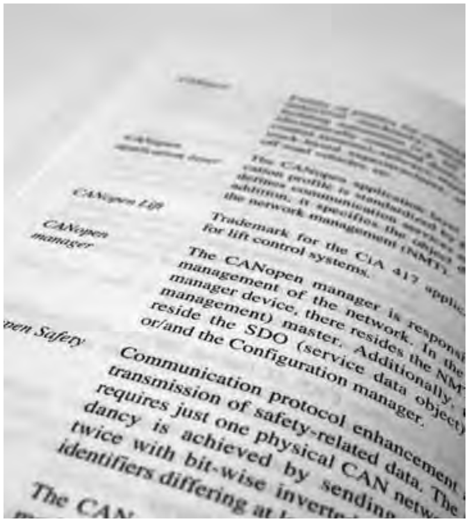
CAN in Automation e. V.
Kontumazgarten 3
DE-90429 Nürnberg
Tel. +49-911-928819-0
Fax +49-911-928819-79
headquarters@can-cia.org
www.can-cia.org

7th edition *2013*

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

7th edition, 2013

Foreword

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

The CAN dictionary covers the CAN data link layer, CAN physical layer 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 may not be sufficiently described yet.

With regard to a more comprehensive successor edition of the CANdictionary, the editors would appreciate comments and proposals (headquarters@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 as ISO 11898. Today, this ISO standard comprises the following parts:

- ISO 11898-1: Data link layer
- ISO 11898-2: High-speed transceiver
- ISO 11898-3: Fault-tolerant transceiver
- ISO 11898-4: Time-triggered CAN
- ISO 11898-5: High-speed transceiver with low-power mode
- ISO 11898-6: High-speed transceiver with selective wake-up capability (*in progress*)

The *classic* CAN data link layer protocol uses one bit-rate for the entire frame. The recently introduced *improved* CAN data link layer protocol (also known as CAN FD) uses for the data-phase a second higher bit-rate, which accelerates the data-transfer. In addition, the improved CAN protocol supports longer data-fields (up to 64 byte). The *improved* CAN data link layer protocol does not support CAN remote frames.

The CAN data link layer is the base of different standardized higher-layer protocols. For commercial vehicle diesel engine powertrain applications, the SAE J1939 series was introduced in the middle of the Nineties. 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 Layer (ISO 15765-2) and the Unified Diagnostic Services (ISO 15765-3) 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.

CAN networks are used in a broad range of applications. In-vehicle networking in any kind of transportation systems (cars, trucks, locomotives, ships, and aircrafts) is the major application field. 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, and the CAN data link layer protocol provides a reliable communication featuring a Hamming distance of 6 (also for the *improved* CAN data frames). This means, five randomly distributed bit errors are detected as well as burst-errors with length of 15 bit. The higher-layer protocols and profiles support interoperability of devices up to the level of off-the-shelf plug-and-play.

A

acceptance filter — The acceptance filter in CAN controller chips is used to select messages that are received depending on the assigned identifier. Most of the CAN controller chips provide a hardware acceptance filter that filters CAN messages assigned with a specific identifier or a range of identifiers. The user-settable filter unburdens the micro-controller from the task of acceptance filtering.

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

acknowledge error — If the message-transmitting node detects the recessive state in the acknowledge slot, it regards that as acknowledge error condition. Acknowledge errors do not cause to bus-off condition. Normally they occur if the network consists of just one node and this node starts transmission of CAN messages.

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

acknowledge (ACK) slot — The first bit of the acknowledge field. It is transmitted recessively by the message-sending node, and it is transmitted dominantly by all receivers, which have performed the CRC (cyclic redundancy check) successfully. If the message-producing node detects this bit as dominant, it knows that there is at least one node that has received the message correctly.

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

application layer The application layer is the communication entity of the OSI (Open System Interconnect) reference model. It provides communication services to the application program.

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

application profile Application profiles define all communication objects and application objects in all devices of a network.

arbitration field The arbitration field is made of the 11-bit or 29-bit identifier, the RTR (remote transmission request) bit, and in case of the 29-bit format also of the IDE (identifier extension) bit, and the SRR (substitute remote request) bit.

arbitration phase The arbitration phase indicates those parts of the *improved* CAN data frame that utilize the bit timing as specified for *classic* CAN. The arbitration phase starts with the SOF and lasts till the sample point of the BRS bit. In addition, the final part of the *improved* CAN data frame, starting with the sample point of the CRC delimiter till EOF complete the arbitration phase. The interframe space is also transmitted with the arbitration bit-time.

Arinc 825-1 This specification by Aeronautical Radio (Arinc) defines a higher-layer protocol dedicated for in-aircraft networking. It is designed similarly to the CANaerospace higher-layer protocol, however it utilizes 29-bit identifier. The physical layer is compliant to ISO 11898-2.

Arinc 826 This specification describes the downloading of software parts to line replaceable units (LRUs). This 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.

automatic re-transmission Corrupted messages (data frames and remote frames) are retransmitted automatically after the error frames are successfully transmitted.

auto bit rate detection The CAN node listens only to the bus traffic, and when a valid message is detected, it acknowledges the received frame. If no valid message is detected, the CAN node switches automatically to the next pre-configured bit-rate. There has to be one and only one node in the network that transmits messages. Some CAN controller chips support automatic bit-rate detection. The same can be achieved by external circuitry.

B

bandwidth	The bandwidth is the value, which denominates the size of information transmitted in a defined time unit.
BasicCAN	A term used in the early days of CAN describing an implementation, which uses just two receive message buffers filled and read out in a ping pong method.
base frame format	The base frame format uses 11-bit identifiers in data frames as well as remote frames.
basic cycle	In TTCAN the basic cycle starts always with the reference message followed by a number of exclusive, arbitration or free windows. One or more basic cycles make the TTCAN matrix cycle.
bit encoding	In CAN, the bits are encoded as non-return to zero coding (NRZ).
bit error	If a bit is transmitted as dominant and received as recessive or vice versa, this is regarded as a bit error condition that causes an error frame transmission in the next bit-time. If a recessive transmitted bit is overwritten by a dominant one in arbitration field and acknowledge 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	Number of bits per time during transmission, independent of bit representation. The bit rate in *classic* CAN is limited to 1 Mbit/s. In the *improved* CAN protocol, the bit rate may be higher in the data phase. In the arbitration phase, the bit rate is still limited to 1 Mbit/s.

bit rate switch At the sample point of the bit rate switch bit in *improved* CAN data frames, the data phase starts. This means at the sample point of the bit rate switch bit the CAN controllers may switch to a higher bit rate. The bit rate switch bit exits in *improved* CAN data frames only.

bit resynchronization Due to local oscillator tolerances it may happen 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 Injections of bits into a bit stream to provide bus state changes that are required for periodic resynchronization.

bit time Duration of one bit.

bit-timing The setting of the bit-timing registers in the CAN controller chip is based on the time quantum, which derives from the oscillator frequency and the node-specific bit-rate pre-scaler.

bridge A device that provides data link layer communication between two networks.

BRS See bit rate switch

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

boot-up message CANopen communication service transmitted whenever a node enters the pre-operational state after initialization.

bus Topology of a communication network, where all nodes are reached by passive links. This allows transmission in both directions.

bus access When the bus is idle, any node may start to transmit a frame. In CAN networks the nodes access the bus by transmitting the dominant SOF (start of frame) bit.

bus analyzer Tool, which 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 may transmit while the other nodes have to delay their transmission. The bus arbitration process used in CAN protocol is CMSA/CD (Carrier Sense Multiple Access/Collision Detection) with AMP (Arbitration on Message Priority). This allows bus arbitration without destruction of messages.

bus comparator Component that converts physical signals used for transfer across the communication medium back into logical information or data signals.

bus driver Component that converts logical information or data signals into physical signals so that these signals can be transferred across the communication medium.

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

bus latency The time between the transmission request and the transmission of the SOF (start of frame) bit. In CAN networks this may be in maximum one message duration minus one bit-time.

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

busload The busload is the ratio of transmitted bits to bus idle bits within a defined time unit. Where 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.

bus monitoring mode In this mode, the CAN controller has switched off the Tx pin. This means no error flag or no ACK slot can be transmitted.

bus-off state The CAN controllers switch to bus-off state when the TEC (transmit error counter) has reached 256. During bus-off state, the CAN controller transmits recessive bits.

bus state Either of the two complementary logical states: dominant or recessive.

CAN	Controller Area Network (CAN) is a serial bus system originally developed by the Robert Bosch GmbH. It is internationally standardized by ISO 11898-1. CAN has been implemented by many semiconductor manufacturers.
CANaero-space	Higher-layer protocol for avionic and aerospace applications.
CAN Application Layer (CAL)	Application layer developed by CiA (CAN in Automation) members providing several communication services and corresponding protocols.
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	Hardware module providing at least one CAN interface.
CAN FD	The *improved* CAN data link layer protocol, also known as CAN with flexible data rate (CAN FD), enables an increased data throughput. The size of the *improved* CAN frame's data field may be lengthened to up to 64 byte. In addition, the data phase of the improved CAN data frame may be transmitted with an increased bit rate. The improved CAN data frame keeps the reliability of the *classic* CAN data frame (Hamming Distance of 6 by adapting the CRC to the size of the data field).
CAN_H	Indicates the CAN high line in CAN-based networks. The CAN_H line of ISO 11898-2 compliant transceiver is in recessive state on 2,5 V and in dominant state on 3,5 V.

CAN identifier The CAN identifier is the main part of the arbitration field of a CAN data frame or CAN remote frame. It comprises 11 bit (base frame format) or 29 bit (extended frame format) and indicates certain information uniquely in the network. The CAN identifier value determines implicitly the priority for the bus arbitration.

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

CAN Kingdom Higher-layer protocol framework optimized for deeply embedded networks. In particular, 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 transceiver is in recessive state on 2,5 V and in dominant state on 1,5 V.

CAN message specification (CMS) Part of the CAN Application Layer (CAL) specification, defining the communication services.

CAN module Implementation of the CAN protocol controller plus the hardware acceptance filter and the message buffers within a micro-controller or application-specific integrated circuit (ASIC).

CAN node Synonym for CAN device.

CANopen Family of 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 application layer
The CANopen application layer and communication profile (CiA 301) is standardized by EN 50325-4. It defines communication services and objects. In addition, it specifies the object dictionary and the network management (NMT).

CANopen Lift
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 manager device, there resides the NMT (network management) master functionality. Additionally, there may reside the SDO (service data object) manager or/and the configuration manager. A CANopen manager owns a CANopen object dictionary and supports also CANopen NMT slave functionality.

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

CANopen Safety Chip (CSC)
This 16-bit micro-controller provides a CANopen Safety protocol firmware implementation. It complies with EN 50325-5 and is certified by TÜV Rhineland up to Safety Integrity Level 3 (SIL 3).

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

CAN transceiver

The CAN transceiver is connected to the CAN controller and to the bus lines. It provides the line transmitter and the receiver. There are high-speed, fault-tolerant, and single-wire transceivers available as well as transceivers for power-line or fiber optic transmissions.

CCP

The CAN Calibration Protocol (CCP) is used to communicate calibration data in engine car applications.

certification

Official compliance test of components or devices to a specific standard. The C&S group performs conformance testing of CAN controller chips. ODVA officially certifies DeviceNet products, and CiA officially certifies CANopen devices.

CiA 102

Additional physical layer specification for high-speed transmission according to ISO 11898-2 using 9-pin D-sub connectors.

CiA 103

Physical layer specification for intrinsically safe capable high-speed transmission according to ISO 11898-2.

CiA 201 to 207

The CAN Application Layer specification (CAL) defines CMS, DBT, NMT, and LMT services and protocols.

CiA 301

The CANopen application layer and communication profile specification covers the functionality of CANopen NMT (network management) slave devices.

CiA 302

Set of additional CANopen specifications, which comprise CANopen manager functionality (part 2), SDO manager functionality (part 5), redundancy concepts (part 6) as well as CANopen router functionality (part 7). In addition, program download (part 3) and network variables (part 4) are described.

CiA 303 Recommendation for CANopen cabling and connector pin assignments, coding of prefixes and SI units as well as LED usage.

(CiA 304) See EN 50325-5.

CiA 305 The layer setting services (LSS) specify services and protocols to set the node-ID or the bit rate via the CANopen network in a master/slave-based communication.

CiA 306 This specification defines format and content of Electronic Data Sheets (EDS) of CANopen devices to be used in configuration tools.

CiA 308 The CANopen technical report defines time measurements such as PDO turn-around time, Sync jitter, and SDO response time. Additionally it defines standard busloads.

CiA 309 This set of specifications defines the services and protocols of TCP/IP-based networks connected to CANopen networks. There are defined protocols for ModbusTCP as well as for ASCII-based commands. Part 2 specifies the ModbusTCP protocol. Part 3 describes a generic text protocol. Part 4 standardizes the access of CANopen networks via ProfibusIO.

CiA 310 The CANopen conformance test plan describes and specifies a lower test for CANopen devices compliant to the CANopen application layer and communication profile CiA 301 as well as the CANopen framework for CANopen managers and programmable CANopen devices CiA 302.

CiA 311 The CANopen XML specification defines the elements and rules for describing device profiles and communication network profiles for devices used in CANopen based control systems.

CiA 312 The set of CANopen device profile conformance test plans specifies all test steps required for checking, whether the implementation of a CANopen device is compliant to the corresponding CANopen device profile.

CiA 313 This set of CANopen profiles specifies test steps for CANopen performance testing as well as a pro forma compliance template.

CiA 314 CANopen framework for PLCs and other programmable devices compliant to IEC 61131-3.

(CiA 400) See CiA 302.

CiA 401 The CANopen device profile for generic I/O modules covers the definition of digital and analog input and output devices.

CiA 402 The CANopen device profile for drives and motion controllers defines the interface to frequency inverters, servo controllers as well as stepper motors. It is internationally standardized in the IEC 61800-7 series.

CiA 404 The CANopen device profile for measuring devices and closed-loop controllers supports also multi-channel devices.

(CiA 405) Former CANopen profile for IEC 61131-3 compatible controllers. Now published in several parts (CiA 302-8, CiA 306-3, CiA 314, and CiA 809).

CiA 406 This CANopen device profile offers a standardized CANopen interface for

	incremental and absolute, linear and rotary encoders.
(CiA 407)	See EN 13149-4/5/6.
CiA 408	The CANopen device profile for hydraulic controllers and proportional valves is compliant to the bus-independent VDMA (Verband Deutscher Maschinen- und Anlagenbau e.V.) device profile fluid power technology – proportional valves and hydrostatic transmission.
CiA 410	The CANopen device profile for inclinometer supports 16-bit as well as 32-bit sensors.
CiA 412	The CANopen device profiles for medical equipment specify the interfaces for x-ray collimators and dosimeter devices.
CiA 413	The CANopen interface profiles specify gateways to SAE J1939, ISO 11992, and other in-vehicle 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.
CiA 414	The CANopen device profile for weaving machines specifies the interface for feeder sub-systems.
CiA 415	The CANopen application profile for sensor systems specifies interfaces for sensors and sensor controllers. It is specified for use in all kinds of road construction machines.
CiA 416	The CANopen application profile for building doors specifies CANopen interfaces for locks, sensors, and other devices used in electronically controlled building doors.

CiA 417 The CANopen application profile for lift control specifies the interfaces 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 The CANopen device profile for battery modules specifies the interface to communicate with battery chargers.

CiA 419 The CANopen device profile for battery charger specifies the interface to communicate with the battery module.

CiA 420 The CANopen profile family for extruder downstream devices defines interfaces for puller, corrugator and saw devices.

CiA 421 The CANopen application profile for train vehicle control systems defines the communication between virtual control systems (e.g. for door control, diesel engine control or control of auxiliary equipment) within locomotives, power cars or coaches.

CiA 422 The CANopen application profile for municipal vehicles (in particular garbage truck superstructures) specifies the interfaces of sub-systems such as compaction unit, weighing unit, etc.

CiA 423 The CANopen application profile for rail power drive systems defines the communication between virtual devices required for the control of diesel- as well as diesel electrical locomotives.

CiA 424 The CANopen application profile for rail door control systems defines the communication between a door controller and the related door units.

CiA 425 The CANopen profile for medical add-on devices defines plug-and-play interfaces for contrast media injectors and electrocardiogram units. CiA 425 is also used as unregistered trademark.

CiA 426 The CANopen application profile for rail exterior lighting defines the communication between an exterior lighting controller and the related exterior lighting units.

CiA 430 The CANopen application profile for rail auxiliary operating systems defines the communication between auxiliary equipment such as power train cooling unit, coolant exposition tank, engine pre-heating unit or battery charger.

CiA 433 The CANopen application profile for rail interior lighting systems defines the communication between an interior lighting controller and interior lighting units.

CiA 434 This set of CANopen device profiles describes the communication between a laboratory automation master and related slave devices such as dilutor unit, dispenser unit, washing unit or heating unit.

CiA 436 The CANopen profile for construction machines defines 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 The CANopen profile for photovoltaic systems defines the integration platform for photovoltaic inverters and sensors as well as other devices.

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

CiA 443 The CANopen device profile for SIIS level-2 devices specifies simple and complex sensors and actuators. This equipment is also known as “Christmas tree”.

CiA 444 The CANopen application profile defines the CANopen interfaces for container-handling machine add-on devices such as e.g. spreaders for cranes or straddle carriers.

CiA 445 This device profile defines the CANopen interface for simple and intelligent radio frequency identification (RFID) devices.

CiA 446 The CANopen device profile for AS-Interface gateways describes CANopen devices, which act as an AS-Interface master in AS-Interface networks.

CiA 447 The CANopen application profile for special purpose car add-on devices specifies the CAN physical layer as well as application, configuration and diagnostic parameters for the add-on devices (e.g. taximeter, blue-light, etc.) used in special-purpose passenger cars.

CiA 450 The CANopen device profile for pumps is based on the VDMA (Verband Deutscher Maschinen- und Anlagenbau e.V.) profile for pumps. It specifies a generic pump, a process vacuum pump, a turbo vacuum pump and a liquid pump.

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

CiA 453 The CANopen device profile ‘power supply’ 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 The CANopen application profile for energy management systems specifies the communication interface for all virtual devices that may take part in energy management control applications. Such energy control applications may be implemented in e.g. light electric vehicles, industrial robots, offshore parks, isolated farms, etc.

CiA 455 The CANopen application profile for drilling machines specifies the control ling of drilling machines with special regard on positioning and tool control.

CiA 456 The CANopen device profile for configurable network components specifies the parameterization of the provided bridge and router functionality for up to 16 CAN ports.

CiA 457 The CANopen device profile for wireless transmission media specifies the gateway functionality between CANopen networks and wireless networks.

CiA 459 The set of specifications specifies the CANopen interface for on-board weighing devices. Such devices are usable on trucks, off-highway or off-road vehicles (including train coaches).

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

CiA 802 This application note provides recommendations for substituting CAN remote frames by other CANopen communication services.

CiA 808 This application note describes the recommended practice as well as application hints for development of the communication between crane and spreader, designed according to the device profile CiA 444.

CiA 809 This application note provides implementation and user guidelines for devices programmable in IEC 61131-3 languages, which communicates to CANopen devices. This includes a description of the programming and network configuration environments.

CiA 810 This application note describes the recommended practice as well as application hints for development of the laboratory automation slave devices, designed according to the device profile CiA 434.

CiA 812 This application note describes use cases of CANopen devices supporting the CiA 315 framework for tunneling CAN messages via wireless networks.

CiA 850 This recommended practice specifies the implementation of the CiA 413 gateway interface for truck-mounted cranes, multi-lifts, and aerial working platforms.

CiA 852 This recommended practice specifies the usage of CiA 401-based operator environments. Operator environments include simple units as well as operator seats with integrated joysticks, foot pedals, pushbuttons, indicators, etc.

classic *CAN* CAN applications based on ISO 11898-1, which do not support the improved CAN data link layer protocol.

CleANopen	Unregistered trademark for the CiA 422 application profile for municipal vehicles.
Client SDO	The Client SDO initiates the SDO communication by means of reading or writing to the object dictionary of the SDO server device.
client/server communication	In a client/server communication the client initiates the communication with the server. It is always a point-to-point communication.
COB	See communication object.
COB-ID	The COB-ID is the object specifying the CAN identifier and additional parameters (valid/-invalid bit, remote frame support bit, frame format bit) for the related communication object.
communication object (COB)	A communication object consists of one or more CAN messages with a specific functionality, e.g. PDO, SDO, Emergency, Time, or Error Control.
communication profile	A communication profile defines the content of communication objects such as Emergency, Time, Sync, Heartbeat, NMT, etc. in CANopen.
configuration manager	The configuration manager provides mechanisms for configuration of CANopen devices during boot-up.
configuration parameter	Parameter in the CANopen object dictionary that 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 message has been received successfully.

conformance test plan — Definitions of test cases that have to be passed successfully in order to achieve conformance to a communication standard. The conformance test plan for CAN is standardized by ISO 16845.

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. The connector pin-assignment for CAN is specified by CiA for CAN and CANopen and by ODVA for DeviceNet.

consumer — In CAN networks a receiver of messages is called a consumer meaning the acceptance filter is opened.

content-based arbitration — Carrier Sense Multiple Access (CMSA) arbitration procedure where simultaneous access of multiple nodes results in a contention.

control field — In *classic* CAN data and remote frames the 6-bit control field contains the four DLC bits, the IDE bit and the reserved bit(s).In the *improved* CAN data frame the 9-bit control field is enhanced by the EDL-, BRS- and ESI-bit.

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. It is always recessive.

CRC error — If the result of the CRC on the receiving CAN node is unequal zero, this will regarded as a CRC error. The corresponding error frame is transmitted after the acknowledge field.

CRC field

The CRC field in data and remote frames contains the 15-bit CRC sequence and the 1-bit CRC delimiter. The CRC sequence is able to detect 5 randomly distributed bit failures in SOF, arbitration, control, data fields, or a burst failure of up to 15 bits. The Hamming distance is specified as 6, not considering stuff-bits. In the *improved* CAN data frames the CRC is sequence is different depending on the length of the data field. The Hamming distance is also specified as 6, but considers the stuff-bits.

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)

The cyclic redundancy check (CRC) is performed by a polynomial implemented in the transmitting as well as in the receiving CAN modules to detect corruption while transmitting CAN data frames or CAN remote frames.

data con-sistency	With regard to network technologies, data consistency means that all devices, 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.
data field	The data field of the CAN data frame contains 0 to including 8 byte of user information as indicated by the DLC.
data frame	The CAN data frame carries data from a producer to one or more consumers. It consists of the start of frame bit, the arbitration field, the control field, the data field, the CRC field, the acknowledge field, the end of frame field.
data length code (DLC)	The 4-bit DLC in the control field of the CAN data frame indicates the length of the data field. In remote frames the DLC corresponds to the data field length in the requested data frame!
data link layer	Second layer in the OSI reference model providing basic communication services. The CAN data link layer defines data, remote, error, and overload frames.
data phase	The data phase indicates those parts of the improved CAN data frame that are transmitted with a higher bit-rate. 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 type	Object attribute in CANopen and DeviceNet defining the format, e.g. Unsigned8, Integer16, Boolean, etc.

DBT The Distributor is part of the CAN Application Layer (CAL) specification defining a method of automatic identifier distribution during boot-up of the network.

DCF See device configuration file.

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

device configuration file (DCF) The device configuration file describes the CANopen parameter of a configured CANopen device in the same file format as the EDS.

DeviceNet CAN-based higher-layer protocol and device profiles definition. 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 defines the device-specific application data and communication capability based on the related higher-layer protocol. For more complex devices these profiles may provide a finite state automaton (FSA), which enables standardized device control.

Diagnostics on CAN The ISO 15765 standard defines the Diagnostic on CAN protocols and services, which are used for the CAN-based diagnostic interface for passenger cars.

DLC See data length code.

dominant bit Bit on the CAN bus lines representing dominant state. It has the logical value

0. A dominant bit overwrites by definition a recessive bit.

double-reception of message If the last bit of the end of frame (EOF) is corrupted at the transmitting node, then a retransmission of the message is caused. Since the receivers have already accepted the message after the last but one bit, they will receive the message twice.

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

DS (draft standard) This kind of standard is not fixed, but it is published. CiA's draft standards are not changed within one year.

DSP (draft standard proposal) This kind of standard is a proposal, but it is published. CiA's draft standard proposals may be changed anytime without notification.

D-sub connectors Standardized connectors. Most common in use is the 9-pin D-sub connector (DIN 41652); its pin-assignment for CAN networks is specified in CiA 102.

EDL See extended data length.

EDS See electronic data sheet.

EDS checker Software tool that checks the conformity of electronic data sheets. The CANopen EDS checker is available on CiA's web-site to be downloaded.

EDS generator Software tool that generates electronic data sheets (available for CANopen and DeviceNet).

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.

Emergency Pre-defined communication service 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 is used to communicate device and application failures.

EN 13149-4/5/6 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 CENELEC standard defining the CANopen application layer and communication profile.

EN 50325-5 CENELEC standard defining 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 according to IEC 61508.

entry category — This is the object attribute in CANopen defining this object as mandatory or optional.

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

error active state — In error active state the CAN controller is allowed to transmit active error frames containing active error flags. If all CAN nodes are in this state, than a network-wide data consistency is guaranteed.

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 entering pre-operational state after inititialization, and it is transmitted if remotely requested by the NMT master (node guarding) or periodically by the device (heartbeat).

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

error delimiter Last segment in error frames made up of 8 recessive bits.

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

error flag First segment in error frames made up of 6 bits of the same polarity. A second error flag transmitted by another node may overlap the first error flag.

error frame Frame to indicate the detection of an error. It is made up of the error flag and the error delimiter.

error globalization Local failures cause the transmission of an error flag, which will be 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 mode.

error passive state In error passive state the CAN controller is only allowed to transmit passive error frames containing passive error flags. Additionally the CAN controller has to wait a certain time after a previous transmission before its own transmission takes place (suspend transmission).

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

error state indicator The error state indicator bit in the *improved* CAN data frame indicates whether the transmitting CAN node is in CAN error active or passive state.

ESI See error state indicator.

event-driven Event-driven messages are transmitted when a defined event occurs in the device. This may be a change of input states, elapsing of a local timer, or any other local event.

event-driven PDO An event-driven PDO is transmitted whenever a device internal event occurs. This event may be the elapsing of the PDO's event timer. 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 defines the frequency of PDO transmission.

expedited SDO This is a confirmed communication service of CANopen (peer-to-peer). 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 has to be 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.

extended data length The extended data length bit distinguishes between *classic* and *improved* CAN data frames.

extended frame format The extended CAN frame format uses the 29-bit identifiers in data frames as well as in remote frames.

F

fault confinement	CAN nodes are able to distinguish short disturbances from permanent failures. Defective transmitting nodes are switched off, meaning the node is logically disconnected from the network (bus-off).
fault-tolerant transceiver	Transceivers as specified in ISO 11898-3 and ISO 11992-1 are capable of communication via one bus-line and CAN ground when one bus-line is broken down, short circuited or termination resistors are not well connected.
finite state automaton (FSA)	An FSA is an abstraction to describe the behavior of a black box. It is composed of a several states, transitions between those states, and actions.
flying master	In safety-critical applications, it may be required that a missing NMT master is substituted automatically by another stand-by NMT master. This concept of redundancy is called flying master.
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 will cause the transmission of an error frame in the very next bit-time.
frame	Data link protocol entity specifying the arrangement and meaning of bits or bit fields in the sequence of transfer.
frame coding	Sequence of fields in the CAN frames, e.g. SOF, arbitration field, control field, data field, CRC field, ACK field and EOF for data frames. The frame coding covers also the bit stuffing.
frame format	The CAN standard distinguishes between the base frame format using 11-

bit identifiers and the extended frame format using 29-bit identifiers.

frame types In CAN, four frame types are used: data frame, remote frame, error frame, and overload frame.

FSA See finite state automaton.

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 First four bits of the CAN identifier in the CANopen pre-defined identifier set indicating the function of the communication object (e.g. TPDO_1 or error control message).

G

galvanic isolation	Galvanic isolation in CAN networks is performed by optocouplers or transformers placed between CAN controller and CAN transceiver chip.
gateway	Device with at least two network interfaces transforming all seven OSI (open system interconnection) protocol layers, e.g. CANopen-to-Ethernet gateway or CANopen-to-DeviceNet gateway.
global error	A global bus error affects all connected CAN devices.
global fail-safe command	The global fail-safe command (GFC) is a high-priority CAN message defined in the CANopen safety protocol. It reduces the reaction time. It shall be followed by the related SRDO.

H

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. CAN provides a hamming distance of 6 (theoretical value for CAN networks). This indicates that five randomly distributed bit failures can be detected. In addition burst errors of up to 15 bit can be detected. CAN provides no bit correction mechanisms.
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 second or third bit of interframe space.
heartbeat	CANopen and DeviceNet use the heartbeat message to indicate that a node is still alive. This message is transmitted periodically.
heartbeat consumer time	The heartbeat consumer time defines 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.
heartbeat producer time	The heartbeat producer time defines the transmission frequency of a heartbeat message.
higher-layer protocol (HLP)	Higher-layer protocols define communication protocols compliant to the transport layer, session, presentation, or application layer as specified in the OSI reference model.
high-speed transceiver	Transceiver as specified in ISO 11898-2 for data rates up to including 1 Mbit/s.

I

identifier	In general, the term refers to a CAN message identifier. See CAN identifier.
identifier extension flag (IDE)	The IDE bit indicates if the following bits are interpreted as control bits or the second part of the 29-bit identifier.
identifier field	The identifier field contains 11 bits in base frame format, and additional 18 bits in extended frame format.
IEC 61162-3	IEC standard for "Digital interfaces for navigational equipment within a ship". Part 3 standardizes the "Serial data instrument network", also known as NEMA 2000.
IEC 61800-7-1/2/3	International standard specifying multiple power drive profiles including CiA 402 and CIP motion. The CiA 402 profile mapping to CANopen and the CIP motion profile mapping to DeviceNet are also specified in this series of standards.
IEC 62026-3	International standard specifying the CAN-based higher layer protocol DeviceNet.
improved *CAN*	The *improved* CAN data link layer protocol supports *classic* CAN frames as well as *improved* CAN data frames. *Improved* CAN data frames are distinguished by means of one reserved bit (recessive) of the classic CAN data frame (dominant).
index	16-bit address to access information in the CANopen object dictionary; for array and records the address is extended by an 8-bit sub-index.
inhibit time	Parameter in CANopen that defines the specified minimal time (inhibit time) for

a transmission of the PDOs and Emergency messages.

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

inner priority inversion If a low-prior message can not be transmitted because of high-prior message traffic on the CAN network and a high-prior transmission request occurs in the device and cannot be passed to the CAN controller due to the still pending low-prior transmission request is called inner priority inversion.

interface profile CANopen profile that describes just the interface and not the application behavior of a device, e.g. gateway devices.

interframe space Three recessive bits make up the interframe space that separates all CAN frames including error and overload frames.

intermission field Synonym for interframe space.

I/O message 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.

ISO 11898-1 International standard defining the CAN data link layer including LLC, MAC and PLS sub-layers.

ISO 11898-2 International standard defining the CAN high-speed medium access unit (see ISO/IEC 7498-1).

ISO 11898-3 International standard defining the CAN fault-tolerant, low-speed medium access unit (see ISO/IEC 7498-1).

ISO 11898-4 International standard defining a time-triggered communication protocol based on the *classic* CAN data link layer protocol.

ISO 11898-5 International standard defining ISO 11898-2 compliant transceivers featuring low-power functionality.

ISO 11898-6 International standard defining ISO 11898-2 and ISO 11898-5 compliant transceivers featuring selective wake-up functionality. This means the transceiver supports partially the CAN data link layer protocol interpreting the CAN-ID and the data field.

ISO 11992 International standard defining a CAN-based application profile for truck/trailer communication. Part 2 specifies the brake and gearing devices, part 3 specifies other devices, and part 4 defines the diagnostics.

ISO 11745-2 International standard defining an application integration framework for ISO 11898 based control systems such as CANopen and DeviceNet.

ISO 11783 International standard defining the CAN-based application profile used in agriculture and forestry machines and vehicles. It is based on the J1939 application profile.

ISO 16844 International standard defining the CAN-based tachograph to be used in trucks and buses.

ISO 16845 International standard that defines the conformance test plan for ISO 11898-1 implementations.

J1939 application profile	The application profile defined by SAE (www.sae.org) specifies the in-vehicle communication in trucks and buses. It defines the communication services as well as the signals including the mapping into CAN data frames by means of PGNs (parameter group numbers).
J2284 bit-timing	Bit-timing definitions by SAE for in-vehicle networks in passenger cars for 250 kbit/s and 500 kbit/s.
J2411 single-wire CAN	Single-wire transmission specification by SAE for CAN networks. The bit rate is limited to 40 kbit/s.

layer-2 protocol	A layer-2 protocol uses the CAN communication services directly without a dedicated higher-layer protocol.
layer-7 protocol	A layer-7 protocol uses CAN communication services in a standardized manner. This allows the reuse of application software without redesigning the CAN communication software.
layer setting services (LSS)	The CANopen layer setting services define communication services for configuring node-ID and bit rate via the CAN network.
life guarding	Method in CAL and CANopen to detect that the NMT master does not guard the NMT slave anymore. This is part of the error control mechanisms.
line topology	Networks, where all nodes are connected directly to one bus line. CAN networks use theoretically just line topologies without any stub cable. However in practice you find tree and star topologies as well.
LLC	See logical link control.
LMT	Abbreviation for layer management.
LMT protocols	Protocols defined in CAL for setting node-IDs and bit rates via the CAN network.
local bus error	A local bus error effects just one or more but not all nodes in the network.
logical link control (LLC)	The LLC sub-layer describes the upper part of the OSI data link layer. It is concerned with those protocol issues that are independent of the type of medium access method.

low-power mode — CAN controller and CAN transceiver may support a stand-by mode requiring lower power than in active mode.

low-speed transceiver — Synonym for fault-tolerant transceivers.

LSS — See layer setting services.

M

MAC	See medium access control.
master	Communication or application entity that is allowed to control a specific function. In networks this is for example the initialization of a communication service.
master/slave communication	In master/slave communication system the master initiates and controls the communication. The slave is not allowed to initiate any communication at all.
matrix cycle	In TTCAN the matrix cycle is made up of one or more basic cycles. Each basic cycle starts with the reference message but may be followed by different windows.
MDI	See medium dependent interface.
medium access control (MAC)	The MAC sub-layer represents the lower part of the OSI data link layer. It services the interface to the LLC sub-layer and the physical layer, and comprises the functions and rules that are related to data en-/de-capsulation, error detection and signaling.
medium-dependent interface (MDI)	The MDI defines the connector, cable and termination resistor requirements.
message	A message in CAN may be a data frame or remote frame.
message buffer	CAN controller chips implement message buffers for frames to be received and/or to be transmitted. The implementation and the use of message buffers is not standardized.

message doubling — See double reception of message.

MilCAN — These CAN-based higher-layer protocols have been defined by a group of interested companies and government bodies associated with the specification, manufacture and test of military vehicles. MilCAN A is based on J1939, and MilCAN B is based on CANopen.

MPDO — See multiplex PDO.

multicast transmission — Addressing, where a single frame is addressed to a group of nodes simultaneously.

multi-master communication — In a multi-master communication system every node may temporarily control the bus communication. This means every node has theoretically the right to access the bus at any time when the bus is in idle state.

multiplex PDO (MPDO) — The MPDO is made of 8 byte including one control byte, three multiplexer bytes (containing the 24-bit index and sub-index), and four bytes of object data.

network-ID	In multiple CANopen network systems this identifier identifies a single CANopen network uniquely. CANopen supports up to 127 networks in hierarchical or non-hierarchical network systems.
network length	See bus length.
network management	Entity responsible for the network boot-up procedure and the optional configuration of nodes. It also may include node-supervising functions such as node guarding.
network variables	Network variables are used in programmable CANopen devices to 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), a US nonprofit organization.
NMT	Abbreviation for network management in CAL and CANopen. See network management.
NMT master	The NMT master device performs the network management by means of transmitting the NMT message. With this message, it controls the state machines of all connected NMT slave devices.
NMT slave	The NMT slaves receive the NMT message, which contains commands for the

NMT state machine implemented in CAL and CANopen devices.

NMT slave state machine — The NMT slave state machine defined in CAL and CANopen supports different states and the highest prior CAN message transmitted by the NMT master controls the transition to the states.

node — Assembly, linked to the CAN network, capable of communicating across the network according to the CAN protocols.

node guarding — Mechanism used in CANopen and CAL to detect bus-off or disconnected devices, which is part of the error control mechanisms. The NMT master sends a remote frame to the NMT slave that is answered by the corresponding error control message.

node-ID — 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 pre-defined connection sets of CANopen or DeviceNet, the node-ID is part of the CAN identifier.

nominal bit rate — The nominal bit rate is the number of bits per second transmitted in the absence of resynchronization by an ideal transmitter.

nominal bit-time — The nominal bit-time can be thought of as being divided into separate non-overlapping time segments.

non-return to zero (NRZ) coding — Method of representing binary signals. Within one and the same bit-time, the signal level does not change.

normal SDO — See segmented SDO.

O

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.

open system interconnection (OSI) reference model — Layered communication model defining seven layers: physical, data link, network, transport, session, presentation, and application layer. In CAN-based networks normally just physical, data link, and application layer are implemented.

operational state — Part of the CANopen NMT slave state machine. In the NMT operational state all CANopen communication services are available.

OSEK/VDX — Set of specifications for communication (COM), network management (NM), real-time operating system (OS), and implementation language (OIL). OSEK/VDX is partly implemented in passenger cars.

OSI reference model — See open system interconnection reference model.

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

overload condition — 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 Last segment in overload frames made up of 8 recessive bits.

overload flag First segment in overload frames made up of 6 bits of dominant value. A second overload flag transmitted by another node may overlap the first overload flag.

overload frame Frame to indicate an overload condition.

P

parameter group (PG)	In J1939, ISO 11783, and ISO 11992, there are defined parameter groups, which specify the content of a specific CAN message.
parameter group number (PGN)	The parameter group number identifies uniquely the parameter group (PG). The PGN is mapped into the 29-bit identifier.
passive error flag	The passive error flag is the first part of the passive error frame made up of six consecutive recessive bits.
PDO	See process data object.
PDO mapping	In PDOs, there may be mapped up to 64 objects. The PDO mapping is described in the PDO mapping parameters.
pending transmission request	There are one or more messages waiting for transmission in the CAN controller because the bus is not idle (node has lost 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, measured in time quanta.
phase segment 1 (Phase_Seg 1)	Part of the bit-time used to compensate for edge phase errors. It may be lengthened by re-synchronization.
phase segment 2 (Phase_Seg 2)	Part of the bit-time used to compensate for edge phase errors. It may be shortened by re-synchronization.
physical layer	Lowest layer in the OSI reference model defining the connectors, bus cables,

	and electrical or optical signals representing a bit value as well as synchronization and re-synchronization.
physical signaling (PLS)	Sub-layer of the 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.
pin assignment	Definition of the usage of connector pins.
PLS	See physical signaling.
pre-defined connection set	Set of CAN identifiers used as default values for different communication protocols in CANopen or DeviceNet.
pre-operational state	Part of the NMT slave state machine. In the NMT pre-operational state no CANopen PDO communication is allowed.
priority	Attribute to a frame controlling its ranking during arbitration. In CAN data and remote frames, the identifier (ID) gives the priority. The lower the ID, the higher is the priority.
process data	Parameter in the CANopen object dictionary that can be mapped into PDOs.
process data object (PDO)	Communication object defined by the PDO communication parameter and PDO mapping parameter objects. It is an unconfirmed communication service without protocol overhead.
producer	In CAN network a transmitter of messages is called a producer.
propagation segment (Prop_Seg)	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 Formal set of conventions and rules for the exchange of information between nodes, including the specification of frame administration, frame transfer and physical layer.

priority inversion Priority inversion occurs if the lower prior object will be processed or communicated before the higher prior object. In not well-designed CAN devices, there may occur inner or outer priority inversions.

receive error counter (REC)	CAN controller internal counter for reception errors. The REC value is readable in some controllers.
receive PDO	The receive process data object (RPDO) is a PDO that is received by a CANopen device.
receiver	A CAN node is called receiver or consumer, if it is not transmitted and the bus is not idle.
reception buffer(s)	Local memory in the CAN controller, where the received messages are stored intermediately.
recessive bit	Bit on the CAN bus lines representing recessive state. It has the logical value 1.
recessive state	By definition, the recessive state will be overwritten by the dominant state.
recovery time	The time between the first bit of the error flag and when the automatic retransmission can be 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 may be required that provide swapping capability in case of detected communication failures.
reference message	In TTCAN, the reference message starts each basic cycle.
remote frame	With a remote frame another node is requested to transmit the corresponding data frame identified by the very same identifier. The remote frame's DLC has the value of the corresponding data

frame DLC. The data field of the remote frame has a length of 0 byte.

remote transmission request (RTR) — Bit in the arbitration field indicating if the frame is a remote frame (recessive value) or a data frame (dominant value).

repeater — 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.

reset — A CAN controller is reset by a command (may be hard-wired). Before the CAN controller transits back to error active state, it has to detect 128 by 11 consecutive recessive bit-times.

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

reset communication — This NMT command resets only the communication objects in CANopen devices to the default values or the permanently stored configured values.

re-synchronization jump width (SJW) — Number of time quanta with which the Phase_Seg 1 may be lengthened or the Phase_Seg 2 may be shortened.

RPDO — See receive PDO.

RTR — See remote transmission request.

S

SafetyBus p — This CAN-based higher-layer protocol and implementation specification by the Safety Network International e. V. is dedicated for safety-related communication within factory automation. It meets the Safety Integrity Level (SIL) 3 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.

safe-guard cycle time (SCT) — The safe-guard cycle time (SCT) defines the maximum time between two periodically transmitted SRDOs.

safety-related object validation time (SRVT) — The safety-related object validation time defines the maximum time between the two CAN messages that make an SRDO.

safety-relevant data object (SRDO) — The safety-relevant data object (SRDO) as defined in the CANopen safety protocol is made by two CAN messages. The second message contains in the data field the bit-wise converted data of the first message.

SCT — See safe-guard cycle time.

SDO — See service data object.

SDO block transfer — SDO block transfer is a CANopen communication service for increasing the speed of downloading data from the CANopen device. In SDO block transfer, the confirmation is sent after the reception of a number of SDO segments.

SDO manager — The SDO manager handles the dynamic establishment of SDO connections. It resides on the very same node as the NMT master functionality.

SDO network indication This function is used to address a remote CANopen device in another CANopen network. This service and protocol establish a virtual channel in order to perform any SDO communication.

segmented SDO If objects longer than 4 byte are transmitted by means of SDO services, a segmented transfer is used. The data is transmitted in segments of up to 7 Byte of application data. The number of segments is theoretically not limited.

server SDO The server SDO receives the SDO messages from the corresponding client and responses each SDO message or a block of SDO messages (SDO block transfer).

service data object (SDO) The SDO is a confirmed communication service that provides access to all entries in the CANopen object dictionary. An SDO uses two 8-byte CAN messages with different identifiers. The SDO may transmit segmented any amount of data. Each segment (segmented SDO) or a number of segments is confirmed (SDO block transfer).

single-shot transmission Some CAN controllers provide a single-shot mode, which means that the message will not be retransmitted automatically when an error has been detected. This mode is required for TTCAN.

single-wire CAN (SWC) Physical layer using only one bus line and CAN ground. The SAE specified a SWC transceiver (J2411).

SI unit International system of units for physical values as specified in ISO 1000:1983.

sleep mode	CAN controller and transceiver may be operated in stand-by or low-power mode not more driving the bus lines.
SOF	See start of frame.
SRDO	See safety-relevant data object.
SRR	See substitute remote request.
SRVT	See safety-related object validation time.
start of frame (SOF)	The very first bit of any data and remote frames. The SOF's state is always dominant.
star topology	In some passenger cars, CAN networks are installed in a star topology terminating the network in the center of the star.
stopped state	Part of the NMT slave state machine. In the NMT state only NMT messages are performed and under some conditions error control messages are transmitted.
stuff-bit	Whenever a CAN transmitter detects 5 consecutive bits of identical value in the bit stream, it automatically inserts a complementary stuff-bit. The CAN receiver excludes the stuff-bits automatically, so that the original message to be transmitted is the very same as the received message. It is used for automatic re-synchronization in the CAN module’s bit-timing circuitry.
stuff error	A stuff error is detected at the bit-time of the sixth consecutive equal bit level in SOF, arbitration, control, data, and CRC field.
sub-index	8-bit sub-address to access the sub-objects of arrays and records in a CANopen object dictionary.

substitute remote request (SRR) — Bit in the extended frame format substituting the RTR bit after the first part of the identifier (11 bit). The SRR's state is recessive.

suspend transmission — CAN controllers in error passive mode have to wait additional 8 bit-times before the next data or remote frame may be transmitted.

SWC — See single-wire CAN.

SYNC counter — The optional parameter SYNC counter is used in CANopen networks to define an explicit relationship between the current SYNC cycle and PDO transmission.

SYNC message — Dedicated CANopen message forcing the receiving nodes to sample the inputs mapped into synchronous TPDOs. Receiving this message causes the node to set the outputs to values received in the previous synchronous RPDO.

sync segment (Sync_Seg) — Part of the bit-time used to synchronize various nodes on the bus. An edge is expected within this segment.

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 message	Standardized message in CANopen containing the time as a 6-byte value given as ms after midnight and days after 1st January 1984.
time quanta	Atomic time unit in a CAN network.
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 include the time stamp in the data field of the very same frame.
time-triggered	Time-triggered messages 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	Physical connection structure of the network (e.g. line, ring, star, and tree topology).

TPDO	See transmit PDO.
transmission buffer(s)	Local memory in the CAN controller, where the message to be transmitted is stored.
transmission request	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 has been transmitted.
transmission type	CANopen object defining the scheduling of a CANopen communication object such as e.g. PDO.
transmit error counter (TEC)	CAN controller internal counter for transmission errors. The TEC value is readable in some controllers.
transmit PDO (TPDO)	The transmit process data object is a PDO that is transmitted by a CANopen device.
transmitter	A node from which a data or remote frame originates. This node remains transmitter until the bus is idle again or until the node loses arbitration.
tree topology	Network topology with trunk line and branch lines. The not terminated branches may cause reflections, which shall not exceed a critical value.
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	Higher-layer protocol defining time-triggered communication in CAN-based

networks. The CAN controllers have to be capable of switching-off automatic retransmission of faulty messages and may be able to capture a 16-bit timer value at SOF transmission in order to transmit the timer value in the very same message.

V

value definition	Detailed description of the value range of a variable in CANopen profiles.
value range	Object attribute in CANopen defining the allowed values supported by this object.

W

wake-up procedure	This procedure is used to wake-up CAN transceiver or CAN module that are in sleep mode.

((Space for CiA publication advertisement))

Imprint

Editors: Christian Dressler
Olga Fischer
Oskar Kaplun
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

7th edition: February 2013

Copies: 2 000

Printed in Germany

CiA
CAN in Automation

CiA's Weekly Telegraph

- Allways up to date
- Informs you about CiA's public activities

CAN *Newsletter Online*
www.can-newsletter.org

- Product and service news
- Dossiers, Features, and Background information

CAN *Newsletter (print)*
www.can-cia.org/newsletter

- Technical articles
- Detailed application reports

CiA *Product Guides*
www.can-productguides.org

- Advertise CANopen, CAN, and J1939 products
- Linked to CiA's online media

For more details please contact the CiA office at publications@can-cia.org

CAN *Newsletter*

I hereby subscribe to the free-of-charge CAN Newsletter for the next four editions (published in March, June, September, and December of every year).

☐ Print version ☐ PDF version

Company

Name

Address

City, ZIP

Phone (Country code - Area code - Number)

Fax (Country code - Area code - Number)

E-mail

URL

Position (see A)

Application (see C)

Business (see B)

Size (see D)

☐ I like to receive CiA's Weekly Telegraph

Please send your subscription form to CAN in Automation (CiA) GmbH, Kontumazgarten 3, DE-90429 Nuremberg, Germany, or fax it to +49-911-928819-79 or e-mail it to headquarters@can-cia.org. You may also subscribe online at www.can-cia.org.