TEKA Kom. Mot. Energ. Roln. – OL PAN, 2010, 10, 432–439

SAFE COMMUNICATION AMONG VEHICLE SUB-ASSEMBLIES ON THE BASIS OF THE EMBEDDED FUNCTIONS OF CAN PROTOCOL

Andrzej Sumorek

Ph.D. Eng., Department of Computer and Electrical Engineering,
Lublin University of Technology, Poland

Summary. CAN networks are the most popular networks integrating the electromechanical nodes in the scope of automotive and industrial applications. Their continuous popularity commenced in the nineties of the 20th century as a result of simplicity of communication and high reliability layer. The purpose of the present study is to present the factors affecting the security of communication directly associated with the protocol specification as well as additional factors increasing the safety constituting the bases for various implementation of hardware and application layers. The factors affecting the error handling minimization applied in competitive solutions and suitable for implementation in CAN protocol are also presented.

Key words: CAN protocol, errors detection, networks

INTRODUCTION

The manufacturers of vehicles and equipment improve the competitiveness of their products by means of three principal methods. The first method consists in the increase of product functionality as a result of the introduction of its new functions or through the improvement of use comfort. Another method consists in the price reduction as a result of reduced material demands. The third method is associated with the reduction of service costs resulting from the failures detection in an earlier or facilitated manner [14].

The achievement of aforesaid goals is possible thanks to the introduction of communication buses into the vehicles and machines instead of conventional cabling. The substitution of conventional cabling associated with the electric and electromechanical elements by the buses with digital protocols causes automatic reduction of electric wiring, reduction of vehicle weight, accelerated communication and increased error tolerance due to the application of digital signals [14].

The complete reliability of a motor vehicle can be potentially achieved by means of reliable mechanical, hydraulic, pneumatic, electric, electronic and electromechanical elements combined together. However the complete reliability is not ensured in this case, because these elements must support each other. Such support is warranted by the buses and communication protocols. Several communication protocols dedicated to automotive applications have been created in recent years; they are characterized with high error tolerance and enable the operation in real time (e. g. FlexRay,

Byteflight). However the manufacturers are still unable to make the decision on the introduction of the control system completely based upon electrical network (X-by-wire). There are the solutions in the scope electronic control associated with the steering wheel (steer-by-wire) or with braking system (brake-by-wire) [2]. However there are no vehicles with completely eliminated mechanical, hydraulic or pneumatic connections and replaced by electric components.

It should be emphasized that the solution which has been created already in 1983 still exists among the communication protocols and buses. CAN (Controller Area Network) protocol has been introduced in the vehicles in the year 1991. CAN has been not completely eliminated by the newer, faster and more secure protocols yet. It has been developed in many specifications and standards in the scope of hardware and application layers as well as constitutes one of the basic systems in the scope of vehicles diagnostics and one of the most popular communication protocols in the scope of industrial facilities. This solution is popular owing to the simple structure of communication frame, flexible network configuration (easy extension and modification), efficient arbitration mechanism, relatively high data transfer and parallel application of the error handling mechanisms. CAN protocol functions which are essential for data transfer correctness and for the correct operation of the whole communication network of the vehicle or facility will be presented in the further part of the present study.

CAN PROTOCOL INTERNAL ERROR SUPPORT

Irrespective of the differences in the scope of communication standard in hardware and application layers or in the network capacity, the communication within Controller Area Network in version 2.0 shall be based upon one of two specifications of the protocol (A and B). Depending on the specification, the layers of protocol stacks are responsible for the basic error handling, the location of said layers are similar to the location in ISO/OSI layer model but their names are different. In the scope of A specification, these tasks are assigned to the Transfer Layer. In B specification, similar tasks are performed by Data Link Layer (DLL) with its inner layers i.e. Logical Link Control (LLC) and Medium Access Control (MAC) [1, 4].

The tasks associated with secure communication imposed onto the protocol to Transfer Layer in A specification directly are: error detection, error signalling, message validation, acknowledgment and fault confinement. The Logical Link Control and Medium Access Control layers from B specification extend the list of tasks by: overload notification, recovery management and frame stuffing/destuffing.

In order to enable the description of the mechanisms of error handling integrated with CAN protocol, the presentation of the construction of the frame (message) and of the basic communication principles. CAN bus it the bus functioning in multicast mode. Its nodes operate in multimaster mode. The frames/messages without the relevant address of sender and recipient are transmitted by each node. There is no direct connection between two nodes. The access to the bus is based upon CSMA/CD method. [2]. The transmission is commenced by the node if the bus is not occupied during the period required to transmit three bits. In case of the transmission commenced by two nodes simultaneously, the arbitration is carried out in course of bits transmission from the arbitration field in communication frame (Fig. 1, 2). The device with the arbitration field containing "more dominant bits" is maintained on the bus. In other words, the arbitration field is characterized by the lower digital value equivalent to the higher message priority [2, 10]. The length of the arbitration field in protocol version 2.0A is equal to 11 bits.

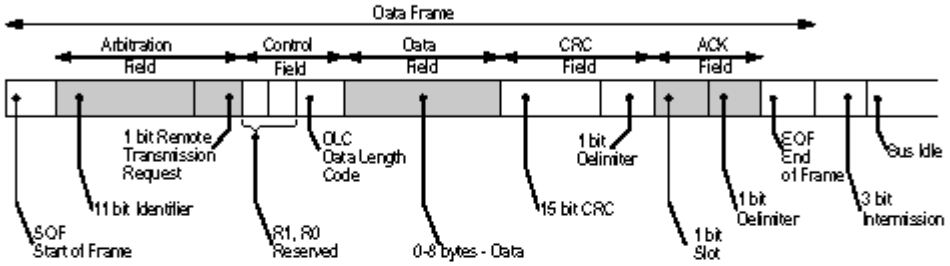

Fig. 1. Construction of data frame – CAN protocol ver.2.0A (standard format)

In the protocol version 2.0B, the arbitration field has been extended to 29 bits. The purpose of the revisions introduced into the specification was to ensure the compatibility with serial communication protocols in automotive applications in United States. The network configuration flexibility has been increased in an indirect manner as a result of greater quantity of identifiers. The opportunity of the cooperation of the nodes conforming with 2.0A and 2.0B specification has been ensured as a result of the introduction of an additional extension bit into the frame of 2.0B version IDE (IDentifier Extension) (Fig. 2). IDE bit makes it possible to identify the specifications of the frames: format standard - dominant bit, standard extended – recessive bit.

Also the operation of the nodes conforming with the both specifications is possible in the same network. The priority of the standard frame will be always higher. The purpose of an additional SSR bit (Substitute Remote Request) is to substitute standard RTR bit (Fig. 2). Furthermore no disturbances into the network with the nodes conforming with standard and extended specifications are not introduced by said bit. SSR bit is always transmitted as a recessive bit control the choice between a standard data frame and extended data frame. The priority of extended data frame will be always higher.

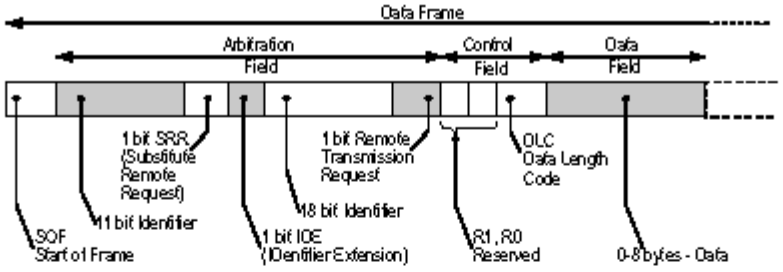

Fig. 2. Changes in the construction of data frame – CAN protocol ver.2.0B (extended format)

As mentioned above, the bus nodes are operated in multimaster mode. The frames generated by the nodes result from an algorithm realized by the node software in application layer. The messages are not addressed to the specific receiving node. All nodes are provided with the access to all messages on the bus. The use of the specific data frame depends on the filters of data in the scope of software or hardware.

However the common accessibility of all messages on the bus for all nodes seems to be the source of potential errors. The stable operation of the whole network could be easily disturbed by

only one incorrectly operating node transmitting the high priority messages. Therefore the authors of protocol introduced the sets of functions improving its error tolerance. Irrespective of the manner of functioning of error detection and correction mechanisms, the network load in the scope of 15 up to 35 % is recommended [3]. Higher loads will result in the occurrence of unacceptable delays for the messages with lower messages.

Five methods of detection are included in CAN protocol specification. Two of them are functioning in the physical layer on the level of single bits and three remaining bits on the message (frame) level. All of them are mutually complementary.

Bit Monitoring. Physical connection between the node and bus is supported by a device called transceiver (Fig. 3). According to its name, the transceiver is a combination of a transmitter and receiver. The transceivers of CAN nodes are able to continue parallel transmission combined with the checking of current bus status. In case of the message with highest priority and in case of the level of the bit being actually read differing from the level of transmitted bit, a Bit Error is displayed.

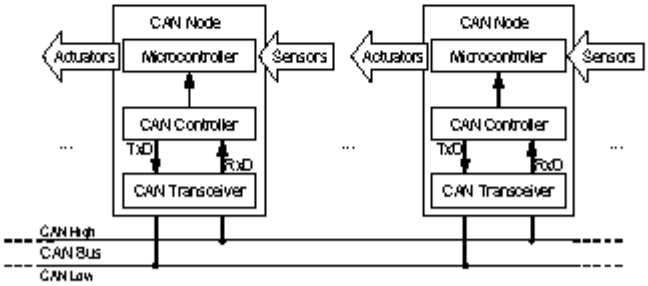

Fig. 3. Part of CAN bus network (transceiver as a part of CAN node) [2]

Bit Stuffing/Destuffing. CAN protocol is a protocol conforming with NRZ (Non-Return-to-Zero) coding [2]. Therefore it is assumed that the bus can operate in dominant or recessive mode during the whole duration of the bit. NRZ coding seems to be more simple than Manchester coding. However the long sequences of zeroes or ones in NRZ conforming code make the identification of duration times of individual bits problematic (Fig. 4). There is a lack of visible limits between individual zeroes or ones being transmitted. Nevertheless NRZ coding is widely applied in the networks with synchronous and asynchronous networks with low transmission rates. In case of Manchester coding the zeroes or ones are coded by the passages between dominant and recessive states as well as by the accompanying duration times of the dominant and recessive states. Therefore it is possible to control the duration time of bit more easily and to achieve higher transmission rates. Manchester coding is applied in synchronous and networks with high transmission rates [5, 9].

Stuffing mechanism in CAN protocol can be described as follows: in a part of frame encompassing start of frame, arbitration field, control field, data field and cyclic redundancy code sequence, every group consisting of five successive bits with the same level and supplemented with an additional sixth bit with the level opposite to the outgoing stream of bits (Fig. 4). This additional bit is eliminated by the transceivers of receiving nodes already in physical layer (destuffing). Therefore in case of six successive bits occur on the bus with the same level, Stuff Error is displayed. Additionally the elimination of excessive constant component on the bus is possible by means of stuffing and destuffing.

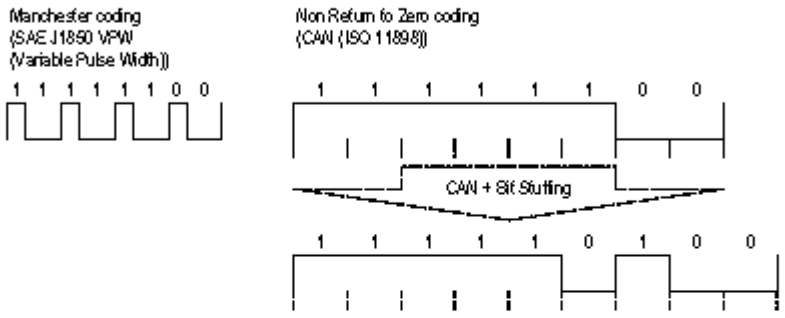

Fig. 4. NRZ and Manchester coding

Cyclic Redundancy Check (CRC). Cyclic Redundancy Check is the method of transmission correctness verification consisting in the calculation of the data of check polynomial on the transmission and receiving side for each data block being transmitted. This method is significantly better than simple parity check and much more credible than in case of ordinary checksum calculated for instance by means symmetrical difference [12]. CRC value is transmitted in data frame between utility data and confirmation field (Fig. 1).

CRC sequence is calculated on the basis of the values of bits not subjected to stuffing and contained in the following fields: start of frame, arbitration field, control field, data field of CAN message. The best results are achieved if the sum of bits in specified fields is lower than 127 [1, 4]. The controlled X value consists of sequence of bits between SOF and the end of data field. The polynomial $P(X)$ is generated on the basis of the equation (1) [1, 4]:

$$P(X) - X^{15} + X^{14} + X^{10} + X^{8} + X^{7} + X^{4} + X^{3} + 1. \tag{1}$$

The controlled X value is divided by the generated polynomial $P(X)$. The reminder of the division is the value which is sent in the field CRC (Fig. 1). Any node detecting the differences between CRC contained in the message and calculated value of CRC, will display CRC Error.

Irrespective of high efficiency of Cyclic Redundancy Check, the reduction of secure level of transmission as a result of introduction of undetectable multi-bit errors is possible owing to mutual impact of CRC and bit stuffing mechanism. The sequences simulating CAN bus functioning with artificially generated dual bit errors may lead to the occurrence of undetectable errors on the level of 10^{-9} detected errors [13].

Acknowledgement Check. In CAN bus message structure, between CRC field and End of Frame bit, two bits have been reserved (ACK Slot & ACK Delimiter) in order to confirm the error free transmission of messages (Fig. 1). ACK Slot bit is transmitted by the frame sender as the recessive bit. Also ACK Delimiter bit is transmitted as the recessive bit. Unless any errors are detected in the message by the message receiving node, a dominant bit is entered into Acknowledgement Slot. Unless the transmitter is able to detect the dominant level in ACK Slot, Acknowledgement Error is recorded. Such status is received as the signal for message retransmission.

Frame Check. Majority fields in CAN frame has a fixed format, i.e. the levels and the time of their occurrence are precisely defined by the standard (SOF, CRC Delimiter, ACK Delimiter, EOF and Intermission) (Fig. 1). In case of an incorrect value in any of these fields (e.g. 16 dominant states in CRC field, two dominant states in ACK field) detected by any controller connected to CAN bus, Frame Error is displayed.

ADDITIONAL ERROR PREVENTION IN HARDWARE AND APPLICATION LAYERS

The capacity consisting in error detection and reliable transmission of messages directly resulting for CAN protocol specification can be improved as a result of the application of reliable solutions in the scope of hardware and software layer.

The first step in the scope of the reduction of errors occurring as a result of the cooperation of CAN nodes consists in the use of **high quality** CAN transceivers and controllers. Such systems are provided with precision clock systems and with powerful computing unit. Therefore it is possible to reduce the errors caused by inaccurately interpreted time of bit duration and to obtain the signal curve slopes with maximum steepness. CAN controller with high operation speed makes it possible to display the errors immediately. Additionally obtained network contains the elements generating the short circuits on the bus less frequently. It is recommended to complete the connections by means of shielded twisted pairs. Owing to the fact that CAN network is base upon differential signal, the distortions are induced in a similar manner on the both conductors and the signal difference remains undisturbed [2]. In case of some bi-wired implementations, the operation in short-circuit or discontinuity of one of wires is allowed. However the operation of the nodes is recommended on the same potential, it can be ensured by means of additional connection by means of screening wire.

In particularly difficult conditions, it is possible to increase the difference of voltages between CAN_High & CAN_Low cables. Such solution is applied in CAN - ISO 11898-3 standard (fault-tolerant). Typically, for ISO 11898 the recessive state is represented by equal potentials of CAN_High & Low, equal to 2,5V (difference of potential equal to 0V) and the dominant state is represented by the potentials CAN_High =3,5V i CAN_Low=1,5V (difference of 2,0 V). In CAN - ISO 11898-3 standard the recessive state is represented by the potential CAN_High =5,0V and CAN_Low=0,0V (difference of 5,0 V), and the dominant state is represented by CAN_High =1,4V and CAN_Low=3,6V (difference of -2,2 V) [2, 11].

In case of necessity to apply CAN protocol in connection with dramatic limitation of hardware errors, the application of **network topology with active elements** can be considered instead of a typical passive bus. The introduction of the central point with installed CAN frames repeater, as in case of Active Start topology of FlexRay protocol would introduce the signal regeneration and accelerate the cutoff of nodes with errors „babbling idiot" [2, 6].

Although there are no symptoms of soon refreshment of CAN protocol, the introduction of the description of hardware layer is the possible direction of activities. **Bus Guardian** module could be incorporated there in order to make CAN bus more deterministic and to control the access to transmission medium, in a manner similar to FlexRay or TTP/C protocols [2, 6, 16].

The increase of tolerance to unexpected error of the devices through the **application layer functions** has been applied in CANopen protocol. CANopen is dedicated to the creation of standardized networks supporting the built-in systems. The nodes functioning control is carried out by means of Node-Guarding & Heartbeat Messages mechanisms. [7]. Node-Guarding consists in "question on operation status" of slave nodes by dedicated node (master) by means of "remote frames" with maximum priority. The nodes are requested by Heartbeat Messages function to display the status in an automatic and periodical manner. The both mechanisms enable quick identification of disconnected or erroneously operating nodes.

CONCLUSIONS

On the basis of error detection functions incorporated in CAN protocol and presented in the study, as well as on the basis of an additional limitation of the impact of disturbances, the following conclusions are possible:

1. Despite the several simultaneously operating error detection and display mechanisms, the level of the protocol errors undetectable in practice corresponds to the level of 10^{-11} in relation to the detected ones [1, 2, 4].
2. The occurrence of higher errors rate is possible. The sequences simulating with artificially generated dual bit errors may lead to the occurrence of undetectable errors at the level of 10^{-9} detected errors [13].
3. Percentage of errors can be limited applying adequate solutions in the hardware layer (cabling, differential voltages, topology) and in application layer (node-guarding & heartbeat messages nodes control).

REFERENCES

Bosch R. GmbH, 1991.: CAN Specification. Version 2.0. Stuttgart.

Bosch R. GmbH, 2008.: Sieci wymiany danych w pojazdach samochodowych. Wydawnictwa Komunikacji i Łączności. Warszawa.

Boys R., 2009.: CAN Primer: Creating your own network. Keil - an ARM Company. San Jose, California.

CAN in Automation GmbH, 2002.: CAN Specification 2.0, Part A, Part B, Addendum. Erlang-Fairhurst G., 2001: Ex 3567 Communications Engineering Course. College of Physical Sciences, University of Aberdeen. http://www.erg.abdn.ac.uk/users/gorry/course/phy-pages/nrz.html.

FlexRay Consortium, 2004.: FlexRay Communications System. Bus Guardian Specification. Version 2.0. http://www.flexray.com/.

IXXAT Automation GmbH Publication, 2008: CANopen Basics - Guarding and Heartbeat. http://www.canopensolutions.com/english/about_canopen/guarding_heartbeat.shtml.

LIN Consortium, 2003.: LIN Specification Package Revision 2.0. Motorola GmbH, Munich, Germany.

Mills A., 2009.: Manchester encoding using RS-232. Rev 2.1. http://www.quickbuilder. co.uk/qb/articles/Manchester_encoding_using_RS232.pdf, United Kingdom.

Pazul K., 2002.: Controller Area Network (CAN) Basics. http://www.cl.cam.ac.uk/research/ srg/HAN/Lambda/webdocs/an713.pdf, Microchip Technology Inc.

Softing AG Publication, 2009.: CAN bus Fault-Tolerant Signal Levels. http://www.softing. com/home/en/industrial-automation/products/can-bus/more-can-bus/fault-tolerant/signal-levels.php?navanchor=3010545, Haar, Germany.

Tervo R., 2001.: CMPE4833 Digital Communications. Error Detection with the CRC. University of New Brunswick, Department of Electrical and Computer Engineering, http://www.ee.unb.ca/tervo/cmpe4833/crc.htm.

Tran E., Koopman P. (advisor), 1999.: Multi-Bit Error Vulnerabilities in the Controller Area Network Protocol. Carnegie Mellon University. Pittsburgh.

Widerski T., 2005.: Samochodowe sieci informatyczne. Poradnik serwisowy, 5/2005, Wydawnictwo Instalator Polski, Warszawa.

Widerski T., Kędzierski J., 2004.: Samochodowe sieci informatyczne. Auto Moto Serwis, 6/2004, Wydawnictwo Instalator Polski, Warszawa, pp. 35-37.

Zimmermann W., Schmidgall R., 2008.: Magistrale danych w pojazdach. Protokoły i standardy. Wydawnictwa Komunikacji i Łączności. Warszawa.

BEZPIECZNA KOMUNIKACJA PODZESPOŁÓW POJAZDÓW DZIĘKI WBUDOWANYM MECHANIZMOM PROTOKOŁU CAN

Streszczenie. Sieci CAN są najbardziej rozpowszechnionymi sieciami łączącymi węzły elektromechaniczne w zastosowaniach motoryzacyjnych i przemysłowych. Ich ciągła popularność zapoczątkowana w latach dziewięćdziesiątych XX wieku wynika z prostoty komunikacji i wysokiej niezawodności. W artykule zaprezentowano czynniki wpływające na bezpieczeństwo komunikacji wynikające bezpośrednio ze specyfikacji protokołu oraz dodatkowe czynniki podnoszące bezpieczeństwo, leżące u podstaw różnych implementacji warstw sprzętowych i warstw aplikacji. Przedstawiono również czynniki wpływające na minimalizację obsługi błędów istniejące w konkurencyjnych rozwiązaniach, a które można zaimplementować w protokole CAN.

Słowa kluczowe: protokół CAN, wykrywanie błędów, sieci