CAN, a Ten Years' Anniversarial Review

Uwe Kiencke, Timo Kytölä

Institute of Industrial Information Systems, University of Karlsruhe
Hertzstr. 16, 76187 Karlsruhe, Germany
Fax: +49-721-755788
Email: kiencke@iiit.etec.uni-karlsruhe.de

ABSTRACT

This paper gives a survey on the historical background of CAN. Additionally, the paper handles different CAN controller implementations and discusses the use of CAN in automotive systems and in automation, including initiatives for open systems in both areas. Although originally developed for in-vehicle purposes, automation area started to apply CAN as soon as the first components became available. Since the introduction of the first CAN chips the sale of CAN components has been remarkably higher in automation than in automotive sector. However, the roles are supposed to change soon due to the growing use of CAN in automotive electronics.

1 Introduction

Automotive companies began in the early 1980's to equip upper class cars with several electronic control units. Since then, the portion of electronics in cars has grown continuously and this trend is expected to go on in the future. At first, electronic systems worked independent of each other. However, it became soon obvious, that an intensive interaction between single systems would offer significant functional improvements. Automotive electronics thus developed into an architecture already applied in automation: distributed real-time control.

Distributed real-time systems require a fast communication link between different subsystems. Requirements for the characteristics of the communication are predominantly driven by control aspects rather than by the classical data transfer. As a consequence, the Controller Area Network (CAN) communication protocol was developed to support efficiently such applications. In order to be accepted by automotive companies, costs for the required communication hardware had to be an order of magnitude lower than the costs for corresponding communication solutions used in automation at that time.

Automation area became interested in CAN in a very early stage. CAN was soon used in various automation applications due to its well-fitting features and low hardware costs. Since the introduction of the first CAN chips the sale of CAN components has been remarkably higher in automation than in automotive sector. Not until now, after 10 years of CAN, a turning-point seems to be reached due to the growing use of CAN in automotive electronics.

This paper gives a survey on the historical background of CAN. Additionally, the paper handles different CAN controller implementations and discusses the use of CAN in automotive systems and in automation, including initiatives for open systems in both areas.

2 Historical Background

The need for a powerful real-time communication between autonomous electronic units was realized in the beginning of 1980's at Robert Bosch GmbH. At this time, three electronic units - engine control, automatic transmission control and anti skid braking - should be implemented for one car. Distributed real-time control involving all those electronic units allows the implementation of functional enhancements like e.g. reduction of motor moment during a transmission activity.

Alternative solutions, use of parallel I/O between electronic units or a centralized approach integrating several control function in one electronic unit, were eliminated because of lacking feasibility. Parallel I/O became economically and technically unreasonable when the need for interaction increased. A centralized approach would require a very powerful microprocessor due to the lacking parallel processing capacity which means higher costs. An additional disadvantage of the centralized approach is the strongly reduced availability by a failure of the electronic unit.

In 1983 Bosch started to develop the CAN communication protocol for real-time control in cars, because the solutions for serial communication available at the time were not in accordance with the requirements of automotive section. Although CAN protocol was first of all developed for real-time communication between electronic units, it was also kept in mind that CAN could be applied throughout the different application areas in cars. This includes a lower data transfer among electronic units of chassis, i.e. multiplexing.

CAN was designed to work in an environment that is stressed with electromagnetic noise. In addition to the reliability ensured by powerful error detection mechanisms, CAN contains a multi-master feature to increase the availability through fast recovery times after detected errors. The bit-wise arbitration scheme allows a non-destructive arbitration when more than one stations try to start sending a message at the same time. This means that in a competition situation always one station, i.e. the one sending the message with the highest priority among competing messages, wins and may complete the transmission alone. Therefore, no communication capacity is wasted for bus access conflicts.

The non-destructive arbitration makes it possible to fulfil response times in a real-time control system with a rather low bus transfer rate. The transfer rate should be kept low to reduce the electromagnetic noise emission caused by the communication bus. A disadvantage resulting from the bit-wise arbitration is that the maximum bus length depends on the used transfer rate: a higher transfer rate corresponds to a shorter bus length.

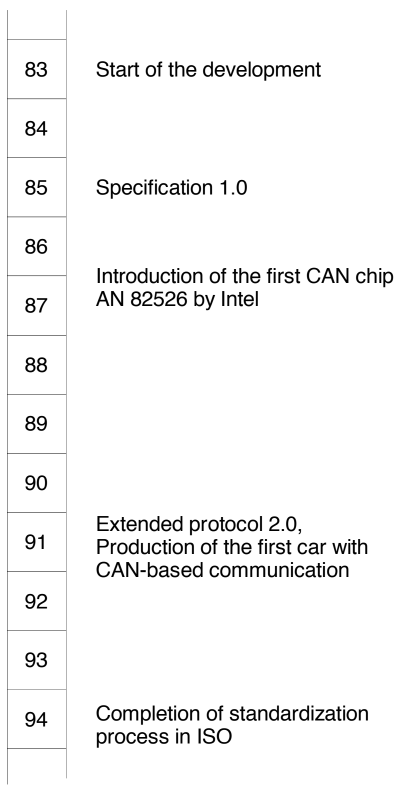

Fig. 1: Historical development of CAN

Figure 1 shows the most important milestones during the historical development of CAN. The standardization of CAN together with two other protocols for in-vehicle communication, VAN and J1850, was completed by the International Standardisation Organisation (ISO) in 1994. However, the initiators for other communication protocols in Europe switched over to CAN mainly due to the availability of CAN chips offered by various semiconductor companies. CAN evolved thus to a de facto standard for in-vehicle communication in Europe.

In 1991, the first luxury-class car, the S-class Mercedes, was equipped with CAN linking five electronic control units together with a transfer rate of 500 kBit/s. In 1996 and 1997 CAN will be applied by the most European automotive companies above all for engine control in upper class cars. The same tendency is also supposed to characterize the US market.

3 CAN Implementations and Components

Different CAN implementations corresponding to the same protocol version are compatible concerning the communication protocol. The implementations differ from each other concerning the interface between the CAN controller and the application microcontroller (μC).

The first CAN implementation, formerly called Full-CAN, was developed as a co-operation between Bosch and Intel. This implementation has a storage buffer for message objects which are configured by the application μC. Within one message object a message with a certain identifier is specified to be received or transmitted. In case of a transmission the μC loads updated data into the data part of the message object and sets a transfer request. The CAN controller then takes care for the transmission of the message. When a message corresponding to a receiver message object is received, the CAN controller writes the data into that message object and informs the μC which can in turn read the received data. CAN implementation with an object storage thus relieve the μC of message acceptance filtering.

The second CAN implementation was introduced as Basic-CAN. This implementation version offers reception and transmission buffers for the application μC. The implementation performs only a limited acceptance filtering: a code and a mask register allow the reception of chosen identifier groups, whereby a further acceptance filtering is left to the μC. This implementation suits well for applications where messages with many different identifiers are to be received by one network node or where the additional acceptance filtering on the μC can be tolerated. The advantage of this implementation is that it requires only a small chip area which in turn reduces the price.

Later CAN implementations combine the above mentioned two approaches. There exist also a CAN implementation which does not require a μC. A component of this kind, called SLIO (Serial Link I/O), can perform simple I/O operations and communicate with other nodes via the bus.

CAN implementations are available by numerous semiconductor companies as stand-alone controllers or integrated with a microcontroller on a single chip. When different types of CAN components are used within one bus, one should ensure that the component types implement the same protocol version or that components with different protocol versions can be used in a compatible manner. In 1991, a new protocol version 2.0 was introduced to support extended identifier format of 29 bits, instead of the former 11.

Figure 2 shows the expected total sale of CAN components and other components used in automation [1]. The sale of CAN chips in automation is expected to be in 1996 for the last time higher than the sale in automotive sector.

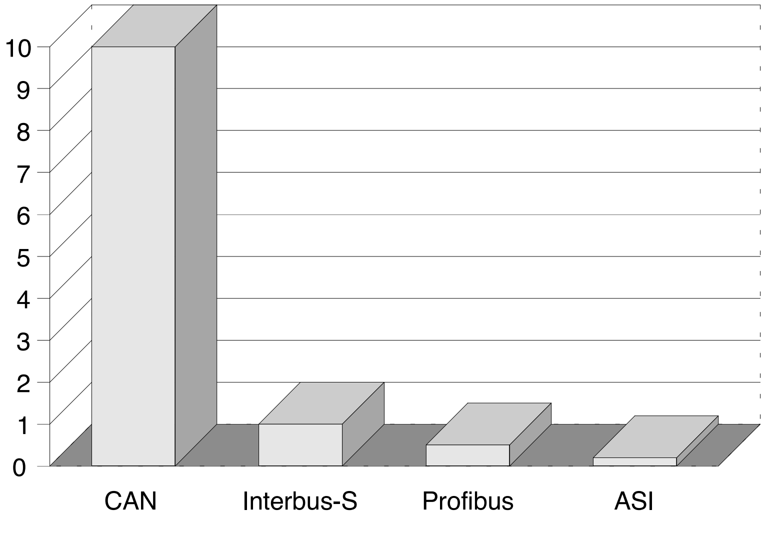

Fig. 2: Total sales of different communication chips in millions

4 CAN in Automotive Sector

As already mentioned, CAN was predominantly designed for real-time control in automotive applications. CAN allows the implementation of control loops among more than one electronic control unit. However, the use of such control loops with CAN is restricted to those cases, where the response times achieved by CAN are sufficient. The response times are limited due to the maximum transmission rate of 1 MBit/s which can be achieved when the bus is not longer than 40 m. The response times are further limited through bus access conflicts when more than one stations start sending a message simultaneously.

Future implementation of enhanced functions, like e.g. vehicle dynamic control and co-pilot system, needs brake and steering actuators with electrical interfaces. In such applications, it might become necessary to implement control loops over the bus with shorter response times than it is possible with currently available communication solutions. This could lead to a development of a more powerful real-time communication bus with deterministic response times [2].

It was during the development ensured that CAN could also be used for networking in other automotive application areas in addition to real-time control. One important area is the communication between different units within chassis and comfort electronics. In this area, also called multiplexing, the wiring harness can be drastically reduced by networking. This makes the installation of electronics easier and can in some cases also reduce costs. Additionally, CAN suits as a communication solution to diagnosis and to an integration of the user interfaces of different mobile communication units.

4.1 OSEK/VDX

As a joint project in the automotive industry, OSEK/VDX aims at a standardization of an open system for distributed automotive electronics. Although originally initiated by CAN users, the specified higher communication layers can be applied also in systems based on other protocols.

OSEK/VDX takes its rise from the German project OSEK [3] and the French project VDX [4]. OSEK is an abbreviation for the German term "Offene Systeme und deren Schnittstellen für die Elektronik im Kraftfahrzeug", in English: open systems and the corresponding interfaces for automotive electronics. VDX is an abbreviation of "Vehicle Distributed eXecutive", a similar project started in French automotive industry which has been harmonised with OSEK. The current OSEK/VDX consortium is made up by the companies BMW, Bosch, Daimler-Benz, Mercedes-Benz, Opel, PSA Peugeot Citroën, Renault, Siemens and VW with the Institute for Industrial Information Systems of the University of Karlsruhe as a co-ordinator. Moreover, since the beginning of 1996 several companies from different countries and industrial areas, i.e. car manufacturers, suppliers, semiconductor industry and software companies, have joined the OSEK/VDX project as associated partners.

Open automotive systems consist of several ECUs connected by communication links. Generally, these control units are supplied by different companies, and they have different microcontroller architectures. The realisation of distributed system functions require that stations exchange messages using standardized interfaces and protocols. A uniform network management has to guarantee the safe operation of the distributed systems. In addition, an operating systems with uniform interfaces offer different application programs available from various suppliers to co-exist in a single processor. The impact of different standards on software is a major cost driver in automotive electronics that must be significantly reduced. This can be achieved by fixing interfaces and higher protocol layers for operating system, communication and network management not just within one subsystem, but for the entire distributed system. With OSEK/VDX this expensive development investment is needed only once, and it is possible to re-use it with minor modifications for various applications.

The open architecture for communicating vehicle systems specified by OSEK/VDX is shown in figure 3. It comprises the areas:

- Real-time Operating System (Execution environment of ECUs),
- Communication (Data exchange between ECUs and within ECUs) and
- Network Management (Configuration registration and monitoring of ECUs).

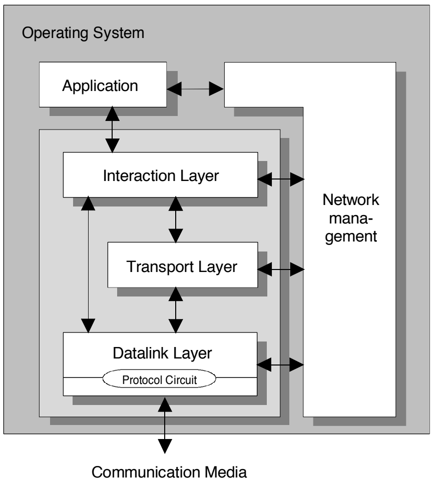

Fig. 3: Layered Model of OSEK/VDX

These areas are considered as competition-non-relevant by the OSEK/VDX partners. Implementations based on the OSEK/VDX specifications will however overcome project- and customer-specific reformulations of the same basic control functions.

The main advantages resulting from using of OSEK/VDX are:

- significant reduction of development costs and time,
- enhanced quality of ECU software,
- integration of functional software modules from different suppliers into one microcontroller becomes possible,
- communication link between ECUs with different controller and network architectures, and
- increased functional performance of the entire vehicle by using all distributed resources and information.

For a more detailed description of OSEK/VDX we refer to the presentation "OSEK/VDX - An open Software Architecture for Communicating Vehicle Systems" [5] in session 4 "Open systems architecture" of this conference and to the proceedings of the 1st OSEK/VDX workshop [6].

5 CAN in Automation

Automation area started to apply CAN as soon as the first components became available. CAN protocol fulfils the communication requirements of a broad scale of automation applications. Therefore, CAN has been used in very different kinds of applications from elevator systems and ships over to medical analysis equipment.

The reason for the fast increasing amount of CAN-based applications in automation is the cost/performance relation: CAN offers a relative high transfer rate and a high-level reliability for a low price in comparison to other protocols. CAN component prices are supposed to become even lower in the future due to increasing sales when CAN is used also in middle and lower class vehicles.

The penetration of CAN into automotive electronics have been rather slow because of an immense cost pressure and a safety critical character of the applications. Additionally, automotive sector has had to go through a learning process to be able to adapt distributed real-time control. Automation area, on the contrary, was already familiar with distributed system architectures and could apply CAN without any reorientation. Therefore, the sale of CAN chips has been remarkably higher in automation than in automotive sector.

Figure 2 showed approximated total sales of CAN chips and components implementing other protocols for automation. The sale of CAN chips in automation is expected to be slightly more than the half of the total sale which still means a very clear leading position for CAN as a communication protocol in automation.

Although CAN can be used either as a field or a sensor/actuator bus, CAN sometimes co-exists with other communication protocols in automation applications. CAN is then applied as a sensor/actuator bus and an other protocol with higher performance is used as a field bus for a transmission of larger data sets [7, 8].

The non-profit international users and manufacturers group "CAN in Automation" (CiA) has made a remarkable contribution to the enhancement of the use and prestige of CAN by providing technical support and product information. Further on, CiA has specified a standard higher level protocol for CAN-based industrial systems, i.e. CAL (CAN Application Layer). This application-independent standard layer provides a means of open communication between modules made by different manufacturers. The specified standard also reduces the effort of communication software development which contributes to more competitive products.

6 Conclusion

Since the introduction of the new protocol version to cover extended identifiers in 1991 CAN protocol has remained unchanged and there are also no modifications in sight today. Although CAN was developed for in-vehicle communication, it has been used more in automation than in automotive sector since the first components came into market about 10 years ago. Automotive area, at the time already familiar with distributed real-time control, started to use CAN in various applications due to its very suitable features and low prices. From now on, however, the sale of CAN components is expected to become larger in automotive sector.

7 References

[1] CiA Press Release, April 1996.

[2] Th. Raith, et al.: "Netzwerke zur Integration von Systemfunktionen der Kraftfahrzeug-Elektronik", it + ti - Informationstechnik und Technische Informatik 37 (1995) 6.

[3] U. Kiencke, et al.: "Open Systems and Interfaces for Distributed Electronics in CARS - OSEK", SAE Technical Paper No. 950291, International Congress and Exhibition, Detroit, Michigan, USA, 1995.

[4] P.-H. Jossa: "VDX, the Distributed Real Time Executive", SAE Technical Paper No. 94A1053, International Congress and Exhibition, Detroit, Michigan, USA, 1994.

[5] U. Kiencke, K.J. Neumann: "OSEK/VDX - An Open Software Architecture for Communicating Vehicle Systems", 3rd international CAN Conference, Paris, France, 1996.

[6] "Proceedings of the 1st International Workshop on Open Systems in Automotive Networks", U. Kiencke (ed.), Karlsruhe, Germany, October 1995

[7] H. Senninger: "CAN-Bus und Profibus teilen sich die Aufgaben", VMEBus, June 1991.

[8] S. Gemperlein: "CAN - mehr als nur ein Fahrzeugbus?", iNet '92 Congress and exhibition for industrial network technics, Proceedings, Stuttgart/Sindelfingen, June 1992.