

# Error Detection Capabilities of Automotive Network Technologies and Ethernet - A Comparative Study -

Mehrnoush Rahmani, Wolfgang Hintermaier  
Automotive Bus Architecture and Networking  
BMW Group Research and Technology  
Email: firstname.lastname@bmw.de

Bernd Müller-Rathgeber  
Institute of Communication Networks  
Munich University of Technology  
Email: mueller-rathgeber@tum.de

Eckehard Steinbach  
Media Technology Group  
Munich University of Technology  
Email: eckehard.steinbach@tum.de

**Abstract**—Coding the payload data and sending the code-word as an overhead of the packet is a very common way to protect data in communication networks. The protection level of the coding technique is chosen depending on the importance of the transmitted data. While high priority, safety critical applications do not tolerate any single bit error in data packets, lower priority services can handle few bit errors. Coding techniques provide error detection and up to a certain level also error correction. Bit errors that are not detectable by coding techniques at the receiver side occur with a residual error probability. The better the coding technique is, the lower is its residual error probability for different bit error rates. Most automotive network systems use Cyclic Redundancy Codes (CRC) mainly in order to detect transmission bit errors. Instead of correcting the identified bit errors which is quite time consuming, usually a retransmission of the damaged data packet is triggered. Similar to automotive network systems, Ethernet, the most applied network technology in local area networks uses the CRC error detection technique. In this work, we present a comparative study of the error detection capabilities of automotive network systems and Ethernet as a possible network system for time critical applications in the car. We evaluate the related residual error probabilities for a reasonable range of bit error rates. Furthermore, several commercial concepts are presented from the automation field that increase the error detection capability of the standard Ethernet technology significantly.

## I. INTRODUCTION

Many different network technologies have been applied in the automotive field during the last years. Each of them is responsible for the transmission of a different type of data. Depending on the importance of the data to be transmitted, the network technologies are equipped by appropriate coding mechanisms. The additional overhead provided by coding, enables end systems to detect and possibly correct transmission failures. Accordingly, there is a trade off between providing good error detection performance by using large coding fields in the network messages and saving throughput capacity in the network. Many research works have been done in order to determine optimal coding techniques for short control messages with high priorities and large messages with lower priorities [4], [5], [1] in decentralized networks such as automotive environments.

Automotive network systems often use Cyclic Redundancy Codes (CRC) or in a few cases checksums and parity codes to protect the payload and header data. CRCs are mostly applied because they require a simple realization in binary

hardware or software and provide a very good performance at error detection. Parity codes and checksums implemented in software are used in systems where the temporal performance is more important than the correctness of the transmitted data, e.g., in RAM memory chips to detect memory defects or in transport protocols.

Statistics in [10] show that the most common transmission failure is the single bit failure, i.e., 79% of all occurred failures are single bit errors. Thus, parity codes and checksums with a good single bit error detection capability suit quite well to low-noise, not safety-critical transmission networks. The physical transmission medium is a very important factor when choosing adequate coding techniques. Wired transmission systems are less affected by the environmental noises and should be endowed with less protective codes while wireless networks are very much disturbed by the harsh environment and should be better protected. A list of different physical transmission media is given in Table I [19]. It can be seen that the error probability which is quite high for cellular radio networks ( $10^{-2}$  to  $10^{-3}$ ) is very low for fiber optic cables (less than  $10^{-9}$ ).

By applying an appropriate coding technique, networks are able to detect most of the transmission errors and thus, reduce the probability of undetectable failures, i.e., the reduction of the residual error probability.

In this paper, we address the problem of the residual error probability for automotive network technologies and focus on the networks' error detection performance. We also study the error detection capability of Ethernet as the most applied network technology in local area networks. In order to examine a possible application of Ethernet for in-vehicle time critical communication systems, we compare its error detection performance with the automotive networks. Finally, we give an overview of the possibilities to improve the transmission reliability of Ethernet for safety-critical, real-time applications.

## II. COMMON ERROR DETECTION TECHNIQUES IN THE AUTOMOTIVE FIELD

Parity check codes, Arithmetic Checksums and Cyclic Redundancy Codes (CRC) have been widely applied in the automotive networks due to their low complexity, simple implementation, and good error detection performance. Due to the time constraints of the automotive network systems,

Physical media	Error probability
Copper double wire	$10^{-4}$ to $10^{-6}$
Twisted pair (Differential)	$\leq 10^{-7}$
Coaxial cable	$\leq 10^{-6}$
Fiber Optics	$\leq 10^{-9}$
Infrared	$10^{-4}$ to $10^{-6}$
Cellular radio	$10^{-2}$ to $10^{-3}$

TABLE I  
ERROR PROBABILITIES OF DIFFERENT TRANSMISSION MEDIA [19]

error correction is not performed by these codes. Instead a retransmission of the corrupted message is triggered. Indeed, parity codes, checksums and CRCs are not adequate at error correction. Other, mathematically more complex codes such as Reed-Solomon codes or convolutional codes are applied where error correction is required.

#### A. Parity check codes

The parity check describes the addition of one check bit to an information bulk in order to achieve an even (even parity) or an odd (odd parity) value [10]. Due to its simple realization and fast performance, the party check is applied where the time constraints are high while the data priority level and the probability of more bit errors are low. Parity check codes detect only an odd number of bit errors and do not correct any failures.

#### B. Arithmetic checksum

Arithmetic checksums improve the error detection capability provided by the parity check codes [10]. The information bytes are added up with carry. The result of the addition is inverted and sent. The receiver performs the same carry based addition and adds the received checksum to the calculated result.

Typical values are 8, 16 and 32 checksum bits.

By using arithmetic checksums, all single errors and most of the burst errors can be detected. Error correction cannot be performed offhand.

#### C. Cyclic Redundancy Codes

CRC provides better error detection performance compared to arithmetic checksums. It is mostly realized in hardware by applying shift registers and bitwise XOR. The original message consisting of  $m$  bits is considered as a binary polynomial  $M(x)$  of degree  $m - 1$ . In order to calculate the CRC ( $C(x)$ ), a generator polynomial  $G(x)$  of degree  $k$  is required. The check codes are then calculated as following:

$$(M(x) \cdot x^k) \bmod G(x) = C(x) \quad (1)$$

The remaining polynomial  $C(x)$  is of degree  $k - 1$  and length  $k$ . It is added to the information bits and a message of length  $m + k$  bits is sent to the receiver. The receiver performs the same calculation on the received information bits and compares its result with the received CRC field. If they do not match an error has occurred [3], [4], [10].

By using the CRC technique all single bit errors can be

detected. Also nearly all double bit errors are detectable when  $G(x)$  has at least three terms. All odd numbers of bit errors can be identified when  $G(x)$  has a factor of  $x + 1$  and burst failures up to  $k$  bit errors can also be tracked down.

### III. RESIDUAL ERROR PROBABILITY

Residual error probability  $P_{re}$  is the probability that the corruption of the transmitted data remains undetected after performing the decoding procedure [3]. The actual  $P_{re}$  depends on the length of the message, the generator polynomial and the bit error probability.

There are several methods proposed in the literature in order to determine  $P_{re}$ : Direct code analysis, transformed code analysis, the Monte-Carlo-simulation and the  $P_{re}$  calculation by Stochastic Automaton [3]. In this paper we focus on the direct code analysis and introduce further estimations for this method. We define upper and lower limits for the actual  $P_{re}$ . In the direct code analysis all possible error patterns are generated explicitly which entails a large computational complexity. The number of error bits is then counted and considered in the  $P_{re}$  calculation.

The hamming distance  $D$  of the generator polynomial is determined according to the derivations in [1].  $D$  is the number of bits which differ between two codewords. The larger  $D$  is, the better is the capability of the code to detect errors.

Accordingly, for the codeword length  $N$ , the hamming distance  $D$  and the bit error probability  $p$  the upper limit of  $P_{re}$  can be calculated as

$$P_{re_{upper}} = \sum_{i=D}^N \binom{N}{i} \cdot p^i \cdot (1-p)^{N-i} \quad (2)$$

The lower bound of  $P_{re}$  is derived in [2] and can be computed as

$$P_{re_{lower}} = 2^{-r} \cdot \sum_{i=D}^N \binom{N}{i} \cdot p^i \cdot (1-p)^{N-i} \quad (3)$$

where  $r$  is the degree of the generator polynomial.

#### IV. EXAMPLES OF AUTOMOTIVE NETWORK TECHNOLOGIES AND THEIR ERROR DETECTION MECHANISMS

Different automotive network technologies support different error detection techniques depending on their functionalities. In the following, some of the current automotive network systems are briefly presented and their coding mechanisms are introduced.

##### A. CAN

CAN (Controller Area Network) is according to [6] a multicast shared serial bus standard, developed in the 1980s by Robert Bosch GmbH, for connecting Electronic Control Units (ECUs) originally for automotive purposes (as a vehicle bus) in electromagnetically noisy environments. Each CAN message can transmit up to 8 bytes payload data. Longer messages are segmented accordingly. Currently, CAN is used in many parts of the car network system, mainly to transmit control data packets. The CAN bus applies a CRC-15 with a hamming distance of  $D = 6$  for 8 byte information data packets to protect the payload data [3]. The CRC generator polynomial  $G(x)$  is

$$G(x) = x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1 \quad (4)$$

##### B. MOST

MOST (Media Oriented Systems Transport) is a network standard developed for interconnecting multimedia components in vehicles [7]. Based on the optical fiber bearer, it provides a networking system at bit rates up to 24 Mbit/s. MOST is mainly applied in ring topology. However, a star topology can also be realized.

The MOST standard specifies three different communication channels. Connection setups and terminations take place over the control channel while data flows over synchronous and asynchronous channels. The payload data is transmitted in packets of 64 bytes where 60 bytes are assigned to the synchronous and asynchronous channels, 2 bytes to the control channel and the rest are header and frame control data.

In order to detect transmission errors, MOST applies a CRC-16, i.e., a generator polynomial of degree 16. However, no information has been provided by the manufacturer about this polynomial so far.

##### C. FlexRay

FlexRay is a deterministic and fault-tolerant bus standard which also supports high data rates for advanced automotive control applications [8]. It supports payload data lengths from 8 bytes up to 254 bytes. There are two CRC generator polynomials supported by FlexRay. A CRC-11 with a hamming distance of  $D = 6$  to protect the header data and a CRC-24 with a hamming distance of  $D = 4$  to protect the header and the payload data. These generator polynomials are given as follows:

$$G(x)_{header} = x^{11} + x^9 + x^8 + x^7 + x^2 + 1 \quad (5)$$

$$G(x)_{header+payload} = x^{24} + x^{23} + x^{22} + x^{19} + x^{18} + x^{16} + x^{14} + x^{13} + x^{11} + x^{10} + x^8 + x^7 + x^6 + x^3 + x + 1 \quad (6)$$

##### D. LIN

LIN (Local Interconnect Network) was developed as a low cost and low complexity automotive network. It is a serial bus similar to CAN and is applied where the transmission capacity and versatility of CAN are not required [9]. The packet lengths of LIN reach 8 bytes. LIN supports an 8-bit arithmetic checksum with carry which performs similar to a CRC with a hamming distance of  $D = 2$ .

#### V. ETHERNET, THE MOST COMMON TECHNOLOGY IN LOCAL AREA NETWORKS

Ethernet (IEEE 802.3) is a well known technology in home and office networks. It was developed around 1980 by Robert Metcalfe and since then constantly extended. As an OSI-Layer 2 protocol, Ethernet specifies the data frame format (see Fig.1) and the physical transmission technology. It provides transmission capacities of up to 10Gbit/s and can use coaxial, copper or fiber optic cables as transmission medium. In the star topology, all nodes are connected with one central device, the star-coupler. There are two possible configurations:

- In a switched star, the node has non-constraining access to the communication medium and the data frames are queued in the star-coupler to avoid collisions.
- In an repeating star, all communication links build together a shared medium.

The access to this medium is mostly based on Carrier Sense Multiple Access with Collision Detection (CSMA/CD).

Ethernet heavily gained impact in automatization applications. Many new plants use Ethernet as an alternative to specially developed field buses. Accordingly, several commercial automatization solutions such as Profinet [11], Powerlink [12] and EtherCAT [13] have been introduced. They all use standard or slightly modified Ethernet hardware and standard Ethernet wiring. Thus, complexity reduction in fabric networks and cost advantage are achieved.

In other transportation sectors, where cost per unit has not that much impact, Ethernet partly replaces traditional network technologies. The new Airbus A380 and his successors use a special adapted Ethernet network after the avionic ARINC 664 Standard called AFDX [14]. Mainly, the rising communication needs due to highly meshed functional ranges and the increasing number of multimedia applications led to this development.

Since this change of paradigm can also be interesting for automotive development, it is important to analyze the adoption of Ethernet in the automotive field.

##### A. Error detection capability of Ethernet

The standardized CRC-32 generator polynomial of the IEEE 802.3 has been analyzed for its effectiveness by several

methods in the literature [1], [5]. Many other polynomials of degree 32 have been proposed that perform better at error detection because of their larger hamming distances. However, due to the Ethernet's previous application field of transferring mostly delay insensitive data packets, reliable transmission has not been required and the IEEE 802.3 CRC has been adapted accordingly. The Ethernet polynomial has a hamming distance greater than or equal to 8 up to a data word length of 91 bits,  $D = 7$  to 171 bits,  $D = 6$  to 268 bits,  $D = 5$  to 2974 bits,  $D = 4$  to 91607 bits and  $D = 3$  to at least 128 Kbits [1]. The polynomial is shown in the following:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1 \quad (7)$$

IEEE 802.3 specifies the first two ISO/OSI layers and the above mentioned CRC-32 to protect the physical layer data. In the case the Internet Protocol (IP) is added to the Ethernet packet on the third layer, the IP-header is additionally secured by the IP-checksum. Transport protocols such as the Transport Control Protocol (TCP) and the User Datagram Protocol (UDP) in the fourth OSI layer also contain a 16-bit arithmetic checksum to protect their header but also the payload data. The format of such an Ethernet packet is shown in Fig.1. Accordingly, in an Ethernet packet with transport

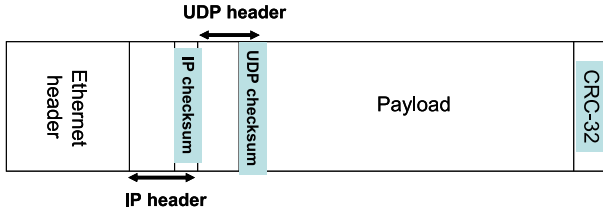


Fig. 1. Ethernet packet format with IP and UDP transport protocol

protocols the payload data is protected by a CRC-32 and a 16-bit arithmetic checksum against transmission errors.

## VI. ANALYSIS OF RESIDUAL ERROR PROBABILITIES OF ETHERNET AND AUTOMOTIVE NETWORK TECHNOLOGIES

By applying the method of direct code analysis (DCA) (See Eq.2 and Eq.3) mentioned in Section III, we calculate upper and lower bounds of the residual error probability ( $P_{re}$ ) for the above mentioned automotive network technologies and Ethernet. Because of the lack of information about the MOST generator polynomial CRC-16, no  $P_{re}$  analysis has been performed for it. Since we analyze the error detection performance for protecting the payload data, we only consider the CRC-24 of FlexRay for the  $P_{re}$  calculation. It is also important to mention that for Ethernet packets only the corresponding CRC-32 has been considered and the effects of the IP and the transport protocols arithmetic checksums have not been taken into account.

According to the Ethernet standard, the shortest allowed payload data length for tagged Ethernet packets is 42 bytes whereas CAN and FlexRay are able to transport 8 byte long

payload data. LIN can handle 8 byte packets where one byte is reserved for the arithmetic checksum.

The results of our calculation are shown in Fig.2. The

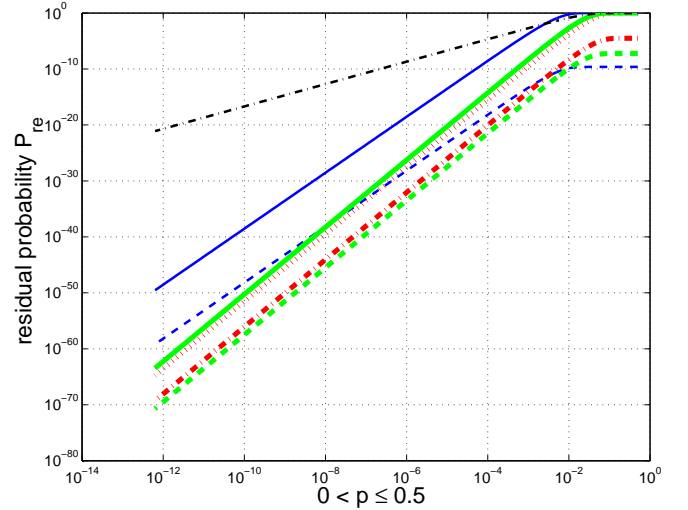


Fig. 2. Analysis of residual error probabilities of Ethernet and automotive network technologies for short data packets depending on the bit error probability  $p$ : The  $P_{re}$  of CAN is computed for a payload field of 8 bytes (Dotted curve: upper bound, thick dashed dotted curve: lower bound), the  $P_{re}$  of FlexRay is calculated for header and payload fields of 5 and 8 bytes (Thick solid curve: upper bound, thick dashed curve: lower bound), the  $P_{re}$  of Ethernet is computed for header and payload fields of 14 and 42 bytes (Thin solid curve: upper bound, thin dashed curve: lower bound), the  $P_{re}$  of LIN is calculated for a payload field of 7 bytes (Thin dashed dotted curve: upper bound).

residual error probability is calculated for the related data fields described in Fig.2. The  $P_{re}$  of the LIN bus is computed as for a CRC with a hamming distance of 2 which represents the  $P_{re}$  upper bound of the LIN bus in Fig.2.

According to Table I, the largest bit error probability of twisted pair cables in a noiseless environment is  $10^{(-7)}$ . The related  $P_{re}$  derived from Fig.2 is  $10^{(-24)}$ . A  $P_{re}$  value of  $10^{(-24)}$  means the appearance of one residual bit error among  $10^{24}$  transmitted bits. Accordingly, for a transmission rate of 100 Mbit/s, one bit error within about  $3.10^8$  years remains undetected which shows the unlikelihood of residual bit errors when transmitting short data packets over the Ethernet. However, in electromagnetically noisy environments the bit error rates from Table I are not valid anymore and the probability of residual bit errors increases significantly. Therefore, we continue our discussion about improving the error detection performance of Ethernet in the following.

Fig.3 shows the performance of the Ethernet CRC for short data packets with 42 byte payload data and for longer data packets with 254 byte payload data. Because of the consistent hamming distance of 5 for the large data range 268 bits to 2974 bits, the residual error probability does not increase much for longer data packets. This result confirms our discussion in Section V about the adaptation of the Ethernet polynomial for longer data packets.

According to Fig.2, the error detection capability of Ethernet

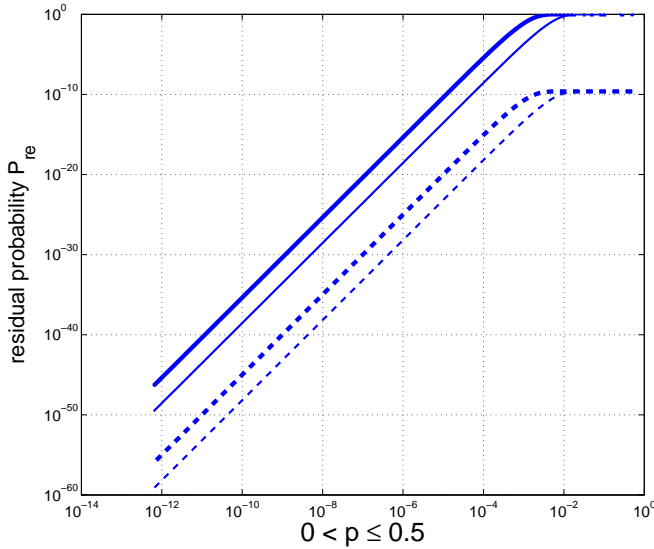


Fig. 3. Analysis of the residual error probability of Ethernet depending on the bit error probability  $p$  for short data packets with a payload field length of 42 bytes (Thin solid curve: upper bound, thin dashed curve: lower bound) and longer data packets with 254 bytes payload data (Thick solid curve: upper bound, thick dashed curve: lower bound).

seems not to be as good as for the FlexRay and CAN standards. In the next section, we introduce possibilities to overcome this issue with only a few small modifications in the Ethernet protocol.

## VII. MECHANISMS TO INCREASE THE ERROR DETECTION CAPABILITY OF ETHERNET

There are several possibilities to improve the performance of the Ethernet CRC in order to achieve lower residual error probabilities, especially for applications in electromagnetically noisy environments. One solution to achieve higher error detection capabilities without changing the Ethernet frame format is the addition of a second coding mechanism to the Ethernet payload.

PROFIsafe [15] from SIEMENS Automation and Drive found a solution for integrating IEC 61508 Safety Integrity Level 3 (SIL 3) [16] or EN954-1 Cat.4 [17], the highest security demand in manufacturing automation. PROFIsafe uses a so called “Black Channel” technique, which means that an additional security mechanism is embedded in the data field of the Ethernet protocol. Thus, it is possible to send standard and fault tolerant Ethernet packets. An up to 4 Byte, i.e. a 32-Cyclic Redundancy Code assures the safety of up to 122 Bytes of data. In addition to the CRC, further efforts can be attempted to meet higher safety levels. A short overview is shown in Table II where the Sequence Number is a sender based counter for each set of data. The 1 Byte long integer value is increased with every transmission to detect packet losses due to transmission errors. The Acknowledgment principle is known from the TCP protocol. The acknowledgment is a special packet that

Failure	Sequence Number	Acknowledgment	CRC
Replay	x		
Lost	x	x	
Insertion	x	x	
Permutation	x		
Tampering			x
Delay		x	

TABLE II  
PROFISAFE FAULT DETECTION

confirms the correct reception of one or more data frames. Since the standard Ethernet protocol does not contain the information fields mentioned above, an additional protocol such as PROFIsafe can be very useful to increase the safety level for short data frames.

Another method for protecting the data transmission is the application of layer 4 protocols with additional CRC. An outstanding example is the Reliable UDP (RUDP) [18] that uses IP and combines the advantages of TCP and UDP. Like PROFIsafe, the header of RUDP consists of a sequence number, an acknowledgment and a 16-CRC in order to achieve higher error detection capabilities. Despite all these higher layer securing mechanisms, the following issues should always be considered:

- A standard Ethernet Network Interface or a switch discard packets with an incorrect Ethernet-CRC without even recognizing the possibility to correct those errors by the additional higher layer CRC.
- Unlike the Ethernet CRC that is computed in hardware, the higher layer CRC is typically calculated in software and requires comparatively more processing time and power.

## VIII. CONCLUSIONS AND FUTURE WORK

### A. Conclusions

This paper presents a theoretical and mathematical comparison between error detection capabilities of different Automotive Network Technologies and Ethernet by comparing their residual error probabilities. We showed that at first view Ethernet performs comparatively inferior at protecting short data packets while it performs quite well at protecting longer packets. For short real-time messages used in motion control or safety critical applications like x-by-wire, additional effort has to be made in order to secure the data against transmission errors. Several commercial, safe and real-time Ethernet approaches have been presented in this work. Among them, PROFIsafe seems to perform the best by applying an additional coding mechanism in the Ethernet frame, a sequence number and acknowledgments.

Accordingly, we can conclude that in addition to big advantages that Ethernet provides such as low cost and high transmission capacities, it also provides at least the same data protection level such as other automotive network systems with only minor modifications.

## B. Future work

In TCP/IP or UDP/IP Ethernet networks, the payload data is additionally protected by arithmetic checksums in TCP and UDP packets. The effect of this additional coding on the residual error probability of Ethernet is currently being analyzed and is one of our future works. Furthermore, we analyze other safety mechanisms to improve the error detection capability of Ethernet more than it has been achieved until today.

## IX. ACKNOWLEDGMENTS

This work has been done in collaboration with the Institute of Information Technology in Mechanical Engineering Department at the Technical University of Munich.

## REFERENCES

- [1] Philip Koopman, 32-Bit Cyclic Redundancy Codes for Internet Applications, *The International Conference on Dependable Systems and Networks (DSN)*, 2002, <http://citeseer.ist.psu.edu/koopman02bit.html>.
- [2] Kamal Merchant, Grenzwerte der Restfehlerwahrscheinlichkeit (Limits of the Residual Error Probability), *DKE Internal Report*, October 2006.
- [3] Frank Schiller and Tina Mattes, "An Efficient Method to Evaluate CRC-Polynomials for Safety-Critical Industrial Communication", *Journal Of Applied Computer Science*, vol. 14, No. 1, 2006.
- [4] T. Chakravarty and Ph. Koopman, Performance of Cyclic Redundancy Codes for Embedded Networks, *M.S. Project Report*, December 2001, <http://citeseer.ist.psu.edu/662901.html>.
- [5] Guy Castagnoli, Stefan Braeuer and Martin Herrmann, Optimization of Cyclic Redundancy-Check Codes with 24 and 32 Parity Bits, *IEEE Transactions On Communications*, June 1993, vol. 41, No. 6, pp. 883-892.
- [6] Robert Bosch GmbH, CAN (Controller Area Network) Specification, Version 2.0, 1991, <http://www.semiconductors.bosch.de/pdf/can2spec.pdf>.
- [7] MOST Cooperation, MOST (Media Oriented Systems Transport) Specification, Version 2.4, 2005, <http://www.mostcooperation.com/downloads/Specifications>.
- [8] FlexRay Consortium, FlexRay Communications System, *Protocol Specification*, Version 2.1, 2005, <http://www.flexray.com>.
- [9] LIN Consortium, LIN (Local Interconnect Network) Specification package, Revision 2.0, 2003, <http://www.lin-subbus.org>.
- [10] Christian Scheurer, Fehlererkennung: Cyclic Redundancy Check Code, Odd/ Even Parity, Checksum, *Technical report*, March 2001, <http://www.mountpoint.ch/unique/project/crc/index.html>.
- [11] International Electrotechnical Commission (IEC), IEC/PAS 62407 Real-time Ethernet PROFINET IO, *Publicly available Specification*, SIEMENS 2005, <http://webstore.iec.ch/webstore/webstore.nsf/artnum/034395>.
- [12] ETHERNET Powerlink Standardization Group (EPG), Real-time Ethernet Powerlink (EPL) IEC PAS 62408, *Publicly available Specification Pre-Standard*, 2005, <http://webstore.iec.ch/webstore/webstore.nsf/artnum/034404>.
- [13] International Electrotechnical Commission (IEC), IEC/PAS 62407 EtherCAT Ethernet Control Automation Technology, *Publicly available Specification*, EtherCAT Technology Group (ETG) 2004, <http://webstore.iec.ch/webstore/webstore.nsf/artnum/034392>.
- [14] ARINC Incorporated, ARINC-Standard 664 Part 7 - Avionics Full Duplex Switched Ethernet (AFDX) Network, *Protocol Specification*, 06.2005, [http://www.arinc.com/cf/store/catalog\\_detail.cfm?itemid=574](http://www.arinc.com/cf/store/catalog_detail.cfm?itemid=574).
- [15] Andreas Peters, Faulttolerant Communication over industrial WLAN, SIMATIC Safety SIEMENS 2006, <http://control-net.fh-duesseldorf.de/download/Peterssiemens.pdf>.
- [16] International Electrotechnical Commission (IEC), 61508 - Functional safety of electrical/electronic/programmable electronic safety-related systems, 2000, <http://webstore.iec.ch/webstore/webstore.nsf/artnum/031512>.
- [17] Deutsches Institut fuer Normung e.V., EN 951-1 Safety of machinery - Safety related parts of control systems - General principles for design, 1997, <http://www2.din.de/index.php?lang=en>.
- [18] T. Bova and T. Krivoruchka, Internet Engineering Task Force INTERNET DRAFT, RELIABLE UDP PROTOCOL, 1999, <http://www.javvin.com/protocol/reliable-udp.pdf>.
- [19] Armin Schön, Und am Anfang war das Ethernet... die Unterlage des Internet, KNF Kongress, November 2001, <http://www.franken.de/fileadmin/mediapool/kongress/2001/ethernet.pdf>.