

Fedor Kállay – Peter Peniak *

THE COMMUNICATION IN MECHATRONIC SYSTEMS

The paper deals with mechatronic systems and communication among mechatronic modules. This article proposes possible solution of usage open CAL protocol above CAN bus, which is independent from any manufacturers and can be widely accepted themselves. The main attention is paid to open CAN/CAL protocols including the viewpoint of their implementation, essential functions and potential integration within the vehicle control system. The required communication functions, communication models, communication objects (PDO, SDO) and applied profiles are being described in the CAL protocol. As an example a double bus CAN system is shown as a typical CAN/CAL solution for mechatronic systems in automotive industry.

1. Introduction

Mechatronics is the cooperation of mechanical engineering (“mecha” for mechanisms), electronic engineering (“tronics” for electronics), and software engineering. The purpose of this interdisciplinary engineering field is the study of automats from an engineering perspective and serves to view the control of advanced hybrid systems (Wikipedia), see Figure 1. The distributed control production system utilises CAN (Controller Area Networks) and VAN (Vehicle Area Networks) to link and integrate autonomous mechatronic modules.

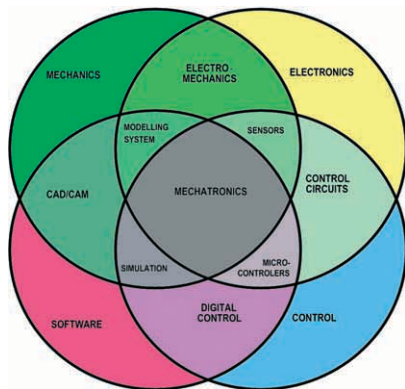


Fig. 1 Mechatronic -interdisciplinary conjunction

Automotive industry was one of the first areas where bus (fieldbus) networks for the communication of mechatronic systems in vehicles and cars was applied. The standard CAN (Controller Area Networks), developed by Bosh and standardized by ISO is still the most popular fieldbus system. There are typically two different types of CAN buses used in cars that differ from each other with available network bandwidth to provide redundancy and to meet the different requirements for end devices. The first bus has a speed of 500kbps and is used for communication between main

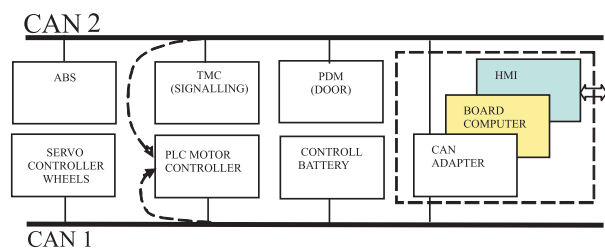


Fig. 2 Example of CAN network application in automotive industry

devices (servomotors, switches, etc.) and the second one with 125 kbps is applied as a communication system for the interconnection of the control subsystems. CAN is used mainly as

- machine control (control of batteries, servomotors of wheels, etc.),
- interconnection bus for various mechatronic systems (ABS, PDM, TMC),
- communication system for system diagnostic.

2. Communication requirements

The networks that are applied in mechatronic systems differ in many ways from standard networks covering needs of typical information systems. Some applications of different network nodes could be introduced in contrast to standard networks, prevailing communication between the machines and control units whereby the accent is put on the communication requirements of mechatronic modules. Different requirements are, therefore, focused on architecture and network protocols. Generally, the specific aspects could be summarized into considerations:

1. *Deterministic communication* is important to provide communication in real time. It is derived from necessity to control all subsystems in real time. Therefore, each node must have the access to the bus within a defined time window that has to be assigned with access method to bus (media).

* Fedor Kállay¹, Peter Peniak²

¹ Dept. of Mechatronic and Electronic, Faculty of Electrotechnical Engineering, University of Zilina, E-mail: Fedor.Kallay@fel.utc.sk

² Peter Peniak, Continental Matador, Puchov, Slovakia

High noise-resistance and fault tolerance in communication. The network must eliminate influence of noise and industrial disturbance so the communication can be able to achieve functions and information without any damage. The network protocols have to allow communication in real time. It is derived from the necessity to control all subsystems in real time. Therefore, each node must have the access to the bus within a defined time window that has to be assigned with a bus access method. The network protocols have to include mechanisms for the communication correction, fail-safe behaviour and auto configuration.

2. *Peer-to-Peer/ Master-slave communication.* The network must support a typical communication for central control (Master-slave/Pooling), see Fig. 3c,b in a cyclic mode, but also provide peer-to-peer communication among mechatronics systems too, so they can cooperate with each other (Fig. 3d) according to real needs. To provide a better utilization and time response, there are also optional communication modes available for communication on-demand such as Change-of-State (Fig. 3a) or Producer-Consumer well known in industrial networks.
3. *Priority communication.* The control process requires communication distributed into hierarchical levels among groups of nodes according to rate of importance and functional classification. For communication in real time the network has to support a division of nodes to the hierarchical levels with defined priority and communication ability.

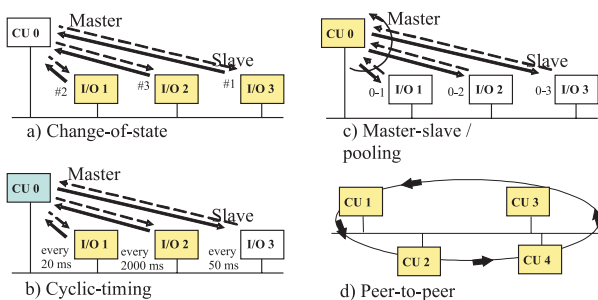


Fig.3. The used communication models for mechatronic modules

3. CAL/CAN Protocols

The application layer of CAN networks has to provide a wide stack of configurational and transfer functions for reliable interactions, including control functions and device profiles. Nowadays, there are several protocols available for CAN networks. Let's enumerate the most used:

- **CAL/CAN open** (*CAN Open Application Layer*) defined by **CiA** organization,
- **CIP** (*Common Industrial Protocol*) from **ODVA** organization (*Device Net*),
- **SDS** (*Smart Distributed Sensors*) from **Honeywell** company.

The main point of this article is to concentrate on the CAL protocol (*CAN Open Application Layer*) that was developed by the **CiA** organization (*CAN in Automation*) in 1993 as a result of the EU research programme. The CiA dealt with open communication of heterogeneous systems via CAN bus. CAL provides applicative, independent, object oriented background for production of distributed mechatronic systems. The CAN/CAL protocol stack according to the ISO model with sub protocols is shown in Table 1.

CAL protocol stack Tab. 1

LAYER	PROTOCOL			
Profile equipment	Profile A	Profile B	Profile C
7	CAL/CAN open (CiA)			
	NMT	DBT	LMT	CMS
-				
2	CAN 2.0			
1	CAN ISO 11898			

- **LMT** (*Layer Management*) - management of particular layers CAN,
- **NMT** (*Network Management*) - management and control of CAN network and net devices,
- **DBT** (*Distributor*) - assignment and distribution of identifiers for network nodes,
- **CMS** (*CAN Message Specification*) - message format and protocol CAN.

The protocol defines several methods for message transfer and communication objects. There are 4 main messages used in CAL communication:

- **AM** (*Administration Messages*) - for initialisation and configuration of network nodes, assignment of CAN identifiers by using sub protocols (LMT, NM, DBT)
- **SDM/SDOs** (*Service Data Messages/Objects*) - to transfer service data above 8B. These messages are also used during device configuration and for copying data buffers with lower priority.
- **PDM/PDO** (*Process Data Messages/Objects*) - defined for communication process in real time via data messages up to 8B size. The messages have high priority and could be applied for cyclic and non-cyclic communication according to the model "producer-consumer".
- **PM** (*Pre-defined Messages*) - messages for auxiliary purposes like communication by way using "Sync Object", "Time Stamp" objects, and service messages like "Emergency".

SDO object utilizes the supporting protocol "**CAL Multiplexed Domain Protocol**" to create a service channel for peer-to-peer communication of network nodes. An "Initiate Domain Request" procedure serves for transferring the messages up to 4B (without fragmentation). The first three octets contain an index (16b) and a sub-index (8b) that refer to the object/item **CMO** (**CMS Object**) in the object dictionary. The protocol enables reading and writing to the dictionary. The transfer of bigger blocks of data is ensured

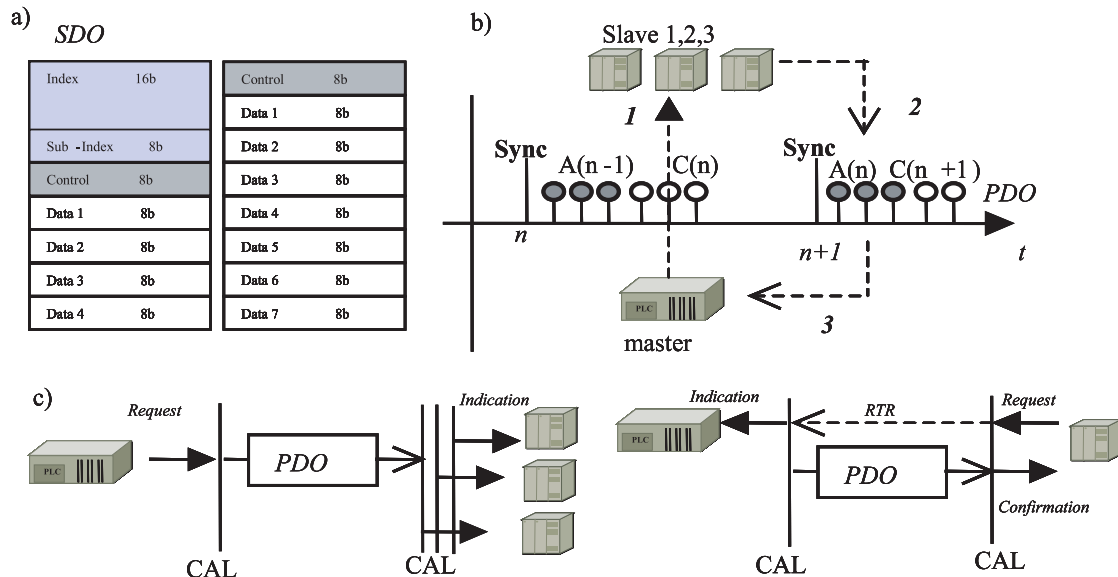


Fig. 4. The communication services of CAN protocol

by the "Acknowledged Fragmented Protocol" which allows transfer of message's fragments (7B) with acknowledging. Both protocols are shown in Figure 4a. A directory owner acts as SDO server and the remote device has a client role. Each device provides two predefined identifiers that are mapped to the SDO server and used for device configuration from network administrators' tools.

PDO object permits access to CMO through variables, events and data structures (domain). These are mapped to assigned messages/objects PDO by using "Index/sub index/data length" in the device object dictionary. The transfer is always initiated by a client with a "Store-and-Immediately-Notify-Event/Read-Event protocol". PDO objects supports communication via "Pooling" with the help of link layer services "CAN remote request" and as well Change-of-State mode with the "Event" object. The Event object is created by the SDO object via a service channel. Therefore, there is no need to transfer additional protocol information together with data transfer. The transfer of data is done synchronously with defined time windows according to "PDO-rate" parameters, which describe the time to send, continuation a periodically generated message "Sync Object". Another option for communication is a master-slave mode, the master generates requests for slave devices C (Command Messages), and slave devices have to answer in following time window (cycle) by sending their I/O values A (Actual Messages), see examples in figure 4b. Figure 4c shows the usage of "Store-and-Immediately-Notify-Event/Read-Event protocol", with PDO message sending from the master. Another option is a request for data sending from the remote node.

4. Assignment of can identifiers, network management and profiles

The assignment of CAN identifiers is one of the most important functions in each CAN network. CAN identifiers are used for

priority defining and addressing purposes for filtering messages among nodes. The original concept for an assignment of the identifiers assumed the static approach with predefined identifiers for typical kind of nodes (I/O device, etc.) and applications. Nowadays there is dynamic way of usage an assignment, so that CAN networks could be applied not only for defined application or object, but also for communication of the open systems. CAN protocol maintains a common list of identifiers (Pool). Each node with help of DMT protocol may ask for dynamic registration of free identifiers for given object or application. There are 1760 identifiers available in Pool for CMO objects. Only small part is predefined for specific purposes. For example 128 CAN identifiers were specified for monitoring of nodes (Node Guarding).

Network management. NMT server enables the network management and monitoring of the network nodes by the "NMT Guarding" function. The server maintains a list of active nodes and pass along "Guard Request" message in pooling mode, see Figure 5b. Each object has to answer for this request with the "Operational State" message. If a node does not answer in a defined time window NMT server creates error message. The same procedure is applied in the network node. If a "Guard Request" message is not received in defined time window then an error message will be generated for running applications.

Device profile is a standard tool for modelling various devices and their communication abilities. CAN defines common directory for all network "Object Dictionary" objects, where each object is addressed by 16-bit index and 16-bit sub index. The directory is divided into 4 parts. The first part "Definition of data types" is used to define all objects, the second one "Communication Profile Area" describes the mapping of application objects to PDO objects and defined SDO channels. The third part "Standardized Profile Area" consists of device profiles, for example for I/O modules, converters, engines. The fourth part "Manufacturer Specific area", con-

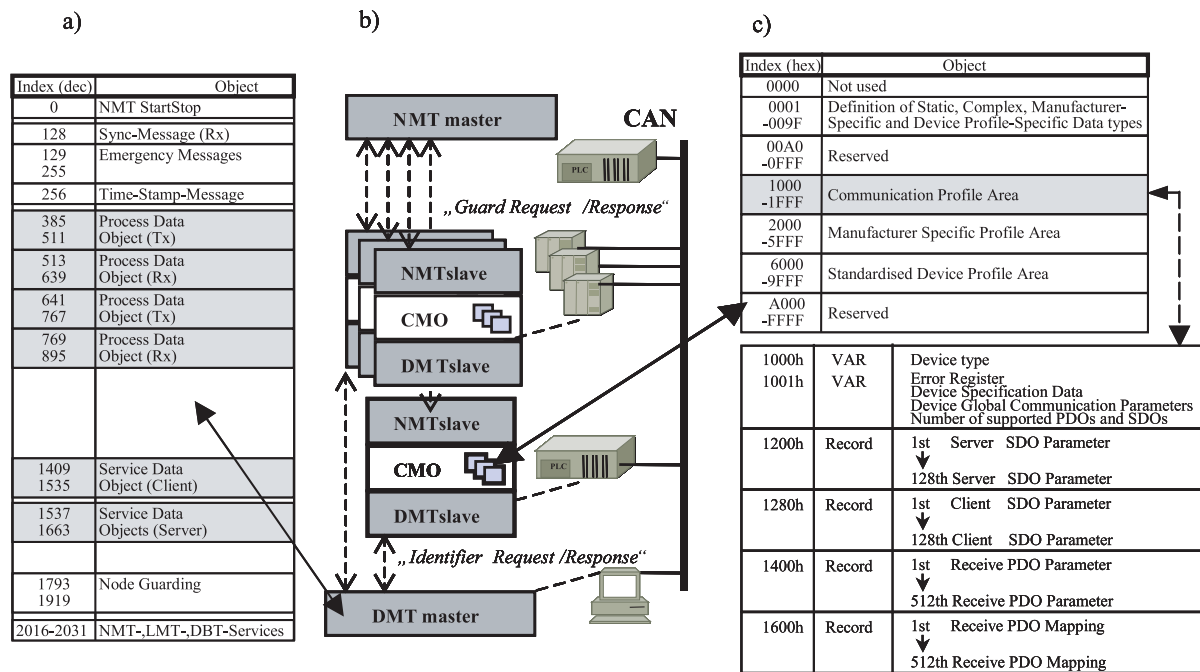


Fig. 5. Supporting services NMT and DBT in CAL protocol

tains profiles of specific manufacturer’s devices. The example of CAN communication profiles is shown in Fig. 5c.

Conclusion

This article deals with the specific communication requirements of the mechatronic modules. The main intention was paid on the possible solutions for communication of the open mechatronic systems. The problem is how to design and establish the communication between heterogenic mechatronic subsystems from various producers. Then a question arises regarding the necessary communication interfaces and a way of their integration. This article proposes one of the possible solutions of applying open CAL protocol above CAN bus that is independent on any manufacturers and can be widely accepted themselves. One of typical applications

was shown on Figure 2 where CAN double bus enables communication for different machatronic modules via standard CAL protocol with the defined communication objects and device profiles. CAL is the one of the perspective solutions for mechatronic applications and modules not only in automotive industry, but also in manufacturing area (CNC machines, robots, embedded devices).

Acknowledgement:

The authors wish to thank for the financial support to VEGA - Scientific Grant Agency of Slovak Republic for the projects No. 1/3086/06 “Research of the New Methods of Modelling-, Control and Simulation of Mechatronic Systems” and VEGA No.1/3123/06 “Research of Commutation Processes in the Power Transistor Structures and Optimisation of their Control at Soft-Switching Mode”.

References

[1] MAHALIK, N. P.: *Fieldbus technology*, Industrial network standard for Real - Time Distributed Control, Springer, 2003
 [2] KÁLLAY, F., PENIAK, P.: *Computers networks LAN/MAN/WAN and their applications (in Czech)*, Grada Publishing a.s., 2003
 [3] SPÁLEK, J. et al: *Decision and management with artificial intelligence support (in Slovak)*, EDIS, Žilina, 80-8070-354-X
 [4] FRANEKOVÁ, M., JANOTA, A., RÁSTOČNÝ, K.: *Trends in segment of communication safety of industrial networks (in Slovak)*, J. AEEE, 4/2005, pp. 250-253, ISSN 1336-1337
 [5] GUTTEN, M., KORENČIAK, D.: *Utilization of neuron networks in monitoring system of power transformers*, Transcom 2003, 5th European Conference of Young Research and Science Workers in Transport and Telecommunications, Section 2, Žilina
 [6] <http://en.wikipedia.org/wiki/Image:MechatronicsDiagram.svg>.