get started with such a system is in the seven-figure range, and the license fees for the operating system and hypervisor add to the overall cost of the system, making it economically unattractive for mid- to low-end car models.

Infineon together with Mediatek have developed in cooperation with design house partners a cockpit solution. The Infineon MCU (micro-controller unit) works as a safety companion to the SoC by Mediatek to meet the ASIL-B (automotive safety integrity level) safety target for automotive clusters. The MCU monitors the content rendered by the SoC and takes over with reduced functionality in case of an error, in addition to normal companion functions such as communication with the vehicle network. The MCU features multiple CAN FD (flexible data rate), CXPI (Clock Extension Peripheral Interface), and LIN ports.

"A modern cockpit supports the driver and increases driving comfort for all vehicle occupants. That is why it is important to us that cost-optimized vehicle models can also be equipped with digital solutions. Together with Mediatek and our partners, we are pleased to pave the way for digital cockpits for all vehicles," said Ralf Koedel, working with Infineon. "The Traveo MCU incorporates our low-power flash memory, multiple high-performance analog and digital peripherals and enables the creation of a secure computing platform."

This launched cockpit solution supports a resolution of 1920 x 720 pixels for clusters and the in-vehicle infotainment display. The MCU is designed for automotive systems such as instrument clusters and head-up displays (HUD). It features a 2,5-D graphics engine, sound processing, two Arm Cortex-M7 cores for primary processing at up to 320 MHz and an Arm Cortex-M0+ core for peripheral and safety processing. The family also features a 720p GFX and a unique 327-ball BGA package. Memory options include 4 MiB of VRAM, 6 MiB of flash memory, and 768 KiB of RAM. *hz*

*Table 2: Combinations between CAN transceiver types and CAN protocol versions (Source: Infineon)*

| CAN protocol | CAN transceiver type | | | Maximal possible bit rate |
|---|---|---|---|---|
| | CAN transceiver | CAN FD transceiver | CAN SIC (XL) transceiver | |
| CAN CC | + | - | - | 500 kbit/s |
| | - | + | - | up to 5 Mbit/s |
| | - | - | + | up to 8 Mbit/s |
| CAN FD | + | - | - | 500 kbit/s |
| | - | + | - | up to 5 Mbit/s |
| | - | - | + | up to 8 Mbit/s |
| CAN XL | + | - | - | 500 kbit/s |
| | - | + | - | up to 5 Mbit/s |
| | - | - | + | up to 20 Mbit/s |

The CAN FD protocol and the CAN XL protocol allow a mixed communication in one network. If CAN FD protocol handler detects a CAN XL data frame, it stops frame detection after the FDF bit and changes into reintegration mode and is waiting until the end of the CAN XL data frame. The CAN XL controller is able to support both frame types. But for both protocols the same arbitration bit-time configuration is needed.

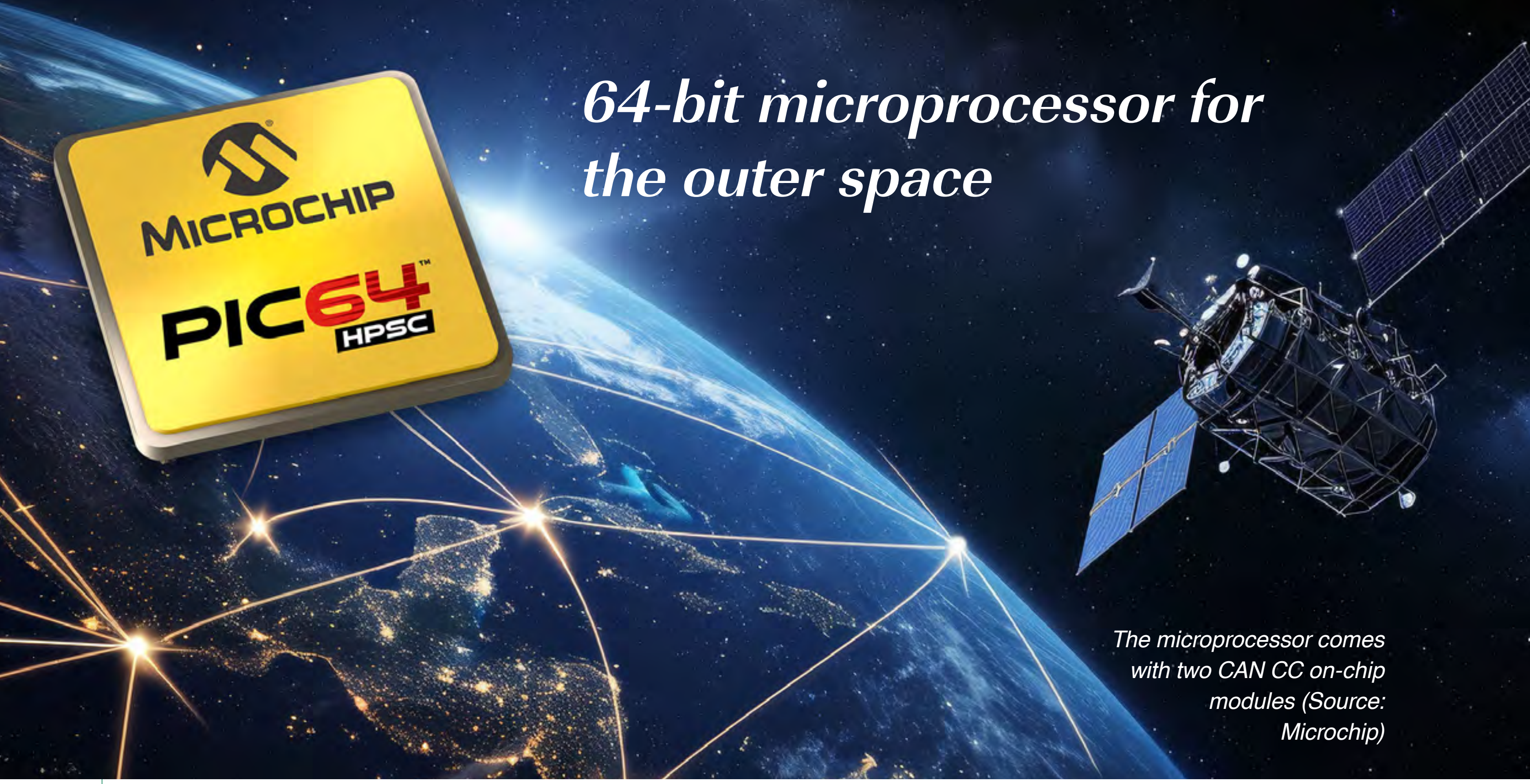
On the physical-layer side, there is the situation that in the CAN SIC XL transceiver FAST mode the differential bus levels might be below the receiver thresholds of CAN FD and CAN SIC transceivers. This has the consequence that from the physical layer point-of-view a reliable mixed protocol communication is only possible, if all nodes are using CAN FD or SIC mode only. The transceiver with the lowest possible bit rate defines the maximum bit rate in the network. An example: Some nodes in the network use CAN FD controllers and CAN FD or CAN SIC transceivers, while other nodes apply CAN XL controllers and CAN SIC XL transceivers. The communication is working properly, if CAN SIC XL transceivers are working in SIC mode only (FAST mode is disabled by configuration). For the arbitration and the data phase, this can be configured in the CAN XL controller. In such a network, CAN SIC transceivers enable bit rates of up to 8 Mbit/s as the maximum possible bit rate for CAN FD and CAN XL communication. The maximum bit rate can be reduced depending on the network topology. Reflections and ringing in the network can reduce the maximum achievable bit rate dramatically. The maximum possible bit rate for each network should be verified via simulation. ◀

*(This article is based on the 18th international CAN Conference (iCC) presentation by Magnus Hell. The complete paper is published in the 18th iCC proceedings 2024; CiA, Nuremberg.)*

**Author**

Magnus-Maria Hell
Infineon Technologies
info@infineon.com
www.infineon.com

# 64-bit microprocessor for the outer space




*The microprocessor comes with two CAN CC on-chip modules (Source: Microchip)*

*In 2022, NASA selected Microchip to develop a High-Performance Spaceflight Computing (HPSC) processor that could provide at least 100 times the computational capacity of previous spaceflight microprocessors. Recently, the chipmaker launched the HPSC family of PIC64 microprocessors.*

The world has changed dramatically in the two decades since the debut of what was then considered a trail-blazing space-grade processor used in NASA missions such as the comet-chasing Deep Impact spacecraft and Mars Curiosity rover vehicle. A report released by the World Economic Forum estimated that the space hardware and the space service industry is set to grow at a CAGR (compound annual growth rate) of 7 % from 330 billion US-$ in 2023 to 755 billion US-$ by 2035. To support a diverse and growing global space market with a rapidly expanding variety of computational needs, including more autonomous applications, Microchip has launched the first products in its planned HPSC family of PIC64 microprocessors.

Unlike previous spaceflight computing solutions, the radiation- and fault-tolerant PIC64-HPSC chips, which Microchip is delivering to NASA and the broader defense and commercial aerospace industry, integrate RISC-V CPUs augmented with vector-processing instruction extensions to support AI/ML (artificial intelligence/machine learning) applications. The MPUs (microprocessor units) also feature a suite of standardized interfaces including two CAN CC (classic) ports. Surprisingly, CAN FD (flexible data rate) is not supported. An ecosystem of partners is being assembled to expedite the development of integrated system-level solutions. This ecosystem comprises single-board computers (SBCs), space-grade companion components, and a network of open-source and commercial software partners.

"This is a giant leap forward in the advancement and modernization of the space avionics and payload technology ecosystem," said Maher Fahmi from Microchip. "The PIC64-HPSC family is a testament to Microchip's longstanding spaceflight heritage and our commitment to providing solutions built on industry-leading technologies and a total systems approach to accelerate our customers' development process."

The radiation-hardened (RH) microprocessor is designed to give autonomous missions the local processing power to execute real-time tasks such as rover hazard avoidance on the Moon's surface, while also enabling long-duration, deep-space missions like Mars expeditions requiring extremely low-power consumption while withstanding harsh space conditions. For the commercial space sector, the radiation-tolerant (RT) PIC64-HPSC RT is designed to meet the needs of low Earth orbit (LEO) constellations where system providers must prioritize low-cost over longevity, while also providing the high-fault tolerance that is vital for round-the-clock service reliability and the cybersecurity of space assets.

The launched microprocessors offer a variety of capabilities, many of which were not previously available for space-computing applications, including:

- Space-grade 64-bit MPU architecture with eight Sifive RISC-V X280 64-bit CPU cores supporting virtualization and real-time operation, with vector extensions that can deliver up to two Tera operations per second (int8) or one Tera floating-point operations per second (bfloat16); ▷

- High-speed network connectivity featuring a 240-Gbit/s TSN-Ethernet switch (time sensitive networking) for a 10-Gbit/s Ethernet connectivity, a scalable PCIe Gen 3, and a CXL 2.0 (compute express link) as well as RMAP-compatible Spacewire ports with internal routers;
- Low-latency data transfers providing remote direct memory access (RDMA) over converged Ethernet (RoCEv2) hardware accelerators to send data from remote sensors without burdening compute performance, which maximizes compute capabilities by bringing data close to the CPU;
- Platform-level defense-grade security implementing defense-in-depth security with support for post-quantum cryptography and anti-tamper features;
- High fault-tolerance capabilities supporting DCLS (dual-core lockstep) operation, the Worldguard hardware architecture for end-to-end partitioning and isolation, and an on-board system controller for fault monitoring and mitigation;
- Flexible power tuning providing dynamic controls to balance the computational demands required by the multiple phases of space missions with tailored activation of functions and interfaces.

“Microchip's PIC64-HPSC family replaces the purpose-built, obsolescence-prone solutions of the past with a high-performance and scalable space-grade compute processor platform supported by the company’s vibrant and growing development ecosystem,” said Kevin Kinsella from Northrop Grumman. “This innovative and forward-looking architecture integrates the best of the past 40-plus years of processing technology advances. By uniquely addressing the three critical areas of reliability, safety, and security, we fully expect the PIC64-HPSC to see widespread adoption in air, land, and sea applications.”

The improved computing-power capability can advance future space missions, from planetary exploration to lunar and Mars surface missions. Representatives from NASA, Microchip, and industry partners like Northrop Grumman shared insights about the HPSC technology and ecosystem at the IEEE Space Compute Conference 2024 in Mountain View, California, in July 2024.

Microchip’s inaugural PIC64-HPSC MPUs were launched in tandem with the company’s PIC64GX MPUs that enable edge designs in industrial, automotive, communications, IoT, aerospace, and defense segments. Samples will be available beginning of 2025. The chipmaker also offers an evaluation platform that incorporates the microprocessor, an expansion card, and a variety of peripheral daughter cards.

*hz*

# Evolution of (D)CAN

*This article describes the development history of CAN and emphasizes the differences between CAN CC (classic), CAN FD (flexible data rate), and CAN XL (extended data-field length) with respect to DCAN IP cores from Digital Core Design (DCD).*

(Source: Adobe Stock)

To understand the current state of CAN developments, we must look back at the beginnings. As a proud member of the DCD's R&D team, I am part of a group with extensive experience in developing CAN IP (intellectual property) cores. Let me guide you through a brief history of CAN, focusing on our contributions with DCAN, DCAN FD, and the latest DCAN XL IP cores.

## The dawn of (D)CAN

Developed out of the need for real-time control and reliable data transmission, CAN (now called CAN CC (classic)) emerged as a pioneer in networking technology. The protocol was officially released in 1986 at the Society of Automotive Engineers (SAE) conference in Detroit (USA). Bosch published the first CAN specification in 1991.

CAN CC features a maximum payload size of 8 byte and supports bit rates of up to 1 Mbit/s. Unlike in other communication networks, CAN frames are not dedicated to a certain receiving node. CAN enables broadcast communication using data/remote frames with a unique identifier. The identifier indicates the content and defines the network access priority. The CAN CC protocol supports two CAN data frame formats. Basically, they only differ in the length of the CAN identifier. The CAN CC base frame format (CBFF) supports the 11-bit CAN identifier and the CAN CC extended frame format (CEFF) supports the 29-bit CAN-identifier.

CAN CC is renowned for its sophisticated error detection and indication capabilities, high immunity to electromagnetic interference, automatic retransmission of corrupted frames, distinction between temporary errors and permanent node failures, and system-wide data consistency checks. Each CAN node has to check, if the network is idle before starting transmission. The network access is managed via a serial communication protocol called Carrier Sense Multiple Access/Collision Detection with Non-Destructive Arbitration. This protocol ensures that data/remote frame collisions are avoided through bit-wise arbitration without any loss of time.

The AUI (attachment unit interface) of CAN CC provides two possible states: "dominant" and "recessive." These states usually (but not always) correspond to logic "zero" and "one," respectively. The network logic employs a "wired-AND" mechanism, where dominant bits (zeros) overwrite recessive bits (ones). If all nodes transmit recessive bits, the network remains in a recessive state. However, if even a single node transmits dominant bits, it forces the network into a dominant state. This mechanism is crucial for arbitration.

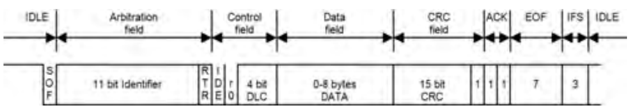



*Figure 1: CAN CC base frame format (Source: DCD)*

Figure 1 illustrates the CAN CC data frame in the CAN CC base frame format. It is composed of several fields, each with a specific role:

1. *Start of frame (SOF):* A single dominant bit indicating the beginning of a frame.
2. *Arbitration field:* Contains the identifier and the RTR (remote transmission request) bit. The RTR bit allows for the transmission of remote frames, which carry the data length code (DLC) for the requested frame but contain no data.
3. *Control field:* Includes the DLC, which defines the number of data bytes sent in the frame. It also contains ▷

either the IDE (ID extension) bit and one reserved bit for future use or two reserved bits, depending on the frame format. Reserved bits must be sent as dominant.
4. *Data field:* The actual data payload of the frame.
5. *CRC field:* Comprises the CRC (cyclic redundancy check) sequence and the CRC delimiter. The CRC is calculated over the destuffed bit sequence from the SOF to the end of the data field. The CRC delimiter is a single recessive bit.
6. *ACK field:* Consists of the ACK slot and delimiter. The transmitter sends two recessive bits, while the receiver acknowledges correct frame reception by sending a dominant bit during the ACK slot. If any node detects an error, it signals a corrupted message in the ACK field, preventing defective nodes from locking the network.
7. *End of frame (EOF):* Seven recessive bits signaling the end of a frame.

After the EOF, an interframe space of at least three bit times is required before the network becomes idle and ready for the next transmission.

## Error handling and synchronization

CAN CC handles errors using two types of error frames:
- *Active error flag:* Six consecutive dominant bits sent after detecting an error.
- *Passive error flag:* Six consecutive recessive bits sent after detecting an error.

These error flags increment internal error counters, which can cause defective nodes to enter an error-passive state and eventually a bus-off state, requiring reconfiguration before they can resume network traffic.

## Bit stuffing and bit timing

To maintain data integrity, CAN CC employs bit stuffing. After five consecutive bits of the same value, the transmitter inserts a bit of the opposite value. Error and overload flags are the only exception, containing six consecutive bits of the same value. The bit symbols are NRZ-coded (non-return-to-zero). CAN CC synchronizes frame transfers by aligning all nodes at the start of each frame with the first falling edge of the SOF bit, a process known as hard synchronization. To ensure accurate sampling throughout the entire data/remote frame, CAN nodes re-synchronize on each recessive-to-dominant edge, necessitating bit stuffing.

The CAN CC physical coding sublayer (PCS) divides the bit time into four non-overlapping time segments to manage synchronization and compensate for signal delays. Each time segment is constructed from an integer multiple of a time quantum ($t_q$). The time quantum is the smallest discrete timing resolution used by the CAN CC node.

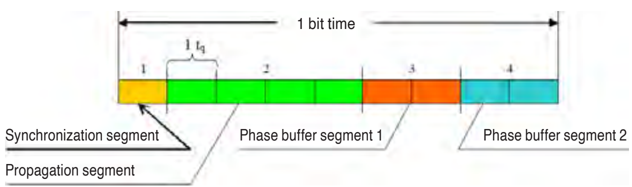



*Figure 2: CAN CC bit timing (Source: DCD)*

Figure 2 shows the CAN CC bit timing:
1. *Synchronization segment:* This 1-$t_q$ segment ensures that any bit state change between the previous and current bit occurs within this segment. It allows receiving nodes to synchronize with the network state change.
2. *Propagation segment:* Compensates for signal delays across the network and must be at least one $t_q$.
3. *Phase buffer segment 1:* Used to compensate for edge phase errors. It must be at least one $t_q$, but can be lengthened during resynchronization to account for oscillator tolerances among different CAN nodes. The extent of lengthening is limited by the synchronization jump width (SJW).
4. *Phase buffer segment 2:* Also used to compensate for edge phase errors. It must be at least one $t_q$, but can be shortened during resynchronization to account for oscillator tolerances. The extent of shortening is also limited by the SJW.

The sample point is the moment, when the network level is evaluated and interpreted as the status of the respective bit. It is located between the phase buffer segment 1 and phase buffer segment 2.

In 2024, Digital Core Design is celebrating its 25th anniversary. Founded in 1999 and after five years, DCD's team has introduced the DCAN IP, a CAN CC IP core. Since then, DCAN has been integrated into millions of vehicles worldwide. Thus, the "CAN journey" lasts for DCD for more than two decades. However, CAN CC has limitations, such as a small payload and a limited transmission speed. These drawbacks led to the development of CAN FD (flexible data rate), which addresses these issues and enhances the protocol's capabilities.

## Enter CAN FD

The next iteration of CAN technology introduced the CAN FD (flexible data rate) frame format, which supports bit rates higher than 1 Mbit/s, reaching up to 8 Mbit/s, and allows payloads up to 64 byte. CAN FD maintains full backward compatibility with CAN CC frames, with the CAN FD frames distinguished by the FDF (flexible data format) formerly reserved in CAN CC.

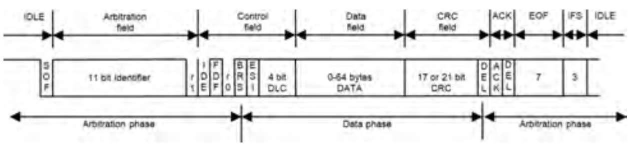



*Figure 3: CAN FD base frame format (Source: DCD)*

Figure 3 shows the CAN FD base frame format. While the order and number of fields in the frame remain the same as in CAN CC, the lengths of certain fields have been adjusted to accommodate the new features and enhancements. The key features and improvements include:
- *Bit rate switch (BRS) bit:* This bit determines whether the bit rate is switched within the frame. By enabling higher bit rates, CAN FD can transmit more data within the same time frame, preventing larger data packets from overloading the network.

Semiconductors

- *Error state indicator (ESI) bit:* This bit indicates whether the sender (transmitter) of the message is in an active error node or a passive error node, providing better error management.
- *Extended data length code (DLC):* The values encoded by the DLC have been extended to indicate longer data payloads. This allows for more efficient data transmission, especially for applications requiring larger payloads.
- *CRC polynomials:* The CRC adjustments ensure data integrity across varying payload sizes.
  - 17-bit CRC: Applied to CAN FD data frames with a data field up to 16 byte.
  - 21-bit CRC: Utilized for larger data fields in CAN FD data frames.

The rest of the frame structure remains similar to CAN CC, allowing CAN FD devices to connect to the same physical network, including use of the same transceivers, connectors, and cables. CAN FD does not support remote frames.

In 2016, to keep pace with the rapidly evolving automotive industry, DCD has designed the DCAN FD based on the Bosch CAN FD specification and the ISO 11898-1:2015 standard. While the Bosch CAN FD was published in 2012 and the ISO 11898 in 2015, DCD's developments followed closely, ensuring compliant implementation. The DCAN FD also underwent the testing against the ISO 16845-1 conformance test specification. Building upon the base of DCAN IP, the DCAN FD incorporates the new CAN FD features. Despite not being a completely new product, but rather an evolution of the DCAN IP, it achieved significant market success due to its enhanced capabilities and reliability.

The DCAN FD was designed with functional safety in mind, meeting the requirements of ISO 26262. It incorporates essential safety mechanisms to ensure the appropriate functional safety level, developed as a Safety Element out of Context (SEooC), and is ready for ASIL-B (automotive safety integrity level) or higher applications. This was the first DCD's IP designed with functional safety compliance. Leveraging our knowledge and experience, we advanced the DCAN FD to the DCAN FD Full. This IP brought following improvements:

- *User experience:* Enhanced user experience with more intuitive interfaces.
- *Hardware-filtering capabilities:* Support for up to 128 base ID filters and up to 64 extended ID filters.
- *Data throughput:* Increased data throughput with two configurable receive FIFOs, up to 32 dedicated transmit buffers, and configurable transmit buffers that can operate in FIFO or queue modes.
- *Event logging:* Dedicated transmit event FIFO and time-stamping according to CiA 603.
- *Data-processing time:* Optimized data processing with two clock domains – one for the CAN clock and another for the CPU clock.
- *Functional safety enhancements:* Further safety improvements ensuring compliance with the latest standards.

At that time, the DCAN FD Full represented the most advanced CAN IP. However, the time marched on, and we faced new challenges: Accommodating larger data packets and achieving faster transmission speeds to meet future demands.

## The advent of CAN XL

As we stepped into 2024, the automotive industry welcomed the advent of CAN XL (extended data-field length), the latest advancement in CAN technology. At DCD, we have announced the release of DCAN XL, which conforms to the updated ISO 11898:2024 standards and the CiA 610-1 specification (in the meantime withdrawn by CiA). This CAN solution represents the pinnacle of DCD's recent achievements and is now available on the market.

The DCAN XL builds on the legacy of our previous CAN solutions and integrates seamlessly into modern automotive networks. It supports the extended payload capacity of up to 2048 byte and can achieve bit rates up to 20 Mbit/s, bridging the gap between CAN FD and Ethernet. DCAN XL also enables to tunnel and to map entire Ethernet frames, thus enhancing data throughput without compromising network timing. At the fifth CAN XL plugfest organized by CAN in Automation (CiA) in Baden-Baden (Germany), DCAN XL has successfully passed the performed interoperability tests within a real CAN XL network.

The key features and enhancements of DCAN XL include:

- *Backward compatibility:* The DCAN XL controller can handle CAN CC and CAN FD frames, ensuring seamless integration with existing systems.
- *Extended payload and bit rates:* With payloads up to 2048 byte and bit rates up to 20 Mbit/s (when using CAN SIC XL transceivers and appropriate topologies), DCAN XL supports higher data transfer requirements.
- *Enhanced CAN-ID field:* The CAN-ID field is divided into an 11-bit priority field and a 32-bit acceptance field (AF), providing improved frame prioritization and acceptance filtering.
- *Protocol-embedded configuration:* CAN XL introduces new configuration options, such as disabling error signaling and enabling PWM (pulse-width modulation) coding, which allows for higher bit rates depending on the physical network design. While PWM is used, the network changes to push-pull type.
- *Improved reliability:* The use of two CRC fields – the 13-bit preface CRC (PCRC) in the control field and the 32-bit frame CRC (FCRC) in the CRC field – ensure enhanced error detection and data integrity. These CRCs can detect any five randomly distributed bit-errors, offering a Hamming distance of 6.

The CAN XL frame structure has been extensively updated compared to CAN CC and CAN FD, yet maintaining backward compatibility. DCD remains committed to provide solutions that meet the evolving demands of the automotive industry.

| SOF | Arbitration Field | Control Field | Data Field | CRC Field | ACK Field | EOF |
|---|---|---|---|---|---|---|

*Figure 4: CAN XL frame format (Source: DCD)*

The Figure 4 to Figure 8 show the CAN XL data frame and its fields in detail. Described are only the fields newly included in CAN XL.

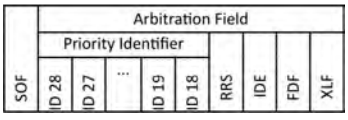



*Figure 5: CAN XL arbitration field (Source: DCD)*

In the CAN XL data-frame structure, several key bits play crucial roles in determining frame types and priorities. The priority identifier sets the priority for CAN XL data frames, ensuring efficient frame handling based on priority levels. The RRS (remote request substitution) bit, is located at the same bit position as the RTR (remote transmission request) bit in CAN CC data frames and the same as the RRS bit in CAN FD data frames. This ensures consistency and compatibility across different CAN generations. The XLF (extended length frame) bit differentiates between CAN FD and CAN XL data frames. In a CAN XL data frame, the XLF bit is always recessive. When the XLF bit is recessive, the FDF (frame data field) bit is also recessive. This configuration is crucial for distinguishing CAN XL data frames from other frame types within the CAN network.

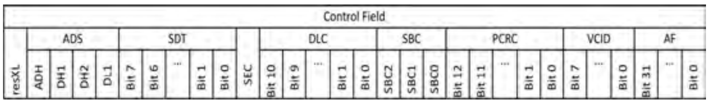



*Figure 6: CAN XL control field (Source: DCD)*

The resXL bit is reserved for future expansion within the protocol's framework. The arbitration to data sequence (ADS) serves dual purposes:

- Transitioning the bit rate from the nominal bit rate to the CAN XL data bit rate;
- Shifting the CAN transceiver mode from arbitration mode to either data TX mode or data RX mode.

ADS comprises ADH, DH1, DH2, and DL1 bits, as shown in Figure 6. ADH, the final nominal bit time before the beginning of the XL data phase, is transmitted as a recessive bit. Subsequent bits, DH1 and DH2, mark the beginning of the XL data phase, are also transmitted recessively. The bit rate shift occurs precisely between ADH and DH1. Receiver synchronization occurs at the transitions from XLF bit to the preceding resXL bit, and from DH1 and DH2 bits to DL1 bit.

SDT (service data unit type), an 8-bit value, originates from the LLC (logical link control) frame, while the SEC (simple extended content) bit is also passed from the LLC frame.

The data length code (DLC), spanning 11 bits (ranging from 0 to 2047), correlates to a data length between 1 byte and 2048 byte. The stuff bit count (SBC), a 3-bit value, indicates the quantity of dynamic stuff bits within the arbitration field, with values ranging from 0 to 3 (Gray-coded). Preface-CRC (PCRC) sequence, derived from CRC, is calculated over the bit stream encompassing the arbitration field, SDT, SEC bit, DLC, and SBC. Dynamic stuff bits, including up to three preceding the FDF bit, are factored into the calculation, whereas static bits such as SOF, IDE, FDF, XLF, resXL, ADS, and fixed stuff bits are excluded.

VCID (virtual CAN network ID), an 8-bit value, and AF (acceptance field), a 32-bit value, are both passed from the LLC frame.

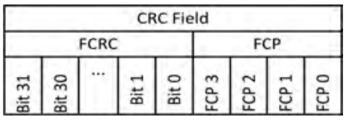



*Figure 7: CAN XL CRC field (Source: DCD)*

The CRC field contains the frame-CRC (FCRC) sequence and the format check pattern (FCP). The relevant bit stream for the CRC calculation is the bit stream consisting of arbitration field, control field, and data field. It excludes the same static bits as the PCRC. Both dynamic and fixed stuff bits are excluded from the CRC calculation.

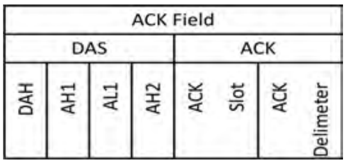



*Figure 8: CAN XL ACK field (Source: DCD)*

The ACK field encompasses the DAS (data to arbitration sequence), the ACK slot, and the ACK delimiter. Notably, DAH, AH1, AH2, ACK slot, and ACK delimiter bits are transmitted as recessive, while AL1 bit stands as dominant. The DAS serves a dual role:

- Transitioning the bit rate from the XL data bit rate to the nominal bit rate.
- Shifting the CAN transceiver mode from either data TX mode or data RX mode to arbitration mode, if the mode transition occurred in the preceding ADS.

DAS comprises DAH, AH1, AL1, and AH2 bits. DAH, the initial bit following the nominal bit time, marks the end of the XL data phase at the XL data bit rate.

## Conclusion

As we reflect the history of CAN, from its beginnings in the 1980s, CAN has remained at the forefront of innovation. The scope of CAN XL exceeds the possibilities of traditional automotive and non-automotive engineering, offering a range of applications that extend beyond conventional boundaries.

Understanding the history of CAN unveils the evolution of Digital Core Design (DCD) and (D)CAN Intellectual Properties (IPs). Our company commemorates its 25th anniversary this year – a quarter-century of innovation. DCD remains committed to provide solutions that meet the evolving demands of the automotive industry. ◀

**Author**

Robert Nawrath
Digital Core Design (DCD)
robert.nawrath@dcd.pl
www.dcd-semi.com

# Single-chip powertrain controller supports CAN XL

*Silicon Mobility, an Intel daughter company, has announced the OLEA U310 field programmable control unit (FPCU). The single system-on-chip (SoC) is able to replace up to six micro-controllers. It can manage simultaneously real-time control tasks and power as well as energy management functions in combination with functional safety and cybersecurity.*

*(Source: Silicon Mobility)*

The design allows original equipment manufacturers (OEMs) to break free from the conventions of EV (electric vehicle) domain controls and move to a highly integrated X-in-1 powertrain that delivers unmatched system performance. It helps to make EVs more energy efficient, lighter, and more cost-effective. The traditional embedded electric architecture is being reinvented to support a software-driven approach. This "software-defined vehicle" concept promises a more sustainable model for car development and a constantly updatable and evolving user experience. However, it requires powerful computational and control solutions that seamlessly integrate hardware and software.

The OLEA U310 is engineered to match the need for powertrain domain control in electrical architectures with distributed software. Built with a hybrid and heterogenous architecture, it surpasses the capabilities of traditional micro-controllers. Up to six micro-controllers can be replaced by a single OLEA U310 in a system combination, where the FPCU is controlling in parallel an inverter, a motor, a gearbox, a DC-DC converter, and an on-board-charger. The SoC embeds multiple software and hardware programmable processing and control units while seamlessly integrating functional safety and cybersecurity into its core design.

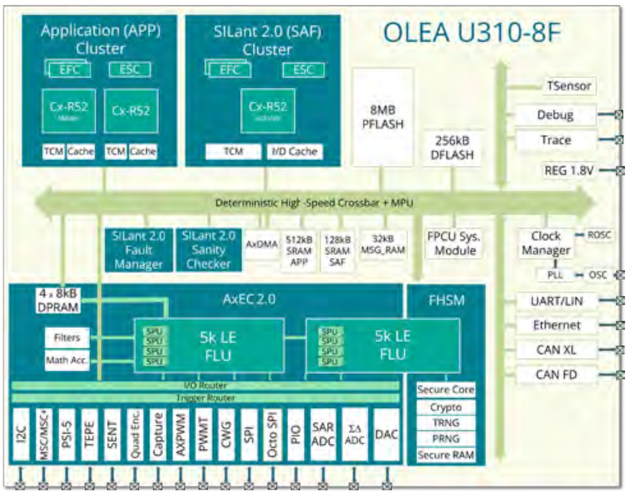


*Figure 1: The SoC comes with four CAN FD and one CAN XL interfaces (Source: Silicon Mobility)*

The OLEA U310 can control up to four traction inverters and their motor in parallel at a speed of 1000 kHz field-oriented control loop with high PWM (pulse-width modulation) precision of hundreds of picoseconds. In addition to the BoM (bill of materials) reduction, early figures show up to five percent energy-efficiency improvement, 25 percent motor downsizing for the same power, 35 percent less cooling need, and up to 30 times passive component downsizing.

The SoC comprising three Cortex-R52 cores supports the following functions:

- AxEC 2.0: The Advanced execution and Events Control is a data processing and real-time control unit based on programmable hardware and configurable peripherals supporting multiple parallel applications.
- SILant 2.0: The Safety Integrity Level (SIL) agent is a set of units and functionalities dedicated to the FPCU and the system functional safety ensuring ISO 26262 ASIL-D compliancy.
- FHSM: The Flexible Hardware Security Module is a sub-system dedicated to the cybersecurity of the FPCU integrating encryption/decryption accelerators and is compliant to EVITA Full and ISO 21434. It is combined with a hardware programmable cluster to support unidentified threats and strengthen security.

The chip features CAN connectivity. It implements four CAN FD controllers and one CAN XL controller as well as one Ethernet port. Additionally, it provides four LIN 2.3 interfaces as well as four SPI ports.

*hz*

# Visualization of terabytes of CAN data at low cost

*With a long-years expertise on CAN data acquisition, the author works closely with end users, typically OEM engineers, across diverse industries (automotive, heavy-duty, maritime, industrial). Here, he explains the integration of the company's data loggers with Grafana telematics dashboards through various data sources.*

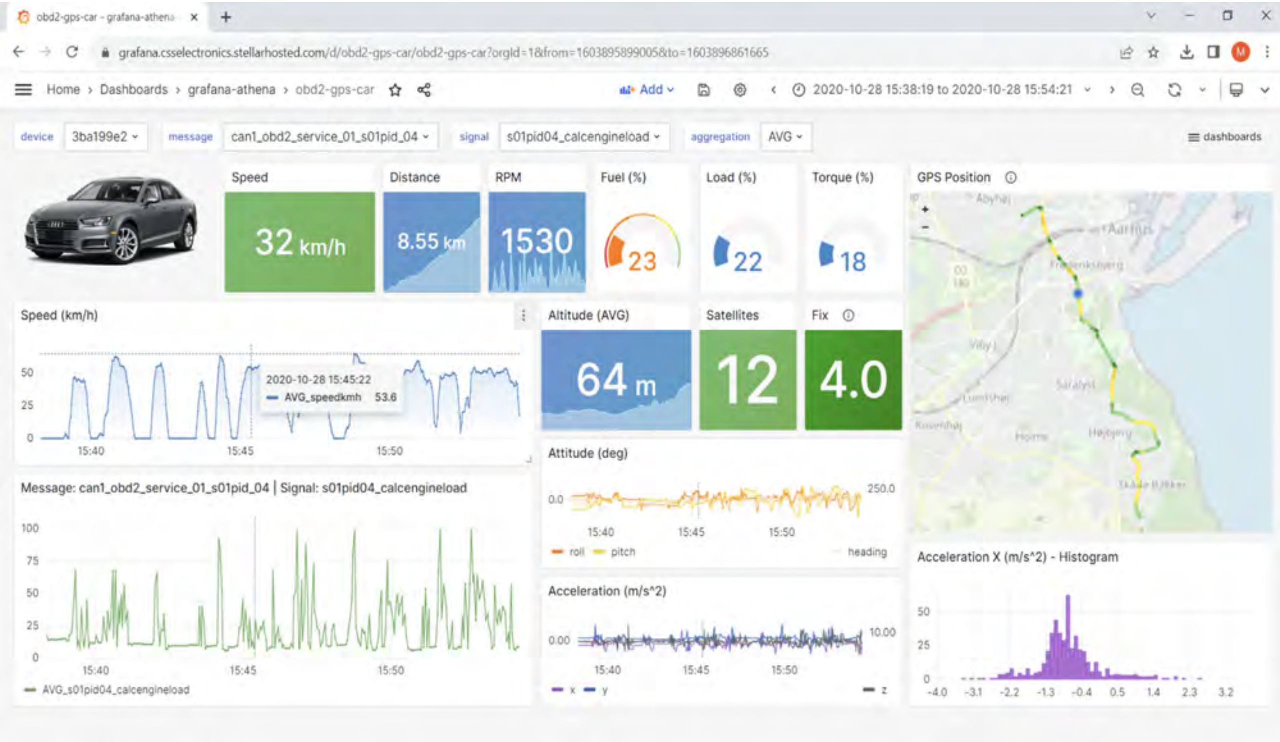



*Figure 1: Grafana dashboard playground (Source: CSS Electronics)*

CSS Electronics develops the CAN data loggers used e.g. by engineers at automotive and industrial manufacturers (OEMs) who need to monitor assets in the field for R&D, diagnostics, or predictive maintenance. A large share of the users need to visualize their data via Grafana dashboards. This article describes the work flow involved in achieving this. It also shows how the Grafana-Athena integration solves some of the unique challenges faced by users. Here, you can check out the company's public Grafana-Athena dashboard playground.

## Five key integration challenges to solve

***1) How to decode and prepare raw binary log files***

The CANedge data loggers record raw CAN data in a binary file format called MDF (*.MF4). This data consists of time-stamped "CAN IDs" and "databytes." In order for the users to make sense of the data, it needs to be "decoded" to human-readable form via software/API tools and a database CAN (DBC) file that contains information on how to interpret data from a specific application (e.g., a truck or car).

***2) How to enable low-cost dashboard visualization across TBs of data***

A single CANedge IoT device connected to e.g. a truck can generate more than 50 GB of data per month. As a result, users need to enable visualization across terabytes of data, which makes it prohibitively expensive to attempt to store the data in e.g. cloud-based time series databases. At the same time, data queries should be reasonably fast to enable users to effectively work with their data.

***3) How to retain the original message time-stamps***

A typical CAN network may comprise 10 to 100 unique CAN messages, each broadcast with a separate time raster. Some messages may be broadcast a few times per second, while others may be broadcast at e.g. 1000 Hz (and higher). The integration must retain these original time-stamps, enabling users to both look at the data across multiple months – and within a single second.

***4) How to enable end users to self-deploy everything***

CSS Electronics does not host servers or services. Rather, the manufacturer enables the end users to set up everything themselves. This is necessary as most end users are very ▷

strict on data sensitivity. It has also the benefit that end users pay a far lower cost for the services. However, it also requires that everything is simple to deploy.

***5) How to stay agnostic***

The CANedge devices enable agnostic data collection: Users can e.g. use a CANedge1 to log data to an SD card for offline usage, or a CANedge2/CANedge3 for data collection via WiFi/LTE to their own S3 bucket. The latter can be self-hosted via e.g. MinIO, or cloud hosted via e.g. AWS, Google Cloud, or Azure. In creating a Grafana dashboard integration, it is key to ensure that the solution can be adapted across these backends.

*Figure 2: CANedge data loggers (Source: CSS Electronics)*

## The solution: Grafana-Athena

To solve the above challenges, CSS Electronics now provides the 'Grafana-Athena' dashboard integration.

Here is a brief explanation of a typical Grafana-Athena setup:

1. A CANedge logger uploads raw CAN/LIN data to an AWS S3 'input bucket'
2. When a log file is uploaded it triggers a Lambda function
3. The Lambda function DBC decodes the data to Parquet files
4. The Parquet files are written to an AWS S3 'output bucket'
5. Grafana visualizes this 'data lake' via the Amazon Athena plugin
6. This setup can be auto-deployed in less than 15 minutes - with zero coding

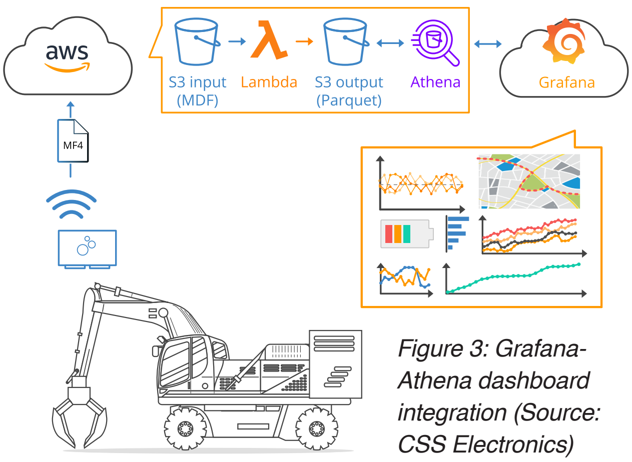



*Figure 3: Grafana-Athena dashboard integration (Source: CSS Electronics)*

## Details on the integration

The Grafana-Athena deployment is fully self-deployed by end users, including users that have never worked with Grafana or AWS. To enable this, CSS provides plug & play AWS CloudFormation stacks that end users can deploy via the company's step-by-step guides.

Users upload their DBC file(s) in their S3 input bucket along with a Lambda function zip. Next, they reference the CloudFormation stack template to deploy the Lambda function, output S3 bucket, Athena and Glue. Once the deployment is complete, users are presented with the relevant credentials for authenticating Athena from within their Grafana Cloud data source setup.

Importantly, end users do not have to worry about the actual construction of the Parquet data lake. This is all taken care of by CSS's premade AWS Lambda function, which also ensures that the data lake is e.g. optimally partitioned by date.

## Five key benefits of Grafana-Athena

Here some of the key benefits of the Grafana-Athena integration are highlighted:

***1) High query performance at low cost***

In the past, CSS enabled users to visualize their data via the company's Grafana-InfluxDB integration. InfluxDB offers high query speeds - but it quickly becomes expensive at larger data volumes. To illustrate this, the manufacturer has created a cost calculator for comparing the costs. If we take outset in a single CANedge device and 100 GB of queryable data, using the previous Grafana-InfluxDB integration would cost circa 150 USD/device/month. With Grafana-Athena, the equivalent cost would be about 3 USD/device/month (i.e. 95 % less). At the same time, the query speed is non-distinguishable between the two integrations for most practical use cases.

The reason why Grafana-Athena is so much cheaper is because the data is stored in an AWS S3 bucket at 0,023 $/GB/month, rather than a database. Further, Athena is a serverless solution that is paid-per-query (5 $/TB), making the combination suitable for visualizing large data volumes in a setup where the data is infrequently queried.

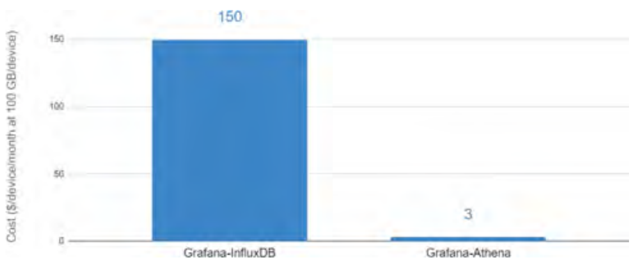



*Figure 4: Cost comparison of Grafana-Athena vs. Grafana-InfluxDB integration (Source: CSS Electronics)*

***2) Deploy in less than 15 min - with zero coding required***

The Grafana-Athena integration involves a lot of resources and AWS user policies that would be too complex to deploy ▷

for end users in practice. However, by preparing agnostic Lambda functions and AWS CloudFormation templates, CSS enables the user to deploy everything in minutes – without writing a single line of code.

***3) Seamlessly analyze data across years - or milliseconds***

The upload log files are automatically decoded and written to an AWS S3 Parquet data lake. The data lake is structured so that every CAN message is contained in a separate folder, which enables users to retain the original time-stamps of every message. In Grafana, users can then leverage standard SQL queries to define each dashboard panel - and seamlessly resample data on-the-fly. In practice this means that end users can effectively visualize data across years and GBs of data - and seconds later zoom in to visualize intra-second observations matching the original time-stamps at which the data was recorded on the CAN network.

***4) Multi-purpose data lake with SQL interface***

Deploying a Grafana-Athena integration opens up other use cases. For example, the S3 based Parquet data lake can be directly used for queries in Python/Matlab, while the Athena integration can be used in e.g. ODBC based Excel reports/dashboards. In other words, the Parquet data lake becomes a 'one-stop shop' for all data analysis requirements.

***5) Optionally deploy with Athena equivalents***

This article focuses on the Amazon Athena integration method. This is because the majority of CSS's end users already use AWS S3 to store data from their CANedge2/ CANedge3 devices. However, multiple equivalent setups can be created with similar implications, for example below:

- *Grafana-ClickHouse:* ClickHouse can be used to similar effect as Athena and offers an open-source alternative that can be self-hosted. This enables visualization from e.g. local disk or self-hosted S3 buckets like MinIO. Data can be visualized in Grafana using the ClickHouse data source plugin.
- *Grafana-BigQuery:* Some of the company's users leverage Google Cloud with S3 interoperability. Here, BigQuery offers a similar setup as Amazon Athena and integrates easily with Grafana via the BigQuery data source plugin.

## Example: Visualize data from prototype vehicle fleet

An automotive OEM engineer deploys the CANedge3 data logger to collect CAN data across 50 prototype vehicles for late-stage diagnostics purposes. The binary log files are automatically uploaded via 3G/4G cellular network to the company's own AWS S3 bucket. Here, the data is automatically processed by a Lambda function using the proprietary DBC file stored in the input S3 bucket, with the decoded data being output to another S3 bucket as Parquet files. The user can now query the data via custom Grafana dashboards, enabling powerful opportunities for diagnostics, benchmarking, and statistical analysis.

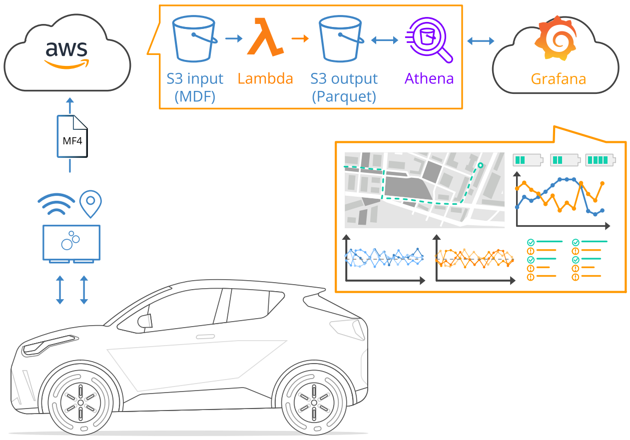



*Figure 5: Data visualization from a prototype vehicle fleet (Source: CSS Electronics)*

## Conclusion

The Grafana-Athena integration enables CSS Electronc's end users to reduce costs by more than 95 % in comparison to their previous integrations. It also serves as a blueprint for other companies that wish to enable Grafana dashboard visualization at low cost across large volumes of data. The complete information about Grafana-Athena integration can be found in the company's intro article. ◀

**Author**

Martin Falch
CSS Electronics
contact@csselectronics.com
www.csselectronics.com