Chapter 5
Applications

Guenter Reichart, Gabriel Leen, Nathalie Courmont, Ralph Knüppel, Christian Schmid and Markus Brockmann

5.1 Electronic system architectures of Automobiles Application of CAN Bus

5.1.1 Bus Systems in Automobiles

Bus systems in automobiles allow the communication, which means the exchange of data between the electronic control units (ECUs), smart sensors and actuators. Depending on the respective requirements, different bus systems are used. Typical requirements consist of required data rate, allowed message length, number of connectable nodes (ECUs), required topologies, requirements on deterministic transmission capability and in further reliability, availability or safety-oriented requirements. Further requirements address aspects of physical characteristics, like tolerance against voltage deviations, temperature stability, impacts on wiring harness (electromagnetic compatibility (EMC), copper wire, plastic or glass fibre,

N. Courmont (✉)
Airbus France S.A.S., 316 Route de Bayonne, 31060 Toulouse Cedex 03, France
e-mail: Nathalie.Courmont@airbus.com

G. Reichart
BMW AG, Petuelring 130, 80788, Munich, Germany

G. Leen
University of Limerick PEI, Limerick, Ireland
e-mail: gabriel.leen@ul.ie

R. Knüppel · C. Schmid
Airbus Deutschland GmbH, Hünefeldstr. 1-5, 28199 Bremen, Germany
e-mail: ralph.knueppel@knueppel-online.de

C. Schmid
e-mail: Christian.Schmid@airbus.com

M. Brockmann
WILO AG, Nortkirchenstrasse 100, 44263 Dortmund, Germany

W. Lawrenz (ed.), *CAN System Engineering,* DOI 10.1007/978-1-4471-5613-0_5,

unshielded twisted pair (UTP) cabling or shielded twisted pair (STP) cabling) and last but not least, cost aspects.

In automotive engineering, bus systems are differentiated according to the so-called Society of Automotive Engineers (SAE) classes:

5.1.1.1 Class A

Bus systems for simple applications with low data rates of up to 10 kbit/s, e.g., sensor data or simple control commands. The main application domains are relatively simple functions without safety relevance in the body domain. The transmitted messages are mainly short and event triggered with a low data rate. The application area is relatively cost sensitive and demands therefore a rather cheap interconnection technology.

5.1.1.2 Class B

Bus systems for applications with data rates of 10 kbit/s and up to 125 kbit/s (e.g., many more complex body functions).

5.1.1.3 Class C

Bus systems for applications, which require real-time behaviour with data rates of 125 kbit/s and up to 1 Mbit/s (engine domain and chassis domain). In these applications, domains at high data rates with defined low latencies of data transmission are required.

5.1.1.4 Class D

Bus systems for the data transmission of long data streams with high bandwidth. These requirements prevail mainly in the area of infotainment and entertainment, e.g., for the transmission of audio/video streams.

International Organization for Standardization (ISO) differentiates bus systems only in two steps:

- Low-speed communication (bit rates < 125 kbit/s) and
- High-speed communication (bit rates > 125 kbit/s).

All these classifications are not really satisfying to adequately describe all the relevant requirements. A classification which is mainly focused on bandwidth is not sufficient to describe the requirements of the different application domains. Due to the development towards higher bandwidth and towards wireless data transmission, this traditional classification concept has to be reconsidered anyway.

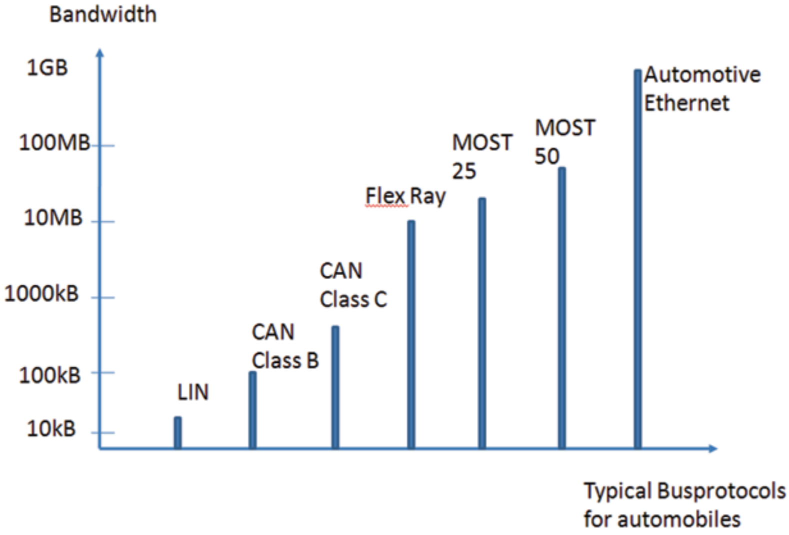

Fig. 5.1 Bus protocols and their bandwidth

The controller area network (CAN) bus protocol is currently applied in two different variants which correspond to class B and class C of the SAE logic.

Figure 5.1 shows typical bus protocols for automotive applications which are either already in use or in development. They are ranked according to their bandwidth.

Simple functions can be covered by the Local Interconnect Network (LIN) bus which allows for a data transmission of up to 20 kbit/s. The low-speed CAN bus can operate with a data rate of 5 kbit/s and up to125 kbit/s and in a network of up to 32 nodes. The strengths of the low-speed CAN are fault tolerance and the possibility to transmit over a single wire connection.

Due to the ever-increasing data rates and the ongoing trends towards higher functional integration along with a decreasing cost advantage of the low-speed CAN versus the high-speed CAN (Class C), low-speed CAN will soon go out of use. There is a trend visible that, in future vehicle architectures, the cost-effective LIN will be used for rather elementary functions. More demanding functions will be realized using the high-speed CAN and/or FlexRay, especially for time-critical or safety-critical applications. Media Oriented Systems Transport (MOST) will be used for multimedia applications in the infotainment and entertainment domain. In a broader perspective, Ethernet will play a significant role for system interconnection and can replace some of the traditional bus protocols. First applications for vehicle flash and diagnostic access are already in the market.

The electronic system architectures which we find today in modern cars will not change suddenly. Different bus protocols will be used even in the coming years since a radical change of the architecture would create huge costs and high quality risks. Even if the goal of the system architect remains to establish a more homogeneous network and therefore a reduction of the number of different protocols, the only solution can be to establish a clear and feasible migration plan.

5.1.2 The Application of CAN in Today's Vehicle Networks

The CAN protocol is, in today's vehicle networks, primarily used for the following three domains:

- Body electronics and active systems of passive safety,
- Chassis domain and driver assistance and
- Engine domain.

The main applications in the body domain deal with the control of windows, doors and flaps, mirror adjustment, control of lights, seat adjustments, climate control and comfort access. For cost reasons, the low-speed CAN plays a significant role but is in competition with the LIN bus. The safety electronics require a fast and safe data transmission; thus, the interconnection of the ECUs is, in most cases, realized by the high-speed CAN.

Chassis control systems as well as driver assistance functions put rather demanding requirements on the safety of the data communication and on the timing. Even if the high-speed CAN does not allow for a deterministic data transmission, high bandwidth can provide a sufficiently low latency in many applications. This implies, however, that only 50% of the maximal data-transfer capacity can be exploited. Experience has shown that, beyond this level, non-deterministic latencies begin to rise. The CAN protocol contains a number of supervisory functions and error recognition concepts:

- Cyclic redundancy check (check of test sums),
- Frame check (check of frame length and structure),
- ACK error (proof of acknowledgment),
- Bit stuffing (error check on bit level by stuff bits) and
- Level monitoring (monitoring of the bus level by the connected ECUs).
- These features and the multi-master concept for the bus access are the foundations why the CAN bus has become a very reliable interconnection technology which has extended, beyond its original scope, the automobile into field bus applications in automation technology.

5.1.3 CAN and AUTOSAR

For the software architecture of ECUs, the international AUTOSAR standard has become increasingly widespread. The acronym AUTOSAR means AUTomotive Open System ARchitecture (see also Sect. 6.2). Within the software, one can speak of architecture if the application level as well as the system basis level is realized in a defined, structured manner. One speaks about an open architecture if the interfaces are standardized and disclosed. Usually, a certain independence from technologies

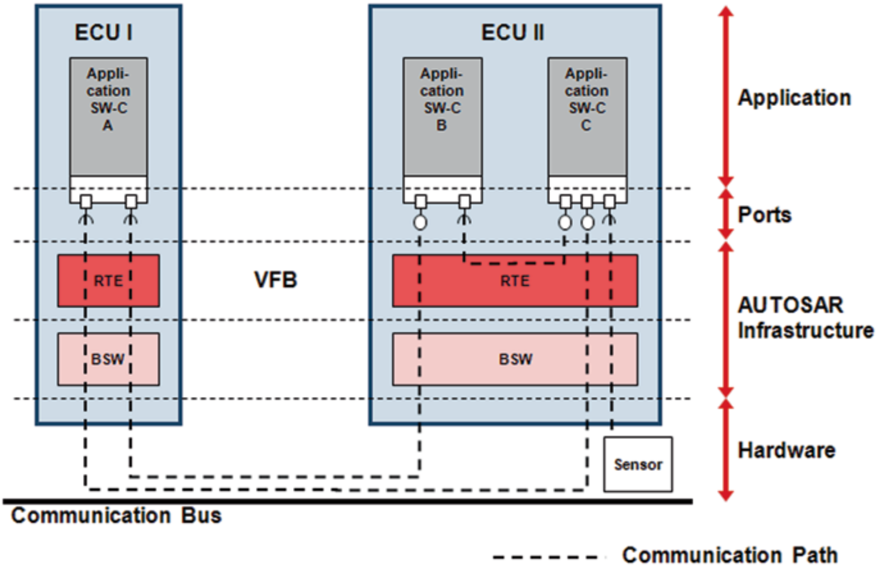

Fig. 5.2 Communication within and between ECUs according to the AUTomotive Open System ARchitecture (AUTOSAR) Standard. (According to Simon Fürst, AUTOSAR Guided Tour 2010)

by the introduction of abstraction layers is also a prerequisite. Based on these characteristics, AUTOSAR allows a transferability of software modules within or between ECUs. Moreover it can support, in the longer run, the exchange of software modules between different original equipment manufacturers (OEMs) if they comply with the AUTOSAR standard.

The system basic functions comprise, e.g., system services (operating system (OS), memory, network, diagnostic and ECU management), the microcontroller abstraction, device driver, driver for communication and communication services, communication hardware abstraction, etc.

The application layer docks on the so-called *Run Time Environment* (RTE) by means of standardized interfaces. The RTE is frequently called a *Virtual Function Bus*, a middleware layer, which allows the communication of the software modules within and between the ECUs (Fig. 5.2).

Ports implement the interface according to the communication paradigm (here: client–server-based).

They are the points of interaction of software components. The communication works through the RTE. The communication layer of the system basis software is encapsulated and not visible at the application layer.

To support the existing variety of bus technologies, one has to establish tailored standard solutions for the system basis functions. These solutions are packed in so-called stacks (e.g., CAN stack and FlexRay stack) and are offered by a number of first and second tiers. Up to now, it was not possible to standardize the system basis functions to such a level that one standard solution would meet the requirements of

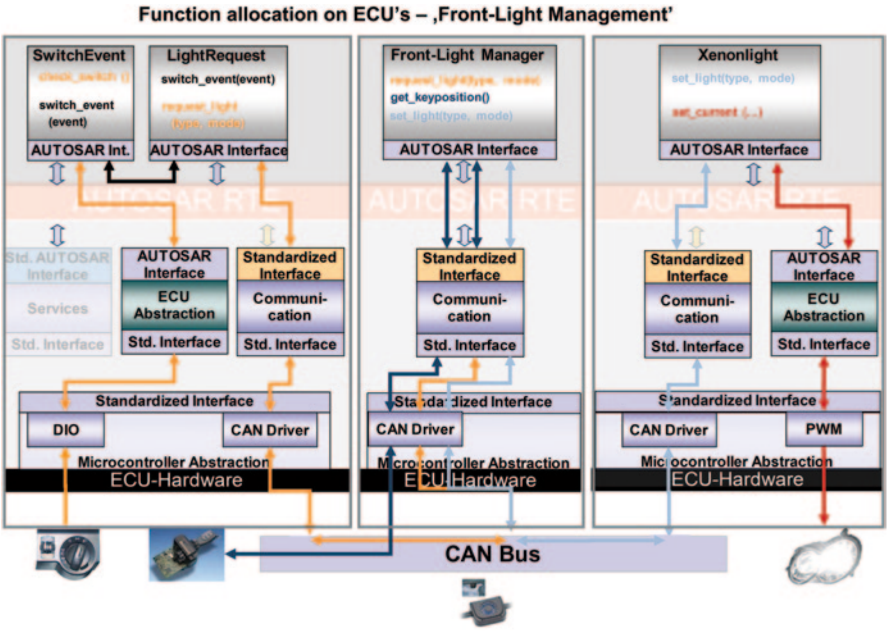

Fig. 5.3 Function allocation on electronic control units (ECUs)

all OEMs. For that reason, those parts of the system basis functions which are still company specific are allocated in the so-called *Complex Device Drivers*.

Figure 5.3 shows a solution for the control of headlights.

The interconnection of ECUs with AUTOSAR software architecture can be realized without any problem.

5.2 Time-Triggered Controller Area Network (TTCAN)—Applications

The following application example is a research educational prototype steer-by-wire and brake-by-wire system which is built on a basic software implementation of the time-triggered controller area network (TTCAN) protocol.

5.2.1 Software Implementation of TTCAN X-by-Wire

A current trend in the automotive industry is to replace certain mechanical components in vehicles with ultra-dependable fault-tolerant electronic systems, referred to as X-by-wire systems. Mechanical components such as drive belts, water pumps, hydraulic brakes and steering columns can be replaced with electronic systems.

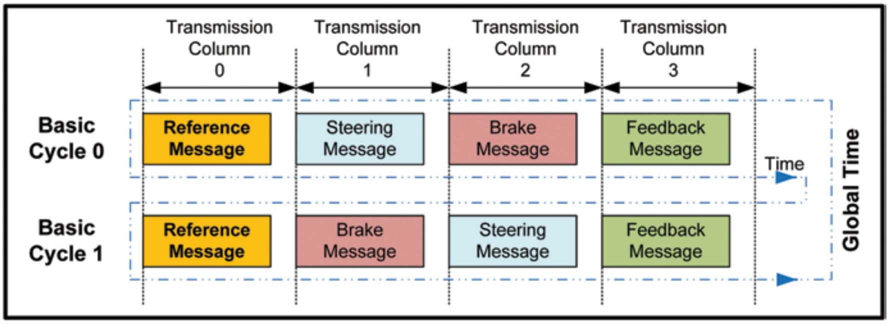

Fig. 5.4 Message cycle matrix

This initiative should result in a lighter, safer, more fuel-efficient and less expensive X-by-wire vehicle which exhibits additional functionality. In such vehicles, there are fewer environmentally unfriendly fluids to contend with, and the systems are self-diagnosing, reconfigurable and easily adapted across vehicle platforms. X-by-wire systems allow the tightest possible integration of distributed functionality within the vehicle, in contrast to the discrete, often disjoint, operation of conventional mechanical systems. The introduction of an X-by-wire vehicle infrastructure facilitates the implementation of many active safety improvements, based on advanced electronic systems; examples include autonomous cruise control, collision avoidance, automated parking assist and autonomous driving. The European SPARC Project is an example of an X-by-wire accident-avoiding vehicle with a Safety Decision Control System (SDCS). A necessary prerequisite to such highly integrated X-by-wire systems is a fault-tolerant communication infrastructure. The following section describes a prototype experimental brake-by-wire and steer-by-wire system based on TTCAN.

5.2.2 TTCAN Network Implementation

At the time this X-by-wire prototype was developed, there were no TTCAN protocol engines available in silicon. As a result, a system based on the Infineon C515C microcontroller and an application layer based on the TTCAN protocol with level 1 synchronization was implemented in software. Figure 5.4 illustrates the TTCAN message matrix used. The cycle matrix consists of two basic cycles. Each basic cycle commences with a reference message, which is followed by either a steering wheel position message or a brake pedal angle message, and terminates with a feedback message. The reference message is used to synchronise the network by resetting the cycle time in each network node. The reference message also contains the current basic cycle count, which is used to help to ensure that all nodes observe the correct schedule pattern. The TTCAN local clock is implemented using the microcontroller's on-chip timers and the TTCAN triggers are implemented using real-time interrupts generated by the overflow of these timers.

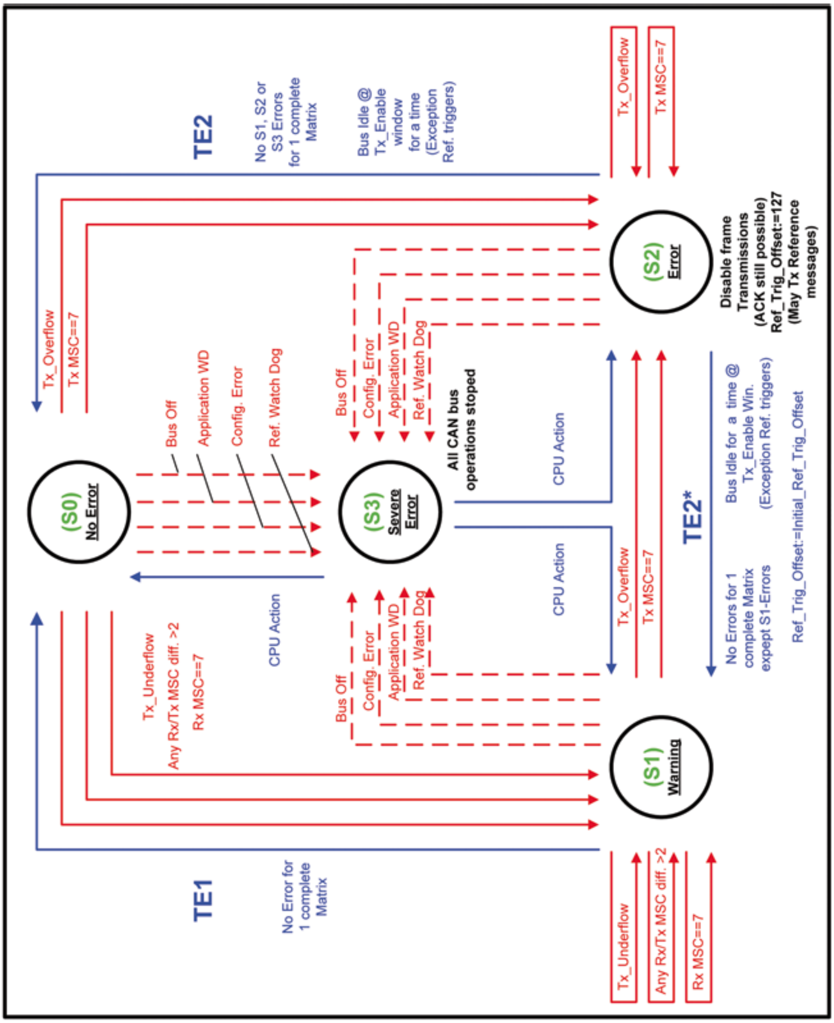
TE1
No Error for 1 complete Matrix
Tx_Underflow
Any Rx/Tx MSC diff. >2
Rx MSC==7
(S1) Warning
(S0) No Error
Tx_Underflow
Any Rx/Tx MSC diff. >2
Rx MSC==7
CPU Action
Bus Off
Application WD
Config. Error
Ref. Watch Dog
(S3) Severe Error
All CAN bus operations stoped
Bus Off
Config. Error
Application WD
Ref. Watch Dog
CPU Action
CPU Action
Tx_Overflow
Tx MSC==7
TE2*
No Errors for 1 complete Matrix expept S1-Errors
Bus Idle for a time @ Tx_Enable Win. (Exception Ref. triggers)
Ref_Trig_Offset:=Initial_Ref_Trig_Offset
(S2) Error
Disable frame Transmissions (ACK still possible) Ref_Trig_Offset:=127 (May Tx Reference messages)
TE2
No S1, S2 or S3 Errors for 1 complete Matrix
Bus Idle @ Tx_Enable window for a time (Exception Ref. triggers)
Tx_Overflow
Tx MSC==7

Fig. 5.5 Time-triggered controller area network (TTCAN) error state machine

Each network node monitors the transmission and reception of relevant messages; this and additional information are used as input to an error state machine. For example, if an expected message is not transmitted or received, a corresponding message status counter (MSC) is incremented. If any MSC reaches a predefined limit, an error is flagged and an appropriate action is taken. In this case, the node is reconfigured and attempts to rejoin the network; see Fig. 5.5 for an overview of the TTCAN error state machine.

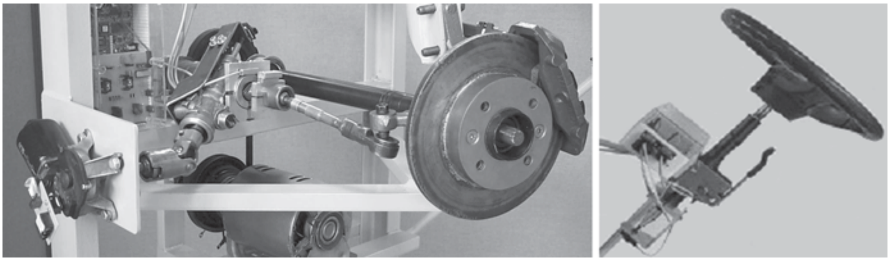

Fig. 5.6 Steer-by-wire system

5.2.3 Steering Implementation

In the prototype, the vehicle steering column was removed and replaced with a position sensor as seen in Fig. 5.6. A rotational absolute encoder is used to measure the angular position of the steering wheel. The sensor measures 128 positions per revolution and, therefore, has a resolution of 2.81 degrees. The sensor is read by the user interface microcontroller, and its position sent over the TTCAN network in the steering wheel position message. More accurate sensors with greater resolution were considered along with the possibility of gearing this sensor for greater resolution; however, for this concept demonstration model, such accuracy was not considered necessary. A real steer-by-wire system would probably require a sensor with a resolution in the order of 0.5 degrees, or better.

At the rack and pinion end (actuator end), a servo-controlled 12-V direct current (DC) motor is used to change the road wheel steering angle. A rotational position sensor connected to the rack and pinion drive provides feedback. Magnetic limit switches at the extremities of motion are used to detect the left and right maximum steering lock positions. Figure 5.6 shows the steering system used. The actuator ECU receives the requested wheel angle via the TTCAN network and rotates the wheels to the required position. In a production implementation, sensors, actuators and communication channels would most likely be replicated to provide redundant backup systems.

5.2.4 Brake Implementation

The system brake pedal uses a linear potentiometer to measure the extent to which the brake pedal is pressed as seen in Fig. 5.7. The voltage drop across the potentiometer is read by an 8-bit analog-to-digital converter (ADC) and its value transmitted over the network in the brake position TTCAN message.

The floating calliper brake unit is modified and incorporates a linear stepper motor to adjust the position of the brake pad, thus applying the braking force. In practice, a servo motor would be more appropriate. The actuator ECU receives the brake pedal position via the TTCAN network and adjusts the position of the brake pad accordingly.

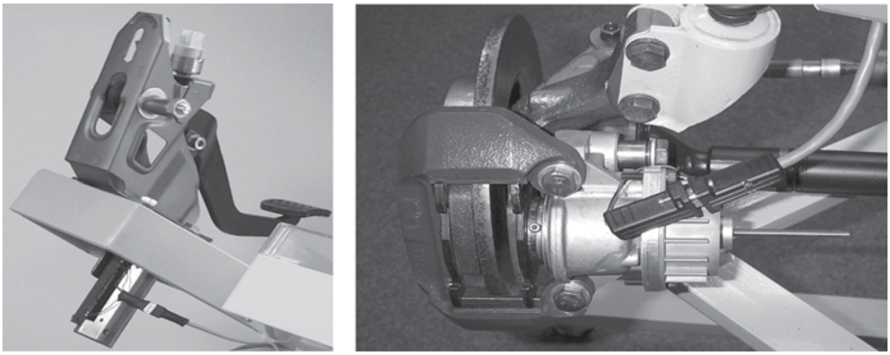

Fig. 5.7 Brake-by-wire system

5.2.5 Feedback Message

No physical force feedback was implemented in this prototype. Nevertheless, a feedback message is sent from the rack and pinion to the steering wheel, containing information relating to the steering angle. This message was included to demonstrate that a feedback message could be easily incorporated into TTCAN's network traffic.

5.2.6 Final System

The final prototype is illustrated in Fig. 5.8. The steer-by-wire and brake-by-wire systems were implemented on a single TTCAN network. The network was operated at 250 Kbaud with a Network Time Unit (NTU) of 1.2 µs. The time windows are 2 ms long. The reference and feedback messages take 0.236 ms each to be transmitted, and the steering and brake messages take 0.276 ms each to be transmitted. This results in a total network bandwidth usage, excluding error frames, of 12.8 %. It should be noted that in this configuration the network is only running at one-quarter of its maximum speed of 1 Mbaud.

5.3 CAN in Aircraft World

5.3.1 Why CAN?

A large amount of information crosses through an aircraft. Many systems coexist ranging from high critical avionics systems (displays, flaps command, engine fire detection, etc.) to cabin systems such as ventilation, water and galleys (kitchens).

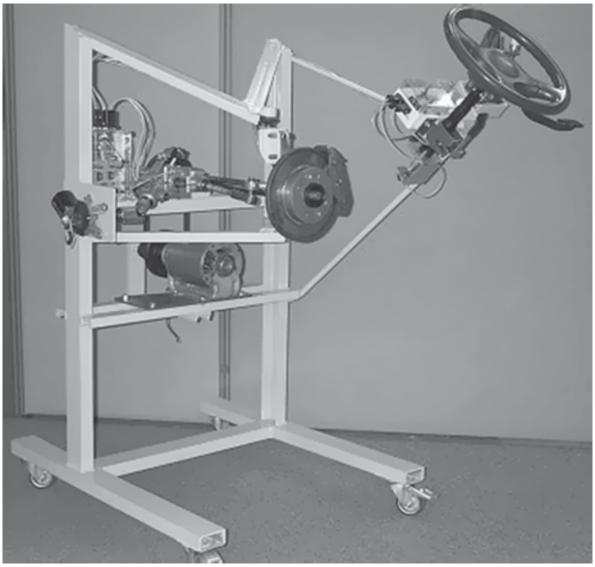

Fig. 5.8 Time-triggered controller area network (TTCAN)-based X-by-wire prototype

Networking is already an old story: aircrafts have been using ARINC 429 since 30 years. Why then a change towards a "non-avionics" network as CAN?

5.3.1.1 From ARINC 429 to CAN

First of all, a few words on the Aeronautical Radio Incorporated (ARINC) label:

The Airlines Electronic Engineering Committee (AEEC) is an international standards organization, comprising major airline operators and other airspace users. The AEEC establishes consensus-based, voluntary form, fit, function and interface standards that are published by ARINC and are known as ARINC Standards. ARINC Standards specify the air transport avionics equipment and systems.

ARINC 429 is very well defined and largely used and known communication system. The first specification was delivered in 1977.

The physical layer is robust to the aeroplane environment and is characterized by:

- Rreturn-to-zero (RZ) bipolar modulation and tri-state modulation consisting of "HI," "NULL" and "LO" states,
- Nnominal voltages values as described in Table 5.1 and
- Cables and nodes with 75 Ω impedance.

Nevertheless, main drawbacks have limited the application and increased wires:

- It has a low bit rate, with high-speed operation at 100 kbits/s and low-speed operation around 12 kbits/s.
- Labels (equivalent to CAN identifiers) are too strictly defined.

Table 5.1 ARNIC 429 emitter voltage values

	HI (V)	NULL (V)	LO (V)
Line A to line B	+10	0	−10
Line A to ground	+5	0	−5
Line B to ground	−5	0	+5

Table 5.2 ARINC 429 word structure

32	31	30	29 11	10	9	8 1
P	SSM		Data	SDI		Label

The label is used to determine the data type of the remainder of the word and, therefore, the method of data translation to use

P parity bit, *SSM* sign/status matrix, *SDI* source/destination identifier

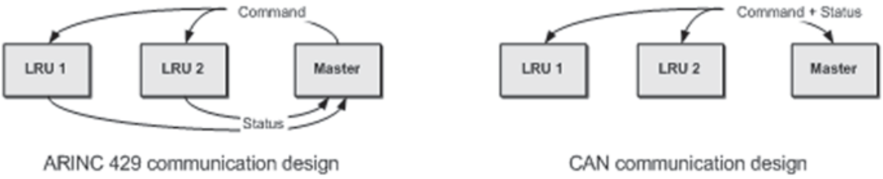

Fig. 5.9 ARINC 429 and controller area network (CAN) communication design

- Components are handled by the aeronautics industry.
- The ARINC 429 word is 32 bits with 20 bits maximum for data field as shown in Table 5.2.

Moreover, the communication happens through one transmitter/multiple receivers. It is highly reliable but increases number of wires (Fig. 5.9).

System designers and aircraft manufacturers therefore decided to apply an already worldwide established important standard for their increasing communication demands - that is why CAN is chosen.

The main advantage seen with CAN is that it is the automotive standard. It is not that airframers "copy" automotive ideas but component obsolescence is a very critical factor. The long life of aircrafts (30 years) but the small amount of units (around 1 per day gets out from Boeing and Airbus assembly lines) pushes us to follow a size market that gives quantities.

The "open" standard, the large number of tools offered and the price have contributed to CAN's success in aircrafts. CAN also offers good error detection and high electromagnetic immunity.

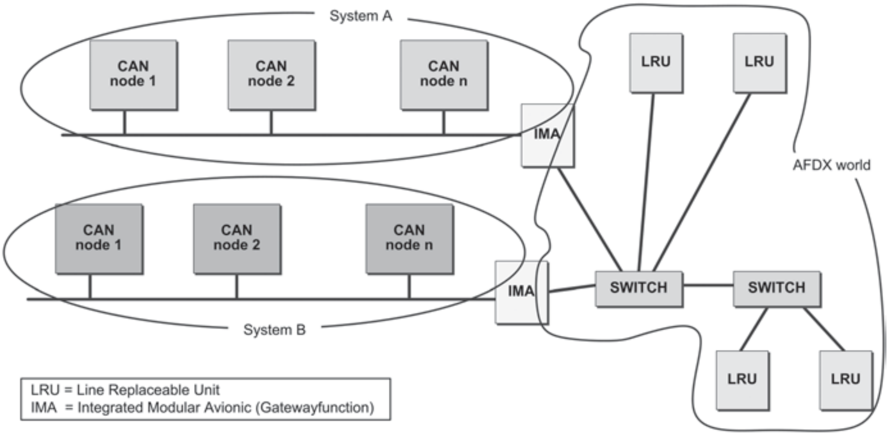

Fig. 5.10 A380 CAN (controller area network) in the global avionics network architecture

5.3.1.2 History and Future …

Two different worlds coexist in aviation: general aviation (GA) with small aircrafts and helicopters and airframers (Airbus and Boeing).

Airframers CAN started with cabin systems (ventilation, smoke detection and water/waste) on A318 and A340, developed by a unique supplier with low bit rate (83 kbps). It was so appealing that the use was largely extended on A380 within avionics high critical systems (power distribution, control panels, engine fire detection, door monitoring, etc.) leading to more than 500 CAN nodes and 75 busses per A380. A380 was also the starting point of the backbone avionics communication with AFDX (avionics full duplex Ethernet; switched Internet ARINC 664). The redundancies are not shown in Fig. 5.10.

The redundancies are not shown on this drawing.

General Aviation Uses CAN for backbone communication and major avionics buses. Therefore, it has to fulfil all the requirements of a flight safety network. National Aeronautics and Space Administration (NASA) has also used CAN for research program.

A specific application layer was developed: *CANaerospace*, the initial version created in 1998. The story does not stop there as the Boeing B787 Dreamliner also hosts a large number of CAN systems, and Airbus as well as Boeing have chosen CAN as a basis for subsystems communication. A CAN standard ARINC 825 is ready for all applications on board aircraft.

Why is an ARINC standard needed, one could ask. The following sections give the answer: aircrafts are specific….

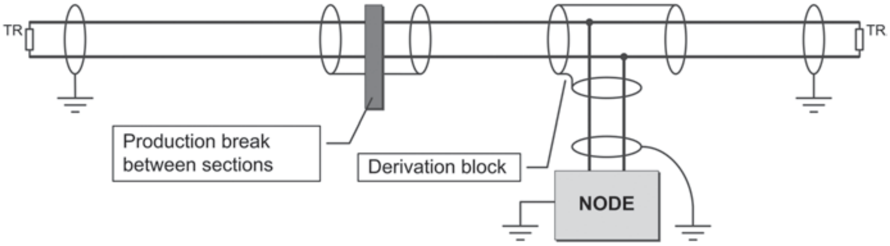

Fig. 5.11 Controller area network (CAN) wires layout

5.3.2 Aircraft-Specific Physical Layer Constraints

Compromise between wire length, number of equipment and baud rate is critical and aircraft systems reach the CAN physical layer limits.

5.3.2.1 EMC and Lightning Stress

Exposition zones are defined depending on the system installation area (pressurised and non-pressurised) and the EMI and lightning stress levels. Maximum bulk current injection (BCI) levels are 75 mA for A380, which are not that far from automotive constraints. Installation without shielding is possible. On A400M military aircraft, the BCI maximum level is 300 mA and installation with efficient and maintained shielding is necessary. Future aircrafts will apply GLAss-fiber REinforced aluminum (GLARE) for weight reduction purpose. This will increase the design constraints even more.

Lightning protection, designed with transorbs, is added in each equipment, which increases the node internal capacitance up to 300 pF.

5.3.2.2 Installation

The total length of A380 cables is 500 km. Routes are defined for cable and side segregation—M1 and M2 with no specific requirements and S1 and S2 for specific signals constraints: for example, few millivolts of audio signals.

The installation of a CAN network is a large part of the CAN adaptation to aircraft world. The constraint of maximum 30 cm stub length can not be met sue to cable routeing weight reduction constraints. This is especially true for cabin installation. Some systems are more than 150 m long, which is out of automotive use.

Some CAN wires go from the cockpit to the aircraft tail (Fig. 5.11) and go through different sections, which are produced in different European sites. The wire connection between sections is called a production break. At this point, the wire is untwisted and impedance is then modified.

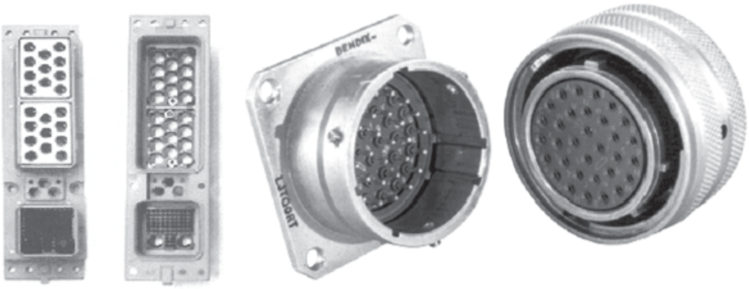

Fig. 5.12 ARINC 600 and EN3646 connector type

Signal quality estimation by simulation and mock-up are run in order to anticipate potential risks for system design.

At node termination, the connector type is a regular avionics one: ARINC 600 or circular EN3646 type as shown Fig. 5.12.

5.3.3 DAL, Safety, Certification

DAL (stands for Design Assurance Level from A to E. DO178B) classifies effects of a functional failure on aircraft safety. Safety analysis is run for all systems and loss of equipment/component (wire, etc.) is classified. Redundancies are built up to reach the system safety requirement. The loss effects classification is as follows:

- E—no safety effect,
- D—minor,
- C—major,
- B—hazardous and
- A—catastrophic.

A loss of a CAN network is no more than major. In specific high critical networks, we have up to three CAN wires and equipment redundancies.

5.3.3.1 Certification

The US Federal Aviation Administration (FAA) and the European Aviation Safety Agency (EASA) are two independent administrations that allow an aircraft to transport passengers. Numerous CRIs (Certification Review Items) will assume that CAN is related to CRI-F40 for A380 and the objectives are:

1. To ensure that the bus perform its intended function under the most demanding conditions and
2. To evaluate effect of abnormal behaviour and ensure the safety consequence.

The following issues are documented for all systems:

- Safety study,
- Data integrity,
- Performance,
- Design assurance,
- EMC,
- System configuration management and
- Continued airworthiness.

For more information, contact the FAA or EASA. Another CRI is related to CAN with the CRI-F09 for critical components; the CAN controller enters this category.

5.3.4 Example: Smoke Detectors Interfaced by a Safety-Critical Aircraft-Based CAN-Bus Network

5.3.4.1 Abstract

Classic architectures of aircraft systems contain equipment using interfaces with digital, analogue or discrete signals. The electrical network to interface the equipment varies between the applications. Some equipment require a dedicated power supply and provides information on an analogue current loop, while others use proprietary digital busses or discrete input/outputs (I/Os) for information exchange.

Initially, CAN was developed for use in the automotive industry, but is nowadays being used in an increasing number of applications. One of these areas is aviation, where in the past 5 years CAN has grown from being an exotic newcomer to an established and widely accepted solution. Within the Fire Protection System on an Airbus, smoke detectors are installed in various areas overall in the pressurised zones of the aircraft like lavatories, equipment bays and cargo compartments. As the CAN bus defines only layers 1 and 2 of the Open Systems Interconnection (OSI) communication model, additional higher layer features are necessary to achieve the level of operational assurance required for a safety-critical application, namely fire protection on an aircraft.

This example is particularly focused on the development of a safety-critical CAN bus network with strict configuration control of smoke detectors in the scope of an aircraft application. In 2003, international airworthiness authorities approved the application in the frame of the Airbus A318 Type certification.

5.3.4.2 Introduction

The objective of the new smoke detection system was to replace the proprietary current modulated supply and communication loop with an open, non-proprietary bus standard. The overall system reliability and performance were aimed to match or

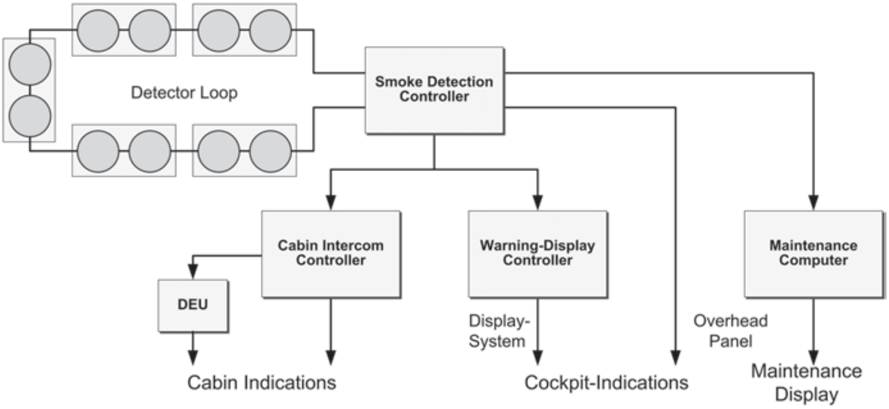

Fig. 5.13 Smoke detection system using proprietary detector supply and communication loop

surpass the existing architecture while keeping development and purchasing costs at a comparative level.

The latter was feasible by reusing the existing smoke detector core and fitting it with an altered communication and power interface (see Figs. 5.13 and 5.14).

The communication medium had to meet a number of requirements for eligibility in a safety-critical application:

- Advanced data integrity and error detection features,
- Deterministic behaviour,
- Operability in challenging EMC environments and
- High degree of flexibility in choice of network size and topology.

Considering the 30-year design life of a modern passenger aircraft, the long-term availability of electronic components was scrutinized in order to minimize the risk of equipment redesign resulting from component obsolescence throughout the life cycle of the aircraft.

The CAN bus was deemed the most suitable communication medium capable of fulfilling the above requirements.

5.3.4.3 Protocol

The CAN protocol, as defined in ISO 11898 [ISO11898], covers layers 1 and 2 of the OSI communication model. The remaining layers, up to layer 7, have to be managed by additional services up to the application. Various standardised higher layer protocols such as CANopen are available and widely used in industrial applications. Instead of selecting a generic high layer protocol, a specific to-type application layer protocol was developed and documented in a System Interface Document [SCHMID] in order to ease compliance with RTCA/DO-178 [DO178B] guidelines.

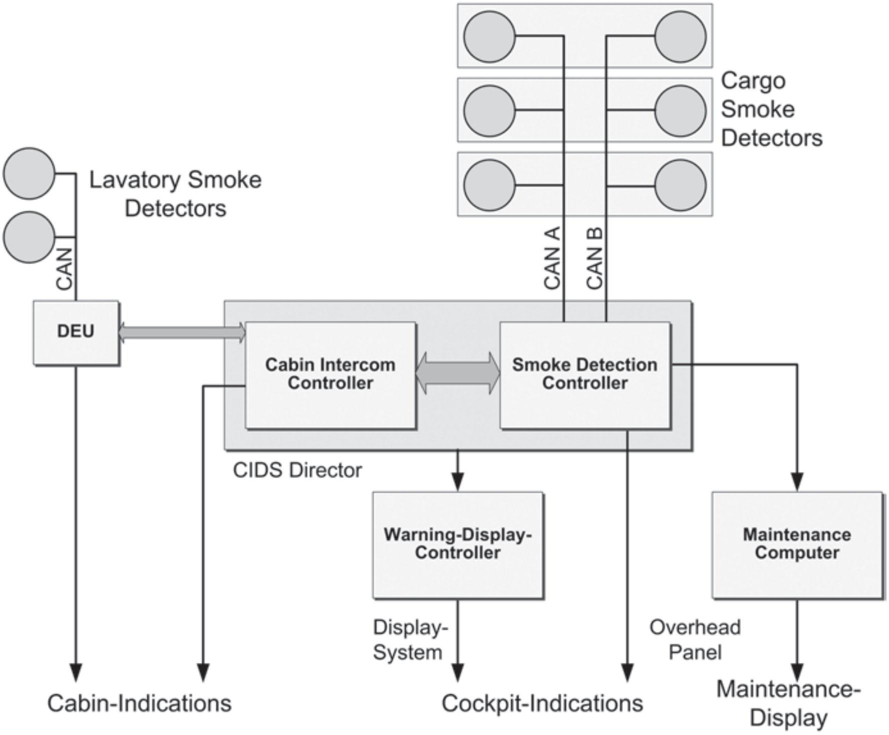

Fig. 5.14 Smoke detection system using an open standard controller area network (CAN) bus-to-interface detector

Analysis of the communication needs to result in the following protocol requirements:

- Every individual smoke detector on the network must be uniquely identifiable,
- Messages generated by a smoke detector must contain information about its identity and
- The detector must support a master–slave communication model.

CAN Identifier The 29-bit extended identifier is utilized and partitioned into the subfields as shown in Fig. 5.15.

Message Type The purpose of the Message Type is to categorize messages according to their overall relative priority and indicate whether the Module ID contains a transmitter or receiver address. Two classes of Message Type, Process Data Object (PDO) and Service Data Object (SDO) are instantiated either as Transmit or as Receive objects, T_PDO and R_PDO as well as T_SDO and R_SDO, respectively. A Transmit Data Object (T_xDO) denotes that Module ID contains the network address of the transmitter, whereas a Receive Data Object (R_xDO) contains the network address of the intended receiver in the Module ID field.

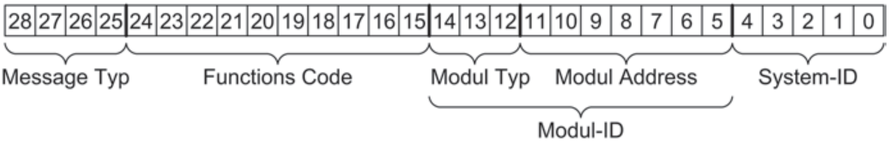

Fig. 5.15 Controller area network (CAN) identifier

Function Code (bits 24…15) Every application function is designated a unique Function Code within its respective Message Type. In addition to describing the next level of arbitration priority, the Function Code is used to transport logical data without the use of the actual CAN data fields. In this case, the Data Length Code (DLC) is 0, enabling efficient use of data bandwidth, particularly for R_PDOs and R_SDOs which contain mostly status requests directed at smoke detectors and do not carry any further information than the request itself.

Module ID (bits 14…5) The Module ID field contains the unique network identification of the CAN node. This may also be a broadcast identification when a message is directed at several nodes simultaneously. Two subfields Module Type and Module Address split the Module ID into equipment classes and their individual addresses. The entire Module Address space may be reused for every Module Type on the network.

System ID (bits 4…0) The System ID is used to tag the CAN identifier with a unique system identification code. All smoke detectors and other fire protection components are assigned a fixed value.

Data Frames A Data Frame is generated by a transmitter to transfer application data to one, or in the case of a broadcast, several receivers. Within the Data Frame, the Data Field consisting of 1–8 bytes carries the application data. A Data Frame may contain an empty Data Field (DLC = 0). In this case, data are carried through the Function Code alone.

A smoke detector's 8-byte status Data Field is as defined in Table 5.3 with the meaning of the data bits in Table 5.4.

5.3.4.4 Network Management

It is of utmost importance that the system configuration and availability of resources (smoke detectors) are known to the network master. Lack of configuration control through the network master device would jeopardize safety and disqualify the system. From a safety assessment point of view, the worst-case condition is an undetected configuration error leading to an incorrect compartment designation in case of fire; an alarm reported in the aircraft's forward cargo compartment while the real fire occurrence is in the aft cargo compartment and vice versa. Such a case is

Table 5.3 Smoke detector data field

Data byte	MSB	LSB	Description	Format
1	7	5	Spare/not used	-
	4	4	Detector Warning	Discrete
	3	3	Prefault threshold exceeded	Discrete
	2	2	Detector standby	Discrete
	1	1	Detector alarm	Discrete
	0	0	Detector failure	Discrete
2	7	0	Trouble shooting data	Binary
3	7	2	Spare/not used	-
	1	0	MSB contamination level	Binary
4	7	0	LSB contamination level	Binary
5	7	2	Spare/not used	-
	1	0	MSB smoke level	Binary
6	7	0	LSB smoke level	Binary
7	7	2	Spare/not used	-
	1	0	MSB temperature	Binary
8	7	0	LSB temperature	Binary

Table 5.4 Meaning of the status bits

Designation	bit	Meaning
Failure	0	The smoke detector is no longer able to detect smoke or to communicate this information in a reliable manner
Alarm	1	Smoke is detected and confirmed
Standby	2	The smoke detector is able to detect smoke and communicate this information in a reliable manner
Prefault	3	The smoke detector optical cell contamination level has exceeded the internal threshold for triggering a corresponding maintenance message
Warning	4	The CAN TX error counter has exceeded 96

classified as catastrophic. A catastrophic event is defined as an occurrence leading to total loss of the aircraft and occupants and must be ruled out with a defined level of probability of failure $< 1 \times 10^{-9}$. Therefore, various network management mechanisms are necessary to ensure proper system configuration during initialization and normal operation.

5.3.4.5 Power-Up Configuration Control

The normal expected configuration of smoke detectors is fixed in a lookup table within the network master's operational software. At power up or system initialization, the current configuration is compared with the expected through a mechanism called Configuration Check Request. During the Configuration Check Request process, the network master broadcasts the Configuration Check Request as an R_PDO

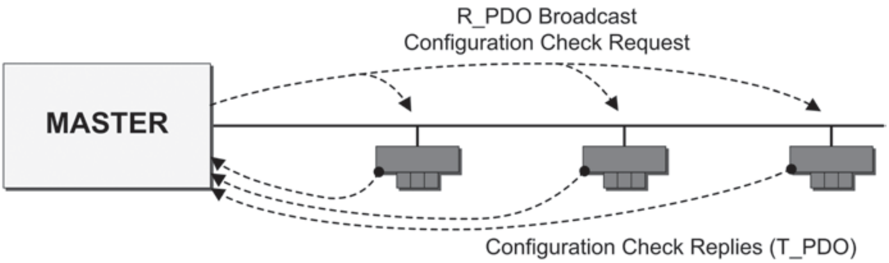
R_PDO Broadcast
Configuration Check Request
MASTER
Configuration Check Replies (T_PDO)

Fig. 5.16 Normal configuration check request/reply process

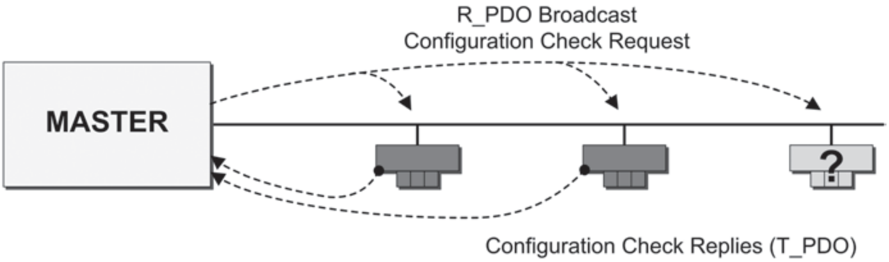
R_PDO Broadcast
Configuration Check Request
MASTER
?
Configuration Check Replies (T_PDO)

Fig. 5.17 Failure of expected smoke detector to reply

with the broadcast Module ID to all smoke detectors. These in turn reply with T_PDOs containing their individual Module Address, enabling the network master to make a comparison of the received replies with the expected replies and, thereby, detect the following failure cases (Figs. 5.16, 5.17 and 5.18):

- The network master is incorrectly configured for the intended installation,
- An expected smoke detector has not replied (missing smoke detector on network) and
- An unexpected smoke detector has replied (excessive smoke detector on network).

Thus, comparison of the configuration present on the network with the expected configuration is a prerequisite for determined network behaviour.

5.3.4.6 Normal Polling Operation

The CAN bus is operated in the master–slave mode (Fig. 5.19). The network master cyclically acquires the status of each smoke detector by an explicitly addressed request frame. Not to be confused with CAN remote-request frames, the request message is a regular data frame of type R_PDO containing the individual Module ID of the subject smoke detector and is replied to by a T_PDO data frame containing the Module ID of the replying transmitter.

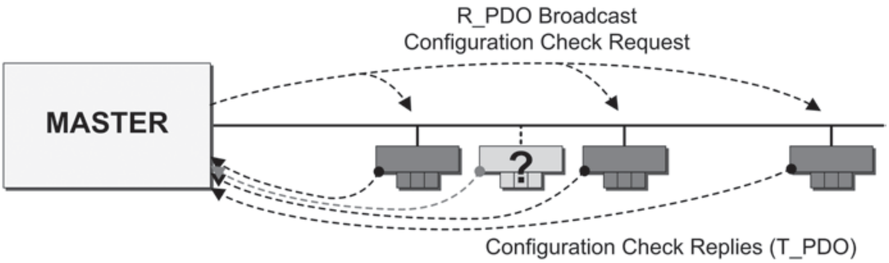
R_PDO Broadcast
Configuration Check Request
MASTER
?
Configuration Check Replies (T_PDO)

Fig. 5.18 Unexpected smoke detector reply

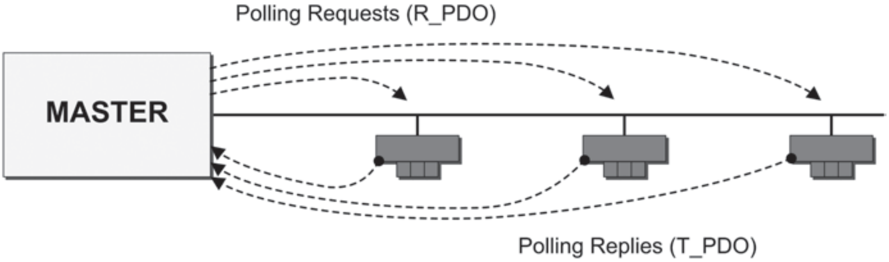

Fig. 5.19 Normal polling operation

Each polling request is monitored by a timeout in which the reply is expected. The polling cycle is repeated every 2 s.

5.3.4.7 Failure Detection/Reconfiguration

The response time of the smoke detector is 60 ms, including internal processing time and retry mechanisms inherent to CAN. A reply is considered timed out by the network master when not received prior to the following polling cycle, 2 s later. An outstanding reply increments a counter C in the network master. The reception of a normal polling reply while the counter is $1 \leq C < 5$ leads to a reset of the counter to 0 and the smoke detector is restored to normal operation. Once the counter reaches 5 outstanding replies (10 s), the smoke detector is declared inoperable and is no longer polled, thereby resulting in a reconfiguration of the system. System determinism is ensured through the request–reply time window and the polling cycle as shown in Fig. 5.20.

In summary, the polling process abides by the following rules:

- Only expected smoke detectors are polled,
- A smoke detector determined missing during power up is not polled,
- A smoke detector is no longer polled following five consecutive timeouts and
- A smoke detector is no longer polled when declared failed.

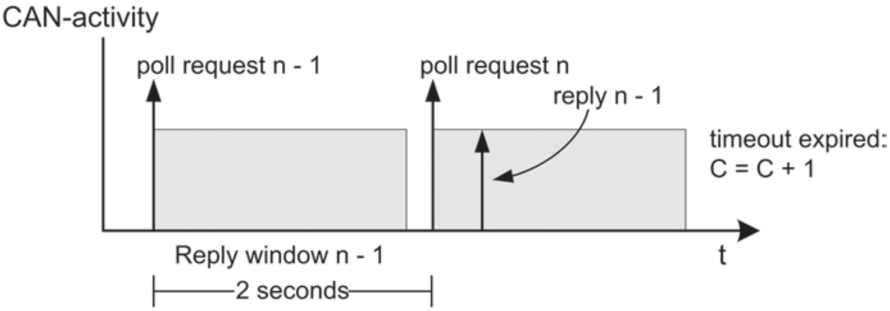

Fig. 5.20 Timeout expired

5.3.4.8 Smoke Detector Monitoring

In addition to the network-based configuration and time monitoring, the smoke detector is monitored for proper functional behaviour by the network master.

Normally, the smoke detector is in the standby condition (bit 2 on data byte 1 is TRUE). In case of alarm, the standby bit becomes false while the alarm condition (bit 1 on data byte 1) is TRUE. These conditions are by definition mutually exclusive and are therefore monitored for proper behaviour. If two consecutive polling replies are received with neither the standby nor the alarm bit set to TRUE, or both bits set to TRUE, the smoke detector is declared failed and is no longer polled. In Boolean terms:

$$Failed = (alarm * s\tan dby) + (\overline{alarm} * \overline{s\tan dby}). \tag{5.1}$$

5.3.4.9 Network Topology

The smoke detectors are connected with the network in a linear bus topology with stubs departing from a central bus line. Bus termination is accomplished through resistors implemented within the network master at one end of the network and the last smoke detector at the other end (Fig. 5.21). Each item of equipment is qualified to operate on a CAN bus of length 150 m, with 32 nodes connected through 2-m-long stubs to the main bus line.

Depending on the aircraft compartment being monitored, either a single- or dual-redundant bus line is incorporated depending on the reliability requirements and whether the compartment is accessible or not during flight. The dual-redundant architecture implies two smoke detectors at each location within a compartment. This is the case for the cargo compartments in the lower deck of the aircraft. Each lavatory, on the other hand, is fitted with a single smoke detector.

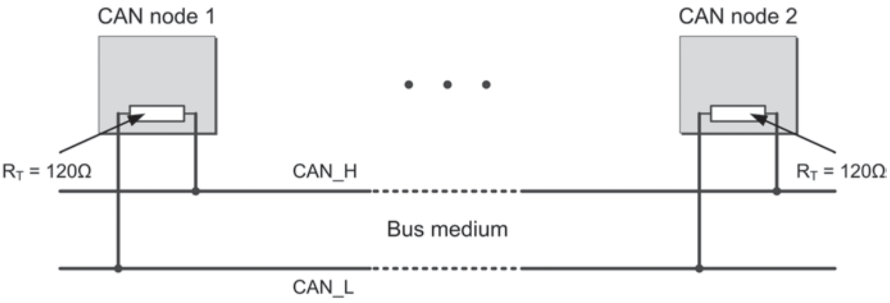

Fig. 5.21 Network topology

5.3.4.10 Development Process

The safety philosophy in aviation defines quantitative safety objectives and assigns acceptable probabilities. The overall probability for a failure with catastrophic consequences must be extremely improbable. This must be demonstrated to the airworthiness authorities for certification. The demonstration is endorsed through a complete, detailed and documented safety analysis, which is one of the integral parts of the software development process.

Guidelines for development of aviation software in the USA are defined in the DO-178B. Since its production by the RTCA, the DO-178B has become a de facto standard. The FAA's Advisory Circular AC20-115B established DO-178B as the accepted means of certifying all new aviation software.

DO178-B is primarily concerned with development processes. As a result, certification to DO178-B requires delivery of multiple supporting documents and records. The quantity of items needed for DO178-B certification, and the amount of information that they must contain, is determined by the level of certification being sought.

The higher the consequences of a potential failure of the software (catastrophic, hazardous-severe, major, minor, or no-effect) are, the higher is the DO178-B certification level. The levels are from A for the highest certification level through B, C and D to E.

This aviation-specific development process had to be followed on an equipment and on a system level.

5.3.4.11 Conclusion

Through clever system design and network management, a CAN bus-based safety-critical smoke-detection system with deterministic behaviour, capable of fulfilling the safety and reliability requirements, was developed and approved by airworthiness authorities. The robustness and reliability of CAN in this airborne application are being closely monitored, with some 1.45×10^7 accumulated flight hours (including multiple equipment factor) having been accumulated in the period between mid-2003 and February 2006.

5.4 The Geniax System Decentralised Heating Pumps

"Geniax" is a real technical revolution in the field of heating technology. It is based on several miniature pumps at the heating surfaces or in the heating circuits instead of using thermostatic regulating valves. The conventional "supply-oriented heating" with one central heating pump is replaced in this way by "demand-oriented heating"—pumping only takes place when heat is needed (see Fig. 5.22, which demonstrates the basic principle of Geniax).

Also new is a central control intelligence for the whole heating system. It maintains the heating system in a hydraulically optimal state and generally improves precision, speed and energy efficiency. Fields of application include new buildings and upgrades of older buildings. The system can be installed in both single- and multi-family houses as well as in commercial properties such as office buildings. The central advantage—besides improved hydraulics and comfort—is the significant reduction in heating energy consumption by an average of 20%.

A further decisive component of the decentralised pump system—besides the miniature pumps and their pump electronics—is a central management unit with an interface to the heat generator: the Geniax server. It is responsible for the coordination of heating needs in the individual rooms and, using the present specifications from the room user interfaces, the management of all components in the entire heating system. The Geniax server's control signals to the pump electronics are used to variably control the pump speed and therefore also the mass flow of the pumps and the heating performance in a needs-based fashion. Beyond this, the server controls the displays of the room user interfaces, monitors all connected components, collects data for diagnostic purposes and controls the heat generator via the 0–10 V interface (see Fig. 5.23).

As already indicated above, the Geniax system is a master–slave system in which all slaves are dependent on the communication with the Geniax server. With this concept, Wilo SE also takes an unusual path for a traditional individual room temperature regulation system, where all the rooms to be regulated typically work on their own.

In the Geniax system, the server works with all the system's information so that individual pumps, if need be, can even be handled according to preference or can perform anticipatory work as a result of learning processes.

As is the case for any master–slave system, the Geniax system requires a suitable communication medium. Since a Geniax pump has a maximum power consumption of approximately 3.5 W, there is no need for a permanent battery supply for the components integrated in the system. This led to the decision to use a cabled system at the start of the newly developed system.

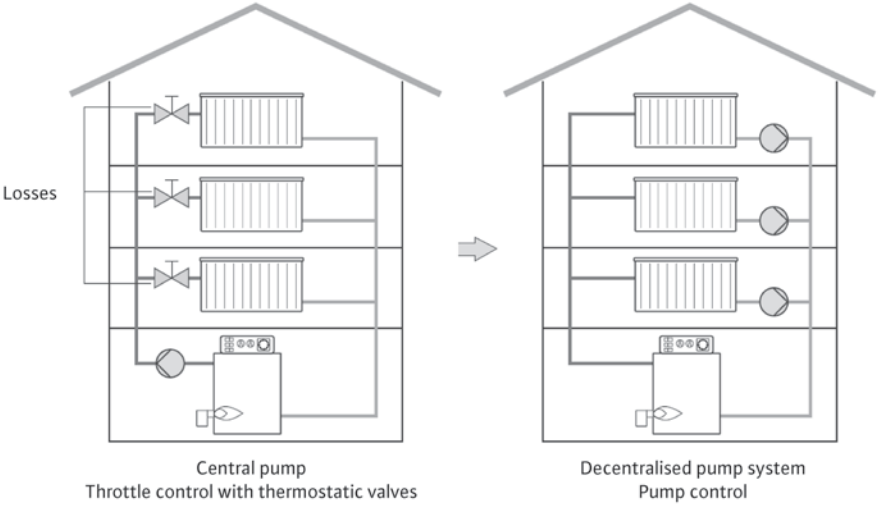
Losses
Central pump
Throttle control with thermostatic valves
Decentralised pump system
Pump control

Fig. 5.22 Basic principle—throttle control compared to pump control

From the many established bus systems, a preselection was made and a benchmark was carried out on this selection.

The selection was to be made among the following bus systems: EIB, LON, BACnet and CAN.

The following reasons finally led to the decision in favour of CAN as the medium for the Geniax system.

- Comparatively low costs per communication node of about 1 €,
- EIB and LON are partially subject to license costs, leading to node costs of more than 10 €,
- Long-term availability of standardized transceivers from the automotive industry,
- The energy-saving functions of the CAN transceiver to be used for further energy saving at every individual node and
- The possibility to develop an exact proprietary protocol to meet the needs of the Geniax system.

Besides the relatively high costs for the hardware design of each communication participant, EIB, BACnet and LON already have a fixed protocol format which cannot be used directly with the Geniax system.

Not the least for this reason, a bus system was selected for the Geniax system which is adaptable when it comes to speed and protocol format and is also becoming better established in other areas of house and building automation.

After the decision was made to use the CAN bus, the transfer speed had to be determined. This would implicitly determine the maximum network expansion. By using low baud rates, a huge expansion of the topology of more than 2,000 m is possible, theoretically. This expansion exceeds the expected network expansion in

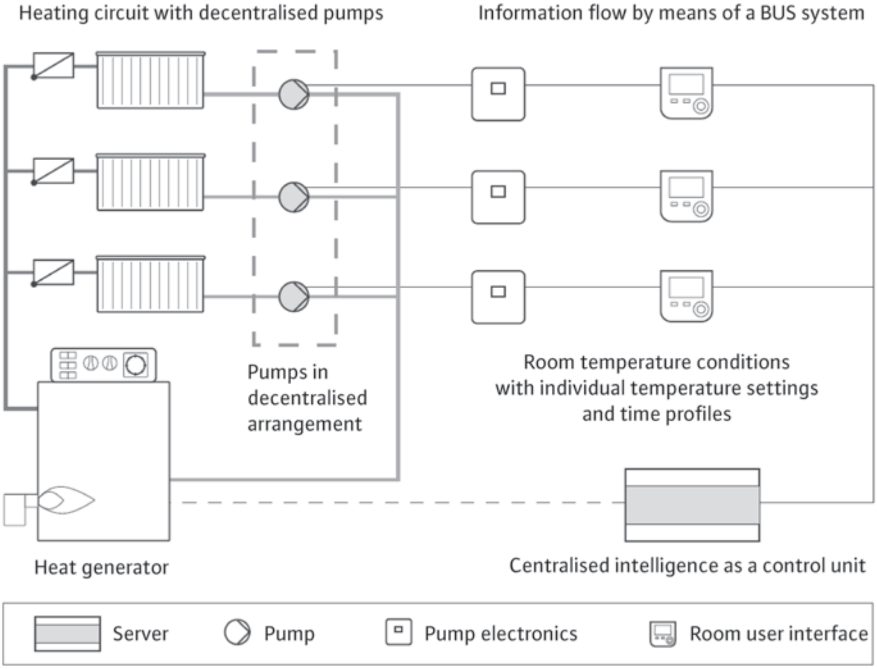
Heating circuit with decentralised pumps
Information flow by means of a BUS system
Pumps in decentralised arrangement
Room temperature conditions with individual temperature settings and time profiles
Heat generator
Centralised intelligence as a control unit
Server
Pump
Pump electronics
Room user interface

Fig. 5.23 Schematic representation of the main system components

a single- or multi-family house. Even for the installation in a functional building, such a greatly expanded system is not supportable due to reliability.

In addition, the greatly extended lines will inevitably lead to voltage drops and the associated earth offset, causing problems for reliable operation and communication between the participants linked by the bus. Beyond this, building construction factors place demands on the installation, which deviate from the linear bus topology, is specified by the CAN standard.

For this reason, the Geniax system is supplemented by the so-called bus coupler. With the help of the bus coupler, it is possible to segment a greatly extended system into logical subsegments. Every subsegment formed with a bus coupler is galvanically isolated from the upstream subsegment and has its own power supply. Due to the galvanic isolation of individual CAN segments, it is possible to implement nearly any topology without violating the CAN principles. This allows even a convoluted star topology to be done without complicated calculations of termination resistance.

This means that clever planning of bus couplers can lead to a system with higher overall availability of the existing heating surfaces than would be the case for traditional operation of a single pump in the basement.

Taking such a system into consideration, if, for example, the power supply in a subsegment fails or there is a short circuit in a particular subsegment, the subseg-

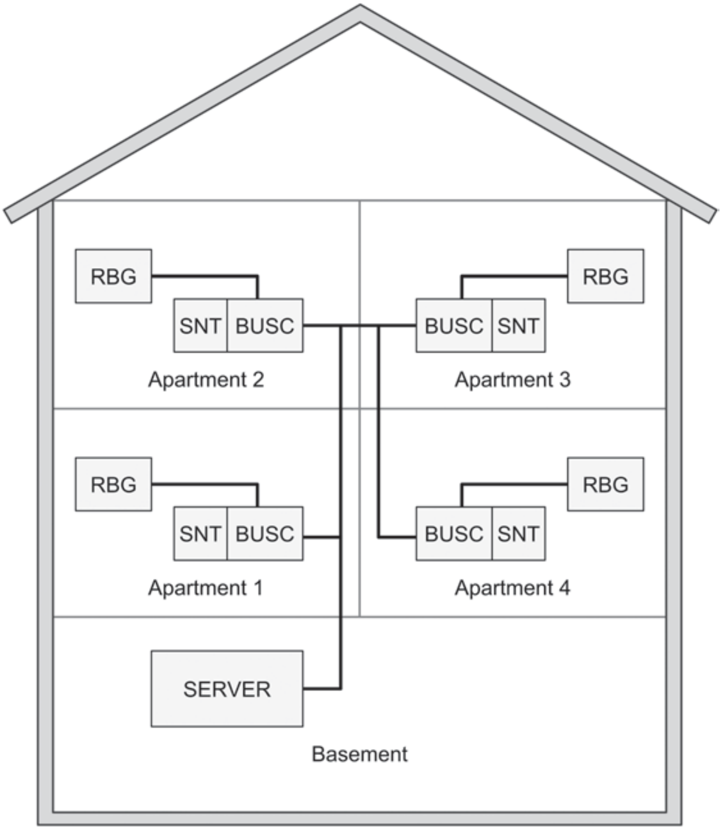

Fig. 5.24 System overview of the controller area network (CAN) topology with bus couplers

ments located upstream of this bus coupler will continue to work without a problem (see Fig. 5.24). Naturally, the Geniax server recognises that there is a problem in the system and passes this information on accordingly.

Besides the physical advantages of using the bus coupler, its protocol-level operation produces system-stabilising effects.

Since every message from every bus coupler must first be completely received before it is forwarded to upstream or downstream subsegments, CAN "Error Frames", for example, in one subsegment are automatically filtered out and, therefore, will not be propagated in the entire system.

To summarise, the following can be stated:

In order to guarantee CAN communication in extended networks with high expected load currents in particular, dividing up the network into smaller segments that are galvanically isolated from each other is a recommendable measure. It is important for the individual CAN segments that all communication nodes within a subsegment are connected to the same reference GND.