# Multiple Data-PDU concept for heterogeneous backbone networks

Holger Zeltwanger, CAN in Automation

**The CAN XL data link layer protocol provides a data field with up to 2048 byte. It can be used for backbone networks. Due to the long data field, this protocol is able to transmit multiple Data-PDUs even in one single CAN XL data frame. This means, a CAN XL network can be shared by several applications using different application layer approaches. This paper shows the options and limits as well as the requirements on the header/footer supporting homogeneous and heterogeneous multiple Data-PDUs.**

In the beginning, Classical CAN networks connected just a few ECUs (electronic control units). Many of the early Classical CAN in-vehicle networks (IVN) used star-like topologies. Step-by-step additional Classical CAN networks were added and linked by means of bridges, routers, or gateways. In the course of time, other communication technologies were invented (e.g. Automotive Ethernet, Flexray, LIN, Most, etc.). They were integrated by means of gateways to the existing Classical CAN in-vehicle network systems. This is the flat network architecture; still used in many passenger cars.

A couple years ago, a domain-oriented IVN architecture was introduced. This approach is logical architecture using domain controllers with dedicated sub-network architectures. These domains include Drivetrain, Body & Comfort, Infotainment, Connectivity, and increasingly ADAS (advanced driver-assistance systems). From software point-of-view this simplifies the system design. It also enables the design of autonomous driving cars. But the domain-oriented architecture is not an approach to optimize the wiring harness. This is, why the so-called zone-oriented IVN architecture is discussed nowadays. This approach uses

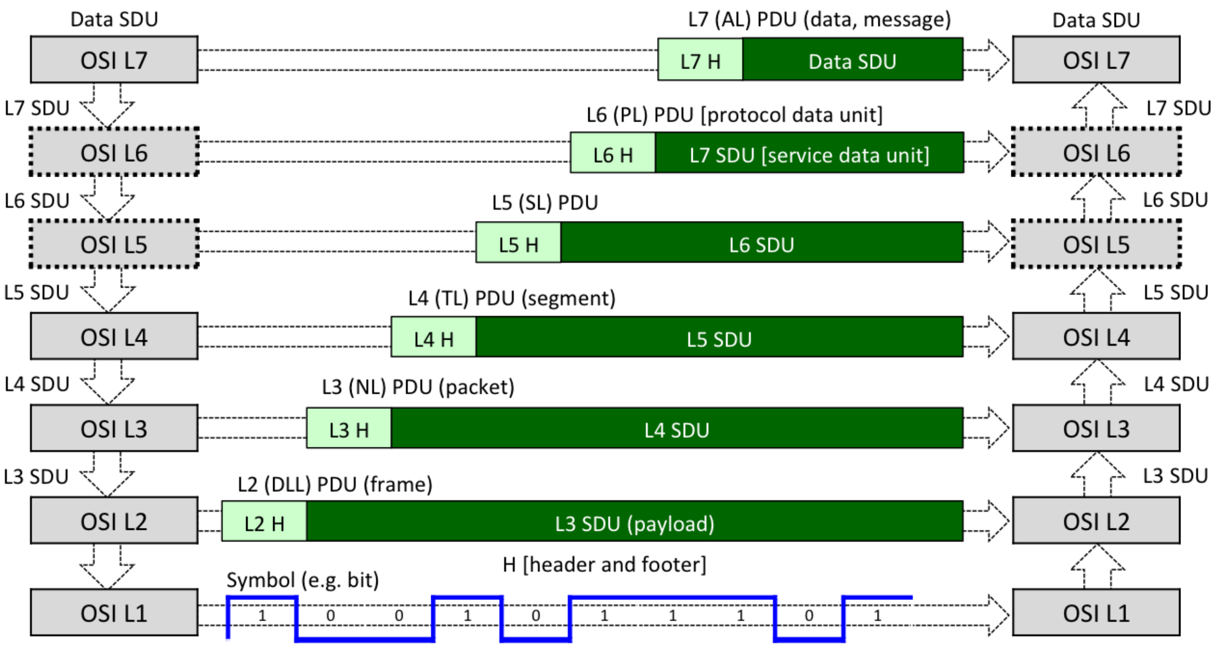


*Figure 1: The 7-layer OSI reference model is the base for all communication systems standardized in ISO; each layer adds a header/trailer to the payload (N-SDU)*

zonal gateways connected by means of high-bandwidth networks and also the sub-layered networks of each zone runs multiple applications.

Due to the fact that there are used many standardized and non-standardized application layers, these zone-oriented networks needs to be shared by different higher-layer protocols. This is more or less the same situation in many application fields – not just in the automotive industry, but also in any kind of mobile machinery ("machine on wheels"), embedded machine control, medical devices, laboratory automation, etc.

**The OSI reference model**

In general, all network technologies standardized in ISO (International Organization for Standardization) should use the 7-layer OSI (Open Systems Interconnection) reference model as standardized in ISO 7498-1:1994 [14]. Some communication technologies such as CAN and Ethernet do not fit perfectly in this model. Nevertheless, it is possible to describe them using an adapted and extended model.

Each layer of the seven OSI layers (see Figure 1) adds to the SDU (Service Data Unit) received from the layer above a trailer and footer. The trailer or the footer is optional. It is even possible that they are not added at all. The SDU plus trailer/footer are the PDU of the next lower layer. The protocol stack gives them as SDU to the next lower layer. This is done for the implemented OSI layers. If one of the OSI layer functions is not needed, it is not implemented. It is so-to-say "empty". No header or footer is added. In CAN-based networks, the CAN controller adds to the SDU from the layer above the SOF (start of frame) bit, some control field bits, the CRC field, the ACK field, and the EOF (end of frame) bits as well as the IMF (intermission field) bits. This is independent of the used data link layer protocol (Classical CAN, CAN FD, or CAN XL).

When receiving a CAN data link layer frame, the protocol stack in the CAN data link layer controller discards the header and footer added by the transmitting entity. The remaining bits an bytes – identifier bits, DLC (data length code) bits, and the data field – is forwarded as SDU to the next layer. This is done in the reverse way for all OSI layers. At the end, the OSI application layer hands over the so-called "payload" to the application software.

"Payload" is not a very precise term. I prefer SDU (given from layer to layer) and PDU (virtual peer-to-peer protocol). Because each layer deals with SDU and PDU, they need to be named different, in order to distinguish between them. Often the OSI layer is abbreviated and used as prefix. Example: AL-PDU or DLL-PDU (also named as L7-PDU resp. L2-PDU). The SDU provided by the application software could be named Data-SDU respectively Data-PDU.

**OSI layer configuration**

The CAN high-speed physical layer as standardized in ISO 11898-2:2016 is scalable in respect to the bit timing. The system designer can configure the bit timing to the application needs. Normally, this is done statically. However, it is also possible to do this dynamically. An example are the CANopen layer setting services (LSS) specified in CiA 305 [2], which are used for example in police cars, ambulances, and other special-purpose vehicles including those for disabled drivers. In case of CiA 305, the CAN network is used to configure the bit timing of the ECUs. One of the predecessor approaches is the Layer Management (LMT) specification in the CiA 200 CAN Application Layer series. CiA 205-1 specified the LMT services and CiA 205-2 specified the LMT protocols [1].

Also the other OSI layers can also provide configurability. A typical example is the network layer, which is responsible for the addressing. Several standardized CAN-based higher layer protocols use a part of the CAN identifier (ID) for this purpose. In SAE J1939-based networks the Source Address is embedded in the CAN-ID. In CANopen and Devicenet [13] networks, the node-ID is embedded in the CAN-ID. These

| Layer | CAN | CXPI | ETH | Flexray | LIN | MOST |
|---|---|---|---|---|---|---|
| 7 | ISO 14229-3<br>ISO 20730-1<br>**ISO 26021-1**<br>ISO 27145-3 | ISO 14229-8<br>ISO 20794-2 | ISO 14229-5<br>ISO 20730-1<br>**ISO 26021-1**<br>ISO 27145-3 | ISO 14229-4 | ISO 14229-7 | ISO 21806-2<br>ISO 21806-14 |
| (6) | Not considered | | | | | |
| (5) | Not considered | | | | | |
| 4 | ISO 15765-2 (DoCAN)<br>ISO 15655-4 | ISO 20794-3 | ISO 13400-2 (DoIP) | ISO 10681-2 | ISO 17987-2 | ISO 21806-4 |
| 3 | | | | | | |
| ISO 14229 series: Unified diagnostic services (UDS)<br>ISO 20730 series: Vehicle interface for electronic periodical technical inspection (ePTI)<br>ISO 26021 series: End-of-life (EOL) activation of on-board pyrotechnical devices<br>ISO 27145 series: Implementation of world-wide harmonized on-board diagnostics (WWH-OBD) | | | | | | |

*Figure 2: Excerpt of ISO standardized higher-layer protocols for in-vehicle networks*

SAs and node-IDs are configurable by means of dedicated management services using the same CAN network. Of course, it is challenging to do this dynamically during runtime of the normal communication. Especially, inconsistent configuration of the connected ECUs should be avoided.

If a protocol stack supports multiple protocols for one layer, a dynamic layer configuration is required. In order to avoid configuration inconsistencies, you need to configure both the producer and the consumer before they can communicate properly. Another option is to embed the configuration information in the SDU respectively PDU. The two-byte Ethertype field in the Ethernet frame is a well-know example. In CAN XL there is a similar field embedded in the CAN XL data frame. In CAN FD and Classical CAN such a configuration information can be embedded in the CAN-ID or the data field.

The OSI Management is standardized in ISO 7498-4. This framework does not specify dedicated services and protocols, but provides general definitions and guidelines. It defines several solutions including system management protocols, special-purpose layer management protocols, or management information carried in normal communication protocols.

**Higher-layer protocols need to be merged**

It would be all very simple, when just one protocol would be used for any OSI layer. No harmonization would be necessary and no consistency problems would exist. However, there are already many different higher-layer protocols for road vehicles standardized (see Figure 2). The automotive industry tries to simplify in-vehicle network architectures by reducing the number of implemented network technologies. But this does not solve the problem of multiple higher-layer protocol approaches. The reducing of the variety of protocols should start with the higher-layer protocols. One higher-layer protocol (application layer to transport layer) approach for in-vehicle network should be sufficient. The lower layers may differ depending on the requirements regarding reliability, robustness, maximum network length, achievable throughput, price, availability of hardware components, etc.

**Multiple Data-PDU implementations**

Autosar has introduced the so-called Multi-PDU concept. The application software respectively the middleware provides multiple Data-PDUs with a 4-byte or 8-byte header/footer as SDU to the OSI application layer. On the receiving side, the application software interprets the headers/footers and passes the received Data-PDUs to different software tasks. The headers/footers provide the configuration information of the Data-PDUs. This SDU embedded information allows a dynamic configuration.

Multiple Data-PDU concepts are also used in application layers based on Classical CAN. Typical examples are J1939-based application profiles including ISO 11783 (also known as Isobus) [16] and IEC 61162-3 (also known as Nema2000) [12]. They specify Parameter Groups (PGs), which are identified uniquely by PGNs (parameter group numbers) mapped into the CAN-ID field. Normally, these PGs are not configurable. It is so-to-say a static multiple Data-PDU already introduced in 1994.

Another example is CANopen introduced in 1994. PDO (process data object) messages can contain multiple process data. In opposite to J1939-based solutions, PDOs are configurable regarding the mapped process data. For this purpose, CANopen SDO messages are used, which can change the PDO mapping parameter set. Of course, the Classical CAN data field is limited to eight byte. This does not allow mapping several longer process data into one data frame.

With the introduction of CAN FD providing a data field of up to 64 byte, the situation has changed. CiA members developed the CiA 602-2 specification [6], which enables the mapping of multiple J1939 PGs into a CAN FD data frame. This multiple Data-PDU concept has been adopted by SAE and will be used in the J1939-22 specification, which is still under development.

The CiA 602-2 multiple Data-PDU messages can be mapped to CAN FD frames with 11-bit identifiers (FBFF: FD Basic Frame Format) and with 29-bit identifiers (FEFF: FD Extended Frame Format). The single Data-PDUs (named Contained PDU) are mapped into the data field. Due to the fact that the CAN FD data field is not organized byte-wise, it can happen that one so-called Padding C-PDU (Contained PDU) is necessary, in order to fill the data field to the length given in the DLC sub-field. This Padding C-PDU complies with Autosar specifications, e.g. the first three bits are “0”.

The header of the single Data-PDU (in CiA 602-2 called C-PDU) has a length of 4 byte. It comprises the TOS (type of service) field, the TL field (trailer length), the Data Page field, the PDUF as well as PDUS fields as specified in SAE J1939-21, and the PL (payload length). The Data-PDU trailer can have a length of 0 byte, 4 byte, or 8 byte. It is intended for cybersecurity and/or functional safety extensions.

**Running different higher-layer protocols on the same network**

In modern truck in-vehicle networks, different application layers are used. In order to run them on the very same CAN FD network, it is necessary to inform the receiving ECUs about the applied higher-layer protocol. In CiA 602-2, this is done by means of the 3-bit protocol indicator (PI), which is made to the 11-bit ID. There are identifiers specified for multiple Data-PDUs, Autosar CAN-NM, SAE address claiming (J1939-81), and the ISO TP protocols standards in ISO 15756-2. When using 29-bit IDs, dedicated PGNs indicate the used protocol. These dedicated PGNs are already assigned by SAE. They include protocol indicators for (multiple Data-PDUs, Autosar CAN-NM, XCP, ISO TP, ISO 11992 subnet addressing [17], and ISO 11783-3 transport protocols). This approach enables that different higher-layer protocols can share one single CAN FD network. Such PIs are embedded parameters in the SDUs.

*Table 1: Excerpt of ISO standardized communication approaches for passenger cars using CAN-based networks*

| Application purpose | AL | TL/NL | DLL | PL |
|---|---|---|---|---|
| UDS on CAN | ISO 14229-3 | ISO 15765-2/4 | ISO 11898-1 (Classical CAN and CAN FD) | ISO 11898-1 (Bit-timing settings are different) |
| EPTI on CAN | ISO 20730-1 | | | |
| Pyrotechnical devices | ISO 26021-1 | | | |
| WWH-OBD on CAN | ISO 27145-3 | | | |
| Keys: AL (application layer), TL/NL (transport and network layer), DLL (data link layer), PL (physical layer) | | | | |

Also in passenger cars and many other applications different higher-layer protocols are used. Sometimes they use separate communication systems; sometimes they share the same network. When they share the same CAN-based network, the usage of the CAN identifier field needs to be harmonized, so that double-use of IDs is avoided.

Table 1 shows an excerpt of ISO standardized communication approaches for passenger cars using CAN-based networks. If they share the same network, a heterogeneous multiple AL-PDU concept needs to be standardized. This means, the protocol stack must be dynamically configurable. As said above, this can be done by separate layer management (configuration) services using additional communication mechanisms or by means of configuration parameters embedded in the SDUs respectively PDUs for each layer, which requires configurability.

For CAN XL and perhaps also CAN FD such embedded layer management can be achieved. The proposed embedded layer management needs standardized protocol indicator (PI) parameters for the relevant OSI layer. To keep this idea generic, it is assumed that any standardized legacy application layer can share the network with any other standardized legacy application layer.

Besides the CAN-related standards listed in Table 1, also other standardized application layers need to be tunneled on the networks connecting zonal controllers and zonal sub-controllers (see Table 2). Of course, also proprietary application layers should be supported by means of a dedicated PI range.

When there are for each OSI layer such PI parameters standardized, it is possible to run OSI application layers on different OSI transport layers. This would enable migration paths from legacy solutions to a harmonized OSI layer solution in the future. Of course, these embedded PI parameters eat some bandwidth. In order to specify this by means of the OSI reference model, an extension of this model would be helpful.

In the CAN XL, there are reserved eight bits for PI purposes. They can be used for the embedded layer configuration. If the other layers do not provide such functionality, all permutations of OSI higher-layer approaches need to be considered. For the planned ISO 26021 (end-of-life activation

*Table 2: Potential other vehicle-related communication application layers to be tunneled on CAN XL based networks*

| Name | Application field | Related specifications |
|---|---|---|
| Body builder network [9] | Commercial vehicles | DIN 4630 (CANopen and J1939) |
| CiA 447 network [5] | Passenger cars | CiA 301, CiA 447 |
| C*le*ANopen [4] | Commercial vehicles | CiA 301, CiA 422 (EN 16815) |
| Commercial IVN gateway | Commercial vehicles | CiA 301, CiA 413, etc. |
| Firecan [10] | Commercial vehicles | DIN 14700-1 |
| ISOBUS [16] | Commercial vehicles | ISO 11783-3 |
| MOST | Passenger cars | ISO 21806-2/-14 |
| SAE J1939 [24] | Commercial vehicles | SAE J1939-21, SAE J1939-22 |
| UDS on CXPI | Passenger cars | ISO 14229-8, ISO 20794-2 |
| Truck/trailer network | Commercial vehicles | ISO 11992-2/-3/-4 |
| UDS on ETH | Passenger cars | ISO 14229-5 |
| UDS on FlexRay | Passenger cars | ISO 14229-4 |
| UDS on LIN | Passenger cars | ISO 14229-7 |
| XCP | Passenger cars | MCD-1 |

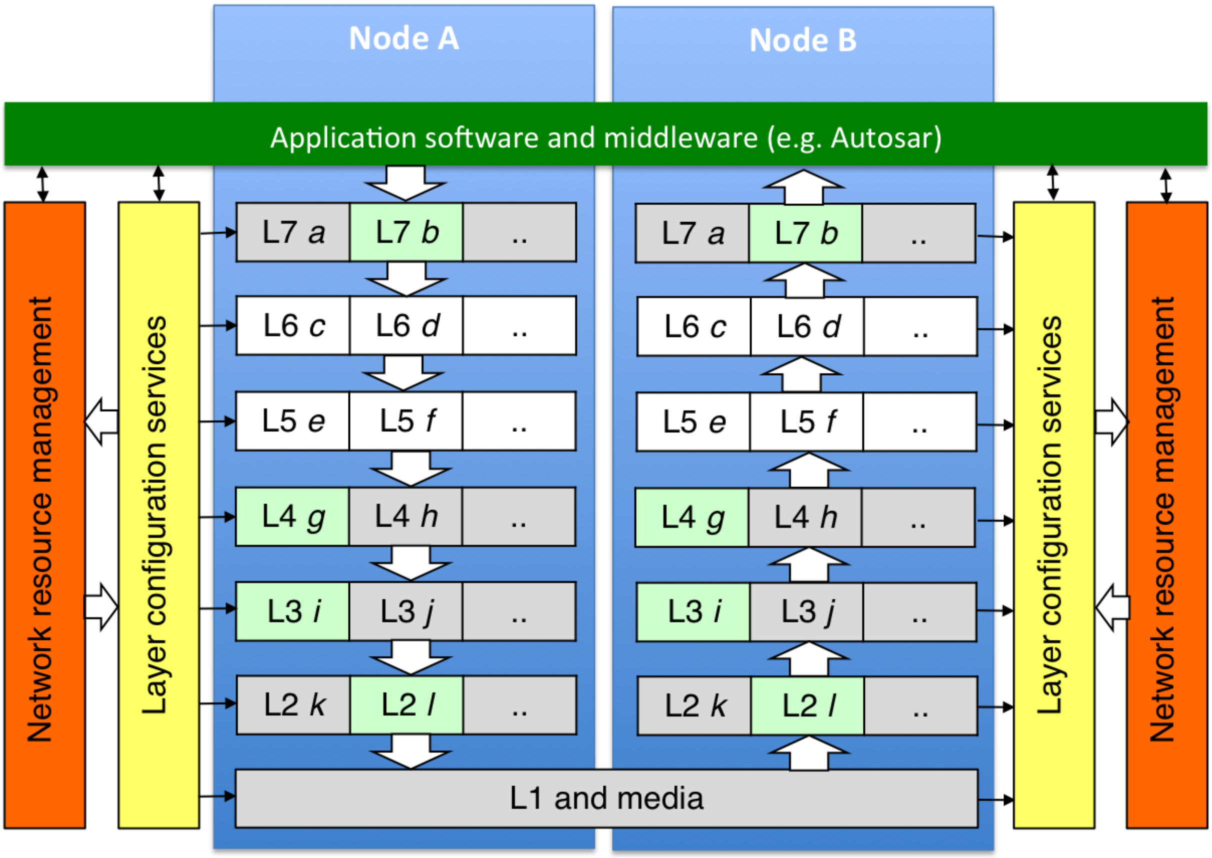


*Figure 3: Model of a multi-protocol stack approach with OSI Management functions (layer configuration and network resource management) defined in ISO 7498-4*

of pyrotechnical devices) two-cascaded approach (any combination of CAN-based and Ethernet-based network segments) four permutations are considered: DoCAN-DoCAN, DoIP-DoIP, DoCAN-DoIP, and DoIP-DoCAN. Not considering single-network implementations.

This is, why I would prefer a single-layer approach for the layer management. This means, each layer has its own PI parameter and not one indicating a set of layers. To simplify this, it would be possible to combine some layers and indicate them jointly. A combination of application, presentation, and session layers seems to be possible. Combining transport and network layers is in my opinion questionable and is not suitable for all application fields.

Figure 3 shows the proposed extension of the OSI reference model. The OSI Layer Management (OLM) specifies just the necessary services. They may be implement by means of protocols using separate communication systems, separate protocols on the same communication system or embedded in the PDUs.

The embedded PI information for the application is given by means of services from the application software via the OSI layer management respectively taken from the protocol of the next lower layer by the OSI layer management of the multi-protocol stack of the receiving ECU. This can be done by means of a not confirmed service (fire-and-forget) as used in CiA 602-2.

## Summary and outlook

There is always a debate on the design approach to be used: top-down versus bottom-up. Historically, CAN-based solutions were designed and standardized bottom-up. When discussing multi-protocol stack solutions, it seems that the top-down is the more appropriated one. I think, for in-vehicle networks this should be done by ISO TC22 SC31. In the long-term, just one higher-layer approach should be used. But it is necessary to provide a migration path from

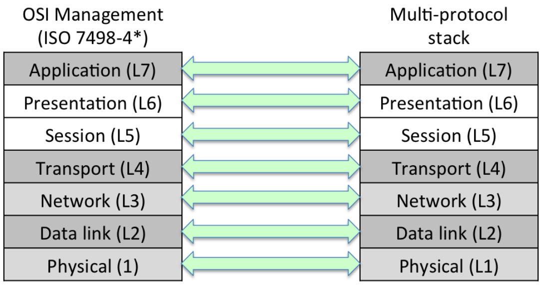


*Figure 3: Model of a multi-protocol stack approach with OSI Management functions (layer configuration and network resource management) defined in ISO 7498-4*

today's multi-protocol situation to a single protocol approach. For the lower layers, there may be several solutions optimized on different requirements (see above).

There is also a need for the management of layer resources. In order to provide scalability and to configure Quality-of-Services (QoS), such resource management is necessary. CiA 309-5 [3] is a first approach to standardize such resource management for CANopen-based networks.

Holger Zeltwanger
CAN in Automation e. V.
Kontumazgarten 3
DE-90429 Nuremberg
www.can-cia.org

**References and literature**


[1] CiA 200 series (seven parts): CAN Application Layer. CiA, Nuremberg (Germany), 1996

[2] CiA 305: CANopen layer setting services (LSS) and protocols. CiA e.V., Nuremberg (Germany), 2013

[3] CiA 309-5: CANopen access from other networks – Part 5: HTTP request mapping (not yet published). CiA e.V., Nuremberg (Germany), 2018

[4] CiA 422 series (six parts): CANopen application profile for municipal vehicles (CleANopen). CiA e.V., Nuremberg (Germany), 2018

[5] CiA 447 series (five parts): CANopen application profile for special-purpose car add-on devices. CiA e.V., Nuremberg (Germany), 2015 to 2018

[6] CiA 602-2: CAN-FD for commercial vehicles – Part 2: Application layer. CiA e.V., Nuremberg (Germany), 2017

[7] CiA 610 series (multiple parts): CAN XL (not yet published). CiA e.V., Nuremberg (Germany).

[8] Decker, Peter: CAN FD with dynamic multi-PDU-to-frame mapping, in 15th iCC proceedings. CiA e.V., Nuremberg (Germany), 2015

[9] DIN 4630: Road vehicles — Data parameter specification for body application units in commercial vehicles. DIN, Berlin (Germany), under development

[10] DIN 14700 series (11 parts): CAN Schnittstelle für Komponenten in Einsatzfahrzeugen. DIN, Berlin (Germany), 2014 to 2019

[11] I-PDU , Specification of I-PDU multiplexer. Autosar, Munich (Germany), 2017

[12] IEC 61162-3: Maritime navigation and radiocommunication equipment and systems – Digital interfaces – Part 3: Serial data instrument network. IEC, Geneva (Switzerland), 2008

[13] IEC 62026-3: Low-voltage switchgear and controlgear – Controller-device interfaces – Part 3: DeviceNet. IEC, Geneva (Switzerland), 2014

[14] ISO 7498-1, Information technology – Open Systems interconnection – Basic reference model: The basic model – Part 1. ISO, Geneva (Switzerland), 1994

[15] ISO 7498-4, Information technology – Open Systems interconnection – Basic reference model – Part 4: Management framework. ISO, Geneva (Switzerland), 1989
[16] ISO 11898 series (three parts): Road vehicles – Controller area network. ISO, Geneva (Switzerland), 2015 to 2017
[17] ISO 11783 series (14 parts): Tractor and machinery for agriculture and forestry – Serial control and communications data network. ISO, Geneva (Switzerland), 2011 to 2019
[18] ISO 11992 series (four parts): Road vehicles – Interchange of digital information on electric connections between towing and towed vehicles. ISO, Geneva (Switzerland), 2014 to 2019
[19] ISO 14229-3: Road vehicles – Unified diagnostic services (UDS) – Part 3: UDS on CAN implementation (UDSONCAN). ISO, Geneva (Switzerland). 2012 (under systematic review)
[20] ISO 15765 series (five parts): Road vehicles – Diagnostic communication over CAN. ISO Geneva (Switzerland), 2016 to 2020)
[21] ISO 20730-1: Road vehicles – Vehicle interface for electronic periodic technical inspection (EPTI) – Part 1: Application and communication requirements. ISO, Geneva (Switzerland), 2019 (under publication)
[22] ISO 26021-1: Road vehicles – End-of-life activation of on-board pyrotechnical devices – Part 1: Application and communication interface(s). ISO, Geneva (Switzerland), under systematic revision
[23] ISO 27145-3: Road vehicles – Implementation of world-wide harmonized on-board diagnostics (WWH-OBD) communication requirements – Part 3: Common message dictionary. ISO, Geneva (Switzerland), 2012
[24] MCD-1: XCP – Universal Measurement and Calibration Protocol. ASAM e.V., Hoehenkirchen (Germany), 2017
[25] SAE J1939 series (multiple parts): Serial control and communications for heavy-duty vehicles. SAE International, Detroit (USA), 2015 to 2019