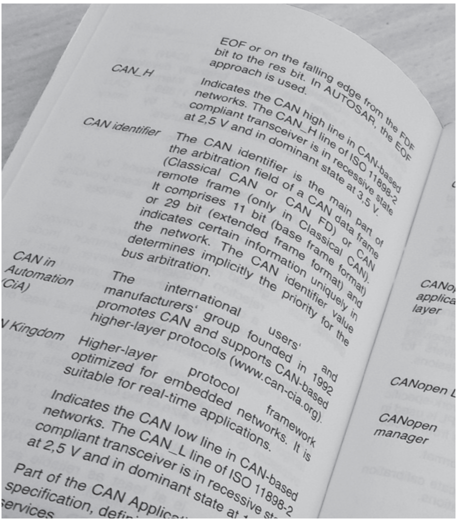
13th edition 2026

CANdictionary

Keywords ◆ Technical terms ◆ Standards



Explains vocabulary and abbreviations used in CAN technology

Covers CAN data link layers, CAN physical layers, and CAN-based higher-layer protocols

Includes a short history of CAN developments and application fields

CAN in Automation

international users' and manufacturer's group e. V.

CANdictionary

13th edition, 2026

Foreword

This dictionary briefly describes vocabulary and abbreviations used in CAN technology. It is not supposed to substitute any standard or specification. CAN newcomers can use the CANdictionary to understand technical articles, handbooks, etc. more easily without consulting standards and specifications.

The CANdictionary covers CAN CC (classic), CAN FD (flexible data rate), and CAN XL (extended data-field length) data link layers, CAN physical layers as well as several CAN-based higher-layer protocols. The editors have tried to include all relevant information. However, users might look for some entries that the editors have not considered or find entries that are not properly described.

With regard to a more comprehensive successor edition of the CANdictionary, the editors would appreciate comments and improvement proposals (pr@can-cia.org).

The editors

Introduction

The internationally standardized, serial bus system controller area network (CAN) was originally developed for in-vehicle networking. In 1986, the CAN data link layer protocol was introduced at the SAE conference in Detroit. In 1993, the CAN protocol and the high-speed physical layer were internationally standardized in the monolithic ISO 11898 standard. In 2003, this document was revised and splitted into two parts. Today, this ISO standard series comprises the following parts:

- ISO 11898-1: CAN data link layer and physical coding sublayer
- ISO 11898-2: CAN high-speed physical medium attachment (PMA) sublayer
- ISO 11898-3: CAN fault-tolerant transceiver
- ISO 11898-4: Time-triggered CAN

The CAN CC (classic) data link layer protocol uses one bit rate for the entire frame. Introduced in 2012, the CAN FD (flexible data rate) data link layer protocol uses a second higher bit rate for the data phase, which accelerates the data transfer. In addition, the CAN FD protocol supports longer data fields (up to 64 byte). The CAN FD data link layer protocol does not support CAN remote frames. The CAN XL (extended data-field length) data link layer, the 3rd CAN protocol generation, so to speak, was released by CiA in 2021. Since 2024, it is internationally standardized in ISO 11898-1 and ISO 11898-2.

The CAN data link layers are the basis of different standardized higher-layer protocols. For commercial-vehicle diesel-engine and powertrain applications, the SAE J1939 series has been introduced in the mid of 90ties. At the same time, DeviceNet (IEC 62026-3) for factory automation and CANopen (EN 50325-4) for embedded control systems were developed. Other standardized higher-layer protocols are the ISO Transport Protocol (standardized in ISO 15765-2) and the unified diagnostic services (UDS) on CAN (standardized in ISO 14229-3) for vehicle diagnostic purposes, the ISO 11783 series for agriculture and forestry machines (also known as Isobus), and the ISO 11992 series for truck to trailer communication. The NMEA 2000 application layer for maritime navigation equipment has been internationally standardized as IEC 61162-3. Several other ISO and IEC standards are based on CANopen and J1939 to specify dedicated device interfaces or complete system approaches.

CAN networks are used in a broad range of application fields. In-vehicle networking in any kind of transportation systems (cars, trucks, locomotives, ships, aircrafts, and even satellites) is the major application. Other applications include industrial machine control, factory automation, medical devices, laboratory automation, lift and door control, power energy generation and distribution as well as many other embedded control systems.

The CAN physical layers using differential voltages are robust against disturbances. The CAN CC and the CAN FD data link layer protocols are able to detect any single bit error. Multiple bit errors are detected with a very high probability. CAN XL features a Hamming distance of 6, meaning five randomly distributed bit errors in the data frame can be detected. The higher-layer protocols and profiles support interoperability of devices up to the level of off-the-shelf plug-and-play.

---

acceptance field (AF)
The 32-bit field of the CAN XL data frame contains address information. This can be a node address or a content indicator.

acceptance filter
The acceptance filter in CAN controller implementations is used to select CAN data and remote frames, which are received depending on the assigned identifier. Most CAN controllers provide a hardware acceptance filter that filters CAN data and remote frames assigned with a specific identifier or a range of identifiers. The user-settable filter unburdens the microcontroller from the task of acceptance filtering.

ACK
See *acknowledgement.*

acknowledgement (ACK)
Receivers check the consistency of the received DF and RFs, acknowledge a consistent frame, and when error signaling is enabled, flag an inconsistent frame by means of an EF.

acknowledgement (ACK) delimiter
The second bit of the ACK field. It is by definition recessive. A dominant state of this bit is regarded as a form error and causes the transmission of an error frame.

acknowledgement (ACK) error
If the frame-transmitting node detects the recessive state in the acknowledgement slot, it regards that as acknowledgement error condition. Acknowledgement errors do not cause a bus-off condition. Normally they occur if the network consists of just one node and this node starts transmission of CAN data or remote frames.

acknowledgement (ACK) field
The ACK field is made of two bits: ACK slot and ACK delimiter.

acknowledgement (ACK) slot — The first bit of the ACK field. It is transmitted recessively by the data or remote frame-sending node. All receivers transmit it dominantly, if no CRC error is detected. If the node producing the data frame or remote frame detects this bit as dominant, it knows that there is at least one node that has received the data frame or remote frame correctly.

active error flag — The active error flag is the first part of the active error frame made up of six consecutive dominant bits.

active error history — This data object in CANopen FD provides all occurred errors, from start of recording till end of recording, independent of the intermediate system down times, and only limited by the available memory, and not limited by the layout of the object dictionary. The listed errors are communication- or application-related.

active error list — This data object in CANopen FD provides only those errors that are currently present in that CANopen FD device. The listed errors are communication- or application-related.

active recessive — This is an intermediate high-speed physical medium attachment (HS-PMA) output drive with a dedicated lower than nominal impedance at transitions from dominant state or level_0 state towards the passive recessive state. It has a dedicated duration.

ADH — *See arbitration to data high bit.*

ADS — *See arbitration to data sequence.*

AF — See *acceptance field.*

application layer — This layer is the 7th layer in the OSI (open systems interconnection) reference model. It provides communica-tion services to the application program.

application objects — Application objects are parameters of the application program visible at the application programming interface (API).

application profile — Application profiles specify all communication objects and application objects of devices connected to the network.

arbitration field — The arbitration field in CAN CC and CAN FD is made of the 11-bit or the 29-bit identifier and the RTR bit (in CBFF and in CEFF) or the RRS bit (in FBFF and in FEFF). The arbitration field of the extended data frames (CEFF and FEFF) also contains the SRR and the IDE bits. In CAN CC and CAN FD, the identifier is used for bus arbitration and content addressing purposes. In CAN XL these functions are separated, such that the priority of the frame is contained in the 11-bit priority ID, whereas the addressing information is relocated to the 32-bit acceptance field (AF).

arbitration mode — This is the operating mode of the physical coding sublayer (PCS), in which it is allowed that dominant bits can overwrite recessive bits.

arbitration phase — The arbitration phase indicates those parts of the CAN FD data frame that utilize the bit timing as specified for CAN CC. The arbitration phase starts with the SOF and lasts till the sample point of the BRS bit. In addition, the final part of the CAN FD data frame, starting with the sample point of the CRC delimiter till EOF completes the arbitration phase. The interframe space (IFS) is also transmitted with the arbitration bit time. During the arbitration phase the nominal bit time is used.

arbitration to data high bit (ADH) — The arbitration to data high bit (ADH) is one of the four bits of the ADS field. It is located at the end of the arbitration phase (bit rate up to 1 Mbit/s) of the control field in the CAN XL frame. Through this bit, the transceiver is switched from slow mode to fast mode. The ADH bit is a recessive bit but at first the bus level stays dominant, and after the mode change command from the CAN XL controller, the transceiver changes from dominant to recessive level with the SIC performance and after a defined time the bus level changes from recessive level to dominant level.

arbitration to data sequence (ADS) The arbitration to data sequence (ADS) is a part of control field. It has two purposes: Switching the bit rate from nominal bit rate (up to 1 Mbit/s) to the XL data bit rate (up to 20 Mbit/s), and switching the CAN transceiver mode from arbitration mode to data TX/RX mode. It is enabled when an appropriate CAN transceiver supporting mode switching is connected. The ADS consists of four bits ADH, DH1, DH2, and DL1. The ADH bit is the last bit of arbitration phase, with nominal bit time before the beginning of the XL data phase with XL data bit rate. The subsequent bits DH1, DH2 and DL1 are the first bits of the XL data phase and are transmitted as recessive bits with the data bit rate.

Arinc 825-1 This specification by aeronautical radio (Arinc) specifies a higher-layer protocol dedicated for in-aircraft networking. It is designed similar to the CANaerospace higher-layer protocol; however, it utilizes a 29-bit identifier. The physical layer is compliant with ISO 11898-2.

Arinc 826 This specification describes the downloading of software parts to line replaceable units (LRUs). The specification is intended for avionic programmable devices.

assembly object This DeviceNet object describes the content of the I/O message.

asynchronous PDO Asynchronous PDO is the historical term for event-driven PDO in CANopen.

attachment unit interface (AUI)	This is the interface between the physical coding sublayer (PCS) as specified in ISO 11898-1 and the physical medium attachment (PMA). PCS and PMA are sublayers of the CAN physical layer.
AUI	See *attachment unit interface.*
AUI state	This is one of the two complementary logical states: dominant or recessive. The dominant state represents the logical 0, and the recessive state represents the logical 1. The terms „dominant“ and „recessive“ for bit values of the medium access control (MAC) frame are used independent of the transceiver mode. In FAST mode the logical 0 is represented by level_0 and the logical 1 is represented by level_1.
auto bit-rate detection	This is a procedure for CAN controllers in listen-only mode with the objective to test several pre-defined bit timing settings, till a complete CAN data or CAN remote frame is received. These bit timing settings are used for further operation.
automatic retransmission	Corrupted data frames and remote frames are retransmitted automatically after the error frames are successfully transmitted.

bandwidth — The bandwidth is the value, which denominates the size of information transmitted in a defined time unit.

base frame format — The base frame format uses 11-bit identifiers in CAN CC data and remote frames (CBFF) as well as in CAN FD data frames (FBFF).

base identifier — The base identifier is made up of a 11-bit identifier, which represents the priority in CBFF and FBFF.

BasicCAN — A term used in the early days of CAN describing an implementation, which uses just two receive frame buffers filled and read out in a ping pong method.

basic cycle — In TTCAN the basic cycle always starts with the reference message followed by a number of exclusive, arbitration or free windows. One or more basic cycles make the TTCAN matrix cycle.

BCH — See *Bose-Chaudhuri-Hocquenghem code.*

bit encoding — This is the representation of a bit on the physical layer. In CAN CC and CAN FD the bits are encoded as NRZ coding. CAN XL nodes can use CAN SIC XL transceivers to support bit rates of 20 Mbit/s and beyond. These transceivers have three modes and can be switched from arbitration mode into the dedicated operating mode (data TX mode and data RX mode) for the XL data phase. This mechanism is called “transceiver mode switching”. To maintain the transceiver mode switching, CAN XL protocol provides the optional PWM encoding function to be linked to the PWM decoding function (see *ISO 11898-1*).

bit error — If a bit is transmitted as dominant and received as recessive or vice versa, this is regarded as a bit-error condition, which causes an error frame transmission in the next bit time. If a recessive transmitted bit is overwritten by a dominant bit in arbitration field and acknowledgement slot, this is not a bit error.

bit monitoring — All transmitting CAN controller chips listen to the bus and monitor the bits that are transmitted by them.

bit rate — This is the number of bits per time during transmission, independent of bit representation. The bit rate in CAN CC is limited to 1 Mbit/s. In CAN FD and CAN XL, the bit rate is higher in the data phase. In the arbitration phase, the bit rate is still limited to 1 Mbit/s.

bit-rate switch (BRS) — At the sample point of the bit-rate switch (BRS) bit in CAN FD data frames, the data phase starts. This means that here the CAN controllers switch to a higher bit rate. The BRS bit exists in CAN FD data frames only.

bit resynchronization — Due to local oscillator tolerances, it happens that one node loses the bit synchronization. Each recessive-to-dominant edge causes the CAN controller to resynchronize itself to the received falling edge.

bit stuffing — This is a frame coding method using injections of bits into a bit stream to provide bus state changes required for periodic resynchronization when using a non-return-to-zero (NRZ) bit representation. Two types of bit stuffing exist: dynamic bit stuffing and fixed bit stuffing. The transmitter adds stuff bits into the outgoing bit stream and receivers de-stuff the data frames and the remote frames.

bit symmetry This is the shortening and lengthening of a single bit time due to effects on the physical medium and in the PMA implementing silicon.

bit time This is the duration of one bit.

bit timing Settings within the bit timing registers in the CAN controller implementation, to derive the duration of a single bit from the local oscillator frequency, via the bit-rate prescaler, assembled in multiples of one time quanta.

boot-up message This CANopen communication protocol is transmitted whenever a CANopen device finalizes NMT state initialization and enters the NMT state pre-operational.

Bose-Chaudhuri-Hocquenghem code (BCH) In coding theory, the Bose-Chaudhuri-Hocquenghem codes (BCH codes) form a class of cyclic error-correcting codes, which are constructed using polynomials over a finite field (also called Galois field). One of the key features of BCH codes is that during code design, there is a precise control over the number of symbol errors correctable by the code. It is possible to design binary BCH codes that correct multiple bit errors. BCH codes are decoded via an algebraic method known as syndrome decoding. This simplifies the design of the decoder for these codes, using small low-power electronic hardware.

bridge This is a network infrastructure entity that provides data link layer communication between two networks.

broadcast transmission This is a communication service performing a simultaneous transmission from one to all nodes.

BRS See *bit-rate switch.*

bus — This is a topology of a communication network, where all nodes are reached by passive links. This allows transmission in both directions. This term also stands for a shared medium of any topology.

bus access — This method clarifies, when and which node is allowed to transmit. When the bus is idle, any node can start immediately to transmit a frame. In CAN networks the bus access is negotiated by means of the bus arbitration.

bus analyzer — This tool monitors the bus and displays the transmitted bits. Bus analyzers are available for the physical layer, the data link layer, and different application layers (e.g., CANopen or DeviceNet).

bus arbitration — If at the very same moment several nodes try to access the bus, an arbitration process is necessary to control which node transmits while the other nodes delay their transmission. The bus arbitration process used in CAN protocol is CSMA/CD with AMP. This allows bus arbitration without destruction of data or remote frames.

bus comparator — This electronic circuitry converts physical (analog) signals used for transfer across the communication medium back into logical (digital) information or data signals.

bus driver — This electronic circuitry converts logical information or data signals into physical signals so that these signals are transferred across the communication medium i.e., the bus.

bus idle — During bus idle state no CAN frame is transmitted and all connected nodes transmit recessive bits.

bus integrating — This is the status of a node, which is waiting for an idle condition on the CAN network.

bus latency This is the time between the transmission request and the sending of the SOF bit. The bus latency depends on the busload, the number of stuff-bits, and the priority of the CAN-ID of the CAN data or remote frame to be transmitted. In CAN CC, the maximum bus latency for the highest prior frame is the time for one CAN data frame with a 64-bit data field and the maximum possible stuff-bits minus one bit time.

bus length This is the network cable length between the two termination resistors. The bus length of CAN networks is limited by the used transmission rate. For CAN CC networks running at 1 Mbit/s, the maximum length is 40 m in theory. When using lower transmission rates, longer bus lines can be used: at 50 kbit/s a length of 1 km is possible.

bus monitoring mode In this mode, the CAN controller switches off the Tx pin. This means no error flag or no ACK slot is transmitted.

bus-off A node is in the bus-off state when it is switched off from the bus due to a request of FCE. In this state a node does not influence the bus. Nodes with disabled error signaling do not enter bus-off state.

bus-off state One of three error states of the CAN error states. The CAN protocol controller switches to bus-off state when the TEC reaches 256. During bus-off state, the CAN protocol controller transmits recessive bits. In the bus-off state, a node neither sends nor receives frames. In the bus-off state, a node does not send any dominant bits.

bus state This is one of the two complementary logical states: dominant (logical 0) or recessive (logical 1).
See also *AUI state*.
See also *MDI state*.

busload

The busload is the ratio of transmitted bits to bus idle bits within a defined time unit. 100 % means that bits are transmitted during the complete defined time unit and 0 % means that the bus is in bus idle state during the complete defined time unit.

CAL See *CAN Application Layer.*

CAN Controller area network (CAN) is a serial bus system originally developed by Robert Bosch. There are several generations of the CAN protocol. We distinguish between CAN CC, CAN FD, and CAN XL. They are internationally standardized in ISO 11898-1:2024. CAN XL, the third protocol generation has been originally developed by CiA members.

CAN Application Layer (CAL) This is an application layer developed by CiA (CAN in Automation) members providing several communication services and corresponding protocols.

CAN bridge This is a functional element that links two CAN network segments using the same CAN data link layer communication services. The physical layers can be different.

CAN CC (classic) This is a data link layer compliant with ISO 11898-1, supporting CBFF (classic base frame format) as well as CEFF (classic extended frame format) and providing a data-field length from 0 byte to 8 byte.

CAN CC controller A CAN CC controller implements the CAN CC data link layer, including the PCS sublayer, only. It sends error frames, when receiving CAN FD or CAN XL data frames.

CAN CC node A CAN CC node comprises a CAN CC controller and a CAN transceiver. This logical entity implements the CAN CC data link layer, the PCS sublayer, and one of the PMA sublayers. An ECU or a device can host several CAN CC nodes.

CAN CC/FD/XL repeater CAN CC/FD/XL repeaters link physically two or more CAN CC/FD/XL network segments. They are used to implement tree or star topologies as well as for allowing long stublines, in order to increase flexibility and to reduce the size and cost of a CAN system.

CAN common ground	Each CAN network requires a common ground that avoids common mode rejection problems. However, there is a chance that there are unwanted loop currents via ground potential.
CAN device	This is a device (e.g. hardware module) that provides at least one CAN interface.
CAN bridge	This is a functional element that links two CAN network segments using the same CAN data link layer communication services. The physical layers can be different.
CAN error states	The CAN data layer provides an FSA to avoid that an erroneous node disturbs the CAN communication permanently. This FSA is called CAN error states and comprises the states error active, error passive, and bus-off.
CAN FD	The CAN FD (flexible data rate) approach enables an increased data throughput compared with CAN CC. The size of the CAN FD frame's data field is lengthened to up to 64 byte. In addition, the data phase of the CAN FD data frame is transmitted with an increased bit rate. The CAN FD protocol is at least as reliable as the CAN CC protocol. The CAN FD data link layer is standardized in ISO 11898-1. It uses FBFF and FEFF frame formats.
CAN FD controller	This is an entity implementing the CAN CC and CAN FD data link layer, including the PCS sublayer. The CAN FD controller performs the communication functions, which are set by the CAN FD protocol.

CAN FD data link layer protocol
The CAN FD data link layer protocol supports CAN CC frames as well as CAN FD data frames. CAN FD data frames are distinguished by the FDF bit, which is recessive in CAN FD data frames and dominant in CAN CC data frames.

CAN FD light
This is a protocol option that covers a subset of the CAN FD functionaliy.

CAN FD light commander node
The CAN FD light commander node sends data frames to CAN FD light responder nodes to initiate a CAN FD light communication.

CAN FD light responder node
The CAN FD light responder node is controlled by a CAN FD light commander node using CAN FD light communication.

CAN FD node
A CAN FD node comprises a CAN FD protocol controller and a CAN transceiver. It implements the CAN FD data link layer, the PCS sublayer, and a CAN PMA sublayer. An ECU (electronic control unit) or a device can host multiple CAN FD nodes.

CAN FD transceiver
CAN FD transceiver is specified in ISO 11898-2. This standard provides parameter sets for bit rates higher than 1 Mbit/s.

CAN frame time-stamp
As specified in CiA 603, a CAN frame time-stamp is captured at the sample point of the SOF, EOF or on the falling edge from the FDF bit to the res bit. In Autosar specifications, the EOF approach is used.

CAN FD transceiver
CAN FD transceiver is specified in ISO 11898-2. This standard provides parameter sets for bit rates higher than 1 Mbit/s, depending on the used network topology. The CAN FD transceiver does not support SIC mode, FAST RX mode, and FAST TX mode.

CAN frame time-stamp
As specified in CiA 603, a CAN frame time-stamp is captured at the sample point of the SOF, EOF or on the falling edge from the FDF bit to the res bit. In Autosar specifications, the EOF approach is used.

CAN_H
Indicates the CAN high line in CAN-based networks. The CAN_H line of ISO 11898-2 compliant transceivers is in recessive state at 2,5 V and in dominant state at 3,5 V. In the CAN XL FAST mode the voltage level is ±500 mV.

CAN_H, CAN_L
This is a pair of ports of the MDI. This definition applies to the PMA.
See also *CAN_H and CAN_L*.

CAN high-speed transceiver
Those transceivers are compliant with ISO 11898-2 for bit rates up to including 1 Mbit/s. It does not support the SIC mode, FAST RX mode, and FAST TX mode.

CAN HS transceiver
See *CAN high-speed transceiver.*

CAN identifier
The CAN identifier is the main part of the arbitration field of a CAN data frame (CAN CC or CAN FD) or CAN remote frame (only in CAN CC). It comprises 11 bit (base frame format) or 29 bit (extended frame format) and indicates certain information uniquely in the network. In CAN CC and CAN FD, the CAN identifier value implicitly determines the priority for the bus arbitration. The CAN XL protocol separates the priority functions (11-bit priority ID) and the addressing (32-bit acceptance field (AF)).

CAN in Automation (CiA)
The international users‘ and manufacturers‘ group founded in 1992 promotes CAN and supports CAN-based higher-layer protocols (http://www.can-cia.org).

CAN Kingdom
This is a higher-layer protocol framework optimized for embedded networks. It is suitable for real-time applications.

CAN_L — Indicates the CAN low line in CAN-based networks. The CAN_L line of ISO 11898-2 compliant transceivers is in recessive state at 2,5 V and in dominant state at 1,5 V. In the CAN XL FAST mode the voltage level is ±500 mV.

CAN message specification (CMS) — This is a part of the CAL specification, defining the communication services.

CAN module — This is the implementation of the CAN protocol controller plus the hardware acceptance filter and the frame buffers within a microcontroller or application-specific integrated circuit (ASIC).

CAN node — This is the implementation of the CAN lower layers (data link layer with physical coding sublayer and physical medium attachment sublayer as well as physical medium dependent sublayer).

CAN protocol controller — The CAN protocol controller is part of a CAN module performing data en-/de-capsulation, bit timing, CRC, bit stuffing, error handling, failure confinement, etc.

CAN repeater — This is a hardware assembly that links two or more network segments using the same physical layer with the very same bit-timing.

CAN SIC transceiver — CAN SIC transceivers are suitable for complex network topologies (including stars). They can reduce the ringing on the bus lines caused by reflections in a network and allow bit rates up to 8 Mbit/s depending on the used cables and connectors. They do not support FAST RX mode as well as FAST TX mode.

CAN SIC XL transceiver — CAN SIC XL transceivers supports different network topologies, such as linear bus, bus with stubs, and stars. The maximum achievable bit rate is 20 Mbit/s. CAN SIC XL transceivers have three modes and are switched from arbitration mode (up to 1 Mbit/s) to FAST TX resp. FAST RX mode (up to 20 Mbit/s). To manage this transceiver mode switching, the CAN XL protocol provides the optional PWM encoding function, to be linked to the PWM decoding function provided by the CAN SIC XL transceiver. In addition, the CAN SIC XL transceiver can be operated in the dataphase also in SIC mode.

CAN switch This is a functional element that links more than two network segments using the same data link layer communication services. The physical layer can be different.

CAN transceiver This component implements the interface between CAN controller and the physical transmission media, converting the digital signals from the CAN controller into (typically differential) signals that are transmitted via the physical transmission media. Depending on the intended transmission media, CAN transceiver implementations comply with standards and specifications such as ISO 11898-2, ISO 11898-3, ISO 11992-1, or SAE J2411.

CAN XL The CAN XL approach, as the third CAN generation, supports data field length from 1 byte to 2048 byte. It provides an 11-bit priority field for network arbitration purposes and a 32-bit ac-ceptance field for addressing purposes. The CAN XL data link layer complies with ISO 11898-1. It supports CBFF (classic base frame format), CEFF (classic extended frame format), FBFF (FD base frame format), and FEFF (FD extended frame format) as well as the XLFF (XL frame format). CAN XL is intended for backbone and sub-backbone network applications. It is designed for an easy integration into TCP/IP network systems.

CAN XL controller	The CAN XL controller implements all three CAN protocol generations (CAN CC, CAN FD, and CAN XL).
CAN XL node	CAN XL node comprises a CAN XL controller and any CAN transceiver. It implements the CAN XL data link layer, the PCS sublayer, and a CAN PMA sublayer.
CANaero-space	This is a higher-layer protocol for avionic and aerospace applications.
CANopen	This term is used to indicate a communication technology. It includes the CANopen CC and CANopen FD application layers and communication profiles as well as the family of CiA profiles for embedded networking in industrial machinery, medical equipment, building automation (e.g., lift control systems, electronically controlled doors, integrated room control systems), railways, maritime electronics, truck-based superstructures, off-highway and off-road vehicles, etc.
CANopen CC application layer	The CANopen CC application layer and communication profile (CiA 301) is standardized in EN 50325-4. It specifies communication services and objects. In addition, it specifies the device's object dictionary and the NMT.
CANopen FD	This is a CAN FD-based application layer and communication profile as specified in CiA 1301, able to make use of the CAN FD data frames. CAN-open FD is the successor of CANopen CC.
CANopen field device	This is an electronic unit comprising printed circuit board, firmware, soft-ware, and an optional enclosure as well as one or more CANopen nodes.
CANopen Lift	This is the unregistered trademark for the CiA 417 application profile for lift control systems.

CANopen manager
: The CANopen manager is responsible for the management of the network. In the CANopen CC and the CANopen FD manager device, there resides the NMT manager functionality. Additionally, there resides in the CANopen CC manager the SDO manager or/and the configuration manager. A CANopen manager implements a CANopen object dictionary and supports the CANopen NMT server functionality.

CANopen network segment
: A CAN-based communication system comprising cabling, connectors, and CANopen field devices is called CANopen network segment.

CANopen node
: This is a node implementing the CANopen application layer services as specified in CiA 301 or CiA 1301.

CANopen router
: This is a functional element that links two or multiple CANopen network segments using the same network layer communication services. The data link and physical layer can be different.

CANopen Safety
: This is a communication protocol enhancement allowing transmission of safety-related data. It is standardized in EN 50325-5. The protocol requires just one physical CAN network. Redundancy is achieved by sending each safety-related message twice with bit-wise inverted content using two identifiers differing at least in two bits.

CANopen Safety Chip (CSC)
: This 16-bit microcontroller provides a CANopen Safety protocol firmware implementation. It complies with EN 50325-5 and is certified by TÜV Rhineland up to SIL 3 (safety integrity level).

CAPL (CAN access programming language)
: CAPL is an ANSI C-based programming language extended by network-specific functions and data types. CAPL is used in CANalyzer and CANoe tools from Vector.

carrier sense multiple access (CSMA) — Carrier sense multiple access (CSMA) is a network access method implemented in the MAC sublayer. It enables a transmitter to use a carrier-sense mechanism for determining whether another transmission is in progress before initiating a transmission. It tries to detect the presence of a carrier signal from another node before attempting to transmit. If a carrier is sensed, the node waits for the transmission in progress to end before initiating its own transmission. By using CSMA, multiple nodes, in turn, send and receive on the same medium. Transmissions by one node are generally received by all other nodes connected to the medium. There are four types of access modes available in CSMA: persistent, non-persistent, P-persistent and O-persistent.

CBFF — See *classic base frame format.*

CC frame — See *classic frame.*

CCP (CAN calibration protocol) — CCP is used to communicate calibration data in engine car applications.

CDCF — This is the abbreviation for Concise DCF. See *device configuration file.*

CEFF — See *classical extended frame format.*

CiA 102 — This document provides guidelines for device and network designers regarding the physical CAN interface.

CiA 103 — This document specifies the physical layer for an intrinsically safe capable CAN interface based on CAN HS transceiver compliant with ISO 11898-2.

CiA 106 — This document recommends the connector pin-assignment for CAN interfaces. This includes the CAN_H and CAN_L pins, the ground pin, and the power supply pins.

CiA 110	This document specifies the electrical and mechanical parameters of common mode chokes, to be used in CAN CC as well as CAN FD-based networks.
CiA 150	This document specifies facilities and services of a power management layer protocol entity on the CAN network. It allows reduction of power consumption in CAN networks by introduction of a network stand-by capability.
CiA 201 to 207	These documents specify the CMS, the DBT, the NMT (network management), and the LMT (layer management) services and protocols of the CAL (CAN Application Layer).
CiA 301	This document specifies the CANopen CC application layer and communication profile. It covers the functionality of CANopen NMT server entity and partly of a CANopen NMT manager entity. CiA 301 is dedicated for devices not using CAN FD. For CAN FD-capable CANopen devices, see CANopen FD (CiA 1301).
CiA 302 series	These documents specify the CANopen CC additional application layer functions, including general definitions (part 1), network management (part 2), configuration and program download (part 3), network variables and process image (part 4), SDO manager (part 5), network redundancy (part 6), multi-level networking (part 7), project management parameters (part 8), and energy saving (part 9).
CiA 303 series	These documents provide device and network design recommendations for the CANopen CC physical layer (part 1), especially for selecting cables. Part 3 recommends communication-related indicators. The recommendations for CANopen connector pin assignments (formerly part 2) have been moved to CiA 106.

(CiA 304) See *EN 50325-5.*

CiA 305 This document specifies the CANopen CC layer setting services and protocols. The LSS manager sets the node-ID or the bit rate in the LSS server entities via the CANopen CC network.

CiA 306 series The first part of this document series specifies format and content of EDSs and DCFs for CANopen CC devices. Part 2 specifies the profile database, and Part 3 describes the network variable handling and tool integration.

CiA 308 The CANopen CC performance specification names and defines communication performance figures used to compare devices and implementations in a specific application environment. Time measurements include PDO turnaround time, SYNC jitter, SDO response time, etc.

CiA 309 series This set of documents specifies the services and protocols for access from other (e.g., TCP/IP-based) networks to CANopen CC networks. The services are mapped to Modbus/TCP (Part 2) as well as to ASCII (Part 3). Part 4 specifies the access of CANopen CC networks from Profinet IO. For cloud-based applications, Part 5 specifies the mapping of the network access services to well-known web-services such as MQTT, HTML, RestFull API, etc.

CiA 310 This document specifies the CANopen CC conformance test plan with regard to the process of verifying that an implementation performs in accordance with a particular standard, specification or environment. A CiA 310 conforming implementation is one that satisfies both static and dynamic conformance requirements.

CiA 311	The document specifies the XML elements and rules for describing electronically CANopen CC devices.
CiA 312 series	This set of device profile conformance test plans specifies all test steps required for checking, whether the implementation of a CANopen CC device is compliant with the corresponding CiA device profile. Part 1 specifies the general definitions. Part 2 is dedicated to I/O modules and Part 4 to contrast media injectors. Part 7 is dedicated to the pedelec profile 1 from CiA 454.
CiA 314	This document provides the CANopen CC framework for programmable logic controllers (PLCs) and other programmable devices compliant with IEC 61131-3.
CiA 315	This specification specifies a generic frame format for the transparent transmission of CAN CC-based messages (arbitration and data field) on a wireless network.
CiA 318	This document specifies the CANopen integration to the RTC (robotic technology component) environment and the mapping of the RTC finite state automaton (FSA) to the CANopen network management FSA. It also describes the RTC-CANopen manager and the ProxyRTCs system integration.
CiA 319	This document provides implementation and configuration specificaton for devices implementing communication services as specified in EN 50325-5 (CANopen Safety).
CiA 320	This document specifies the sleep and wake-up handling of CANopen devices.
(CiA 400)	See *CiA 302-7.*

CiA 401 series These documents specify the CANopen interface for modules with generic analog and digital inputs and outputs (part 1). The series includes CANopen interface for joysticks and similar devices, e.g. foot-pedals (part 2). There is a mapping to CANopen CC and to CANopen FD specified.

CiA 402 series This set of documents specifies the CANopen device profile for drives and motion controllers. It covers frequency inverters, servo controllers as well as stepper motors. Part 1 provides general definitions. Part 2 and Part 3 correspond to IEC 61800-7-201 respectively IEC 61800-7-301. Part 4 specifies the safety functionality. Part 5 specifies PDOs for CiA 402 compliant devices, which control asynchronous and synchronous motors. Part 6 specifies the CANopen FD mapping for frequency converters, servo drives and stepper motors as well as for multiple-axes systems.

CiA 404 series This set of documents specifies the CANopen device profile for measuring devices and closed-loop controllers. This also includes multi-channel devices. Part 1 specifies the generic objects and generic PDO mapping. Part 2 specifies EUROMAP specific parameters.

(CiA 405) Former CANopen profile for IEC 61131-3 compatible controllers. Now published in several parts (CiA 302-8, CiA 306-3, CiA 314).

CiA 406 series This set of documents specifies the communication and application parameters, the functional behavior and parameters (part B), the CANopen CC communication and mapping parameters (part C), the CANopen FD communication and mapping parameters (part F), and the mapping of CiA 406-B process data into J1939 parameter groups (part J) for different types of rotary and linear encoders such as incremental and absolute, normal, and high resolution, single and multi-sensor (linear only) encoders.

(CiA 407)	See *EN 13149-4/-5/-6.*
CiA 408	This document specifies the CANopen device profile for fluid power technology including hydraulic controllers and proportional valves, which are compliant with the bus-independent VDMA (Verband Deutscher Maschinen- und Anlagenbau e. V.) device profile fluid power technology – proportional valves and hydrostatic transmission.
CiA 410 series	This document specifies the CANopen device profile for one- and two-axis inclinometers, as well as the safety functionality of inclinometers. Part B specifies the functional behavior and parameters, part C specifies the mapping to CANopen CC and part F the mapping to CANopen FD. Part J specifies the mapping of CiA 410 to J1939.
CiA 412 series	This set of documents specifies CANopen profiles for medical devices (Part 2: Automatic x-ray collimator; Part 6: Dose measurement system).
CiA 413 series	The CANopen interface profile series for truck gateways specifies gateways to CAN-based in-vehicle networks using SAE J1939, ISO 11992, and other J1939-based networks. The CANopen network is mainly used for truck or trailer-based body applications, e.g., as in refuse collecting vehicles, truck-mounted cranes, and concrete mixers.

Part 1 specifies the general definitions, part 2 specifies the application objects for braking and running gear and part 3 for equipment other than brakes and running gear. Part 5 specifies the application objects for superstructures, part 6 specifies a framework for J1939 parameter groups, and part 8 specifies the framework for HMI (human machine interface) control.

CiA 414 series The CANopen CC device profiles for for weaving machines specify the general definitions (part 1) and the interfaces for feeder sub-systems (part 2), including “pre-measuring feeders” as well as “weft feeders”.

CiA 415 This CANopen CC application profile specifies interfaces for sensors and sensor controllers. Such CANopen devices are intended for use in all kinds of road-construction and earth moving machines.

CiA 416 series This set of documents, CANopen CC application profile for building door control, specifies general definitions such as start-up procedures and sys-tem security (part 1), virtual devices overview (part 2), and pre-defines communication objects and application data objects (part 3). The application profile is dedicated for locks, sensors, and other devices used in electronically controlled building doors.

CiA 417 series This set of documents, CANopen CC application profile for lift control systems, specifies general definitions (part 1), the functionality of the virtual devices (part 2), the pre-defined PDOs for lift application 1 (part 3), and the detailed application data objects (part 4). The application profile is dedicated for car controllers, door controllers, call controllers and other controllers as well as for car units, door units, input panels, and display units, etc.

CiA 418 This CANopen CC device profile for battery modules specifies the interface to communicate with battery chargers.

CiA 419 This CANopen CC device profile for battery chargers specifies the interface to communicate with the battery module.

CiA 420 series This CANopen CC profile family for extruder downstream devices specifies interfaces for puller (part 2), corrugator (part 3), saw devices (part 4), co-extruder (part 5) as well as for calibration tables (part 6).

CiA 421 series This set of documents, CANopen CC application profile for train vehicle control systems, specifies the communication between virtual devices within locomotives, power cars as well as coaches. It includes the general definitions (part 1), the virtual rail vehicle auxiliary operating system (part 4), the virtual power drive system to the in-vehicle network (part 5), the vehicle linkage system (part 9), the virtual exterior lighting system (part 10), the virtual interior lighting system (part 11), and the virtual door control system (part 12).

CiA 422 series This set of documents, CANopen CC application profile for municipal vehicles, specifies general definitions (part 1), the functionality of the virtual devices (part 2), the pre-defined TPDOs, RPDOs, and SDOs (part 3) as well as the application objects (part 4) for the interfaces of sub-systems such as compaction unit, weighing unit, etc.

CiA 423 series This set of documents, CANopen CC application profile for rail vehicle power drive systems, specifies the communication between virtual devices required for the control of diesel as well as diesel electrical locomotives. It includes the general definitions (part 1), the traction controller (part 2), the diesel engine control unit (part 3), the transmission control unit (part 4), the diesel engine safety control unit (part 5), the speed sensor unit (part 6), the diesel engine signal unit (part 7), the particle filters control unit (part 10), the starter unit (part 11), and the oil refill unit (part 12).

CiA 424 series This set of documents, CANopen CC application profile for rail vehicle door control systems, specifies the general definitions (part 1) for the communication between a door controller (part 2) and the related door units (part 3).

CiA 425 series This set of documents, CANopen CC application profile for medical diagnostic add-on modules, specifies plug-and-play interfaces for contrast media injectors (part 2). The general definitions are specified in part 1. CiA 425 is also used as an unregistered trademark.

CiA 426 series This set of documents, CANopen CC application profile for rail vehicle exterior lighting control, specifies the communication between an exterior lighting controller (part 2) and the related exterior lighting units (part 3). The general definitions are specified in part 1.

CiA 430 series This set of documents, CANopen CC application profile for rail vehicle auxiliary operating systems, specifies auxiliary operating system controller (part 2) and the communication between auxiliary equipment such as power train cooling control unit (part 3), coolant expansion tank unit (part 4), power train cooling fan control unit (part 5), engine pre-heating unit (part 6), hydrostatic signal unit (part 7), and the generator unit (part 9). The general definitions are specified in part 1.

CiA 433 series This CANopen CC application profile for rail vehicle interior lighting control specifies the communication between an interior lighting controller (part 2) and interior lighting units (part 3). The general definitions are specified in part 1.

CiA 434 series This set of documents, CANopen CC profiles for laboratory automation systems, specifies the communication between a laboratory automation controller unit and related functional units such as dilutor, dispenser, and pump units (part 2) as well as heating, cooling, and shaking units (part 3). The general definitions are specified in part 1.

CiA 436 This CANopen CC profile for construction machines specifies the integration platform for sensor, engine, and transmission systems as well as for the driver/worker user interface and the implement systems (e.g., crane).

CiA 437 series This set of documents, CANopen CC application profile for grid-based photo-voltaic systems, specifies the integration platform for photovoltaic controller, inverters, tracking systems and sensors as well as other devices. The corresponding general definitions, pre-defined communication objects and the profile data objects of this profile are specified respectively in part 1, part 2, and part 3.

CiA 442 This CANopen CC device profile for motor starters is based on the IEC 61915-2 root profile for starters and similar equipment.

CiA 443 This CANopen CC profile for SIIS level-2 devices specifies interfaces for simple and complex sensors and actuators used in so-called “Christmas trees” on the ocean ground.

CiA 444 series This set of documents specifies the CANopen CC interfaces for container handling system add-on devices such as spreaders for cranes (part 2) or straddle carriers (part 3).

CiA 445 This device profile specifies the CANopen CC interface for simple and sophisticated radio frequency identification (RFID) reader/writer devices.

CiA 446 This CANopen CC interface profile for AS-Interface gateways specifies CANopen devices, which act as an AS-Interface controller in AS-Interface networks.

CiA 447 series This set of documents, CANopen CC application profile for special-purpose car add-on devices, specifies the CAN physical layer as well as application, configuration, and diagnostic parameters for functional units such as taximeter, roof-bar, etc. General definitions, the virtual device definitions, the application data objects, and the pre-defined CAN-IDs and communication objects are specified respectively in parts 1 to 4. Part 5 specifies the application profile specific tests for CiA 447 devices.

CiA 450 This CANopen CC device profile for pumps is based on the VDMA (Verband Deutscher Maschinen- und Anlagenbau e. V.) profile for pumps. It specifies process data for generic pumps and for liquid pumps.

CiA 452 This document specifies the CANopen CC interface for drives controlled by programmable logic controllers (PLCs) using PLCopen motion control.

CiA 453 This CANopen CC device profile for power supplies specifies an interface for AC/AC, DC/DC, AC/DC, and DC/AC converters.

It is suitable for programmable and non-programmable power supply devices with single or multiple outputs that are voltage-, current- or power-controlled.

CiA 454 series This set of documents, CANopen CC application profile for energy management systems, specifies the communication interface for all virtual devices, which take part in energy management control application. Such energy management control applications are implemented in e.g., light electric vehicles, robots, offshore parks, isolated farms, etc. The general definitions, the pre-defined communication parameters and general application objects, as well as the PDO communications are specified respectively in parts 1, 2, and 3. It also includes specifications for the energy bus controller (part 4), voltage converter (part 5), battery pack (part 6), drive control unit (part 7), HMI unit (part 9), generator unit (part 13), and load unit (part 14).

CiA 455 This CANopen CC profile specifies the control of drilling machines with special regard on positioning and tool control.

CiA 456 This device profile specifies the CANopen CC interface for configurable network components that provide CAN bridge/switch functionality with up to 16 CAN ports including one configurable CANopen port.

CiA 457 This document specifies the CANopen CC interface profile for devices providing a gateway function to wireless transmission media ports.

CiA 458 This device profile specifies the CANopen CC interface for energy measuring devices including energy consumption and production, in particular for energy recovering.

CiA 459 series This set of documents specifies the CANopen CC interface for on-board weighing devices. Such devices are usable on trucks, off-highway, or off-road vehicles (including train coaches). Part 1 includes the general specification and functional overview, part 2 specifies the communication parameters, and part 3 specifies the application data objects.

CiA 460	This profile specifies the CANopen CC interface of a service robot controller device, which is compliant with the robotic technology component (RTC) specification.
CiA 462	This profile specifies the CANopen interface for devices that identify existence, dimension, orientation, or movement of items in their environment (e.g., optical camera (2D or 3D), laser device, radars). Often those devices are called vision sensors or object detection devices.
CiA 463 series	This set of documents, device profile for IO-Link gateway, specifies the mapping of the IO-Link gateway to CANopen CC (part C) and CANopen FD (part F). The operating principles and the application data for IO-Link interface devices are specified in part B.
CiA 510	This document specifies the mapping of CANopen CC application layer protocols such as SDO and EMCY to dedicated J1939 parameter groups. In particular, this includes the mapping of SDO client and SDO server protocols.
CiA 601 series	This set of documents specifies the usage of CAN FD hardware implementations, comprising the CAN FD physical interface implementation (part 1), CAN FD system design recommendations (part 3), and cable recommendations (part 6).
CiA 603	This document specifies the time-stamping when transmitting or receiving CAN CC or CAN FD data frames. This time-stamping enables a synchronized time-base in a network segment. It complies with the Autosar specification.

CiA 611 Part 1 specifies the CAN XL higher-layer functions such as the usage of the SDT field of the CAN XL data link layer. The SDT defines the type of service data, mapped to the data field of the CAN XL LLC sublayer, and how the management information such as addressing, virtualization, or data size are mapped on dedicated LLC frame fields.
Part 2 specifies the multi-PDU protocol for SDT 08_h (see part 1), which supports the mapping of multiple messages (PDUs) compliant to different higher-layer protocols into a single CAN XL data frame.

CiA 702 This document specifies implementing of LSS FD services (see CiA 1305) in a CANopen CC network and describes its beneficial usage. It provides hints with regard to the implementation and usage of the LSS services as well.

CiA 710 This document specifies the bootloader mode for CANopen devices as well as the switching between the bootloader mode and the application mode.

CiA 801 This technical report describes the recommended practice and gives application hints for implementing automatic bit-rate detection in CANopen devices.

CiA 802 This document provides recommendations for avoiding CAN remote frames when using CANopen CC communication services.

CiA 808 This document describes the recommended practice and gives application hints for implementing the connection of crane and spreader compliant with the CiA 444-1 and CiA 444-2 specifications.

CiA 810 This application note recommends and suggests how to implement and handle laboratory automation devices, which are compliant to CiA 434 profile family.

CiA 812 This application note describes use cases for CANopen CC devices supporting the CiA 315 framework for tunneling of CAN CC data and remote frames via wireless networks.

CiA 814 This document provides implementation hints for the CiA 417 bootloader (see CiA 417).

CiA 850 This document recommends how to implement a CiA 413 gateway interface for truck-mounted cranes, multi-lifts, and aerial working platforms.

CiA 852 This document recommends how to implement the CANopen interface for CiA 401-based operator environments. Operator environments include simple remote-control units as well as operator seats with integrated joysticks, foot pedals, pushbuttons, indicators, etc.

CiA 890 This document provides recommendations how to represent the SI units and prefixes in parameter specifications. It applies to all CiA specifications and harmonizes the coding of SI units and prefixes within CiA documents.

CiA 910 This document specifies the CAN simulation model. Part 1 defines general terms and use cases for CAN simulation, which are used in the requirement specifications of the part 2 (simulation models of the PMA sub-layer).

CiA 1301 The CANopen FD application layer and communication profile specification maps its communication objects to CAN FD data frames. This means, the PDOs have a maximum length of 64 byte. The USDO communication service enables broadcast and multicast communication.

CiA 1305 This document specifies the LSS and the layer setting protocols for CANopen FD devices. LSS is used to inquire or to change the settings of three CANopen FD device parameter sets (Network-ID and/or node-ID, bit timing, and LSS address compliant to the identity object) of the physical layer, data link layer, and application layer via the CAN network.

CCiA 1310-1 This document specifies the CANopen FD conformance test plan. Part 1 specifies test cases to proof the conformity of devices compliant to the CiA 1301 document.

CiA 1311 This document provides definitions and specifications for electronic descriptions of CANopen FD devices. Electronic device descriptions are used to teach tools the functionality of a device. This includes also host controllers, which configure other CANopen FD devices. Specifically, the first and the only part of this document specifies the XML Schema Definition for the CANopen FD profiles. It is based on ISO 15745-1:2003/Amd 1:2007.

CiA application profile A CiA application profile specifies the device parameters of an entire control application and the mapping to a CAN-based application layer (e.g., CANopen CC, CANopen FD, J1939). This set of defined functional entities and communication interfaces enables virtual devices building a logical network system for a dedicated application (e.g., lift control system) to achieve interoperability. This includes specification of process data, configuration parameters, and diagnostic information as well as communication relations between virtual devices.

CiA device profile A CiA device profile specifies the device parameters of one device and the mapping to a CAN-based application layer (e.g., CANopen CC, CANopen FD, J1939).

This set of defined functional entities and communication interfaces enables logical devices to exhibit the same behavior. This includes specification of process data, configuration parameters, and diagnostic information.

CiA interface profile — A CiA interface profile specifies the parameters of an entity with gateway functionality and the mapping to a CAN-based application layer (e.g., CANopen CC, CANopen FD, J1939). It describes just the interface and not the application behavior of a device, e.g., a gateway device.

classic base frame format (CBFF) — This format of CAN CC data frames and remote frames uses an 11-bit identifier. The frames are transmitted with one single bit rate and support data field lengths of zero up to eight byte.

classic extended frame format (CEFF) — This format of CAN CC data frames and remote frames uses a 29-bit identifier. The frames are transmitted with one single bit rate and support data field lengths of zero up to eight byte.

classical frame — This is a data frame or remote frame using the classic base frame format (11-bit identifier) or the classic extended frame format (29-bit identifier).

CleANopen — This is the unregistered trademark for the CiA 422 application profile for municipal vehicles.

CMS — CAN based message specification (CMS) as specified in CAL. This is one of the service elements of the application layer in the CAN Reference Model. CMS is a language that can describe how the functionality of a module can be accessed at its CAN interface.

COB — See *communication object.*

COB-ID In CANopen and CAL, the COB-ID specifies the CAN identifier and additional parameters (valid/-invalid bit, remote frame support bit, frame format bit) for the related COB.

commander/ responder communication In a commander/responder communication system the commander initiates and controls the communication. The responder is not allowed to initiate any communication at all.

communicating state This is a CANopen FD NMT server FSA state (see CiA 1301) in which a device in principle owns the ability to communicate. It covers NMT states pre-operational (no PDO transmission allowed), operational (all communication services available) and stopped (only NMT and error control).

communication object (COB) In CANopen, a communication object is a specific communication function, e.g., PDO, SDO, USDO, EMCY, TIME, or error control mapped to one or more CAN CC respectively CAN FD data or remote (only CAN CC) frames as specified in CiA 301 (or EN 50325-4) respectively CiA 1301.

communication parameter This CANopen device parameter determines the behavior of the CANopen device at its communication interface.

communication profile A communication profile specifies the content of communication objects such as EMCY, TIME, SYNC, heartbeat, NMT, etc. in CANopen.

configuration manager The configuration manager provides mechanisms for configuration of CANopen devices during boot-up.

configuration parameter This parameter in the CANopen object dictionary configures the application behavior of the device.

confirmed communication Confirmed communication services require a bi-directional communication, meaning that the receiving node sends a confirmation that the protocol is received correctly.

conformance testing — Conformance testing proofs that a component or device complies to a specific standard or specification. There are test houses, providing conformance tests for CAN controller chips. ODVA tests DeviceNet products on conformity, and CiA does the same for CANopen devices. There are also third-party test houses for Isobus devices.

conformance test plan — A conformance test plan comprises definitions of test cases. The conformance test plans for CAN CC/FD implementations are standardized in ISO 16845-1.

conformance test tool — A conformance test tool is the implementation of a conformance test plan.

connector — Electro-mechanical component used to make a connection between a device and the CAN bus-line or to extend bus cables. CiA specifies the connector pin-assignment for CAN and CANopen in the CiA 106. ODVA specifies the connector pin-assignment for DeviceNet.

consumer — A receiver of data and remote frames is called a consumer; meaning the acceptance filter is opened.

consumer heartbeat time — In CANopen, this determines the time interval required by the monitoring node to verify, whether a monitored node is alive or not. In case the monitored node transmits no heartbeat message within that time interval, it is regarded as not alive.

contained PDU (C-PDU) — The C-PDU in SAE J1939-22 consists of the 3-bit TOS field, the service header (5 bit, 13 bit, 21 bit, or 29 bit) and the C-PDU payload (0 byte to 60 byte).

contention-based arbitration — This is an arbitration procedure, in which simultaneous access of multiple CAN nodes results in a contention, which is non-destructively resolved, giving the frame with the highest priority immediately bus access.

control field IIn CAN CC data and remote frames, the 6-bit control field contains the four DLC bits, the IDE bit, and the reserved bit(s). In the CAN FD data frame, the 9-bit control field is enhanced by the FDF, BRS and ESI bit.

controller (functional) A functional controller (e.g., lift door controller) is a virtual device (entity), which controls other virtual devices (e.g., lift door units). This is a virtual device controlling other virtual devices (e.g., lift door controller).

controller area network (CAN) CAN is a serial network that allows nodes to communicate with each other. A CAN communication is described by means of the 7-layer OSI reference model. CAN covers just the two lower layers: data link layer and physical layer.

CRC See *cyclic redundancy check.*

CRC delimiter The CRC delimiter bit is the last bit in the CRC field of the CAN data frame or CAN remote frame (only in CAN CC). It is always recessive.

CRC error A consumer verifies the correct reception of a CAN frame by calculating the CRC checksum, according to the generator polynomial, and taking the CRC checksum provided by the producer into account. As result of this calculation, consumers get the information whether they received the CAN frame correctly or whether an error occurred during the frame transmission (CRC error). In CAN FD frames, a mismatch between the counted stuff-bits and the received stuff count is treated as a CRC error. The corresponding error frame is transmitted after the acknowledgement field.

CRC field The CRC field contains the CRC sequence followed by a recessive CRC delimiter. In CAN FD frames, the CRC field also contains the stuff count. The 15-bit CRC sequence is used for CAN CC frames. The 17-bit and 21-bit CRC sequences are respectively used for CAN FD frames with up to 16-byte or longer than 16-byte data fields.

CSMA — See *carrier sense multiple access.*

CSMA/CD + AMP — The carrier sense multiple access/collision detection with arbitration on message priority is the bus arbitration method used in CAN. This method arbitrates simultaneous bus access requests.

cyclic redundancy check (CRC) — CRC is performed by a polynomial implemented in transmitting and in receiving CAN modules to detect corruption while transmitting CAN data frames or CAN remote frames (only in CAN CC).

DAH	See *data to arbitration high bit.*
DAS	See *data to arbitration sequence.*
data bit rate	This is the number of bits per time during the data phase of a CAN FD and CAN XL frame. The data bit rate is independent of bit encoding/decoding.
data bit time	This is the duration of one bit in the data phase of a CAN FD and CAN XL frame. The data bit time has the same length as the nominal bit time or is shorter than the nominal bit time. It is defined by a number of data time quanta in the bit.
data consistency	With regard to network technologies, data consistency means that all nodes, which are connected to the same network, have the same state of knowledge. Network-wide data consistency is guaranteed for all error active CAN nodes by means of globalization of local errors. In CAN XL, the automatic retransmission of faulty frames can be disabled.
data element	This is an entity of a data object, representing a parameter in the CANopen object dictionary. It specifies a basic and unique information of a data (e.g., the type and size) and is addressed uniquely by means of an 8-bit sub-index.
data field	The data field of the CAN data frame has a granularity of one byte and varies in the range of zero byte to including eight bytes of user data in CAN CC data frames or zero byte to including 64 byte in CAN FD data frames. In CAN XL the data field is made of 1 to up to 2048 byte. The size of the data field is indicated by the DLC.
data frame (DF)	The CAN data frame carries user data from a producer to one or more consumers. It consists of the SOF bit, the arbitration field,

the control field, the data field, the CRC field, the ACK field, and the EOF field.

data frame (DF) — The CAN data frame carries user data with application content from a producer to one or more consumers. It consists of the SOF bit, the arbitration field, the control field, the data field, the CRC field, the ACK field, and the EOF field.

data high bit (DH) — The data high bit (DH), including two subsequent bits DH1 and DH2 of the 4-bit ADS field, are the first bits of the CAN XL data phase and are transmitted recessive. These bits belong to control field of a CAN XL frame.

data length code (DLC) — The 4-bit DLC in the control field of a CAN CC frame or a CAN FD frame indicates the data field length. In remote frames (only CAN CC) the DLC corresponds to the data field length in the requested data frame (not the transmitted). In CAN XL, the DLC sub-field has a length of 11 bit.

data link layer — This is the second layer in the OSI reference model providing basic communication services. The CAN data link layer specifies data, remote (only in CAN CC), error, and overload frames.

data low bit (DL) — The data low bit (DL), including 1 bit (DL1), is the last bit of the 4-bit ADS field that belongs to the control field of a CAN XL frame. This bit is transmitted in dominant state.

data object — This is a parameter (variable, array, or record) in the CANopen object dictionary, which is identified uniquely by a 16-bit index and an 8-bit sub-index.

data phase — This phase indicates those parts of the CAN FD data frame or the CAN XL data frame that are transmitted with a higher bit rate as in the arbitration phase. It is wrapped by the arbitration phase, and starts with the sample point of the BRS bit and lasts till the sample point of the CRC delimiter.

data RX mode This is the operating mode of the physical medium attachment (PMA) sublayer, in which the bus states can be different from the bus states in the arbitration mode.

data to arbitration high bit (DAH) The DAH bit is the first bit with the nominal bit time after the end of the XL data phase. It is one of the four bits of the DAS field. It is located at the start of arbitration phase (bit rate up to 1 Mbit/s) of the ACK field in the CAN XL frame. Through this bit, the transceiver is switched from fast mode to slow mode. At the beginning of the DAH bit, the transmitter changes from level_0 to active recessive and after the signal improvement time, the transmitter changes from active recessive to passive recessive state.

data to arbitration sequence (DAS) The data to arbitration sequence (DAS) is a part of the ACK field and has two purposes: switching the bit rate from the XL data bit rate to the nominal bit rate, and switching the CAN transceiver mode from the data TX/RX mode to the arbitration mode, if the mode is switched in the preceding ADS. The DAS consists of the DAH, AH1, AL1, and AH2 bits. The DAH bit is the first bit with the nominal bit time after the end of the XL data phase with the XL data bit rate. The edge AH1 to AL1 is used in the CAN XL controller for synchronization in the arbitration bit rate.

data TX mode This is the operating mode of the physical medium attachment (PMA) sublayer, in which it can drive the bus states differently than it drives them in the arbitration mode.

data type This is the object attribute in CANopen and DeviceNet defining the format, e.g., Unsigned8, Integer16, Boolean, etc.

DBT The distributor (DBT) is part of the CAL specification, defining a method of automatic identifier distribution during network boot-up.

DCF See *device configuration file.*

default value This is the object attribute in CANopen defining the pre-setting of not user-configured objects after power-on or application reset.

destination address mode (DAM) In the DAM mode of a CANopen CC MPDO, a multiplexer identifies the object (16-bit index and 8-bit sub-index) in the MPDO consumer's object dictionary.

device configuration file (DCF) The device configuration file describes the CANopen CC parameters of a configured CANopen CC device in the same file format as the EDS. EDS and DCF are specified in the CiA 306-1. A compact device description is provided in the Concise DCF, specified in CiA 302-3.

DeviceNet This is a CAN-based higher-layer protocol and device profiles specification. DeviceNet was designed for factory automation and provides a well-defined CAN physical layer in order to achieve a high off-the-shelf plug-and-play capability. The DeviceNet specification is maintained by the ODVA (www.odva.org) non-profit organization.

device profile A device profile specifies the device-specific application data and communication capability based on the related higher-layer protocol. For more complex devices, a device profile specification can provide an FSA, which enables a standardized device control.

DF See *data frame.*

DH See *data high bit.*

Diagnostics on CAN The ISO 15765 standard series specifies the diagnostic on CAN protocols and services, which are used for the CAN-based diagnostic interface for passenger cars as specified in ISO 14229-3 (UDSonCAN). Part 2 of this standard series specifies the CAN-based network/transport layer services and protocols. Part 5 specifies the CAN diagnostic link connector including bit rates for CAN CC and CAN FD.

DIN 4630 This German standard in English language specifies the CAN-based network connecting body builder ECUs to telematics. There are two implementation options: One uses the J1939-21 application layer, the other is based on the CANopen CC application layer. DIN has submitted this standard to ISO (see ISO 25200).

DIN 14700 This set of German standards specifies the so-called FireCAN application profile. It specifies the CAN CC interfaces for fire-fighting truck specific devices. The application layer uses communication services similar to CANopen CC.

DIN 14704 This German standard in English language specifies the J1939-based in-vehicle network gateway for fire-fighting trucks.

DL See *data low bit.*

DLC See *data length code.*

dominant bit *A dominant bit overwrites by definition a recessive bit. It has the logical value 0.*

double-reception of data frames This is the probability in CAN that the very same CAN data or remote frame is transmitted twice. If the last bit of the EOF is corrupted at the transmitting node, then a retransmission of the data or remote frame is caused. When the receivers accept the frame after the last but one bit, they receive the frame twice.

DR (draft recommendation) This kind of recommendation is not finalized, but it is published. CiA's DRs are normally not changed within one year.

DS (draft specification) This kind of specifications is a pub-lished draft by CiA. It is normally not changed within one year.

DSP (draft specification proposal) — This kind of specification is a proposal, which is CiA internally released. DSPs can be changed anytime without notification.

D-sub connectors — These are standardized connectors. Most common in use is the 9-pin D-sub connector (DIN 41652); its pin-assignment for CAN/CANopen networks is recommended in CiA 106.

edge An edge is a transition of a bus state from dominant to recessive (rising edge) or from recessive to dominant (falling edge).

EDS See *electronic data sheet.*

EDS checker This software tool checks the conformity of electronic data sheets. The CANopen EDS checker is integrated into CiA's CANopen CC conformance test tool.

EDS generator This software tool generates electronic data sheets (available for CANopen and DeviceNet).

EF See *error frame.*

electromagnetic compatibility (EMC) Electromagnetic compatibility (EMC) is a characteristic of electrical equipment to function acceptably in their electromagnetic environment. Accordingly, the equipment does not emit levels of electromagnetic energy that cause electromagnetic interference in other equipment. The aim of EMC is the correct operation of different equipment in a common electromagnetic environment.

electronic data sheet (EDS) The electronic data sheet describes the functionality of a device in a standardized manner. CANopen and DeviceNet use different EDS formats. It is specified in CiA 306-1 for CANopen devices.

electro static discharge (ESD) Electrostatic discharge (ESD) is a sudden and momentary flow of electric current between two electrically charged objects. It is caused by contact, an electrical short or dielectric breakdown. The ESD occurs when differently-charged objects are brought close together or when the dielectric between them breaks down, often creating a visible spark.

EMC See *electromagnetic compatibility.*

EMCY	See *emergency message.*
EMCY consumer	This is a CANopen entity, receiving EMCY messages from other CANopen nodes.
EMCY producer	This is a CANopen entity, transmitting the EMCY message.
emergency message (EMCY)	This is a predefined communication service and protocol in CANopen mapped into a single 8-byte data frame containing a 2-byte standardized error code, the 1-byte error register, and 5-byte manufacturer-specific information. It provides status information of the CANopen EMCY FSA (finite state automaton) and detailed information about detected communication and application errors.
EN 13149-4/5/6	This is a set of CENELEC standards defining a CANopen application profile for passenger information systems, which was developed in cooperation with the German VDV. It specifies interfaces for a range of devices including displays, ticket printers, passenger counting units, main onboard computers, etc.
EN 50325-4	This is a CENELEC standard defining the CANopen application layer and communication profile, which is further developed in the CiA 301 specification.
EN 50325-5	This CENELEC standard specifies the CANopen Safety protocol. The CANopen framework for safety-relevant communication is an add-on to the CANopen application layer and communication profile. The CANopen Safety protocol is designed to allow safety-related communication based on CAN according to IEC/EN 61508. It is approved by German authorities and fulfils the requirements to build systems requiring SIL 3 (safety integrity level) according to IEC 61508.

entry category This is a data object attribute in CANopen defining this object as mandatory, conditional (mandatory for certain conditions), or optional.

end of frame (EOF) Seven recessive bits make the EOF field of CAN data and CAN CC remote frames.

EOF See *end of frame.*

error active state One of the three CAN error states. In error active state the CAN communication of that node is not restricted. If all nodes are in error active state, there is a network-wide data consistency. In error active state the CAN controller is allowed to transmit active error frames containing active error flags.

error code CANopen specifies error codes transmitted in emergency messages.

error control message The CANopen error control messages are mapped to a single 1-byte CAN data frame assigned with a fixed identifier that is derived from the device's CANopen node-ID. It is transmitted as boot-up message before leaving the NMT state initialization and prior to entering the NMT state pre-operational. It is also transmitted periodically by the device (heartbeat) or, if remotely requested (only in CAN CC implementations) by the NMT manager (node guarding).

error counter Each CAN controller implements two error counters, one for received data or remote frames and one for transmitted data or remote frames. They are increased and decreased user-transparently by implemented rules as specified in ISO 11898-1. They are used to determine the current error state of the CAN module (error active, error passive, and bus-off).

error delimiter This is the last field of error frames made up of 8 recessive bits.

error detection capability There are five different failure detection mechanisms in the CAN protocol, which allow the detection of nearly any error in CAN frames. The probability of non-detected failures depends on error rate, bit rate, busload, number of nodes, and error detection capability factor.

error flag This is the first field of error frames comprising 6 bits of the same polarity. A second error flag transmitted by another node can overlap the first error flag partly. Therefore, the maximum length of an error flag is 12 bit.

error frame (EF) This is the CAN frame to indicate the detection of an error condition. It comprises an error flag and an error delimiter.

error globalization Local failures cause the transmission of an error flag, which are regarded as a stuff error forcing the other nodes to transmit error flags. This means the local failure is globalized, so that network-wide data consistency is guaranteed for nodes in error active state. In CAN XL, the error globalization can be switched off.

error passive state This is one of three CAN error states. In error passive state the CAN communication of that node is restricted. CAN controllers are only allowed to transmit passive error frames containing passive error flags. Additionally, CAN controllers in this state have a waiting time in addition to the interframe space (IFS), after they are allowed to start a CAN frame transmission (suspend transmission). If there is one node in error passive state, the network-wide data consistency is not more guaranteed.

error signaling The error signaling is provided by means of transmitting error frames.

error state indicator (ESI) The error state indicator bit in the CAN FD data frame indicates whether the transmitting

	CAN node is in CAN error active (dominant) or passive (recessive) state.
ESD	See *electro static discharge.*
ESI	See *error state indicator.*
event-driven	Event-driven messages are transmitted when a defined event occurs in the device. This can be a change of input states, elapsing of a local timer, or any other local event.
event-driven PDO	An event-driven PDO in CANopen is transmitted whenever a device internal event (e.g., elapsing of PDO's event timer) occurs. If an event-driven PDO is received the protocol software immediately updates the mapped objects in the object dictionary.
event timer	The event timer is assigned in CANopen to one PDO. It determines the frequency of PDO transmission.
expedited SDO	This is a confirmed communication service in CANopen (peer-to-peer) and is specified in CiA 301. It is made up by one SDO initiate message of the client node and the corresponding confirmation message of the server node. Expedited SDOs are used if not more than 4 byte of data is transmitted.
explicit message	The explicit message is a confirmed communication service in DeviceNet used for configuration purposes. It supports segmented transfer in order to transmit information longer than 8 byte.

FAST RX mode	In this mode, the CAN SIC XL trans-ceiver drives the bus state recessive and the receive thresholds are adjusted to distinguish between the bus states level_0 and level_1.
FAST TX mode	In this mode, the CAN SIC XL trans-ceiver drives the bus states level_0 and level_1, which are not able to overwrite each other.
fault confinement	CAN nodes are able to distinguish short disturbances from permanent failures. Defective transmitting nodes are switched off, meaning the node is logically discon-nected from the network (bus-off).
fault confine-ment entity (FCE)	This is a supervisor entity fulfilling the fault confinement.
fault-tolerant transceiver	Transceivers as specified in ISO 11898-3 and ISO 11992-1 are capable of communi-cation via one bus-line and CAN ground when one bus-line is broken down, short circuited or termination resistors are not well connected.
FBFF	See *FD base frame format.*
FCE	See *fault confinement entity.*
FCP	See *format check pattern.*
FCRC	See *frame CRC.*
FD	See *flexible data rate.*
FD base frame format (FBFF)	This format of CAN FD data frames uses an 11-bit identifier. The data frames can be transmitted with a flexible bit rate and include no or up to 64 byte of data.
FD data bit rate	This is related to the CAN FD data bit rate, which specifies the number of bits per time during the FD data phase, independent of the bit encoding/decoding.

FD data bit time — This is the duration of one bit in the FD data phase.

FD data phase — This is the phase, in which the FD data bit time is used.

FD enabled — This means a CAN node is able to receive and transmit FD frames as well as CC frames.

FD extended frame format (FEFF) — This format of CAN FD data frames uses a 29-bit identifier. The data frames can be transmitted with a flexible bit rate and include no or up to 64 byte of data.

FDF — See *FD format indicator.*

FD format indicator (FDF) — This bit distinguishes between CAN CC frames (dominant) and CAN FD frames (recessive). In frames with 11-bit identifiers, FDF comes after the IDE bit. In frames with 29-bit identifiers, it comes as the first bit of the control field.

FD frame — *This includes data frames using FBFF or FEFF format.*

FD intolerant — A CAN node that is only able to receive or to transmit CC frames; it cannot transmit FD frames and the reception of FD frames leads to error frames.

FD tolerant — A CAN device that is not able to transmit or to receive FD frames. However, it does not destroy the CAN FD frame by an error frame.

FEFF — See *FD extended frame format.*

field device — This is an independent physical entity of an automation system, which hosts zero, one or several CANopen devices, and performs specific functions such as controlling, actuating, sensing, and/or data transferring.

finite state automaton (FSA) — This is an abstract machine that can be in exactly one of a finite number of states at any given time. It can transit to other states in response to specified events.

flexible data rate (FD) — This is related to CAN FD, which uses two bit rates: arbitration bit rate and data phase bit rate.

flying manager — In safety-critical applications, it is required that a missing NMT manager is substituted automatically by another stand-by NMT manager. This concept of redundancy is called flying manager.

format check pattern (FCP) — The format check pattern (FCP), comprising four bits, is a part of the CRC field in the CAN XL frame. The FCP sequence located directly before the point where the bit rate is switched back from the data bit rate to the nominal bit rate and where the physical layer is signaled to switch back into the arbitration mode. It provides a synchronizing edge before the transition, from the FCP2 bit to the FCP1 bit. The FCP3 bit and FCP2 bit are transmitted as recessive bits. The FCP1 bit and FCP0 bit are transmitted as dominant bits. The FCP0 bit is the last bit of the XL data phase.

form error — A corruption of one of the pre-defined recessive bits (CRC delimiter, ACK delimiter, and EOF) is regarded as a form error condition that causes the transmission of an error frame in the very next bit time.

frame — This is a protocol data unit of the data link layer specifying the arrangement and meaning of bits or bit fields in the sequence of transfer across the transmission medium.

frame coding — This is a sequence of fields in the CAN frames, e.g., SOF, arbitration field, control field, data field, CRC field, ACK field, and EOF for CAN CC data frames. The frame coding also covers the bit stuffing.

frame CRC (FCRC) — The 32-bit frame CRC (FCRC) is a part of the CRC field in the CAN XL frame. A receiver detects an FCRC error when the calculated FCRC sequence does not equal the received one, or when it detects an error in the format check pattern (FCP).

frame field — This is a sub-part of a frame, e.g., control field, data field, CRC field, and ACK field.

frame format — The ISO 11898-1 standard distinguishes between the base frame format (CBFF and FBFF) using 11-bit identifiers and the extended frame format (CEFF and FEFF) using 29-bit identifiers as well as the XLFF with 11-bit priority identifiers and 32-bit acceptance fields.

frame type (FTYP) — In CAN, four frame types are used: data frame, remote frame (only in CAN CC), error frame, and overload frame.

FSA — See *finite state automaton.*

FTYP — See *frame type.*

FullCAN — A term used in the early days of CAN describing an implementation, which features single receive and transmit buffers for a number of IDs.

function code — The first and most significant four bits of the CAN identifier in the CANopen CC and the CANopen FD pre-defined connection set, which indicate the function (e.g., SDO or USDO request, TPDO or EMCY).

functional element — This is a hardware and/or software entity implementing a specific functionality.

FUP message — It is mapped into a CAN data frame containing the remaining part of the current value of a time-base plus the value of transmit delays of the preceding SYNC message specified by Autosar. The mapping of the FUP message into a CAN data frame is specified in CiA 603.

galvanic isolation Galvanic isolation in CAN networks is performed by optocouplers or transformers placed between CAN controller and CAN transceiver chip.

gateway This is a funtional element that links two or more network segments using different application layer communication services. Such an entity with at least two network interfaces transforms all seven OSI protocol layers, e.g., CANopen-to-Ethernet gateway or CANopen-to-DeviceNet gateway.

GFC See *global fail-safe command.*

global error A global bus error affects all connected CAN devices.

global fail-safe command (GFC) The global fail-safe command (GFC) is a high-priority message specified in the CANopen Safety protocol (see *EN 50325-5*). It is used to switch the SRLDs into the safe state, which improves the overall system reaction time in case of an error. It is followed by the related SRDO.

GND See *ground.*

ground (GND) The ground is considered as the common reference point in the electronic circuits to measure the voltage against any point of the circuit. In many single-voltage digital and analog circuits, the negative (zero) power supply is GND. All the electrical components in a circuit are connected to the GND, as the common connection, in order to perform correctly.

Hamming distance In general, the Hamming distance between two strings of equal length measures the number of errors that transformed one string into the other. In general, CAN CC and CAN FD provide a Hamming distance of 6 (theoretical value for CAN networks). In some unlike scenarios with a very low probability, the Hamming distance is shortcut to 2 (a single bit error is detected). The CAN XL protocol with two cascaded CRCs provides a true Hamming distance of 6. A Hamming distance of 6 indicates that five randomly distributed bit failures are detected. CAN protocols do not provide bit correction mechanisms.

handle This is the label of one or multiple logical link control (LLC) frames (LPDU) or data link layer service data units (LSDU). The data link layer‘s interface data comes from the higher open systems interconnection (OSI) layers (network layer or transport layer). This explanation applies to the scope of ISO 11898-1, only.

hard synchronization All CAN nodes are internally hard synchronized to the falling edge of the SOF bit detected on the bus. Hard synchronization is performed during bus idle, suspend transmission and the third bit of interframe space.

heartbeat CANopen and DeviceNet use the heartbeat message to indicate that a node is still alive. The device transmits this message periodically. In CANopen, this message includes the NMT server’s FSA status as a confirmation of the requested NMT transition.

heartbeat consumer This is a CANopen functional element receiving heartbeat messages from other CANopen nodes.

heartbeat producer — This is a CANopen functional element transmitting the heartbeat message.

higher-layer protocol (HLP) — Higher-layer protocols specify communication protocols compliant with the transport layer, session, presentation, or application layer as specified in the OSI reference model.

HLP — See *higher-layer protocol.*

HS-PMA legacy implementation — This is the name for implementations with function coverage compliant with ISO 11898-2:2003.

ID See *identifier.*

IDE See *identifier extension flag.*

identifier (ID) This is the unique label reflecting the priority of a particular frame.

identifier-based arbitration This is the CSMA/CR arbitration procedure resolving bus contention when multiple nodes simultaneously access the bus.

identifier extension flag (IDE) This bit distinguishes whether the data or remote frame uses the base frame format (dominant) or the extended frame format (recessive). Thus, the IDE bit indicates if the following bits are interpreted as control bits or the second part of the 29-bit identifier.

identifier field In CAN CC and CAN FD frames, the identifier field contains 11 bits in base frame format, and additional 18 bits in extended frame format. Here, the identifier is used for both arbitration and addressing purposes. In CAN XL these functions are separated. The CAN XL protocol separates the priority functions (11-bit priority ID) and the addressing (32-bit acceptance field (AF)).

idle This is an operating condition of the bus after the completion of a frame until the next frame starts. In bus idle, the AUI is in recessive state. The period of bus idle can be of arbitrary length.
The bus is recognized as idle by receivers and by error-active transmitters, when the third bit of intermission is seen recessive; by error-passive transmitters when th 8^{th} bit of suspend transmission time is seen recessive, or when the bus integrating state is left. When the bus is idle, any CAN node can access the bus for transmission.

idle condition This is the detection of a sequence of 11 consecutive sampled recessive bits on the bus.

IEC 61162-3 This is the international standard for digital interfaces for navigational equipment within a ship. Part 3 standardizes the CAN-based serial data instrument network, also known as NMEA 2000.

IEC 61800-7 series This international standard specifies power drive profiles including CiA 402 and CIP motion. The CiA 402 profile mapping to CANopen (61800-7-201/-301) and the CIP motion profile mapping to DeviceNet (61800-7-202/-302) are also specified in this series of standards.

IEC 62026-3 This is the international standard for the CAN-based DeviceNet application layer.

IMF See *intermission field.*

index This is the 16-bit address to access information in the CANopen object dictionary; for arrays and records the address is extended by an 8-bit sub-index.

information processing time (IPT) This time is given as number of time quanta required for the calculation of the subsequent bit level. It begins at the sample point and is less than or equal to Phase Segment 2.

inhibit time This is the parameter in CANopen that determines the minimal time that elapses between transmission of PDOs (using the very same CAN-ID) or the EMCY messages (using the very same CAN-ID).

initialization state See *NMT state initialization.*

inner priority inversion Occurs, if a low-prior CAN data or remote frame is not transmitted, because of high-prior frame traffic on the CAN network and a high-prior transmission request occurs in the device and is not forwarded to the

CAN controller due to the still pending low-prior transmission request.

integrating Status of a CAN node waiting on an idle condition after starting the protocol operation during bus-off recovery or after a protocol exception event.

inter-frame space This is the time between two frames comprising the IMF and bus idle time. For error-passive nodes, which are the transmitters of the previous frame, it also contains the suspend transmission time.

intermission field (IMF) This is the 3-bit field after the EOF. Detection of a dominant bit at the third IMF bit is interpreted as SOF.

I/O message This is the communication object in DeviceNet transporting application objects representing inputs or outputs. I/O messages are mapped to one or more CAN data frames supporting segmented transfer.

IPT See *information processing time.*

ISO 11783 series This is the international standard series specifying the CAN-based application profile used in agriculture and forestry machines and vehicles (Isobus). It is based on the J1939 application profile.

ISO 11898-1 This is the international standard specifying the CAN CC, CAN FD, and CAN XL data link layers (including LLC and MAC) and PCS.

ISO 11898-2 This is the international standard specifying the CAN high-speed PMA sublayers. Since ISO 11898-2:2024, the document includes the so-called CAN HS (high-speed) transceivers (max. 1 Mbit/s), CAN FD transceivers (max. 5 Mbit), CAN SIC transceivers (max. 8 Mbit/s), and CAN SIC XL transceivers (max. 20 Mbit/s). Since ISO 11898-2:2016, the standard also includes the specification for the low-power mode (formerly specified in ISO 11898-5) and selective wake-up functionality (formerly specified in ISO 11898-6).

ISO 11898-3 This is the international standard specifying the CAN fault-tolerant, low-speed PMA sublayer. It is not more recommended for new designs.

ISO 11898-4 This is the international standard specifying a time-triggered communication protocol, which can be adapted to the CAN data link layer protocols.

ISO 11898-5 ISO 11898-5:2007 represented an extension of ISO 11898-2:2003, dealing with functionality for systems requiring low-power consumption features while there is no active bus communication. This standard is now included in ISO 11898-2.

ISO 11898-6 This was the international standard specifying selective wake-up functionality for CAN high-speed transceivers. Transceivers compliant with this standard partly implement the CAN (FD) protocol. This standard is now included in ISO 11898-2.

ISO 11992 series This international standard series specifies the CAN-based communication between truck and trailers. Part 1 specifies the physical medium attachment sublayer (transceiver). Part 2 and part 3 provide the specification of the J1939-based messages for braking/running gear devices respectively all other purposes. Part 4 specifies the diagnostics services.

ISO 15745-2 This is the international standard specifying an application integration framework for ISO 11898-based control systems such as CANopen and DeviceNet.

ISO 15765 series The series of standards specifies the so-called ISO transport protocol (TP) in

Part 2, and in Part 5 the CAN FD interface on the vehicle-side to be connected to external tools.

ISO 16844-4 This international standard specifies the CAN-based data communication between the display unit and other devices connected to an SAE J1939-based in-vehicle network. The provided requirements and recommendations cover physical, data link, network, and application layers. Additionally, it specifies dedicated J1939 parameter groups.

ISO 16845-1 This is the international standard that specifies the conformance test plan for ISO 11898-1 implementations.

ISO 16845-2 This is the international standard that specifies the conformance test plan for CAN high-speed transceivers compliant with ISO 11898-2.

ISO 26021 series This series of international standards specifies the CAN-based interface for the end-of-life activation of on-board pyrotechnic devices.

J1939 application profile
The application profile developed by the nonprofit SAE organization specifies the in-vehicle communication in trucks and buses and other heavy-duty commercial vehicles. It specifies the communication services mapped into parameter groups (PGs).

J2284 series
This is a set of bit timing specifications by SAE for in-vehicle networks in passenger cars for 250 kbit/s and 500 kbit/s. Part 4 and Part 5 specify the use of CAN FD (arbitration phase and data phase at 500 kbit/s and 2 Mbit/s, and at 500 kbit/s and 5 Mbit/s).

J2411
This is a single-wire transmission specification by SAE for CAN networks. The bit rate is limited to 40 kbit/s. It is not more used for new designs.

layer-2 protocol — A layer-2 (means OSI layer) protocol uses the CAN communication services directly from the application software.

layer-7 protocol — A CAN-based layer-7 (means OSI layer) protocol uses CAN communication services. Doing this in a standardized manner allows the reuse of application software without redesigning the CAN communication software.

layer management entity (LME) — The LME is a part of the entity layer, which manages resources and parameters residing in its layer protocol entity.

layer setting services (LSS) — These are services and protocols to configure CAN bit-rate settings, the node-ID of the CANopen (FD) device, or the network-ID of the CANopen FD device. LSS are specified in CiA 305 (CANopen CC) respectively CiA 1305 (CANopen FD).

least significant bit (LSB) — LSB is the bit with the lowest significance in a multi-bit binary number, which represents the first place (usually the farthest to the right) of the binary number.

life guarding — This is the method in CAL and CANopen CC to detect that the NMT manager does not guard the NMT server anymore. This is a part of the error control mechanisms.

line topology — Network topology, where all nodes are connected directly to one bus line. CAN networks use line topologies without stub-cables (daisy chain), in order to minimize reflections. However, in practice even tree and star topologies are found.

LLC — See *logical link control.*

LLC service data unit (LSDU)	This is the service data unit, which is transmitted between the LLC sublayer and the MAC sublayer. There is no LLC protocol data unit specified, it is implementation-specific.
LME	See *layer management entity.*
LMT protocols	These are protocols specified in CAL for setting node-IDs and bit rates via the CAN network.
local bus error	A local bus error affects just one or more but not all nodes in the CAN network.
logical device	This is the logical entity of a CANopen device providing status, control, and diagnostic information to the CANopen device in a pre-defined format.
logical link control (LLC)	This is a sublayer of the CAN data link layer connecting the higher OSI layers with the MAC sublayer.
logical network system	This is a network system not mapped to physical network segments.
low-power mode	In this mode (stand-by or sleep mode) the transceiver is not capable of transmitting or receiving frames, except for the purposes of determining if a WUP (wake-up pattern) or WUF (wake-up frame) is being received.
low-speed transceiver	This is a synonym for fault-tolerant transceivers as standardized in ISO 11898-3.
LSB	See *least significant bit.*
LSDU	See *LLC service data unit.*
LSS	See *layer setting services.*
LSS manager	This is a layer management entity that configures the node-ID of the CANopen (FD) device, the network-ID of the CANopen FD device, or the CAN bit-rate settings.

LSS master Outdated term, see *LSS manager.*

LSS server This is a layer management entity in a CANopen device that evaluates the LSS messages sent by the LSS manager.

LSS slave Outdated term, see *LSS server.*

MAC See *medium access control.*

MAC protocol data unit (MPDU) The MAC protocol data unit (MPDU) is the protocol exchanged between two CAN controller entities.

manager This is a communication or application entity that is allowed to control a specific function. In networks, this is for example the initialization of a communication service.

matrix cycle In TTCAN (ISO 11898-4), the matrix cycle is made up of one or more basic cycles. Each basic cycle starts with the reference message and is followed by different CAN data frame slots.

MDI See *medium dependent interface.*

MDI state This is a state of the MDI, which is dominant or recessive if the PMA sub-layer is in arbitration mode, or is in level_0 or level_1 state. The dominant state represents the logical 0 and the recessive state represents the logical 1. During simultaneous transmission of dominant and recessive bits, the resulting bus state is dominant. When no transmission is in progress, the bus is idle. During the idle time, it is in recessive state. The level_0 state represents the logical 0, and the level_1 state represents the logical 1.

medium access control (MAC) The MAC sublayer represents the lower part of the OSI data link layer. It links the LLC sublayer and the PCS physical sublayer, and comprises the functions and rules that are related to data en-/decapsulation, and error detection.

medium dependent interface (MDI) This is the electrical interface consisting of CAN_H and CAN_L, that defines the signal transfer between the PMD sublayer and the PMA sublayer.

message — This is an entity of the application layer protocol, which is mapped to CAN data and remote frames in CAN-based communication systems. In most CAN-based application layers, CAN remote frames are not used or it is not recommended to use them.

message buffer — CAN implementations provide message buffers for data frames to be received and/or to be transmitted. The implementation and the usage of message buffers are not standardized.

MilCAN — These CAN-based higher-layer protocols, as specified by a group of interested companies and government bodies, are intended for military vehicles. MilCAN A is based on J1939, and MilCAN B is based on CANopen CC.

minimum time quantum (mtq) — This is the smallest time quantum that can be configured for the specific node. It is equal to one CAN clock period.

most significant bit (MSB) — MSB is the bit with the highest significance in a multi-bit binary number, which represents the last place (usually the farthest to the left) of the binary number.

MPDO — See *multiplex PDO.*

MPDU — See *MAC protocol data unit.*

MSB — See *most significant bit.*

mtq — See *minimum time quantum.*

multicast transmission — This is an addressing schema, where a PDU is addressed to a group of nodes simultaneously. A broadcast is a special case of multicast, whereby a single PDU is addressed to all nodes simultaneously.

Multi-PDU — This is a communication entity comprising several contained PDUs (C-PDUs).

multiplex PDO (MPDO) In CANopen CC, the MPDO is made of eight bytes including one control byte, three multiplexer bytes (containing the 24-bit index and sub-index), and four bytes of object data.

network-ID In systems that integrate several CANopen networks, this number identifies CANopen networks uniquely. CANopen supports up to 127 networks in hierarchical or non-hierarchical network systems as specified in CiA 302-7 and CiA 1301.

network length See *bus length.*

network management (NMT) This is the CANopen application layer function responsible for the network boot-up procedure, control of the node's FSA, and the optional configuration of nodes.

network variables These are application parameters that represent not yet assigned process data of programmable CANopen devices. Network variables can be mapped into PDOs after programming the device.

NMEA 2000 This is a combined electrical and data specification for a marine data network for communication between marine electronic devices such as depth finders, nautical chart plotters, navigation instruments, engines, tank level sensors, and GPS receivers. The J1939-based application profile has been developed by NMEA (National Marine Electronics Association), US non-profit organization. It is standardized in IEC 61162-3.

NMT See *network management.*

NMT manager The NMT manager device in CAL and CANopen performs the network management by means of transmitting the NMT message. With this message, it controls the state machines of all connected NMT server devices.

NMT manager FSA This is the FSA of a CANopen device with NMT manager functionality. It covers the states NMT manager initial (indicates FSA start), NMT manager startup capable device (no or limited NMT manager functionality is provided), NMT manager inactive (no or limited functionality e.g., scanning for NMT manager capable devices is provided), NMT manager active (entire supported functionality is active), and NMT manager final (indicates FSA end).

NMT master Outdated term, see *NMT manager.*

NMT message This is a COB, which is provided by the NMT manager entity and is evaluated by NMT server entities, for controlling the transitions of the NMT server FSA.

NMT reset application This NMT command resets all CANopen objects to default values or to the permanently stored configured values.

NMT reset communication This NMT command resets only the CANopen communication objects to the default values or to the permanently stored configured values. This NMT state is divided in sub-states waiting for node-ID, resetting, and request boot-up.

NMT server The NMT server receives and evaluates the NMT message, which contains commands for the NMT state machine implemented in CAL and CANopen devices.

NMT server state machine The NMT server state machine specified in CAL and CANopen supports different states. The NMT manager controls the transition to the states via the highest prior CAN frame (ID = 0) transmitted.

NMT slave Outdated term, see *NMT server.*

NMT startup capable device This is the CANopen device, which is able to enter the NMT state operational after the NMT state initialization autonomously (self starting).

NMT state initialization This is the NMT server state in CANopen that is reached automatically after power on and communication or application reset.

NMT state operational This is a part of the CANopen NMT server state machine. In this state, all CANopen communication services are available.

NMT state pre-operational This is a part of the CANopen NMT server state machine. In this state, no PDO communication is allowed.

NMT state stopped This is a part of the CANopen NMT server state machine. In this state, only NMT messages are performed and, under certain conditions, error control messages are transmitted.

node This is an assembly, linked to a communication network, capable of communicating across the network according to a communication protocol specification. A node operating in a controller area network (CAN) is called a CAN node.

node clock This is a time base to coordinate the bit-time-related state machines in CAN nodes.

node guarding This is part of the error control mechanisms used in CANopen CC and CAL to detect bus-off or disconnected devices. The NMT manager sends a remote frame to the NMT server that is answered by the corresponding error control message. This mechanism is not supported in CANopen FD.

node-ID This is the unique identifier for a device required by different CAN-based higher-layer protocols in order to assign CAN identifiers to this device, e.g., in CANopen or DeviceNet. Using the predefined connection sets of CANopen or DeviceNet, the node-ID is part of the CAN identifier.

nominal bit rate This is the number of bits per time transmitted in the absence of resynchronization by an ideal transmitter, independent of the bit-encoding/decoding.

It is used in CAN CC data frames and in the arbitration phase of the CAN FD and CAN XL data frames. CAN error frames and CAN overload frames use it, too.

non-return-to-zero (NRZ)	This is the bit coding method of representing binary signals, in which the binary low and high states are represented by numerals 0 and 1; i.e., within one and the same bit time, the signal level does not change, where a stream of bits having the same logical value provides no edges.
normal-power mode	This is the mode, in which a transceiver is capable of transmitting and receiving frames.
normal SDO	See *segmented SDO*.
NRZ	See *non-return-to-zero.*

object dictionary — The object dictionary is the heart of any CANopen device. It enables access to all data types used in the device, to the communication parameters, as well as to the process data and configuration parameters addressable using a 16-bit index and an 8-bit sub-index.

OF — See *overload frame.*

open systems interconnection (OSI) reference model — This is an abstract representation of a communication system in seven layers, each performing a specific subset of functionalities required for the communication system. This communication model defines seven layers: physical (1), data link, network, transport, session, presentation, and application (7) layer. In CAN-based networks normally just physical, data link, and application layer are implemented. The physical layer and data link layer are specified in ISO 11898-2 and ISO 11898-1. The OSI reference model is specified in ISO/IEC 7498-1 and ISO/IEC 10731.

operational state — See *NMT state operational.*

OSEK/VDX — This is a set of specifications for communication (COM), NMT, real-time operating system (OS), and OSEK implementation language (OIL). OSEK/VDX is partly implemented in passenger cars.

OSI reference model — See open systems interconnection reference model.

outer priority inversion — If a CAN node wants to transmit two high-prior data or remote frames and is not able to send the second message directly after the intermission field, it can happen that a lower-prior data or remote frame is transmitted by another node in between. This is called outer priority inversion.

overload condition — An overload condition occurs in situations when the CAN controller transmits an overload frame: e.g., dominant value in the first two interframe space bits, dominant value in the last bit of EOF, bit failure in last bit of error or overload delimiter.

overload delimiter — This is the last field of overload frames made up of 8 recessive bits.

overload flag — This is the first field of overload frames made up of six dominant bits. A second overload flag transmitted by another node can overlap the first overload flag.

overload frame (OF) — This is the CAN frame to indicate an overload condition. It is made up of the overload flag and the overload delimiter. The overload flag corresponds to that of the active error flag. The overload delimiter is the same as the error delimiter. Overload flags do not affect the error counters.

packet This is an entity of the network layer protocol.

padding sub-field This is the sub-field of the data field containing meaningless bits in a fixed format in order to pad the data field to a defined byte limit.

parameter field This is a sub-part of a structured parameter.

parameter group (PG) This is an assembly of suspect parameters in J1939, ISO 11783, and ISO 11992. This message is identified by the PG number (PGN). The PG is mapped to one or more CAN data frames.

parameter group number (PGN) The parameter group number (PGN) uniquely identifies the PG. The PGN is mapped into the 29-bit CAN identifier field.

passive error flag The passive error flag is the first part of the passive error frame made up of six consecutive recessive bits.

passive recessive This is the final high-speed physical medium attachment (HS-PMA) output drive with a nominal impedance, also known as recessive.

PCRC See *preface CRC.*

PCS See *physical coding sub-layer.*

PDO See *process data object.*

PDO consumer This is the CANopen functional entity receiving PDO messages from other CANopen nodes.

PDO mapping In CANopen, up to 64 objects can be mapped in a PDO. The PDO mapping is described in the PDO mapping parameters.

PDO producer This is the CANopen functional entity transmitting PDO messages.

PDU See *protocol data unit.*

pending transmission request There are one or more data or remote frames waiting for transmission in the CAN controller because the bus is not idle (node loses arbitration).

PG See *parameter group.*

PGN See *parameter group number.*

phase error The phase error of an edge is given by the position of the edge relative to the sync segment. It is measured in time quanta.

phase segment 1 (Phase_Seg 1) This is the part of the bit time used to compensate for edge phase errors. It can be lengthened by resynchronization.

phase segment 2 (Phase_Seg 2) This is the part of the bit time used to compensate for edge phase errors. It can be shortened by resynchronization.

physical coding sub-layer (PCS) This is a sublayer of the OSI physical layer. It receives from and sends to the transceiver circuitry the bit stream and performs the bit en/decoding, controls the bit timing and synchronization.

physical layer (PL) This is the lowest layer in the OSI reference model. It is divided in three sublayers: PCS (physical signaling), PMA, and PMD.

physical medium attachment (PMA) This is a sublayer of the OSI physical layer. It specifies the functional circuitry for bus line transmission/reception. It converts logical signals to physical signals and vice versa.

physical medium dependent (PMD) This sublayer of the OSI physical layer includes optional common mode choke, termination, network, ESD protection, and signal improvement circuitry. It also specifies optional protection circuitry, galvanic isolation, connectors, cables, etc.

PL	See *physical layer.*
PMA	See *physical medium attachment.*
PMA comparator	A PMA comparator is an electronic device or an integrated circuit (IC) that commonly is used for converting analog signals to digital signals, in order to transfer across the communication medium. A comparator including two analog input terminals and one digital output compares two input voltages and indicates the larger one, then determines the output digital (binary) voltage, whether is 0 or 1.
PMD	See *physical medium dependent.*
pre-defined connection set	This is the set of CAN identifiers used as default values for different communication protocols in CANopen or DeviceNet.
preface CRC (PCRC)	The preface CRC (PCRC) is a 13-bit field, which belongs to the control field of a CAN XL frame. This is one of the two CRCs for additional protection over the transmitted data. As the other CRC field (FCRC), PCRC is able to detect any five randomly distributed bit-errors.
pre-operational state	See *NMT state pre-operational.*
priority	This is the attribute of a frame controlling its ranking during arbitration. In CAN data and remote frames (only in CAN CC), the identifier (ID) gives the priority. The lower the ID, the higher is the priority. A high priority increases the probability that a frame wins the arbitration process.
priority identifier	This is an 11-bit sub-field in the arbitration field of a CAN XL frame, which provides the assigned priority of the CAN XL data frame. The assignment needs to be unique in a CAN network.
priority inversion	Priority inversion occurs, when lower prior data frames are processed or

communicated before the higher prior data frames. In not well-designed CAN nodes, inner or outer priority inversions can occur.

process data This is an application parameter that represents values from process interface inputs or values to the process interface outputs. It is a parameter in the CANopen object dictionary that is mapped into PDOs.

process data object (PDO) This is a CANopen communication object defined by the PDO communication parameter and PDO mapping parameter objects. It is an unconfirmed communication service without protocol overhead. A PDO contains up to 64 byte of data. For example, it provides such process data as commands, status information, or measured data.

producer In CAN-based networks, a transmitter of messages is called a producer.

producer heartbeat time In CANopen, the producer heartbeat time determines the transmission frequency of a heartbeat message.

propagation segment (Prop_Seg) This is a part of the bit time used to compensate physical delay times within the network. These delay times consist of the signal propagation time on the bus line and the internal delay times in the nodes.

protocol This is a formal set of conventions and rules for the exchange of information between nodes, including the specification of frame administration, frame transfer and physical medium attachment (PMA).

protocol data unit (PDU) This is a unit of data specified in a protocol, consisting of protocol control information and user data. A PDU is exchanged between peer entities of an OSI layer implementation. This term is also used in Autosar and other standards and specifications.

protocol exception event	This is either an exception from the formal set of conventions or rules to be able to tolerate future new frame formats, or a reaction to errors when CAN XL is used with error signalling disabled.
pulse width modulation (PWM)	This is a modulation method for controlling analog devices, such as motors, lights, actuators, by using a modified digital signal, which is similar to an analog signal. The aim of PWM is controlling a load by switching (ON and OFF) between 0 % and 100 % of the supply that generates variable-width pulses. Long-time remaining ON generates a high-amplitude signal and long-time remaining OFF generates a low-amplitude signal. If the PWM switching frequency is selected precisely, the output analog-like signal is smooth.
pulse width modulation long phase time (PWML)	This is a PWM configuration parameter that defines the long-time period of the PWM symbol. A PWM symbol consists of a long-time period and a short-time period. The sum out of both period times results in the nominal PWM symbol length.
pulse width modulation offset time (PWMO)	This is a PWM configuration parameter, which is used for phase alignment in case of very odd bit-rate relations between arbitration and data bit rate.
pulse width modulation short phase time (PWMS)	This is a PWM configuration parameter that defines the short-time period of the PWM symbol. A PWM symbol consists of a long-time period and a short-time period. The sum out of both period times results in the nominal PWM sym-bol length.
PWM	See *pulse width modulation.*
PWM decoding (PWMD)	This is a PMA sublayer function, which decodes the PWM bit-streams into the NRZ bit-streams between PMA and AUI sublayers, in order to control the physical behavior of the PMA sublayer implicitly through the output bit stream. CAN XL

nodes provide the PWM decoding function to switch the operating modes of the CAN transceivers.

PWM encoding (PWME) This is a PCS sublayer function, which encodes NRZ-bit-streams into the PWM-bit-streams between PCS and AUI sublayers, in order to control the physical behavior of the PMA sublayer implicitly through the output bit stream. CAN XL nodes provide the PWM encoding function to switch the operating modes of the CAN transceivers.

PWMD See *pulse width modulation decoding.*

PWME See *PWM encoding.*

PWML See *pulse width modulation long phase.*

PWMO See *pulse width modulation offset time.*

PWMS See *pulse width modulation short phase.*

radio frequency (RF) — Radio frequency (RF) is a type of electromagnetic field in the approximate frequency range from 20 kHz to 300 GHz. It is used for data transmission through electromagnetic signals among wireless equipment.

re-arbitration — Re-arbitration means re-starting the transmission of a CAN data or remote frame that lost bus arbitration.

receive error counter (REC) — This is a CAN controller internal counter for reception errors. The REC value is readable in some controllers.

receive PDO (RPDO) — This is a process data object that is received by a CANopen device.

receiver — A CAN node is called receiver or consumer, if it is not transmitting and the bus is not idle.

reception buffer(s) — This is the local memory in the CAN controller, where the received messages are stored intermediately.

recessive bit — A recessive bit is overwritten by a dominant bit. It has the logical value 1.

recessive state — By definition, the recessive state is overwritten by the dominant state.

recovery time — This is the time between the first bit of the error flag and the time point when the automatic retransmission is started. In error active nodes, the maximum recovery time is 23 bit times, in error passive nodes it is 31 bit times.

redundant networks — In some safety-critical applications (e.g., maritime systems), redundant networks are required that provide swapping capability in case of detected communication failures.

reference message — In TTCAN, the reference message starts each basic cycle.

remote frame (RF) With an RF (only in CAN CC) another node is requested to transmit the corresponding data frame identified by the same CAN-ID. RF's DLC has the value of the corresponding data frame DLC. RF's data field has a length of 0 byte.

remote request substitution (RRS) In CAN FD frames, the RRS bit is transmitted at the position of the RTR bit in CAN CC frames. It is transmitted dominantly, but receivers accept recessive and dominant RRS bits.

remote transmission request (RTR) In CAN CC, this is the bit in the arbitration field indicating if the frame is a remote frame (recessive value) or a data frame (dominant value).

repeater This is a passive component that refreshes CAN bus signals. It is used to increase the maximum number of nodes, to achieve longer networks (>1 km) or to implement tree or meshed topologies.

reserved bit XL format (resXL) This is a reserved bit in arbitration field for future expansion of the protocol that is dominant.

reset A CAN controller is reset by a command (is hard-wired). Before the CAN controller transits back to error active state, it detects 128 occurrences of the idle condition (11 consecutive recessive bit times) on the bus.

reset application See *NMT reset application.*

reset communication See *NMT reset communication.*

resXL See *reserved bit XL format.*

retransmission This means re-starting the transmission of a CAN data or remote frame that has not been successfully transmitted for any reason except lost bus arbitration.

RF See *remote frame.*

RPDO	See *receive PDO.*
RRS	See *remote request substitution.*
RTR	See *remote transmission request.*
RXD	This is one port of the AUI used to transmit the actual state of the physical medium, in binary format, to the PCS.

safe-guard cycle time (SCT) — This determines the maximum time between two periodically transmitted SRDOs (see *EN 50325-5*).

safety-related data object (SRDO) — This is a COB for safety-related data transfer as standardized in the CANopen Safety protocol (EN 50325-5). It is made of two parts mapped to two CAN CC data frames. The second data frame contains the bit-wise inverted data of the first data frame.

safety-related logical device (SRLD) — This is a CANopen device participating in the safe communication as specified in EN 50325-5.

safety-related object validation time (SRVT) — This determines the maximum time between the two parts of an SRDO (see *EN 50325-5*).

SafetyBus p — This CAN-based higher-layer protocol and implementation specification by the Safety Network International e. V. is dedicated to safety-related communication within factory automation. It meets the SIL 3 (safety integrity level) according to IEC 61508.

sample point — The sample point is the point of time at which the bus level is read and interpreted as the value of the respective bit. Its location is between Phase_Seg 1 and Phase_Seg 2. In CAN FD, for bit rates exceeding the 1 Mbit/s, the secondary sample point is considered.

SAP — See *service access point.*

SBC — See *stuff bit count.*

SCT — See *safe-guard cycle time.*

SDO — See *service data object.*

SDO block transfer SDO block transfer is a CANopen CC communication service for increasing the speed of uploading/downloading data to/from a CANopen CC device. It is specified in CiA 301. In SDO block transfer, the confirmation is sent after the reception of a number of SDO segments.

SDO client The CANopen CC SDO client initiates the SDO communication by means of reading or writing to the object dictionary of the CANopen CC SDO server.

SDO manager In CANopen CC, the SDO manager handles the dynamic establishment of SDO connections. It resides on the very same node as the NMT manager functionality.

SDO network indication This service is used to address a remote CANopen CC device in another (not directly accessible) CANopen CC network. This service establishes a virtual channel in order to perform any SDO communication (see *CiA 302-7*). The SDO services are specified in CiA 301.

SDO server The CANopen CC SDO server receives the SDO request (see *CiA 301*) from the corresponding CANopen CC SDO client and sends the SDO response (expedited and segmented SDO transfer) or a block of SDO segments (SDO block transfer).

SDT See *SDU type.*

SDU See *service data unit.*

SDU type (SDT) This is an 8-bit field in the control field of a CAN XL frame that indicates, which higher-layer protocol is used by the sender.

SEC See *simple extended content.*

secondary sample point (SSP) When in a CAN FD data frame sending node the transmitter delay compensation is enabled, the bus state is sampled at the SSP.

segment — There are two meanings:

a) segments of a bit (synchronization, propagation, phase 1, and phase 2)
b) segments of the OSI transport layer protocols (e.g., first, initial, consecutive).

segmented SDO — If objects (parameters) longer than 4 byte are transmitted by means of SDO services (see CiA 301 for CANopen CC), a segmented transfer is used. The data is transmitted in segments of up to 7 byte of application data. Theoretically, the number of segments is not limited.

service access point (SAP) — An SAP refers to the boundary between the data link layer and the network layer in the OSI reference model. It identifies a particular user service that sends and receives a specific class of data. This user service allows different classes of data to be routed separately to their corresponding service handlers.

service data object (SDO) — The SDO is a confirmed communication service (see CiA 301) that provides access to all entries in the CANopen CC object dictionary. The SDO request and the SDO response are mapped to one or multiple CAN CC data frames with an 8-byte payload. The SDO downloads or uploads messages with any amount of data. Each segment (segmented SDO) or a number of segments (SDO block transfer) is confirmed.

service data unit (SDU) — These are data packets, which are received by a lower OSI layer, from an upper OSI layer.

SIC — See *signal improvement capability.*

SIC mode — This is the mode, which is implemented by SIC or SIC XL transceivers. Such transceivers can suppress ringing on the bus lines by means of dynamic impedance adjustments.

signal improvement capability (SIC) — This is the capability of CAN transceivers, which supports the suppression of ringing on the bus lines. Such transceivers are suitable for complex network topologies with not terminated stub lines.

simple extended content (SEC) — This is a 1-bit sub-field of the control field in a CAN XL frame, which indicates whether the CAN XL data frame uses add-on functions specified by CiA or an international standardization body.

single-shot transmission — Some CAN controllers provide a single-shot mode, which means that the CAN data or remote frame is not retransmitted automatically when an error is detected. This mode is required for TTCAN.

single-wire CAN (SWC) — This is a physical layer using only one bus line and CAN ground. The SAE specified a SWC transceiver in J2411.

SI unit — International system of units (SI), also known as physical units, are standardized in ISO 80000-1:2013. CiA supports this standard in all of its specifications.

SJW — See *synchronization jump width.*

sleep mode — CAN controller and transceiver are operated in stand-by or low-power (sleep) mode no longer driving the bus lines.

SOF — See *start of frame.*

source address mode (SAM) — In the SAM mode of a CANopen MPDO, a multiplexer (16-bit index and 8-bit sub-index) refers to the MPDO producer. The MPDO producer can use the scanner list (parameters to be sent). The MPDO consumers use a dispatcher list indicating which source multiplexer references to which destination multiplexer.

SRDO — See *safety-related data object.*

SRLD — See *safety-related logical device.*

SRR — See *substitute remote request.*

SRVT — See *safety-related object validation time.*

SSP — See *secondary sample point.*

star topology — In some passenger cars, CAN networks are installed in a star topology terminating the network in the center of the star.

start of frame (SOF) — This is the very first bit of CAN data frames and CAN remote frames. The SOF state is always dominant.

stopped state — See *NMT state stopped.*

stuff-bit — Whenever a CAN transmitter detects five consecutive bits of identical value in the bit stream (except in EOF and error/overload delimiter), it automatically inserts a complementary stuff-bit. The CAN receiver excludes the stuff-bits automatically, so that the original bit stream to be transmitted is the very same as the received one. It is used for automatic re-synchronization in the CAN module's bit timing circuitry. In CAN FD, in the CRC field, there is a fixed stuff-bit any fifth bit. This fixed stuff-bit has the opposite value of the previous bit.

stuff bit count (SBC) — This is a 3-bit field of the control field in a CAN XL frame, which indicates the number of stuff bits in a frame before the CRC field, not including fixed stuff bits.

stuff count — In CAN FD frames, the stuff count is at the beginning of the CRC field. It consists of the stuff bit count modulo 8 in a 3-bit Gray code followed by a parity bit.

stuff error — A stuff error is detected at the bit time of the sixth consecutive equal bit level in SOF, arbitration, control and data fields, as well as in the CRC sequence.

stuff-rate — This is the repetition rate of stuff bits in a bit sequence, in which the stuff bits are inserted at fixed positions.

sub-index	This is the 8-bit sub-address to access the sub-objects of arrays and records in a CANopen object dictionary.
substitute remote request (SRR)	This bit is transmitted only in CEFF and in FEFF after ID-bit 18, at the position of the RTR bit in CBFF or of the RRS bit in FBFF. The SRR is transmitted recessively, but receivers accept recessive and dominant SRR bits.
suspend transmission	CAN controllers in error passive mode wait additional eight bit times before the next data or remote frame (only in CAN CC) is transmitted.
SWC	See *single-wire CAN.*
SYNC	See *synchronization message.*
SYNC consumer	This is the CANopen functional element receiving the SYNC message.
SYNC counter	The optional parameter SYNC counter is used in CANopen networks to determine an explicit relationship between the current SYNC cycle and PDO transmission.
SYNC message	This message is mapped into a CAN data frame, containing one part of the current value of a time-base as specified in CiA 603.
synchroniza-tion message (SYNC)	This is a dedicated CANopen message (see *CiA 301* and *CiA 1301*) forcing the receiving nodes to sample the inputs mapped into synchronous TPDOs. Receiving this message causes also the node to set the out-puts to values received in the previous synchronous RPDO. Thus, this message also triggers synchronous actuation in different CANopen nodes.
SYNC producer	This is the CANopen functional element transmitting the SYNC message.
sync segment (Sync_Seg)	This is the part of the bit time used to synchronize various nodes on the bus. An edge is expected within this segment.

synchronization jump width (SJW) This is the number of time quanta with which the Phase_Seg 1 is lengthened or the Phase_Seg 2 is shortened.

system clock This is the time base to coordinate the state machines in CAN implementations.

system variable This is an application parameter that represents undefined shared process data of field devices with multiple CANopen devices.

TEC	See *transmit error counter.*
termination resistor	In CAN high-speed networks with bus line topology, both ends are terminated with resistors (120 Ω) in order to suppress reflections.
thick cable	The thick cable is specified in the physical layer definitions of the DeviceNet specification. This cable is used for networks longer than 100 m.
thin cable	The thin cable is specified in the physical layer definitions of the DeviceNet specification. This cable is used for drop lines and networks shorter than 100 m.
TIME	See *time message.*
TIME consumer	This is the CANopen functional entity receiving the TIME message.
time message (TIME)	This is the standardized message in CANopen containing the time as a 6-byte value given in ms after midnight and days after January 1, 1984.
TIME producer	This is the CANopen functional entity transmitting the TIME message.
time quantum/time quanta (tq)	This is the atomic time unit in a CAN network, derived from the local oscillator frequency.
time stamp	Some CAN controllers provide the possibility of assigning time information to each received message. For TTCAN level 2 it is also required that the transmitting node captures the time and includes the time stamp in the data field of the very same frame.
time-triggered communication	This is a CAN data frame scheduling option, where the frame is transmitted in a defined time slot. In order to achieve a networkwide synchronization the auto-

matic retransmission of frames needs to be disabled. A time-triggered communication avoids bus-arbitration collisions of CAN data and remote frames.

time-triggered messages	They are transmitted in pre-defined time slots. This requires a global time-synchronization and the avoidance of automatic retransmission of faulty messages. Time-triggered communication for CAN is standardized in ISO 11898-4 (TTCAN).
topology	This is the physical connection structure of the network (e.g., line, ring, star, and tree topology).
TOS	See *type of services.*
TPDO	See *transmit PDO.*
tq	See *time quantum/time quanta.*
transceiver	This is an electronic circuitry, usually an integrated circuit (IC), implementing the PMA sublayer. It connects a CAN node to a CAN network and consists of a bus comparator and a bus driver.
transceiver mode	This is an operating mode of the physical medium attachment (PMA) sublayer.
transmission buffer(s)	This is the local memory in the CAN controller, where the message to be transmitted is stored.
transmission request	This is an internal event in the CAN controller to transmit a message.
transmission time capture	In TTCAN level 2 it is required to capture the time when the SOF bit of the reference message is transmitted.
transmission type	This is the CANopen object defining the scheduling of a CANopen communication object such as PDO.
transmit error counter (TEC)	This is the CAN controller's internal counter for transmission errors. The TEC value is readable in some controllers.

transmit PDO (TPDO)	This is a process data object that is transmitted by a CANopen device.
transmitter	This is a node from which a data frame or a remote frame (only in CAN CC) originates. It remains transmitter until the bus is idle again or until the node loses arbitration.
transmitter delay (TD)	This is the delay from the CAN FD controller's transmission flip-flop (FF) to its receiving flip-flop. When the CAN FD controller sends a bit, this bit appears at the CAN FD controller's receive pin after TD. TD includes the microcontroller internal delay, the transceiver delay, and the delay on the ECU. This term is specified in ISO 11898-1. In CiA 601-1 the term transmitting node delay has the same meaning.
transmitter delay compensation (TDC)	At bit rates higher than 1 Mbit/s in the data phase of CAN FD frames, the transmitting node compensates the TD when comparing its transmitted bits to the delayed received bits. The TDC mechanism determines the secondary sample point (SSP).
tree topology	This is a network topology with a trunk line and branch lines. The not terminated branches cause reflections, which can lead to malfunctions.
TSEG1	This value includes the propagation segment as well as the Phase_Seg 1 of a bit time.
TSEG2	This value is the same as the Phase_Seg 2 of a bit time.
TTCAN protocol	This is a higher-layer protocol defining time-triggered communication in CAN-based networks. The CAN controllers are capable of switching-off automatic retransmission of faulty messages and are able to capture a 16-bit timer value at SOF transmission in order to transmit the timer value in the very same message. It is standardized in ISO 11898-4.

TXD This is one port of the attachment unit interface (AUI) driven by the physical coding sublayer (PCS) to control how the physical medium attachment (PMA) influences the actual state of the physical medium.

type of services (TOS) The 3-bit TOS field of a C-PDU indicates whether it is a J1939 mapping C-PDU or a padding C-PDU.

unit (functional) Unit specifies a virtual device that is controlled by another virtual device. For example, in a lift control system, the car door unit is controlled by the car door controller.

universal service data object (USDO) The USDO communication services introduced in CiA 1301 are intended for configuration and diagnostic tasks in CANopen FD systems. It offers confirmed communication between one USDO client and one or several USDO servers. The services provide read and write access to one or several sub-indices in the USDO servers' object dictionaries. Transfer of any data size is possible. The inherent routing capability enables data transfer between different CANopen FD networks i.e., the remote access to CANopen FD field devices in other network segments.

USDO See *universal service data object*.

USDO client The CANopen FD USDO client initiates the USDO communication by means of reading or writing to the object dictionary of the addressed CANopen FD USDO servers.

USDO server This is the responder to an USDO read or write service request in CANopen FD.

value definition This is the detailed description of the value range of a variable in CANopen profiles.

value range This is the object attribute in CANopen defining the allowed values supported by this object.

VCID See *virtual CAN network ID.*

virtual CAN network ID (VCID) This identifier is an 8-bit field of the control field in a CAN XL frame. It enables running up to 256 logical networks by the same SDT, on one single CAN XL physical network segment. This field is also an OSI layer management information as described in ISO 7498-4:1998.

virtual device This is a functional entity within a logical device compliant to a CiA application profile.

wake-up frame (WUF)	Wake-up frame (WUF), as specified in ISO 11898-2, is a dedicated CAN CC frame with a defined ID that enables a high-speed physical media attachment (HS-PMA) to decode CAN frames in either CBFF or CEFF, in normal- and low-power modes.
wake-up pattern (WUP)	Wake-up pattern (WUP), as specified in ISO 11898-2:2016, is a mechanism that restarts selective wake mode in low power (sleep) mode. A transceiver, which is in sleep mode, detects a WUP on the high-speed CAN and changes the mode of operation accordingly.
wake-up procedure	This procedure is used to wake-up a CAN transceiver or a CAN module that is in sleep mode.
WUF	See *wake-up frame.*
WUP	See *wake-up pattern.*

XL	See *extended data length.*
XL data bit rate	This is the number of bits per time during XL data phase, independent of bit-encoding/decoding.
XL data bit time	This is the duration of one bit in the XL data phase.
XL data phase	This is a phase, in which the XL data bit time is used.
XLF	See *XL format indicator.*
XLFF	See *XL frame format.*
XL frame	This is a data frame using the XL frame format.
XL frame format (XLFF)	This is a format for data frames using an 11-bit identifier, including up to 2048 data bytes and dedicated configuration fields for higher layers, e.g., the SDT field and the VCID field. The bit rate is normally switched to a higher bit rate at the start of the data phase. At the end of the data phase, it is switched back to the nominal bit rate.
XL format indicator (XLF)	This is a 1-bit field of the arbitration field in the CAN XL frame, which distinguishes between FD frames and XL frames. The XLF bit is recessive. When a node transmits a recessive XLF bit and it samples a dominant XLF bit, it interprets this as an arbitration lost situation.

Imprint

Editors:	Olga Fischer Oskar Kaplun Maryam Khaghani Dr. Martin Merkel Thilo Schumann Holger Zeltwanger Reiner Zitzmann

Publisher:	CAN in Automation e. V. Kontumazgarten 3 DE-90429 Nuremberg Phone +49-911-928819-0 Fax +49-911-928819-79 headquarters@can-cia.org www.can-cia.org
13th edition:	2026
Copies:	500

Printed in Germany

CAN in Automation (CiA) e. V.
Kontumazgarten 3
DE-90429 Nuremberg
Phone: +49-911-928819-0
Fax: +49-911-928819-79
headquarters@can-cia.org
www.can-cia.org