ESPOO 2004 VTT RESEARCH NOTES 2265

Jarmo Alanen, Marita Hietikko & Timo Malm

Safety of Digital Communications in Machines

Prioritising occupational safety

Työsuojelurahasto
Arbetarskyddsfonden
The Finnish Work Environment Fund

VTT TIEDOTTEITA – RESEARCH NOTES 2265

Safety of Digital Communications in Machines

Jarmo Alanen, Marita Hietikko & Timo Malm

VTT Industrial Systems

ISBN 951–38–6502–9 (soft back ed.)
ISSN 1235–0605 (soft back ed.)

ISBN 951–38–6503–7 (URL: http://www.vtt.fi/inf/pdf/)
ISSN 1455–0865 (URL: http://www.vtt.fi/inf/pdf/)

JULKAISIJA – UTGIVARE – PUBLISHER

VTT, Vuorimiehentie 5, PL 2000, 02044 VTT
puh. vaihde (09) 4561, faksi (09) 456 4374

VTT, Bergsmansvägen 5, PB 2000, 02044 VTT
tel. växel (09) 4561, fax (09) 456 4374

VTT Technical Research Centre of Finland, Vuorimiehentie 5, P.O.Box 2000, FIN–02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 456 4374

VTT Tuotteet ja tuotanto, Tekniikankatu 1, PL 1307, 33101 TAMPERE
puh. vaihde (03) 316 3111, faksi (03) 316 3495

VTT Industriella System, Tekniikankatu 1, PB 1307, 33101 TAMMERFORS
tel. växel (03) 316 3111, fax (03) 316 3495

VTT Industrial Systems, Tekniikankatu 1, P.O.Box 1307, FIN–33101 TAMPERE, Finland
phone internat. + 358 3 316 3111, fax + 358 3 316 3495

Technical editing Maini Manninen

Otamedia Oy, Espoo 2004

Alanen, Jarmo, Hietikko, Marita & Malm, Timo. Safety of Digital Communications in Machines. Espoo 2004. VTT Tiedotteita – Research Notes 2265. 93 p. + app. 1 p.

Keywords serial mode transmission, safety-related communications, safety bus, machines, HAZOP

Abstract

The utilisation of digital communications in safety-related machine control systems has been widely extended during the last ten years. This new technology brings about an additional safety engineering challenge compared to a single controller case where only simple wired communication is needed to execute safety-related functions.

The scope of this report is safety-related serial communications in machine automation. Standards and guidelines that include information dealing with safety-related communications and the design of safety-related communication systems are introduced. The typical message error types or threats relating to serial mode transmission as well as defence methods against these threats are also introduced.

There are several safety buses available for safety-related machine and automation applications. The basic information about these safety buses is given in this report. This information includes methods against possible transmission errors. Most of the safety bus solutions are commercially available from several suppliers. Some safety bus solutions that are not commercially available are also described.

A documentation and analysis tool to support the safety analysis of bus-based communication systems at signal level is presented. The tool is based on database software, and the analysis method is based on Hazard and Operability study (HAZOP). This tool was developed within this project and tested with two case studies consisting of distributed control systems in machine automation applications. The advantages of using this tool are presented.

A serial mode wireless communication is gaining ground in safety-related machine applications, and therefore the wireless message transmission is also considered. It was noticed that the safety analysis framework described in this report is applicable in the case of wireless communication as well. Wireless communication does not bring any new message error types; only the probability of the error types will possibly change. Therefore, the same defence methods against message errors are also true in the case of wireless systems.

Preface

This report is an output of the KETU project (The implementation and maintenance of safety-related functions in machine control systems based on field bus and wireless communication). The KETU project was carried out from 2001–2004. The project belongs to the national occupational accident prevention programme.

Financial support for the project was provided by the Finnish Work Environment Fund and Finnish industry. The companies that co-operated and gave financial support to the project were Sandvik Tamrock, Timberjack, Metso Minerals, Kalmar Industries and Kone Cranes.

The final format of this report was achieved with help, discussions and comments from Jari Karjalainen, Risto Tiusanen, Maarit Kivipuro and partners from the companies.

We would like to thank all those who have participated in this project.

Tampere, October 2004,

Authors

Contents

List of acronyms

ALARP	As Low As Reasonably Practicable
AP	Application Part
BER	Bit Error Ratio
BIA	Berufsgenossenschaftlichen Instituts für Arbeitssicherheit, Institute for Occupational Safety of Accident Insurance Institutions
BIP	Bus Interface Part
CAN	Controller Area Network
COB-ID	Communication object identifier (=CAN identifier)
COTS	Commercial Off-The-Shelf
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CSP	Communicating Sequential Processes
E/E/PE	Electrical/Electronic/Programmable Electronics
EMI	ElectroMagnetic Interference
EUC	Equipment Under Control
FCS	Frame Checking Sequence
FMEA	Failure Mode & Effects Analysis
GFC	Global Fail Safe Command
HAZOP	Hazard and Operability studies
HDLC	High level Data Link Control
HWIL	Hardware-in-the-loop
I/O	Input/Output
LOTOS	Language for Temporal Ordering Specification
MD	Management Device
MTTF	Mean Time To Failure
NDA	Non-Disclosure Agreement
OSI	Open Systems Interconnection
PDO	Process Data Object
PES	Programmable Electronic System
PLC	Programmable Logic Controller
RAM	Random Access Memory
RAMS	Reliability, Availability, Maintainability and Safety
RTOS	Real-Time Operating System
SDL	Specification and Description Language
SDO	Service Data Object
SIL	Safety Integrity Level
SRDO	Safety-related Data Object
SRECS	Safety-related Electrical Control System
TCP/IP	Transmission Control Protocol / Internet Protocol
TTP	Time-Triggered Protocol
UML	Unified Modeling Language
WCET	Worst Case Execution Time

1. Introduction

Utilisation of digital communications in safety-related machine control systems brings about an additional safety engineering challenge compared to a single controller case where no communications is needed to execute safety-related functions. Additional function block introduced by the communication system is illustrated in Figure 1.

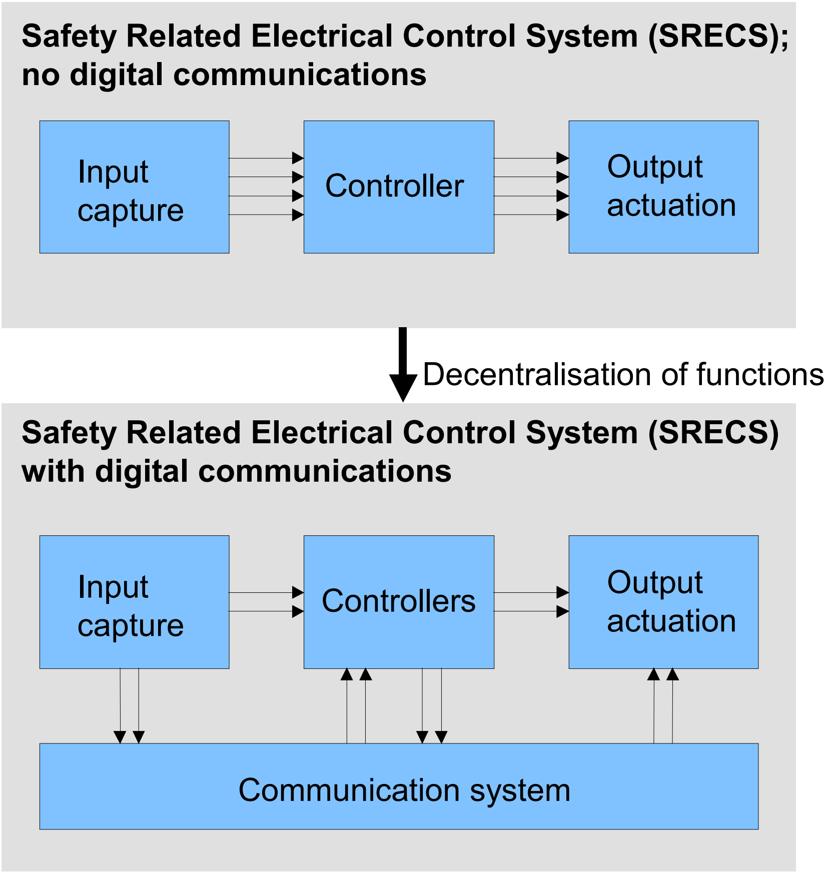

Safety Related Electrical Control System (SRECS); no digital communications
Input capture
Controller
Output actuation
Decentralisation of functions
Safety Related Electrical Control System (SRECS) with digital communications
Input capture
Controllers
Output actuation
Communication system

Figure 1. Safety-related Electrical Control System (SRECS) with and without digital communications.

Figure 1 provides a generic view of complex communication system by presenting the case of a distributed control system. Figure 1 does not illustrate well the traditional use of communications, like remote control and monitoring of machines, in which case there is simply a communication channel between two subsystems: the operator platform and the machine control system.

As the scope of this report is safety-critical serial communications in machine automation, we try to specify more precisely than in Figure 1 what is included in the communication system and what is left for the other parts of the control system. To do this, we will first take a look at a centralised system without communications and then decentralise it onto three controllers to find out what is needed because of the decentralisation. As an example, a centralised system with six processes is presented in Figure 2. The system consists of three input signals (signals 4, 5 and 15), one output signal (signal 11) and several signals between processes. A single controller executes all processes.

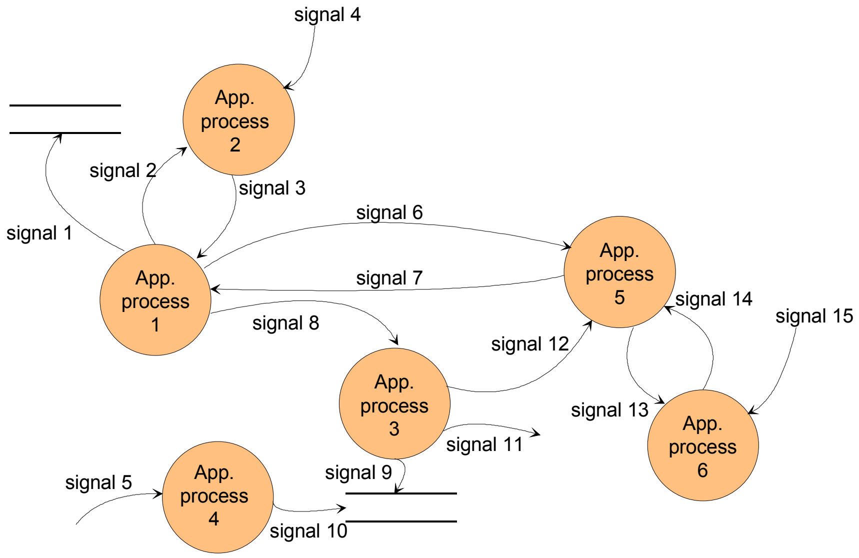


Figure 2. A centralised system with no serial communications.

The decentralised version of the same control system is presented in Figure 3.

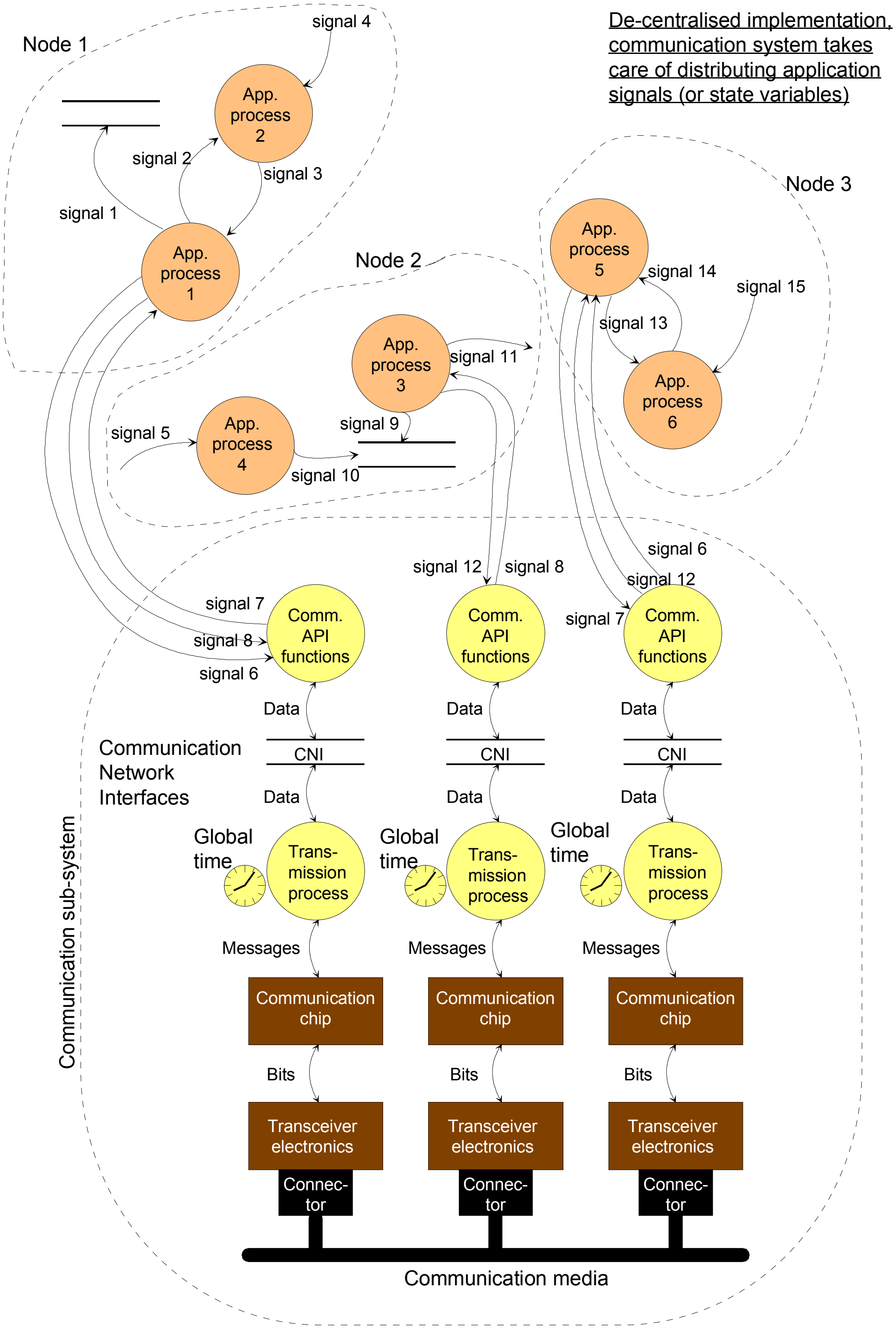


Figure 3. The control system of Figure 2 distributed to three controllers.

Figure 3 clearly shows the software and hardware items that are of additional concern to the safety engineers in distributed systems:

- Communication media (the cabling)
- Connectors
- Transceiver circuitry
- Communication chip (often included in the microcontroller); sometimes implemented by software
- Communication software (protocol stack and its software driver)
- Arrangement of global time (if needed).

Faults in any of these or a random error, like EMI, may cause message errors that appear as safety-critical failures in the system. A list of root causes that may produce message errors is presented in Figure 4. Most of the root causes are derived from EN 50159-2 [13]. Figure 4 also illustrates the message level threats (message errors) according to EN 50159-2, except that an additional threat, inconsistency, is added to support distributed control systems with multiple receivers for a single state variable. Furthermore, the defences against the threats, as given in EN 50159-2, are presented. An additional layer of defences is added on top of the EN 50159-2 defences. The additional layer is called architectural defences in Figure 4. These are the defences that are mostly needed in systems where the continuous state of the system is the safe state, like drive-by-wire systems.

The application signals are extracted from the messages. The message errors can be transformed into a set of signal deviations to reflect the guide words presented by the HAZOP method [24]. Table 1 illustrates this transformation.

Table 1. EN 50159-2 threats and the corresponding HAZOP guide words.

EN 50159-2 threat	HAZOP guide word
Repetition (duplication, replication or babbling idiot)	More (in message rate), As well as
Deletion (all or only part of the messages or part of the message content disappear)	No, Part of, Less (in message rate)
Insertion (incorrect messages, for example data from wrong source)	As well as
Incorrect sequence (failure in event ordering of messages, for example due to priority inversion)	Before, After
Corruption	More (in value), Less (in value)
Delay (too long latencies)	Late
Too early messages[2]	Early
Excessive jitter[2]	-[1]
Masquerade (mixing safety-related message with non-safety-related; authentication error)	Other than
Inconsistency[2] (two or more receivers may have inconsistent view of the transmitted data or receivers may be in different states)	Other than
Notes: 1. A new HAZOP guide word, (excessive) fluctuation, could be introduced. 2. Not listed in EN 50159-2.	

As a consequence, we now have a generic "interface" (signal deviations represented by HAZOP guide words) to deal with the communication-related safety-critical errors at application level. The analysis of the communication-related signals does not differ from the analysis of normal signals, except that there may be some deviations that are relevant to communication-related signals but not to normal signals.

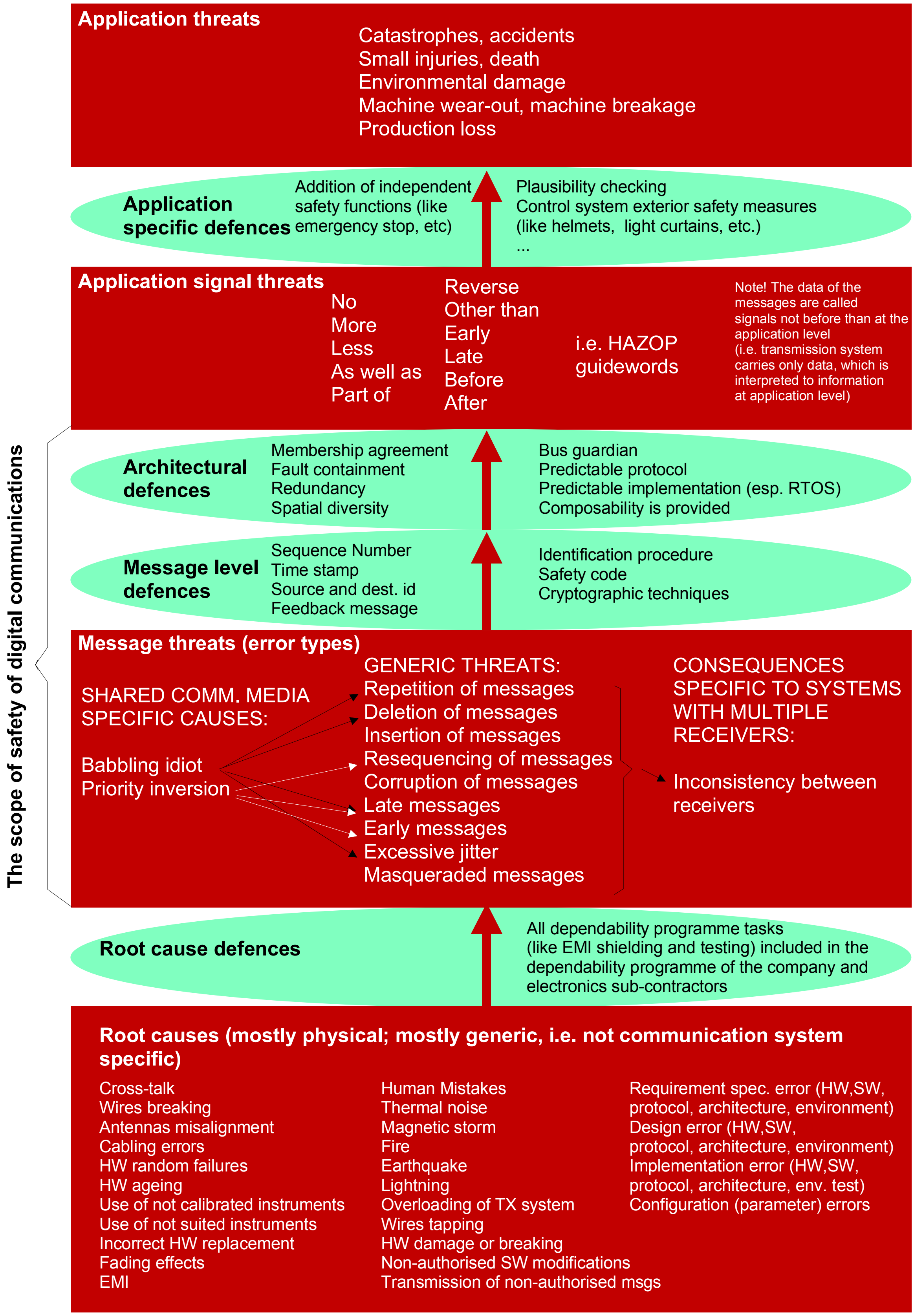

Application threats
Catastrophes, accidents
Small injuries, death
Environmental damage
Machine wear-out, machine breakage
Production loss
Application specific defences
Addition of independent safety functions (like emergency stop, etc)
Plausibility checking
Control system exterior safety measures (like helmets, light curtains, etc.)
...
Application signal threats
No
More
Less
As well as
Part of
Reverse
Other than
Early
Late
Before
After
i.e. HAZOP guidewords
Note! The data of the messages are called signals not before than at the application level (i.e. transmission system carries only data, which is interpreted to information at application level)
Architectural defences
Membership agreement
Fault containment
Redundancy
Spatial diversity
Bus guardian
Predictable protocol
Predictable implementation (esp. RTOS)
Composability is provided
Message level defences
Sequence Number
Time stamp
Source and dest. id
Feedback message
Identification procedure
Safety code
Cryptographic techniques
Message threats (error types)
SHARED COMM. MEDIA SPECIFIC CAUSES:
Babbling idiot
Priority inversion
GENERIC THREATS:
Repetition of messages
Deletion of messages
Insertion of messages
Resequencing of messages
Corruption of messages
Late messages
Early messages
Excessive jitter
Masqueraded messages
CONSEQUENCES SPECIFIC TO SYSTEMS WITH MULTIPLE RECEIVERS:
Inconsistency between receivers
The scope of safety of digital communications
Root cause defences
All dependability programme tasks (like EMI shielding and testing) included in the dependability programme of the company and electronics sub-contractors
Root causes (mostly physical; mostly generic, i.e. not communication system specific)
Cross-talk
Wires breaking
Antennas misalignment
Cabling errors
HW random failures
HW ageing
Use of not calibrated instruments
Use of not suited instruments
Incorrect HW replacement
Fading effects
EMI
Human Mistakes
Thermal noise
Magnetic storm
Fire
Earthquake
Lightning
Overloading of TX system
Wires tapping
HW damage or breaking
Non-authorised SW modifications
Transmission of non-authorised msgs
Requirement spec. error (HW,SW, protocol, architecture, environment)
Design error (HW,SW, protocol, architecture, environment)
Implementation error (HW,SW, protocol, architecture, env. test)
Configuration (parameter) errors

Figure 4. Cause-consequence model for communication related errors.

Figure 4 also includes an insight into the distributed systems specific communication errors by introducing “babbling idiot” fault mode and priority inversion fault mode. These are consequences of design or specification faults. Furthermore, a specific communication error, inconsistency, is depicted to be a consequence of one of the generic message errors. However, inconsistency is included in the list of HAZOP deviations, as it addresses a special error case, which would otherwise be unobserved. In this specific error case, receiving a message correctly and in correct time could manifest an error if another receiver does not receive the message correctly. This error case may not be found in the analysis of a signal just by interpreting the generic message errors (threats) to HAZOP deviations.

The system may also include defences against the application signal deviations in different domains (for example, within the control system or in the working environment). Those defences are application specific and are beyond the scope of this report.

We started at the root causes and ended up at the application level. However, we intend to do the safety analysis of the communication subsystem from the middle of the cause-consequence scenario by analysing the consequences of the deviations of the communications related signals. The analysis is done to find out what signals, and hence messages, may produce safety critical failures. The two flow charts, presented in Figure 5 and Figure 6, illustrate the procedure to do this type of safety analysis.

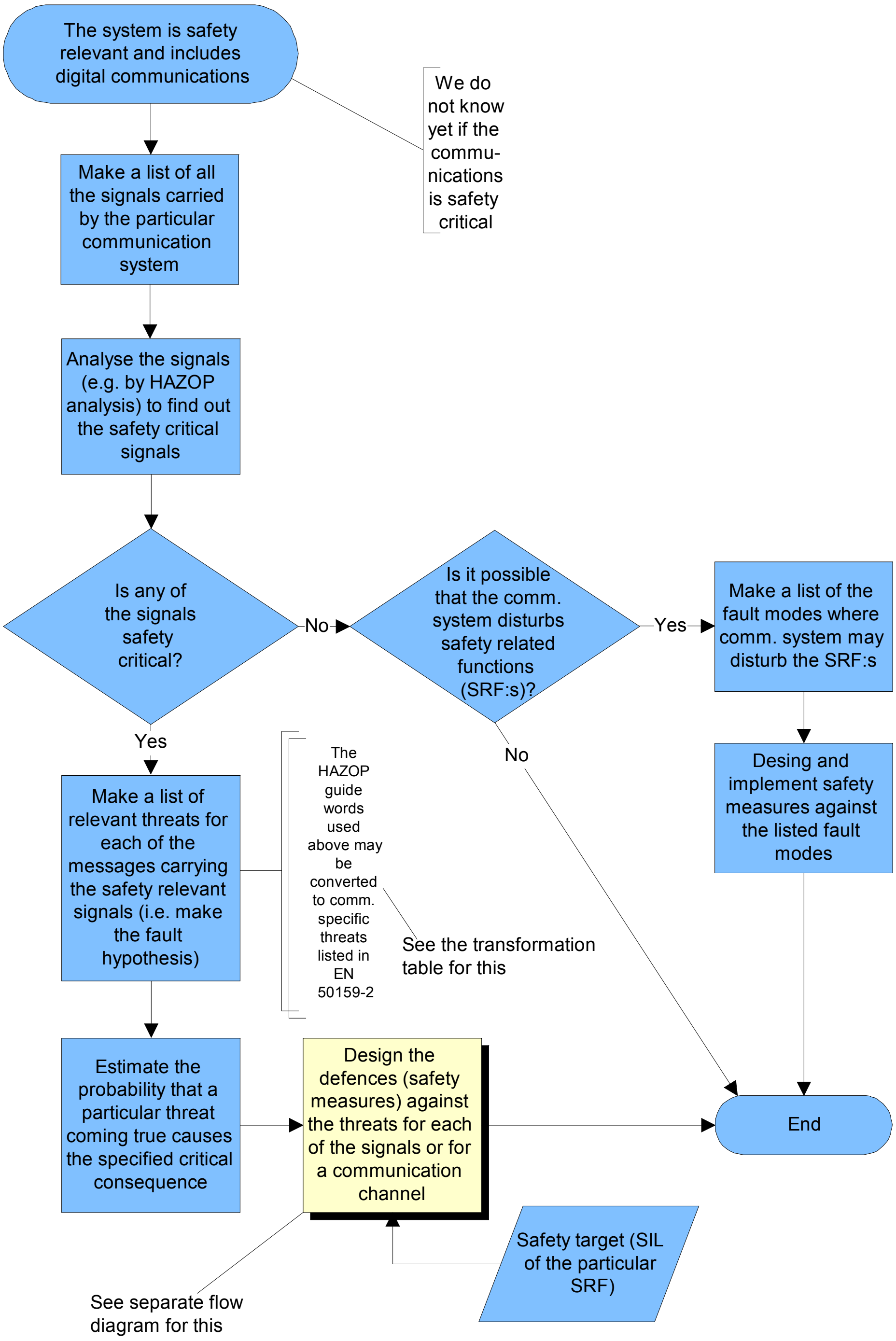

The system is safety relevant and includes digital communications
We do not know yet if the commu-nications is safety critical
Make a list of all the signals carried by the particular communication system
Analyse the signals (e.g. by HAZOP analysis) to find out the safety critical signals
Is any of the signals safety critical?
No
Is it possible that the comm. system disturbs safety related functions (SRF:s)?
Yes
Make a list of the fault modes where comm. system may disturb the SRF:s
Yes
No
Make a list of relevant threats for each of the messages carrying the safety relevant signals (i.e. make the fault hypothesis)
The HAZOP guide words used above may be converted to comm. specific threats listed in EN 50159-2
See the transformation table for this
Desing and implement safety measures against the listed fault modes
Estimate the probability that a particular threat coming true causes the specified critical consequence
Design the defences (safety measures) against the threats for each of the signals or for a communication channel
End
Safety target (SIL of the particular SRF)
See separate flow diagram for this

Figure 5. Safety analysis of communication-related application signals (continues in Figure 6).

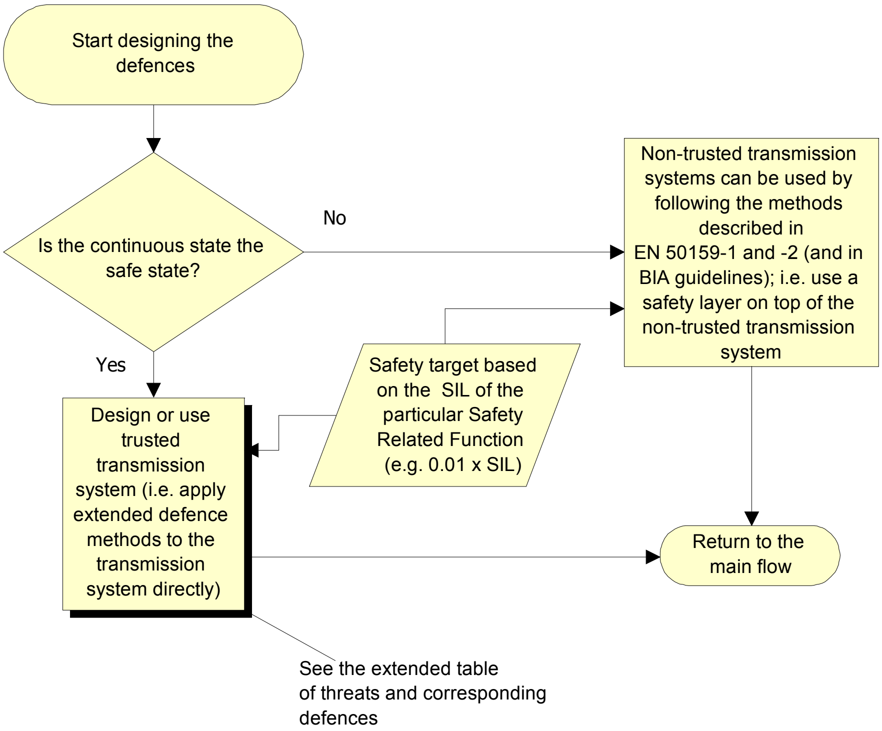

Start designing the defences
Is the continuous state the safe state?
No
Non-trusted transmission systems can be used by following the methods described in EN 50159-1 and -2 (and in BIA guidelines); i.e. use a safety layer on top of the non-trusted transmission system
Safety target based on the SIL of the particular Safety Related Function (e.g. 0.01 x SIL)
Yes
Design or use trusted transmission system (i.e. apply extended defence methods to the transmission system directly)
Return to the main flow
See the extended table of threats and corresponding defences

Figure 6. Designing the defences against threats in communication system (continued from Figure 5).

If the continuous state of the system is the safe state, the design of the communication system is affected mostly by the set of questions described in Table 2.

Table 2. Questions that most affect the communications architecture if the continuous state is the safe state for at least one of the signals.

Question	Actions, if the answer is YES
Are any of the signals such that unpredictable communication latency is a safety-critical fault mode? (In other words, is the system a hard real-time system?)	Design or use communication protocol and architecture that provides predictability (for example, apply time-triggered communications)
Must a single fault be tolerated?	Increase reliability by redundancy to support fault-tolerance
Do the communication players use shared media?	Consider how to defend against “babbling idiot” and priority inversion error types
Is the system a distributed system? (In other words, there are multiple receivers for a message. Note! communications can be pure point-to-point also in networks with multiple nodes)	Apply membership control and inconsistency control (if inconsistency is a safety-critical fault mode)

The threats and corresponding defences referred to in Figure 6 are presented in Table 3. Table 4 includes descriptions of these defence methods as well as threats against which each defence method can be used.

Table 3. Extended EN 50159-2 defence methods.

DEFENCE THREAT[1]	Sequence number	Timestamp	Time out	Source and destination identifier	Feedback message (acknowledgements)	Identification procedure	Safety code (CRC)	Cryptographic techniques	Redundancy (replication)	Membership control	Atomic broadcast	Apply time-triggered architecture	Apply bus guardian	Prioritisation of messages	Inhibit times	Hamming distance applied to node addresses or message identifiers
Repetition	•	•							♦			♦	♦		♦	
Deletion	•		♦		♦				♦			♦				
Insertion	•			•	•	•	♦[2]		♦							♦
Incorrect sequence	•	•							♦			♦				
Corruption					♦		•	•	♦							
Late		•	•		♦							♦		♦	♦	
Early		•										♦				
Excessive jitter		♦										♦		♦	♦	
Masquerade					•	•	♦[2]	•								♦
Inconsistency										♦	♦					

Notes:

1. ♦ Are not supplied by EN 50159-2.

2. Valid, if the CRC calculation includes data that is not in the message itself, but is known by the transmitter and receiver(s) a priori (for example, a message key and expected send time).

Table 4. Descriptions of the defence methods against threats. [31, 3, 27, 11]

Defence method	Description	Used against this threat
Sequence number	Each message has a consecutive number. In the simplest case the message includes a toggle bit.	Repetition, deletion, insertion, incorrect sequence
Time stamp	Each message has a time code, which describes the sending time.	Repetition, incorrect sequence, delay
Timeout (for example, watchdog)	Receiver accepts messages only when they arrive in time or during a predefined time window. Usually exception handling is used to react upon delayed messages.	Deletion, delay
Source and destination identifier	Each message has a source and/or destination address or other code.	Insertion
Feedback message (acknowledgements and echoes)	After receiving a message the module sends a positive or negative acknowledgement or after receiving a message the module sends the whole message or a checksum back.	Insertion, masquerade
Identification procedure	The members of the network check the identity of the other members prior to the start of the system or prior to the transmission of a specific message. Identity may include, for example, information about software and hardware versions.	Insertion, masquerade
Safety code (for example, CRC cyclic redundancy check)	The method adds into the message a checking code; also other types of data consistency checks are available.	Corruption
Cryptographic techniques	Authentication is applied and cryptographic code is added to the message to protect against malicious attacks.	Corruption, masquerade
Redundancy (replication):	The messages are transferred periodically even though no changes in values have occurred; a message may be replicated (for example, sent twice with the other message inverted); the communication subsystem may be replicated.	Repetition, deletion, insertion, incorrect sequence, corruption
Membership control	The members of the network monitor each other and execute exception handling in case of malfunction in one of the members.	Inconsistency
Atomic broadcast	Communication protocol with atomic broadcast ensures that all messages are delivered in the same order to all correct processors in the system and all consumers of the data have a consistent view of data (all accept the data or all reject it).	Inconsistency
Time-triggered architecture	Messages are scheduled in regard to time. The time schedule is often pre-fixed by the system designer.	Repetition, deletion, incorrect sequence, corruption, timing errors, excessive jitter
Bus guardian	Transmission of messages is controlled by a hardware that opens and closes the access path for the transmitter to the communication media.	Repetition
Prioritisation of messages	The messages are prioritised to enable safety-critical messages to access the bus with minimum delay.	Late, excessive jitter
Inhibit times	Similar to bus guardian, but can be implemented by software at the communication subsystem; after transmitting a certain message, that particular message is put in "quarantine" for a given period of time before it can be transmitted again by the particular transmitter.	Repetition, late, excessive jitter
Hamming distance applied to node addresses or message identifiers	The node addresses or message identifiers are selected so that any single bit failure in the address or in the identifier produces a non-used address or identifier and can thus be noticed by the receivers.	Insertion, masquerade

Along with transferring run-time process signals, the communication system is used for the following purposes:

- To pass communication related parameters and configuration information (like node numbers and other membership information, priorities, etc.)

- To pass application related parameters and configuration information

- To download programs (software)

- To upload blocks of application related data (like error logs, data acquisition blocks, etc.)

- To control starting, stopping and restarting of the members in the network (network management).

Analysis of such communications with the HAZOP method is not practical but other analysis methods must be applied. Such methods are presented, for example, in the Swedish PALBUS project [33] (see also http://www.sp.se/electronics/RnD/palbus/).

The main scope of this report is to present the safety-related standards that address communications (Chapter 2) and to present the available safety-related communication systems (Chapter 3). The presentation of the safety-related communication systems is based on the concept presented in Figure 4 (the message level defences and architectural defences are discussed separately). The defences suggested in Table 3 are considered in particular.

A short review of safety-related wireless communication is also provided (Chapter 4). In Chapter 0, a HAZOP based analysis method and analysis tool for digital communications is presented.

2. Related standards and guidelines

2.1 EN 50159

EN 50159 is a member of a set of railway safety standards. The scope of EN 50159 is safety-related communication in the context of railway systems. To better illustrate the context of EN 50159, some of the most important railway safety standards are listed in Table 5.

Table 5. EN 50159 related railway safety EN standards.

EN number	Title	Scope
EN 50126	Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS)	Complete railway system
EN 50129	Railway applications - Safety-related electronic systems for signalling	Complete signalling system
EN 50128	Railway applications - Software for railway control and protection systems	Individual subsystem or complete signalling system
EN 50159	Railway applications - Communications, signalling and processing systems Part 1: Safety-related communication in closed transmission systems Part 2: Safety-related communication in open transmission systems	Complete railway system

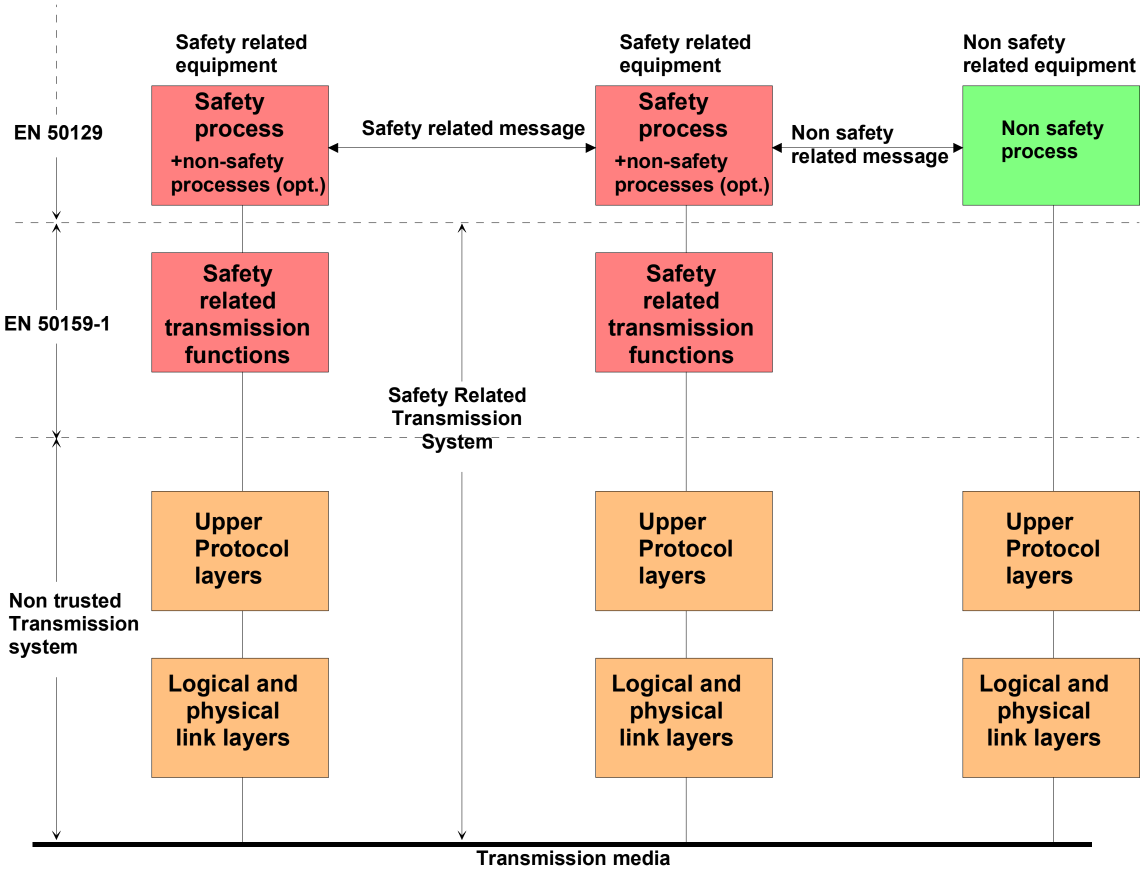
Application of EN 50159 requires adherence to EN 50126 and EN 50129 in the case of railway applications, but EN 50159 is also practical in general cases, for example, in machine automation. In fact, EN 50159 is often referenced in literature and papers dealing with the safety of digital communications beyond the scope of railway applications.

EN 50159 covers safety-related communication on top of a closed (part 1) or open (part 2), non-trusted, transmission system. It does not set any safety requirements on the underlying transmission system. It does, however, define an additional safety layer on top of the transmission system that has to take care of the safety precautions (defences), like additional CRC checks or time stamping, to reach the specified safety integrity level. EN 50159 does not clearly state the level of the safety layer in regard to the OSI model. Therefore, the safety layer services may be implemented in the application layer or in the underlying layers.

EN 50159 presumes that a safe state can be defined for the system. The safe state is entered if the safety layer indicates a fatal communication error. Transmission system is not used to enter the safe state. EN 50159 standards have also been issued as IEC standards with the number, IEC 62280.

2.1.1 EN 50159-1

Figure 7 presents the layered architecture model of EN 50159-1 (closed transmission systems). [12]



Safety related equipment
Safety related equipment
Non safety related equipment
EN 50129
Safety process +non-safety processes (opt.)
Safety related message
Safety process +non-safety processes (opt.)
Non safety related message
Non safety process
EN 50159-1
Safety related transmission functions
Safety related transmission functions
Safety Related Transmission System
Non trusted Transmission system
Upper Protocol layers
Upper Protocol layers
Upper Protocol layers
Logical and physical link layers
Logical and physical link layers
Logical and physical link layers
Transmission media

Figure 7. EN 50159-1 communication model using non-trusted closed transmission system [12].

According to EN 50159-1, a transmission system is considered closed if it fulfils the following conditions:

- Only approved access is permitted.
- The maximum number of connectable participants is known.
- The transmission media is known and fixed.

EN 50159-1 focuses on the safety-related transmission functions, as depicted in Figure 7, not on the underlying transmission system or on the application-related issues like data contents or device types. The safety-related transmission functions are supposed to add a safety code to a message (for example, an additional CRC, different from that of the non-trusted transmission system) and safety procedures (like source identifier, time stamps and sequence counts). The safety code is a quantitative measure and the safety procedures are qualitative measures. In other words, for the safety code a mathematical analysis must be made to ensure that its residual error probability meets the safety integrity requirement (SIL)[1] with the given message rate. EN 50159-1 suggests using the simple pessimistic equation to calculate the residual error probability of a safety code:

$$P_{US} = 2^{-C}, \text{ where} \tag{1}$$

P_{US} = residual error probability of the safety code, and

C = the length of the safety code in bits.

In some applications, the pessimistic residual error probability may be impractical and a more realistic figure for the residual error probability may be needed. In that case, the bit error ratio (BER) of the communication channel must be known and monitorable. The mathematical analysis of the safety code reveals the interdependece between BER and the residual error probability. However, care should be taken to also consider the effect of a synchronisation slip error, if message frames are separated from each other by monitoring fixed 'frame start' -marks; a synchronisation slip will cause a random number of bit errors making the BER value virtually much higher. For example, the common HDLC protocol is susceptible to a synchronisation slip; even a single bit error may cause a synchronisation slip that is not detected by the receiver of the frames.

The other safety procedures, like source identifier, are qualitative measures, and therefore quantitative SIL requirements cannot be specified for them. Instead, the SIL requirement is met by applying appropriate procedures defined in EN 50129. In a general case (where EN 50129 is not applicable), IEC 61508 or its application specific derivative could be used[2].

The safety layer consumes some bytes of the available data bytes of a message. A safety message consists of the normal transmission frame plus the safety code and some bytes for other safety procedures like source identifier and sequence count (see Figure 8).

[1] Normally, one percent of the total safety budget is allocated for the safety code. For example, if the SIL requirement is $< 10^{-7}$ critical dangerous failures per hour, the requirement for the critical message error rate is $< 10^{-9}$ 1/h. The critical error rate depends on the rate of critical messages and on the residual error probability.

[2] This is a suggestion by the authors of this report and is not stated in EN 50159-1.

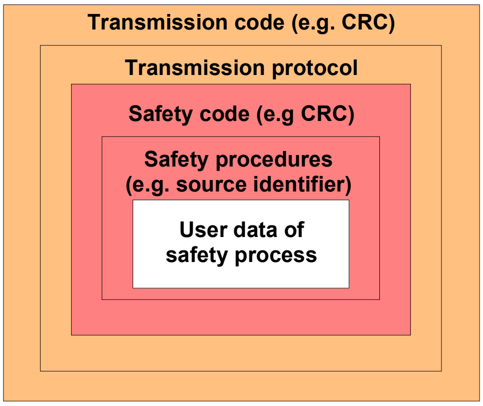


Figure 8. The EN 50159-1 model of a safety-related message.

EN 50159-1 is not as comprehensive as EN 50159-2 in supplying the fault hypothesis (expected communication threats) and the list of defences against the communication errors. EN 50159-1 and EN 50159-2 could be merged together to form a single standard with the majority of the text coming from EN 50159-2; the authors of this report recommend to adopt the guidelines of EN 50159-2, where applicable, also in case of closed transmission systems of machine automation.

2.1.2 EN 50159-2

EN 50159-2 deals with safety-related communications on top of an open transmission system. Open transmission systems bring about an additional communication threat: authentication error. Therefore, in addition to the safety layer, a “security layer” called Safety-related Access Protection Process is introduced (see Figure 9).

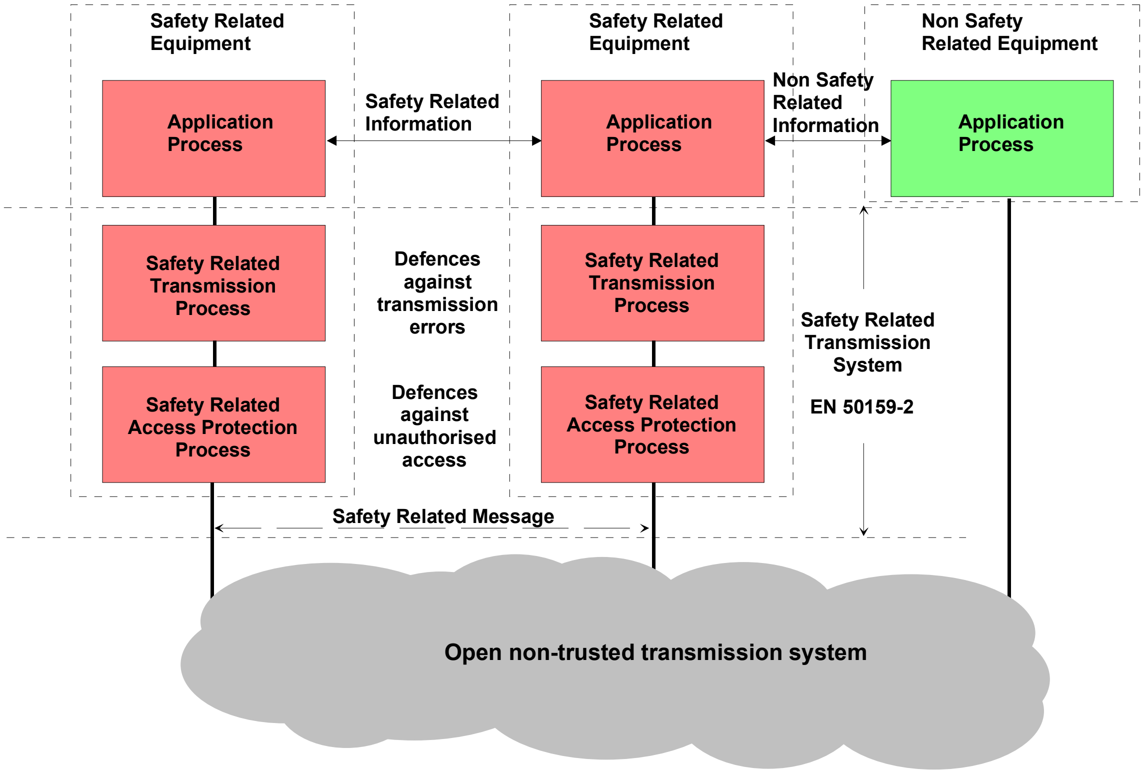

Safety Related Equipment
Safety Related Equipment
Non Safety Related Equipment
Application Process
Safety Related Information
Application Process
Non Safety Related Information
Application Process
Safety Related Transmission Process
Defences against transmission errors
Safety Related Transmission Process
Safety Related Transmission System
EN 50159-2
Safety Related Access Protection Process
Defences against unauthorised access
Safety Related Access Protection Process
Safety Related Message
Open non-trusted transmission system

Figure 9. EN 50159-2 communication model using non-trusted open transmission system.

The safety-related message includes an additional portion for the access protection procedure (see Figure 10).

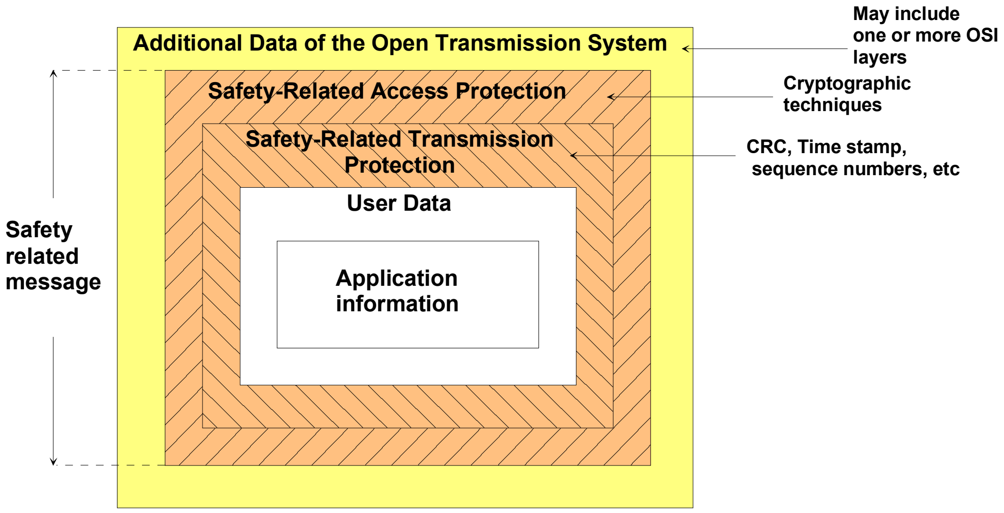

Additional Data of the Open Transmission System
May include one or more OSI layers
Safety-Related Access Protection
Cryptographic techniques
Safety-Related Transmission Protection
CRC, Time stamp, sequence numbers, etc
User Data
Application information
Safety related message

Figure 10. The EN 50159-2 model of a safety-related message.

EN 50159-2 discusses the fault hypothesis of digital communications more comprehensively than EN 50159-1 and supplies a “standardised” set of communication threats and a “standardised” set of defences against the threats (see Table 6). The threats as wells as the defences are discussed in detail so that the communication system designer can find practical work guidance. The access protection issues especially are dealt with at length.

Table 6. Threats and defences matrix of EN 50159-2.

	Defences							
Threats	**Sequence Number**	**Time stamp**	**Time out**	**Source and destination Identifier**	**Feedback message**	**Identification Proc.**	**Safety code**	**Cryptographic techniques**
Repetition	X	X						
Deletion	X							
Insertion	X			X [2]	X [1]	X [1]		
Resequence	X	X						
Corruption							X [3]	X
Delay		X	X					
Masquerade					X [1]	X [1]		X [3]

Notes:

1) Application dependent

2) Only applicable for source identifier

Will only detect insertion from invalid source

If unique identifiers cannot be determined because of unknown users, a cryptographic technique shall be used, see clause 6.3.8 of EN 51059-2

3) See section 7.3 and annex A2 (of 50159-2)

Along with the threats and defences, EN 50159-2 supplies a list of hazardous events (root causes) that may cause the listed threats. An amplified list of the root causes can be found in Figure 4 in Chapter 1 of this report

2.1.3 Considerations for machine automation

As stated earlier, EN 50159 is referenced well beyond railway systems and is also applicable in machine automation, especially as IEC 61508-2 will (and does already) make reference to EN 50159 (see Chapter 2.3). EN 50159 sets a general fault hypothesis (which must be extended in the case of distributed real-time systems) and is well suited as a guideline for systems, where a steady safe state can be defined. For systems, where the continuous operation is the safe state (like steer-by-wire systems with relatively high vehicle speed), EN 50159 is not applicable alone, but the complete transmission system must be designed to meet the high availability requirements and the consequent SIL level. In practice, this leads to systems with redundant communication media, protocol chips and possibly redundant modules.

It is also recommended here to adopt EN 50159-2 in the case of a closed transmission system and harmonise it with the special requirements or mitigations from 50159-1 on closed transmission. Therefore, EN 50159-2 is also used as the basic building block in handling safety-related communication in KETU-project. (This report is the result of KETU-project.) Chapter 1 provides an insight as to how EN 50159-2 is exploited in the KETU-project.

Furthermore, the whole set of railway equipment safety standards listed in Table 5 is recommended study, if similar standards for the particular machine application do not exist. The life cycle model and the example of a RAMS (Reliability, Availability, Maintainability and Safety) specification presented in EN 50126, in particular, are also valuable in the case of machine automation.

2.2 BIA Guidelines

The German organisation BIA (Berufsgenossenschaftlichen Instituts für Arbeitssicherheit, Institute for Occupational Safety of Accident Insurance Institutions) has issued guidelines for the test and certification of safety-relevant digital communication buses [16]. The BIA guidelines are well referenced and are currently the only practical guideline to assess safety relevant digital bus systems for certification to comply with the European Union Machine Directive (89/392/EEC).

The BIA guidelines support four different architecture models. The models are presented in Figure 11.

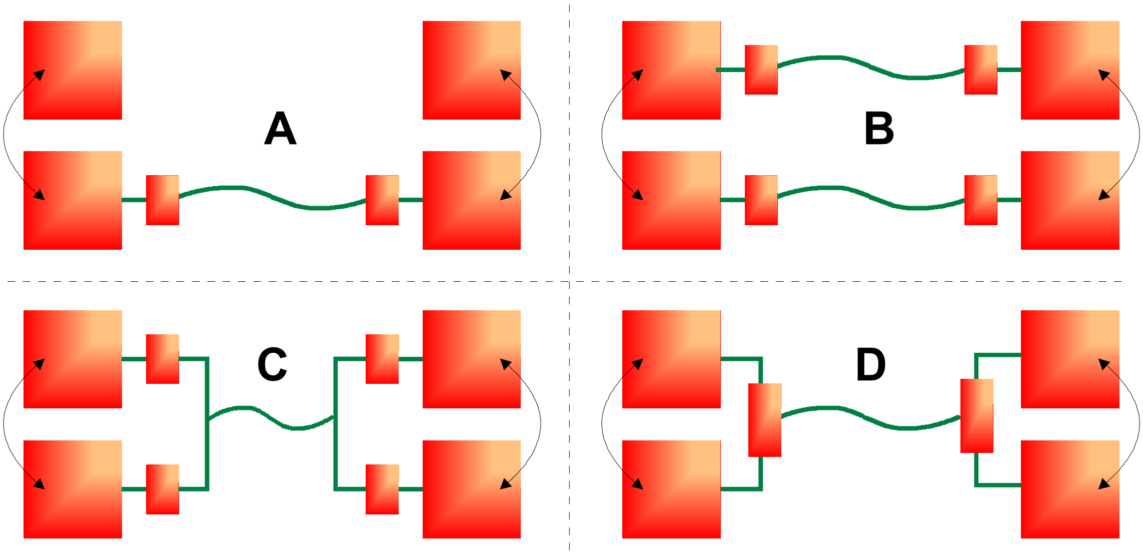


Figure 11. Architecture models of BIA guidelines.

Explanations of the models are as follows:

- Model A: A single channel system
- Model B: A two channel system
- Model C: A two channel system, but the transmission media is not duplicated
- Model D: Dual safety layers, but single transmission system

After introducing the architecture models, the BIA guidelines supply a list of transmission errors and a list of defence methods against the transmission errors. The lists are practically the same as those in EN 50159-2 (see table Table 6). The list of defence methods provides only a short insight into the typical (and recommended) abatement methods; except for the data integrity mechanism a more through discussion is provided. Equations for three data integrity mechanism cases are supplied:

1. Data integrity assurance for models A and D, where the CRC (Cyclic Redundancy Check) of the underlying mechanism is not taken into account, but an additional safety code (for example, CRC as well) is assumed to contribute all the abatement against corruption errors.

The following equation is defined for the failure rate of critical corruption errors:

$$\Lambda = 3600R(p)v(m-1)\times 100\text{, } \textit{where} \tag{2}$$

Λ = the failure rate (1/h) of critical corruption errors multiplied by factor of 100 to leave room for other safety-critical failures,

v = number of safety-relevant messages per second,

R(p) = residual error probability of the safety code (CRC of the safety layer) and

m = number of safety-relevant nodes.

2. Data integrity assurance for models B and C, where the CRC of the underlying mechanism is taken into account and the message is sent twice (once via each channel).

The following equation is defined for the failure rate of critical corruption errors:

$$\Lambda = 3600 R(p)^2 v(m-1) \times 100 \text{, } \textit{where} \qquad (3)$$

Λ = the failure rate (1/h) of critical corruption errors multiplied by factor of 100 to leave room for other safety critical failures,

v = number of safety-relevant messages per second,

R(p) = residual error probability of the transmission code (CRC of the transmission layer) and

m = number of safety-relevant nodes.

3. Data integrity assurance for models A and D, where the CRC of the underlying mechanism is taken into account and an additional safety code (for example, CRC as well) is also applied.

The following equation is defined for the failure rate of critical corruption errors:

$$\Lambda_{Sys} = \Lambda_{HW} + \Lambda_{EMI} + \Lambda_{TC} < \Lambda_{CORRUPTION,TARGET} = \frac{\Lambda_{SRF,TARGET}}{100} \text{, } \textit{where} \qquad (4)$$

Λ_{SYS} *= Critical failure rate (1/h) of corruption errors,*

Λ_{HW} *= Failure rate (1/h) of transmission system (HW) fault (without CRC checker fault),*

Λ_{EMI} *= Failure rate (1/h) of EMI caused faults that are not noticed by the CRC,*

Λ_{TC} *= Failure rate (1/h) of transmission code checker faults,*

$\Lambda_{CORRUPTION,TARGET}$ *= upper bound for the maximum critical failure rate (1/h) of corruption errors and*

$\Lambda_{SRF,TARGET}$ *= upper bound for the critical failure rate (1/h) of the corresponding Safety-related Function (SRF).*

$$\Lambda_{HW} = \lambda_{HW} R_{US} \text{, } \textit{where} \qquad (5)$$

λ_{HW} *= 1/MTTF (Mean Time To Failure) and*

R_{US} = maximum residual error probability of the superimposed safety code (for example, additional CRC at safety layer).

$$\Lambda_{EMI} = f_w R_{UB} R_{US}\text{, } where \tag{6}$$

f_w = occurrence of corrupted messages on the transmission system,

R_{UB} = residual error probability of the commercial bus system and

R_{US} = maximum residual error probability of the superimposed safety code (for example, additional CRC at safety layer).

$$\Lambda_{TC} = R_{US} \times k \times 1/T\text{, } where \tag{7}$$

R_{US} = maximum residual error probability of the superimposed safety code (for example, additional CRC at safety layer),

k = the relation of the hardware failures of the code checking mechanism to the whole hardware failures of the communication chip, and

T = Time span, if more than a well-defined number of corrupted messages were received within this time, the safe fall back state will be entered.

For other defence methods, BIA guidelines do not provide such a quantitative assessment. This is understandable, as it is difficult to calculate failure rates for the other defence methods, which are more qualitative in nature. Functional calculations, of course, have to be performed, for example, for the timeout defence method and for the subsequent action latencies.

After supplying a set of general requirements to overcome the transmission errors, the BIA guidelines concentrate on setting the environmental test specifications for the bus devices.

The BIA guidelines are best suited for a vendor who supplies safety buses with its devices and wants to tag its products with an EN 954-1 category label. However, the equations to calculate the residual error rates for the corruption errors are also particularly applicable also for a work machine manufacturer. This ensures the conformity of the data integrity mechanism of the manufacturer's embedded communication bus with the relevant requirements derived, for example, from Machine Directive and its accompanying standards.

The BIA guidelines are best apprehended in conjunction with EN 50159-1 and EN 50159-2 (see Chapter 2.1).

2.3 IEC 61508-2 Work

IEC 61508-2 [23] addresses communication issues very lightly. Data communications are addressed primarily in Chapter 7.4.8 (Requirements for data communications) of IEC 61508-2. However, as Chapter 7.4.8 is short it is printed below:

"7.4.8 Requirements for data communications

7.4.8.1 When any form of data communication is used in the implementation of a safety function then the probability of undetected failure of the communication process shall be estimated taking into account transmission errors, repetitions, deletion, insertion, re-sequencing, corruption, delay and masquerade (see also 7.4.8.2). This probability shall be taken into account when estimating the probability of dangerous failure of the safety function due to random hardware failures (see 7.4.3.2.2).

NOTE The term masquerade means that the true contents of a message are not correctly identified. For example, a message from a non-safety component is incorrectly identified as a message from a safety component.

7.4.8.2 In particular, the following parameters shall be taken into account when estimating the probability of failure of the safety function due to the communication process:

a) the residual error rate (see IEV 371-08-05);

b) the rate of residual information loss (see IEV 371-08-09);

c) the limits, and variability, of the rate of information transfer (bit rate);

d) the limits, and variability, of the information propagation delay time.

NOTE 1 It can be shown that the probability of a dangerous failure per hour is equal to the quotient of the residual error probability and the message length (in bits) multiplied by the bus transmission rate for safety-related messages and a factor of 3600.

NOTE 2 Further information can be found in IEC 60870-5-1 and in EN 50159-1 and EN 50159-2."

(It can be seen that this particular chapter refers to IEC 60870-5-1 [22] and EN 50159 for further guidance.)

As a consequence of this superficial discussion of digital communications, a task group IEC SC65A MT13/TG1 was formed to:

- "review the requirements of IEC 61508 in respect to digital communication systems

- assess whether the existing requirements in IEC 61508 are sufficient

- develop proposals on what guidance is needed to facilitate the use of digital communication systems in E/E/PE safety-related systems and their compliance with IEC 61508". (Citation from IEC SC65A MT13/TG1 report, January 2002.)

It should be noted that a communication system is a combination of hardware and software. Therefore, in principle, IEC 61508-2 (hardware) and IEC 61508-3 (software) methods for designing safety into the safety-related systems should suffice to cover communication systems as well without any special conduct. However, as there is also a need to use commercial, non-trusted, fieldbuses, LANs etc. in safety-critical systems, the approach of EN 50159 (see Chapter 2.1) is also advocated. The suggestion of IEC SC65A MT13/TG1 is to support the two approaches (Figure 12 and Figure 13) towards safety-related data communications (note that EN 50159 was also referenced in the original version of IEC 61508-2):

1. *"All subsystems (hardware and software) used by the communication process meet the relevant requirements of IEC 61508. The term 'white channel' was adopted to describe this approach. In this case, the requirements of IEC 61508-2 and IEC 62508-3 will ensure that the likelihood of dangerous failures of the communication process due to both transient errors (caused by e.m.i. etc.) and random hardware failures are sufficiently low in relation to the safety integrity level of the safety function."* (Citation from the July 9th, 2002 minutes of the IEC SC65A MT13/TG1 meeting.)

This approach is necessary in the case of fail operational systems (in other words, in systems where the continuous operation is the only safe state and thus fault tolerant operation is required).

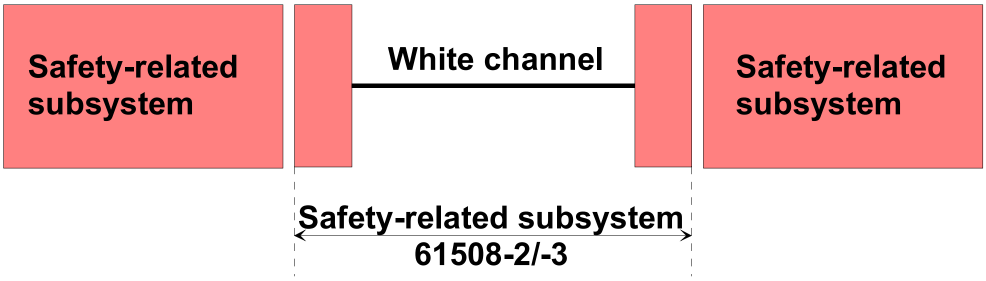


Figure 12. IEC SC65A MT13/TG1 approach 1 illustrated.

2. *"All measures necessary to assure safety-related data transmission are implemented in the subsystems communicating with each other (sender and receiver). There are no safety specific requirements on any other parts of the communication system. The term 'black channel' was adopted to describe this approach. In this case, the measures necessary to ensure the safety integrity of the data communication process shall be implemented in the E/E/PE safety-related subsystems, which interface with the communication channel. This is the approach of EN 50159-1,-2 (in voting for publication as IEC 62280). In this case there needs to be an explicit requirement in IEC 61508-2 to include the probability of dangerous failure of the communication process in the overall estimation of the probability of failure of a safety function."* (Citation from the July 9th, 2002 minutes of the IEC SC65A MT13/TG1 meeting.)

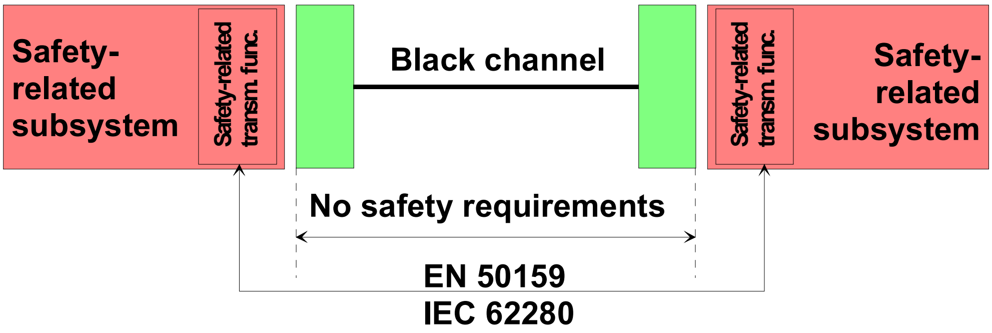


Figure 13. IEC SC65A MT13/TG1 approach 2 illustrated.

Although IEC SC65A MT13/TG1 developed a guideline document to be issued as an additional annex or guideline document for IEC 61508, the final decision was not to issue such guidelines, only to update the necessary chapters (mainly Chapter 7.4.8) of IEC 61508-2. The suggested contents for the updated Chapter 7.4.8 are as follows:

7.4.8 Additional requirements for data communications

7.4.8.1 When data communication is used in the implementation of a safety function then the failure measure of the communication process shall be estimated, taking into account *transmission errors, repetitions, deletion, insertion, re-sequencing, corruption, delay and masquerade. This failure measure shall be taken into account when estimating the failure measure of the safety function due to random failures (see 7.4.3.2.).*

NOTE The term masquerade means that the true contents of a message are not correctly identified. For example, a message from a non-safety component is incorrectly identified as a message from a safety component.

7.4.8.2 The measures necessary to ensure the required failure measure of the communication process (see 7.4.8.1) shall be implemented according to the requirements of this standard and IEC 61508-3.

This allows 2 possible approaches:

a) the communication channel shall be designed, implemented and validated according to IEC 61508 throughout (so-called 'white channel' see Figure 12), or

b) parts of the communication channel are not designed or validated according to IEC 61508 (so-called 'black channel' see Figure 13). In this case, the measures necessary to ensure the probability of dangerous failure of the communcation process shall be implemented in the E/E/PE safety-related subsystems which interface with the communication channel in accordance with IEC 62280 / EN 50159-1,-2 as appropriate.

Some updates on Chapters 7.4.2.2 and 7.4.3.2 and on Table A.1 of IEC 61508-2 were also suggested. The updates will be minor.

IEC SC65A MT13/TG1 supplies no specific requirements for certification of safety-related buses or other communication systems, but the normal IEC 61508 certification procedures are applicable.

2.4 EWICS Guidelines

The organisation EWICS TC7 (European Workshop on Industrial Computer Systems Technical Committee 7) works in the field of Programmable Electronic Systems safety, reliability and security. The membership of EWICS covers representatives from regulators, industrial users, researchers and members of standards committees. It has members from the most European countries, covering various fields of interests and affiliations. The aim of EWICS is to assess the state-of-the-art methods and tools for critical software development and maintenance in industrial environments. In addition, the aim of EWICS is to develop standards and guidelines for the development and assessment of safe and secure systems, and to disseminate information and knowledge in this field.

EWICS TC7 has produced a guideline on achieving safety in distributed systems [1]. This guideline concentrates on safety-critical distributed systems, which means systems that may have catastrophic consequences to their embedding environments. The objective of the guideline is to provide guidance on achieving safety in industrial computer-based distributed systems over the system life cycle. It is intended to assist those involved in safety-related distributed systems throughout their life cycles, to be used with other guidance and standards. Among other things, the guideline provides assistance on how to develop distributed systems that conform to IEC 61508 and IEC 61131 standards.

The guideline is divided into three parts. The first part introduces the guideline, including definitions. The intention of the second part is to review the activities of a distributed system life cycle and, for each activity, expose those aspects, which particularly have significant safety-related implications. The third part includes generic aspects of distributed systems, which have impact on safety. Within each section of part 2 and part 3 the following items are identified:

- Safety Aspects: Which issues are of particular interest.
- Constraints: What are the design, development and exploitation limits and restrictions.
- Qualities: How to assess that the work has been done well.
- Guidelines: Guidelines and hints on how to do it.

As an example of the contents of the guideline, the system requirements specification part is described in the following chapters.

2.4.1 Safety Aspects

The following aspects are presented in the EWICS guideline:

- Identification of the safety functions to be performed by the system.

- Identification of the functional architecture of the safety-related part of the system, which shows the interrelationships between the safety functions. This is the logical architecture, which is to be mapped into the physical architecture of the system.

- Identification of the extent to which operator(s) and maintainer(s) can participate in fulfilling the safety-critical functions.

- Identification of the start-up and shut-down requirements as they relate to safe operation of the system.

- Identification of safe states.

- For each safety-related function determination, of:

  - the logic executed by the function,

  - the criticality of the function (from the system hazard analysis),

  - integrity, confidentiality and availability of the function and associated information,

  - reliability requirements,

  - fail safety requirements, and

  - analysis of interaction with other functions.

- Identification of the map between safety functions and the partially specified architecture (see the constraints in chapter 2.4.2 below).

- Safety requirements for initialisation and termination of the system should be explicitly addressed.

2.4.2 Constraints

The requirements may need to encompass multiple levels of abstraction. This may result from:

- Domain standards (for example, certain standards may mandate the use of component redundancy to meet particular levels of criticality, or they may even prescribe a specific architecture).

- The development being performed may be a “retrofit” or modification to an existing system, and therefore the existing architecture has to be used.

- Physical distribution of the equipment under control (EUC) to be controlled by the system may require the corresponding distribution of the system.

- Consideration should be given to use of specific architectures on the basis of previous experience, for example, it may be easier to get a specific architecture certified if the developers can say “it's just like the one we did before but with these small changes” (so called ‘design patterns’).

- The distribution of the system architecture may be already (partially) defined by earlier design decisions (for example, in case when the spiral lifecycle model is being followed).

- Project scope should be broad enough to include safe states.

2.4.3 Qualities

For a common set of requirements specification qualities like completeness, consistency, unambiguity, verifiability, see for example, [30]. In addition:

- If the system is an enhancement to an existing one make sure that:

    - the requirements specifications of the existing part meets the level of the safety integrity required from the new system,

    - the new safety requirements of the (extended) system are consistent with existing ones, and

    - the new safety requirements are verifiable with respect to the existing requirement.

- The scope of the safety requirements should be sufficiently general to cover all identified system hazards.

2.4.4 Guidelines

For guidance on how to identify safety functions refer to for example, [30]. In addition:

- Analyse the map between safety functions and what is known of the distributed architecture. This should focus on avoiding hazardous event sequences and reducing associated risk. For example, this will include descriptions of the ways functions are realised by distributed sub-functions and the synchronisation characteristics of the sub-functions and associated failure modes. This analysis should be driven by the classification of motivations for the introduction of system distribution (and associated hazards).

- It should be possible to find a map from the safety functions to components in the partially defined architecture, which enables demonstration that the required safety levels and other safety attributes can be achieved.

2.5 ISO work relating to earth-moving machines

International Organisation for Standardisation (ISO), technical committee 127 and sub-committee 3, in particular, is being prepared a standard concerning Machine Control Systems (MCS) of earth-moving machinery. This is a draft international standard 15998.2, which describes performance criteria and tests for MCS used in earth-moving machines. The standard sets out guidance for systems that are comprised of electrical, electronic or programmable electronic components. The Annex D of this standard includes requirements for bus-systems for the transmission of safety-related messages. [26]

3. Presentation of Off-the-Shelf Implementations

3.1 DeviceNet Safety

3.1.1 Description

DeviceNet Safety is a recent initiative to facilitate DeviceNet with safety features. Open DeviceNet Vendors Association, ODVA (http://www.odva.org), carries out the work. The basic DeviceNet protocol is maintained, but additional safety features are defined for the safety devices. A DeviceNet network can include both normal DeviceNet devices and DeviceNet Safety devices. The topology and the communication media are not affected by DeviceNet Safety. A DeviceNet Safety network can consist of up to 64 nodes. Latency times of 20 ms, from input capture to output actuation, are reported to be possible.

DeviceNet Safety boasts of suitability to Category 4 of EN 954-1 or SIL3 of IEC 61508 with 1% safety budget consumption. To achieve such a high level of safety, redundancy must be applied for input capture, control program execution and for output actuation. Therefore, starting from the CAN controller, all hardware (including the controller CPU) must be replicated either by using two (or more) devices or by using two-channel devices. The cable and the transceivers are not replicated. Due to single channel transmission, a basic DeviceNet Safety network does not support systems where the continuous state is the safe state (in other words, where a steady safe state cannot be identified). It is, of course, possible to replicate such DeviceNet Safety networks to introduce replicated communication media and therefore increase reliability performance and tolerance of a single fault to a level, which might enable the usage of DeviceNet Safety in safety-related systems with no steady safe state (see the example architecture D in Figure 14).

Besides redundancy, to achieve the required reliability, DeviceNet Safety applies an additional safety protocol layer on top of the normal DeviceNet. The safety protocol consumes two bytes of the maximum eight DeviceNet data bytes. The two trailing bytes, in other words, 16 bits, are used for a sequence count (2 bits) and for a redundancy check (CRC-S1, 12 bits). The remaining 2 bits are reserved for future purposes.

A single DeviceNet Safety network can embrace several independent safety circuits or safety chains as they are called in the DeviceNet introduction [34].

3.1.2 Fault Hypothesis

DeviceNet Safety applies the fault hypothesis of EN 50159-2 [13].

3.1.3 Message Defences

Table 7 provides the abatement methods of DeviceNet Safety against the standard set of message errors.

Table 7. Message Defences in the context of DeviceNet Safety.

Against	Description
Repetition	Sequence count (with 2 bits)
Deletion	Time expectation (periodic transmission) Acknowledgement of safety-critical messages Time out Re-sending
Insertion	Sequence count (with 2 bits)
Incorrect sequence	Sequence count (with 2 bits)
Corruption	CRC-S1 (12 bit cyclic redundancy check) Acknowledgement (data is returned to the producer, in other words, to the transmitter) Redundancy and cross-checking of redundant messages
Timing errors	Periodic sending Reply time out Prioritisation (with CAN message identifiers)
Masquerade	Unique safety CRC Unique safety sequence count Unique CAN message identifier Message size checking Automatic checking of duplicate node addresses (in basic DeviceNet protocol)
Inconsistency	None besides the normal consistency control of CAN protocol

3.1.4 Architectural Defences

The basic architectural defence with DeviceNet Safety is redundancy. The architectures presented in Figure 14 are examples of redundant architectures.

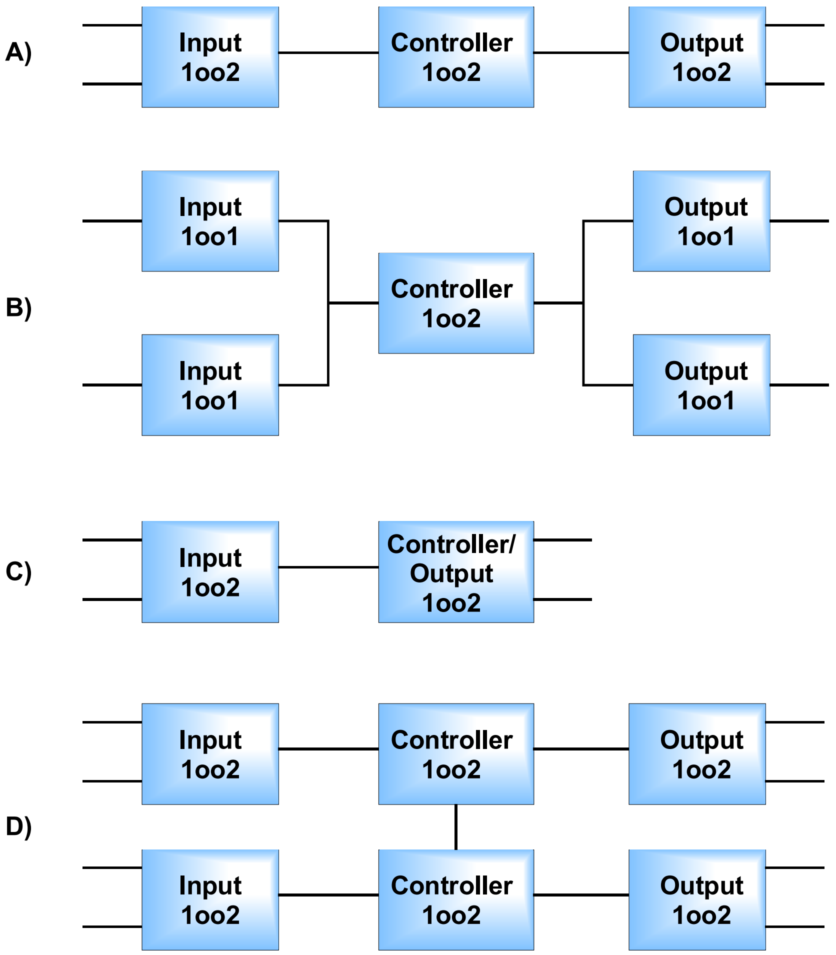


Figure 14. DeviceNet Safety architecture examples (1oo2 means "one out of two"): A) two-channel input, output and controller modules; B) two-channel controller, two one channel input and output modules; C) two-channel input module and a combined two-channel controller and output module (for example, to implement safety stop); D) architecture that tolerates single physical layer faults.

3.1.5 Off-the-Shelf Availability

Three global industrial automation and machine safety companies, Rockwell Automation, OMRON Corporation and SICK AG, are collaborating with ODVA on the development of an open protocol for safety communications including the development of DeviceNet Safety. According to SICK AG, the companies expect to introduce the first DeviceNet Safety solutions in 2004.

3.2 PROFIsafe

3.2.1 Description

Profisafe is constructed on standard Profibus DP. Both Profibus and Profisafe modules can exist in the same network. The safety features are located above Profibus layers (above OSI layer 7) to so-called safety layer. Therefore, both fail-safe and standard messages look the same. In Profisafe, messages inside the process data there are some added features that ensure the correct communication. Figure 15 shows how the safety information is integrated into the standard message. The first row shows standard messages, which are sent in a sequence. The second row shows, what is inside a single message. The third row shows how the safety information is located inside the standard process data of a single message. The added safety features in this example are as follows: fail-safe data, status, consecutive number and CRC2. The fail-safe data is the actual fail-safe message. Status can express for example a failure. Consecutive number makes it possible to control the correct sequence of the messages. CRC2 increases the effectiveness of the basic FCS (frame checking sequence). FCS or CRC2 monitors the underlined blocks. Profibus also uses CRC1 and CRC3 for monitoring the bus system. CRC1 is 2 bytes long and it is calculated across F-parameters, such as password address, watchdog time, CRC2 length, SIL (safety integrity level) and profile. It is calculated at least daily and it gives a base value for calculating CRC2. CRC3 is 2 bytes long and it is calculated across individual device parameters, such as the detection zone of a safety device or the constant speed of a motor. It may give a value for calculating CRC1 [8].

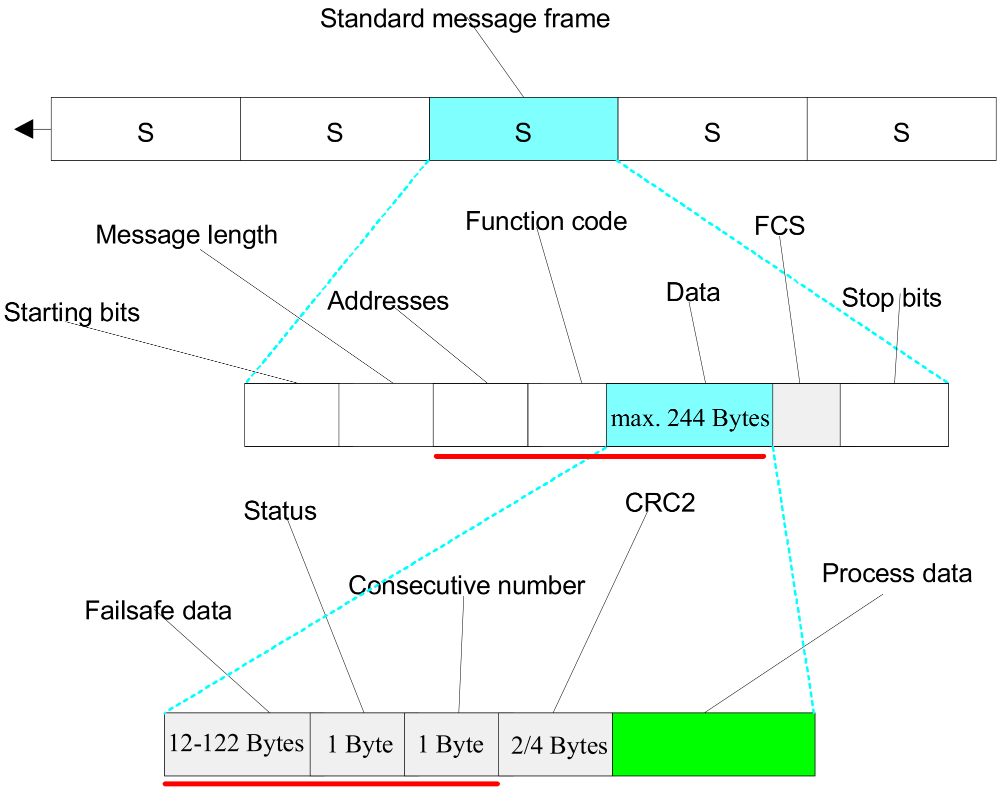


Figure 15. The structure of a Profisafe standard message.

The Profisafe modules have hardware and software based safety features in input, output and safety logic operations. Therefore, the modules differ from ordinary Profibus modules. All modules are certified. In Profisafe some parts, such as ASICs, bus drivers, lines, repeaters, links and the slave interface of modular slaves, are not considered to be safety relevant and such parts can be the same as in standard Profibus.

Figure 16 shows a typical Profisafe configuration. In the figure the modules, which start with letter F (fail-safe), are Profisafe modules and the others are standard Profibus modules.

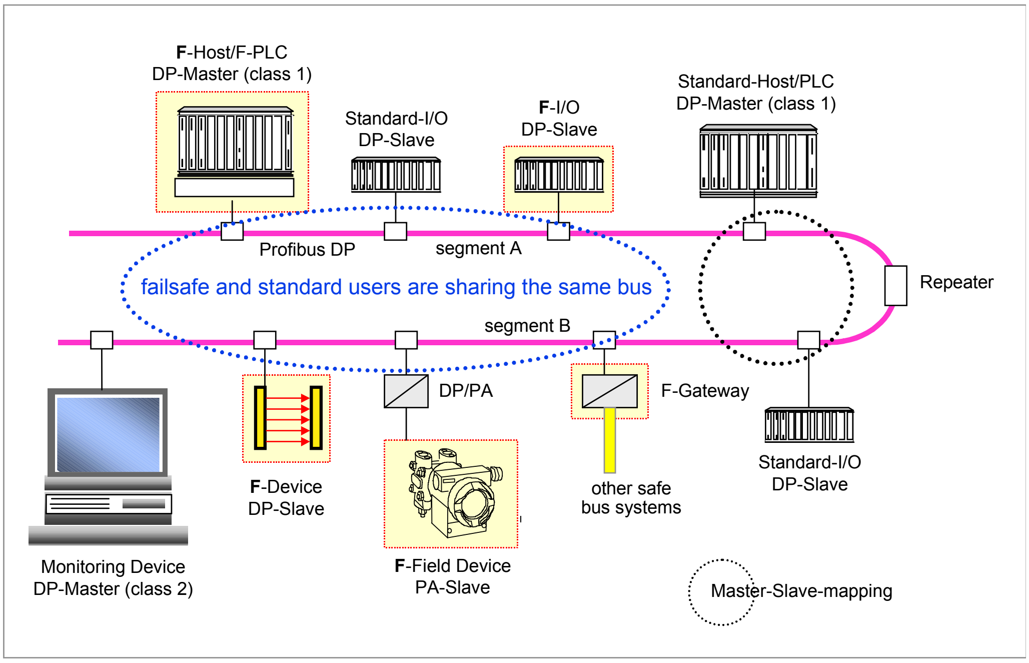


Figure 16. A typical Profisafe configuration [8].

3.2.2 Fault Hypothesis

Profisafe is designed keeping in mind the risks described in EN 50159 standard. Profisafe is also designed to fulfil the requirements stated in EN 954-1 category 4, IEC 61508 SIL 3 and DIN 19250 AK6.

3.2.3 Message Defences

Table 8 provides the abatement methods of Profisafe against the standard set of message errors.

Table 8. Message Defences in the context of PROFIsafe.

Against	Description
Repetition	Profisafe uses a 1 byte-long source-based consecutive number to identify the message. CRC2 is calculated across watchdog time and therefore old messages are detected.
Deletion	Consecutive numbering helps to detect that a message is missing. It is expected that the messages are sent at certain intervals and also acknowledgements are expected. Since the messages are sent periodically the message is soon repeated and old information is overwritten.
Insertion	Insertion of a message can be detected in several ways depending on the message. The message may have the wrong consecutive number, may not be sent at correct time, the addresses, password address and CRC2 (calculated also across addressees) may be wrong.
Incorrect sequence	Consecutive numbering reveals the wrong sequence. Since the messages are repeated periodically any wrong message is quickly overwritten.
Corruption	16-bit CRC (up to 12 byte messages) or 32-bit CRC (up to 122 byte messages) detects failures in fail-safe messages. In addition to these, there is also a 2 byte FCS (frame checking sequence) provided by Profibus. The hamming distance in standard Profibus is 4.
Timing errors	The messages are supposed to be received at a certain time and an acknowledgement should come at right time. Consecutive numbering may reveal also an error.
Masquerade	The structure of a standard Profibus message is simpler than a Profisafe message. The message can be detected by message type, addresses, password address, watchdog timer, consecutive number, CRC2 and time expectation with acknowledgement. The messages are repeated and therefore an old message is quickly overwritten.
Inconsistency	All messages have only one receiver and therefore only two modules share the information. Each message also needs an acknowledgement.

3.2.4 Architectural Defences

When the Profibus/Profisafe network is defined, each fail-safe module calculates CRC1 and CRC3 of the configuration. If the configuration changes during use, fail-safe receiving module detects the fault in CRC2. All messages and their acknowledgements should always be sent at a certain time.

3.2.5 Off-the-Shelf Availability

Profisafe is a commercial bus system. In summer 2004, the Profibus product catalogue (http://www.profibus.com/productguide.html) did not exhibit any actual Profisafe device. However, such devices as remote I/O terminals and motion control drives exist on the market. Profisafe components must pass a certification procedure to enter the market.

3.3 CANopen Framework for Safety-Relevant Communication

3.3.1 Description

CANopen Safety -profile is defined by CAN in Automation (CiA) and holds a document number DSP 304 V1.0 (CANopen Framework for Safety-related

Communication). DSP 304 does not define any specific profiles for safety-related devices. However, it allows implementation of safety features along with any CANopen device profile. The main scope of DSP 304 is to introduce an additional communication object, SRDO (Safety-related data object) along with the normal CANopen communication objects PDO (Process data object) and SDO (Service data object). SRDO can be roughly described as a duplicated PDO with bitwise inverted data in the duplicated message. No data bytes from the CAN data frame are used for the safety features.

In total, 64 safety-related communication objects can be used in a system. Consequentially, there is a maximum of 64 message producers; the number of consumers is not limited. In a CANopen safety network, normal and safety devices may coexist.

3.3.2 Fault Hypothesis

DSP 304 does not supply any systematic fault hypothesis. Message corruption seems to be the major error type concerned. Furthermore, configuration parameter corruption, timing errors and excessive bus-load are somewhat addressed.

3.3.3 Message Defences

Table 9 provides the defence methods of a CANopen safety framework DSP 304 against the standard set of message errors.

Table 9. Message Defences in context of CANopen safety.

Against	Description
Repetition	None
Deletion	Time expectation (periodic sending of SRDO's)
Insertion	None
Incorrect sequence	None, except that DSP 304 requires that the application must check that the two messages of a SRDO comes in chronological order (higher priority identifier first)
Corruption	Duplication of the information with bitwise inverted data in the duplicated message
Timing errors	Time out (periodic sending) Excessive delay between the original and the duplicated message indicates excessive bus load Possibility to use global fail-safe command (GFC), which works as a sort of interrupt to speed up the processing of the subsequent SRDO. GFC is sent with fixed high priority identifier (COB-ID = 1)
Masquerade	The two messages that build up the SRDO must use identifiers that differ in two bit positions
Inconsistency	None besides the normal CAN consistency control

3.3.4 Architectural Defences

The only architectural defence in DSP 304 is the checking of SRDO configuration parameters. This is done by defining two objects, Configuration Valid and Safety Configuration Checksum. The configuration tool that is used to perform the configuration of the safety devices must update the Safety Configuration Checksum object if it updates the SRDO related parameters. The Configuration Valid entry enters the “non valid” state immediately if any of the SRDO parameters have been updated. The configuration tool sets it to “valid” after reading back the parameters and the checksum. In runtime, the application shall check the SRDO parameters (of each safety device) by reading the parameters and checking the resulting checksum against the checksum stored in the Safety Configuration Checksum object. The Configuration Valid object must also be checked.

3.3.5 Off-the-Shelf Availability

Products to support CANopen Safety start to emerge onto the market. For example, Elobau GmbH provides a safety bus system called eloSafe, which is based on CANopen. EloSafe systems consists of eloSafe sensors and sensor islands, actuators, shot bolt units, an emergency stop switch and diagnostics unit. EloSafe is implemented with port GmbH’s CANopen safety module. Furthermore, a company called Bernstein AG provides emergency stop switches and safety switches with CANopen Safety interface. Janz Computer AG provides a CANopen Safety configuration tool.

A consortium called CANopen Safety Chip (CSC) is going to provide a chip that supports CANopen safety and will support safety integrity level 3 (SIL 3) according to IEC 61508.

3.4 EsaLAN

3.4.1 Description

EsaLAN (Elan Safety Local Area Network) is an open and CAN-bus based system, which has been developed exclusively for safety-related tasks. The system is constructed from a central control unit, either a directly connected or decentralised terminal station, and intermediate stations between them. Any commercially available sensors and actuators can be connected to the stations via short spur lines.

EsaLAN is based on a multi-master system. This means that the system may consist of two or more masters, which are all able to access the bus.

A special feature of the system is a terminal station. This is a station, which is actually connected at the physical end of the system. A special interplay between the central control unit and the terminal station offers highly dynamic monitoring of the bus line. The reaction time (also in the case of a fault) is 15 ms. The central control unit contains the parameterisation and programming interface of the system and the connection facilities for gateways to operational field bus or control systems.

The EsaLAN system has been on the market since 1998. The system was validated in 1999 against the standard EN 954-1 category 4 and IEC 61508 SIL 3. EsaLAN is already being used in applications including industrial automation systems, robots and materials handling systems. [32]

3.4.2 Fault Hypothesis

EsaLAN applies the fault hypothesis of EN 50159-2 [13].

3.4.3 Message Defences

Table 10 provides the defence methods of EsaLAN against the standard set of message errors.

Table 10. Message Defences in the context of EsaLAN. [31]

Against	Description
Repetition	In addition to system redundancy, the telegram structure on the bus level in the EsaLAN system is implemented so that from eight possible data bytes only three data bytes are used, and in each status message of bus participants the information on the current indication of counter, current indication of counter as inverted, and security byte is transmitted.
Deletion	As above.
Insertion	As above.
Incorrect sequence	Sequence count + sequence count inverted.
Corruption	15-bit CRC. System redundancy. Signal duplication.
Timing errors	Time window.
Masquerade	Isolated architecture (including only safety modules).
Inconsistency	In addition to system redundancy, the telegram structure on the bus level in the EsaLAN system is implemented so that from eight possible data bytes only three data bytes are used, and in each status message of bus participants the information on the current indication of counter, current indication of counter as inverted, and security byte is transmitted.

3.4.4 Architectural Defences

EsaLAN system has a redundant structure. Every individual station is composed of two independent CAN microcomputer systems with their own isolated power supply. Both systems co-operate via a so-called link in a special safety manner. This internal connection causes a continuous cross-wise data comparison in every station. All inputs of the EsaLAN system are monitored cyclically. Correct functioning of the outputs (semiconductor) is ensured by a cyclical test of the semiconductors.

3.4.5 Off-the-Shelf Availability

Schmersal Company in Germany supplies EsaLAN systems. The distributor in Finland is Advancetec. Schmersal is the main developer of the EsaLAN system. EsaLAN is both proprietary and open, and operates independently of operational control. All commercially available safety switching devices can be connected to it and connections are made to the input and output stations of the system. [4]

3.5 SafetyBUS p

3.5.1 Description

SafetyBUS p is an open safe bus system for serial transfer of safety-related information. It is based on an event-driven bus procedure, in other words, data is only sent when the status at the I/O or field module has changed. SafetyBUS p is a multi-master system based on the proven CAN bus system. A SafetyBUS p network can consist of up to 64 subscribers. A subscriber can be a programmable safety system, a decentralised I/O module or any other safety device. It uses CAN as the underlying field bus system. [29]

SafetyBUS p divides the units involved in safe bus traffic into classes, depending on their function. Each SafetyBUS p network has one Management Device (MD), which conducts the configuration and also the control of the network. It triggers connection monitoring of all the subscribers, sets transmission rates and starts or stops individual I/O-groups. Introducing I/O-groups enables faults to be restricted to a local level.

The structure of each SafetyBUS p subscriber is such that each subscriber has a Bus Interface Part (BIP) and an Application Part (AP), which are linked via an interface. The SafetyBUS p related safety measures are realised in the BIP, and the respective application is implemented in the AP.

SafetyBUS p and its components have been approved to category 4 of EN 954-1 and SIL 3 of IEC 61508. It also meets the requirements of AK6 in accordance with DIN 19250. The system uses the 3-channel diverse structure of the processing units. Each channel processes all inputs and outputs separately. Each channel compares the result with the two adjacent channels. Input and output signals are only valid if all three channels reach the same result.

3.5.2 Fault Hypothesis

SafetyBUS p applies the fault hypothesis of EN 50159-2 [13].

3.5.3 Message Defences

Table 11 provides the defence methods of SafetyBUS p against the standard set of message errors.

Table 11. Message defences in the context of SafetyBUS p [5].

Against	Description
Repetition	Event counter. (In the case of event-driven telegrams, this type of error is detected by including an event counter. The counter status on the issued telegram is compared with the counter status received in the acknowledgement telegram. This means that response telegrams can clearly be assigned. In the case of connection test telegrams, a key code is incorporated into the telegram instead of the event counter. The rest of the procedure is identical to that of the event counter.)
Deletion	Event counter. Echo (In most cases, the recipient will send an acknowledgement telegram to confirm that it has received the message. If the acknowledgement telegram fails to appear, it can be assumed that the telegram has been lost en route. Acknowledgement telegrams are generally monitored for time.)
Insertion	Event counter. Echo. Identifier for sender and receiver.
Incorrect sequence	Event counter.
Corruption	16-bit CRC. Echo.
Timing errors	Timeout. (Entering timeout periods when programming the bus system enables dynamic monitoring of communication. The Cycle Timeout is used to monitor telegrams within the connection test. The Event Timeout is used to check whether the recipient has acknowledged the bit data in time. The Domain Timeout monitors the transfer of data fields.)
Masquerade	Isolated architecture.
Inconsistency	See “deletion”.

3.5.4 Architectural Defences

The security of the bus system is based primarily on a safe communication protocol, which includes security mechanisms such as CRC checksums, echo mode, connection/addressing tests and time monitoring. The concept behind SafetyBUS p means that bus subscribers must be safe modules, in other words, they must be self-monitoring, performing all necessary checks and corresponding reactions independently. From the safety point of view, the bus cable itself is regarded as non-safe [29].

3.5.5 Off-the-Shelf Availability

There are a large number of SafetyBUS p components available. One of the companies offering products for SafetyBUS p is Pilz International. SafetyBUS p has become one of the standard bus systems for safe automation technology. Further information on the SafetyBUS p components and devices are available via the SafetyBUS p Club International e.V. (http://www.safetybus.com). It constitutes of companies that develop and use systems and components for fail-safe automation.

3.6 AS-interface Safety at Work

3.6.1 Description

AS-interface (AS-i) is applicable in systems, which connect simple on/off information and (small) power between switches, PLC and output units. An AS-interface Safety at Work system is composed of a standard ASI network along with a safety monitor(s) and safety-related slaves (max. 31). Safety-related slaves are connected to safety devices or switches and safety outputs. The safety monitor controls the communication between modules all the time and if it detects a failure, it starts a fault-handling procedure and de-energises its two output relays. Safety PLC is not required in the system. The network is controlled by a master unit, which sends a request to a slave unit, which answers immediately. If the answer is not correct, it may answer again and then the master sends a request to the next slave. Figure 17 shows a typical AS-i configuration, which may have standard PLC and other standard modules together with a safety monitor and safety-related slaves.

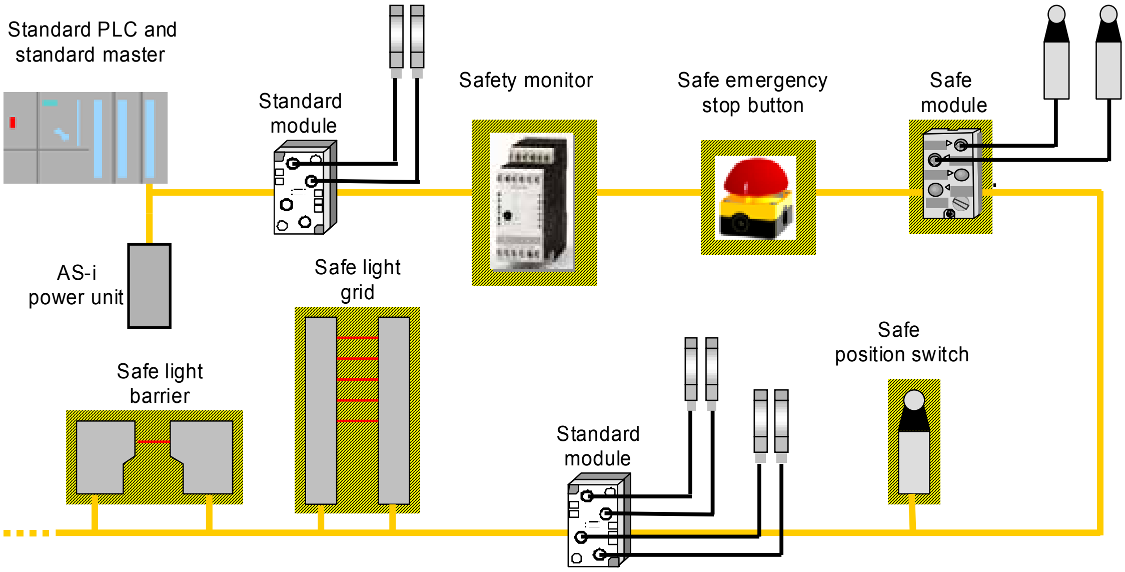


Figure 17. A typical AS-i configuration [7].

Figure 18 shows how a safety monitor is checking all messages from its code table. When the input of a safety-related slave is "ON", one row from the code table is sent and then at the next turn the next row is sent. When the input is "OFF", zero sequence is sent regularly.

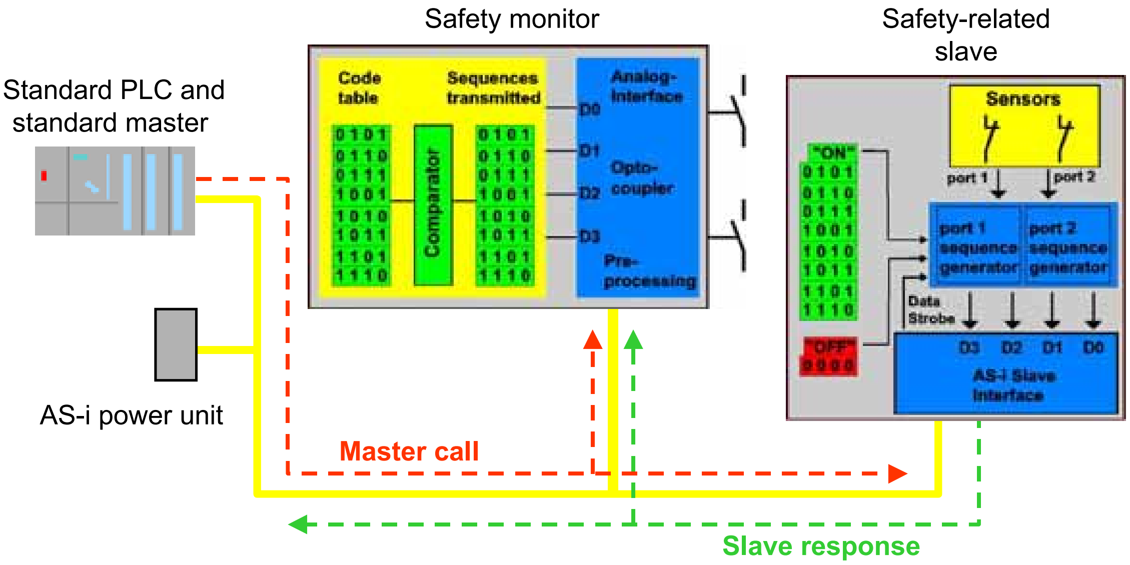


Figure 18. The monitoring system AS-i Safety at Work system [7].

Figure 19 shows, how to calculate the maximum response time (35 ms) after the input of the slave unit has been triggered.

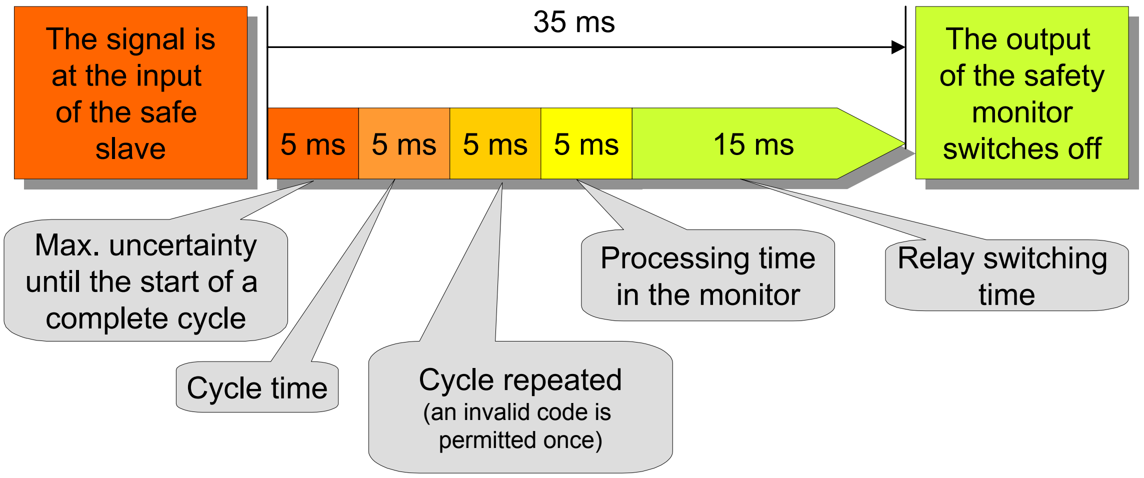

The signal is at the input of the safe slave
35 ms
5 ms
5 ms
5 ms
5 ms
15 ms
The output of the safety monitor switches off
Max. uncertainty until the start of a complete cycle
Cycle time
Cycle repeated (an invalid code is permitted once)
Processing time in the monitor
Relay switching time

Figure 19. The maximum response time in AS-i Safety at Work [7].

AS-i Safety at Work is designed to fulfil the requirements stated in EN 954-1 category 4. AS-i is intended for simple networks and it can be connected to other networks with specific gateways. Since the messages are short and simple, they can be repeated quickly (150 µs). The main safety feature in an AS-i Safety at Work system is the code tables. Only the correct slaves can generate the expected codes. Each message has also a parity bit.

3.6.2 Fault Hypothesis

AS-i Safety at Work applies the fault hypothesis of EN 50159-2.

3.6.3 Message Defences

Table 12 provides the defence methods of AS-i Safety at Work against the standard set of message errors.

Table 12. Message Defences in the context of AS-i Safety at Work.

Against	Description
Repetition	Slave unit sends safe state messages in a specific sequence. If the bit sequence is not expected, the monitoring unit generates a fault-handling situation. The "OFF" state messages are zero bits and the system generates a proper safety function. If the zero bits are repeated, the safety function remains on.
Deletion	If one message is deleted, it will be sent again. If it is not received the monitoring unit detects that a message is missing from the sequence and generates a fault-handling situation.
Insertion	If one message is inserted, the monitoring unit detects the false sequence. The master unit and monitoring unit control the transmissions and the correct order. If a message containing the "OFF" signal is received, a safety function is executed.
Incorrect sequence	The monitoring unit detects an incorrect sequence. Messages containing the "OFF" signal are similar and therefore the order has no meaning.
Corruption	Each message has a parity bit, which detects, for example, single bit failures. The message from a slave contains only 7 bits: startbit, information (4 bits), parity bit and endbit. The message from the master unit contains 14 bits: startbit, controlbit, slave address (5 bits), information (4 bits), parity bit and endbit. Since the messages are short, any massive fault detection method is not applicable. The code sequences also help to detect a failure. A hazardous situation can be generated only if the information bits, which should be zero, are changed into expected code bits and the parity bit is also acceptable. This means that in messages usually 2 to 4 bits out of 5 bits should be continuously wrong in a specific way to cause a hazardous situation.
Timing errors	All slaves may cause a fault-handling situation if the delay exceeds 40 ms.
Masquerade	The master unit controls the transmissions. The monitoring unit can also detect a fault in transmission. Only the master unit receives messages from the slave units and the slave units send messages only to the master unit. Each unit has an individual address.
Inconsistency	All messages have only one receiver. In addition, the monitoring unit receives and monitors all messages.

3.6.4 Architectural Defences

The master unit controls the transmissions. It asks for information from all slave units one after the other. If the monitoring unit detects faulty messages, it may cause a fault-handling situation. Only the monitoring unit and safety-related slaves can be considered as “fail-safe”.

3.6.5 Off-the-Shelf Availability

The AS-Interface web page (http://www.as-interface.com/db/_search.asp) lists about 40 AS-i Safety at Work products from five vendors. The components can be added to a standard AS-i network.

3.7 Interbus Safety

3.7.1 Description

Interbus Safety is a special "safety-related expansion" for Interbus, which is widely used in machine and industrial applications. It is developed by Phoenix and supported by many automation manufacturers. The topology of Interbus is an active ring. This means that all devices are actively integrated into a closed transmission path. Interbus works using the master-slave principle. It has fixed telegram length and is therefore deterministic. All bus devices include repeater functionality.

Interbus Safety uses the SafeControl concept, which is independent of both the bus system and the host system. The basis for this is concept is the SafeControl unit. It must be installed directly after the Interbus master and therefore it receives all the I/O information of the connected devices. A safety protocol is used between this SafeControl unit and the connected safety I/O devices that guarantees the required safety of the data transfer and can only be interpreted by the connected safety devices. This characteristic enables safe simultaneous operation of standard and safety devices in the bus system. [6, 28]

The Interbus Safety protocol extends the standard Interbus system to include a safe transmission channel, which, according to references, transmits application data up to category 4 of EN 954-1 or SIL 3 of IEC 61508. [2, 6].

The user can choose, depending on the application, either a so-called "one-cable solution with integrated safety technology" or a "two-cable solution". In the latter, one bus cable is used for standard signals and the other for safety signals. The safety-related functions are integrated directly into the safe application program in the form of blocks, which are linked to safe inputs and outputs.

Each device in the Interbus Safety system has its own transmitting and receiving unit, thus making point-to-point coupling of Interbus devices possible. In the Interbus system, data is automatically assigned to devices using their physical location in the system (plug-and-play function). The physical layer of Interbus is based on the RS 485 - standard.

Interbus works according to the summation frame method (Figure 20) that uses only one protocol frame for messages from all devices. The bus master acts as the coupling to the higher-level control or bus system. The method provides a high level of efficiency during data transmission and enables data to be sent and received simultaneously (full duplex operation). The summation frame method ensures that the process image is consistent for all devices, because all the input data originates from the same point of scan time and all the output data is accepted by the devices simultaneously.

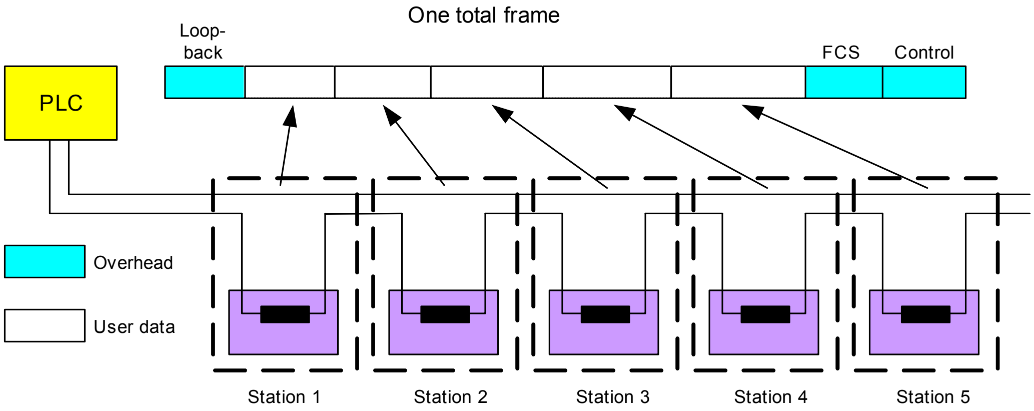


Figure 20. Summation frame method [2].

The cycle time, which means the time required for I/O data to be exchanged once with all the connected modules, depends on the amount of user data in an Interbus system. The summation frame has a set length and thus the cycle time also remains constant.

The master/slave structure prevents bus access conflicts. The bus master ensures transmission reliability by using the loop back word. If there is a break in transmission of more than 25 ms, this is interpreted by all devices as a system interrupt. The devices switch to a defined safe reset state.

The summation frame protocol enables the integration of safety data into the normal Interbus data flow (Figure 21). Only the safe components (IOs and monitoring unit) can evaluate this added safety protocol.

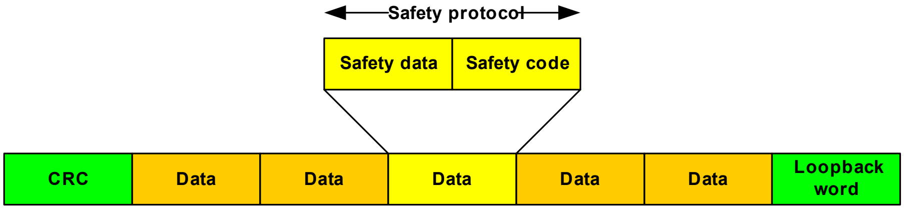


Figure 21. Integrating safety data in the Interbus data flow. [6]

3.7.2 Fault Hypothesis

Interbus Safety applies the fault hypothesis of EN 50159-2.

3.7.3 Message Defences

Table 13 provides the defence methods of Interbus Safety against the standard set of message errors.

Table 13. Message defences in the context of Interbus Safety [31].

Against	Description
Repetition	A serial number is given for each new message. Redundancy.
Deletion	A serial number is given for each new message. In addition, there is a time control for the appearance of the right number. Redundancy.
Insertion	The sequence of a message with a serial number for the message. Redundancy.
Incorrect sequence	A serial number is given for each new message. Redundancy.
Corruption	A serial number is given for each new message. 16-bit CRC per message. Redundancy.
Timing errors	A watchdog integrated to both sides will detect a delayed message.
Masquerade	The sequence of a message with a serial number for the message.
Inconsistency	Not known.

3.7.4 Architectural Defences

The safe distributed units have an entirely redundant structure that includes a fail-safe comparator. The output units generate an active status for the output only if the control system and the monitoring system (SafeControl) request it simultaneously. All data is checked again for possible errors. Each distributed unit has also an internal safety architecture that makes it possible to detect component failures, system interrupts or the absence of data immediately. In addition, each distributed safety-oriented unit has an internal watchdog. It is only reset on receipt of authorised and error-free data information.

3.7.5 Off-the-Shelf Availability

The technology is developed by Phoenix Contact and Interbus Club, which also offer more detailed information on Interbus Safety.

3.8 TTP/C

3.8.1 Description

TTP/C is a hard real-time protocol based on Time-Triggered Architecture (TTA), which is a hard real-time architecture designed for safety-relevant distributed control systems. The letter 'C' in the abbreviation of TTP/C denotes class C category according to the SAE classification of multiplex buses. TTA was designed in the University of Vienna by Dr. Herman Kopetz and his research team. TTA is based on MARS architecture, which was conceived in the University of Vienna in 1979. The Vienna group has done a lot of work with hard real-time distributed architectures, for example, in two European projects: X-by-wire (http://www.vmars.tuwien.ac.at/projects/xbywire) and TTA (http://www.vmars.tuwien.ac.at/projects/tta). To promote the TTP technology a company called TTTech Computer-technik AG (http://www.tttech.com/) was founded in 1998. A group called TTAgroup (http://www.ttagroup.org/index.htm) was also founded in the same year to provide a forum to exchange information and to prepare guidelines and standards to support TTP. Currently, the TTP group has the following core members: Airbus, Audi, Delphi, Honeywell and TTTech. Associate and affiliate members include among others NEC and Xilinx.

TTP/C is designed for safety-critical systems, like steer-by-wire and brake-by-wire systems, but it is well suited for any safety-relevant distributed control systems where the continuous state of the system is the safe state and where the system is characterised as a hard real-time system. The TTP/C specification addresses safety by applying fixed scheduling of messages (time-triggered approach). Event initiated messages must be delayed until the pre-allocated timeslots for their transfer takes place. Therefore, the bandwidth for the event messages is fixed in the configuration phase. However, the bandwidth (period of timeslots) for event messages does not have to be the same as for the periodic messages. A fault tolerant clock synchronisation algorithm arranges the global time base needed for the time-triggered operation. A bus guardian protects against the babbling-idiot fault mode.

Another property concerning safety is the support for scalable redundancy. The TTP/C may be duplicated (the protocol chip provides two channels) (see Figure 22). Furthermore, the nodes may also be duplicated or replicated (see Figure 23). However, the system may include nodes that are not replicated and nodes that use only one bus to transfer its data.

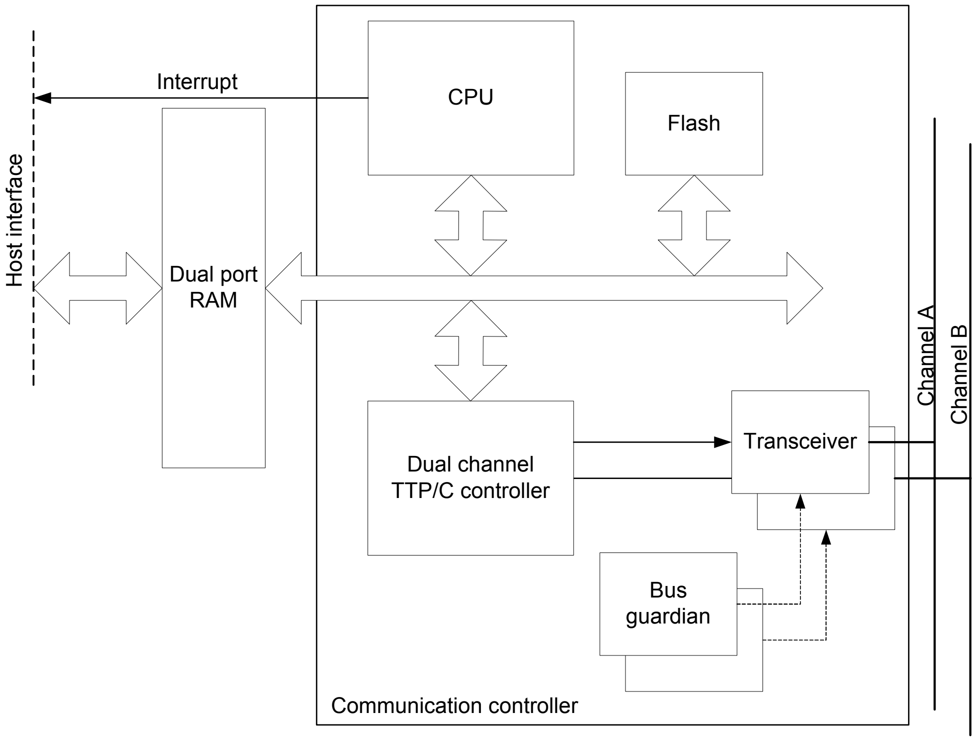


Figure 22. TTP/C node architecture.

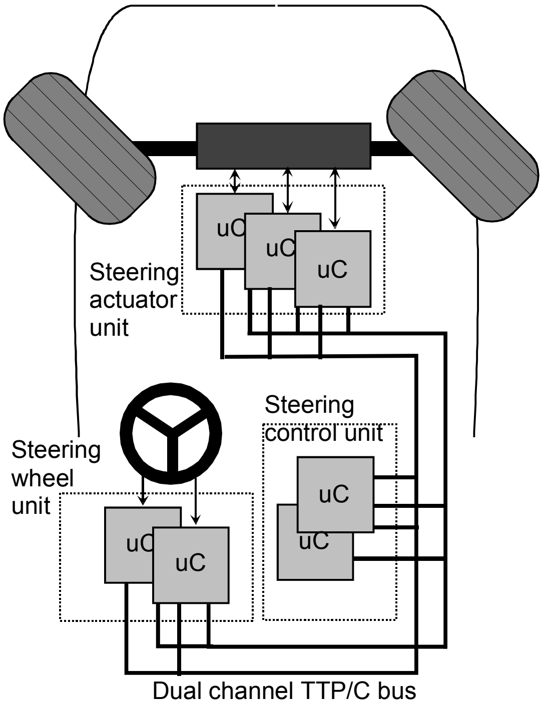


Figure 23. Steer-by-wire prototype architecture [36].

The key distinguishing property of TTP/C is composability. Composability is a property, which ensures that a subsystem has exactly the same properties in the system as it has independently or in other systems. In practice, this means that the communication network interface (CNI) is fixed in value domain and in temporal domain; in other words, if the CNI is implemented as a dual port RAM, its contents (the message memory locations and their size and number of such items) is fixed and its time related behaviour is deterministic. Composability is a property that is contradictory to flexibility. Composable systems cannot be highly flexible and highly flexible systems are not composable. Composability requires more work in system configuration phase, but it alleviates the testing of the system, as the changes in the application code of a node do not affect other nodes if the CNI:s of the nodes remain the same on all nodes. In non-composable systems, changes in one node may cause time-related changes to the bus traffic and therefore, to the behaviour of other nodes. Furthermore, in non-composable architectures, the properties of a subsystem depend on its system context. Consequentially, in non-composable systems, comprehensive testing is required if changes are done to the application software of a single node. Comprehensive testing is also required if a non-composable subsystem is reused in a new system of different type.

TTP/C also addresses membership (node guarding) issues, in other words, the nodes of the system monitor the status of other nodes and report it to the bus. Therefore, a node that does not know itself to be faulty can notice its faultiness from the feedback of the other nodes. Such a facility is not available intrinsically, for example, in the CAN protocol, but has to be provided by the upper layer protocols. Table 14 lists the main properties of TTP/C.

Table 14. TTP/C facts.

Attribute	Description
Bus speed	5 mbit/s asynchronous and 25 Mbit/s synchronous with the currently available chips
Cable length	Not known (depends on the physical media selected; there is no bit arbitration methods similar to CAN which would set restrictions on the length of the bus)
Protocol	Time Division Multiple Access (time-triggered communications only possible)
Topology	Bus or star or multistar
Number of nodes	Not known
Media	Optical and electrical physical layers possible; for example, CAN transceivers or RS 485 can be used
Other notes	Deterministic, support of redundant channels, support of fault tolerant global time, bus guardian protects against babbling idiots, composability is provided

As stated above, TTP/C supports only time-triggered communication. Nevertheless, event-triggered communication can be implemented on top of the TTP/C protocol. For example, CAN emulation can be provided facilitating the usage of legacy CAN software. Composability is not lost with the CAN emulation as the CAN messages are embedded into the TTP/C frames, which are transferred periodically. The usage of TCP/IP on top of TTP/C is also planned.

TTP/C has been selected by Honeywell to be used in its APEX integrated cockpits for aviation aircraft and helicopters. The drive-by-wire system of a concept car called FILO from Bertone uses TTP/C. The drive-by-wire system for FILO is supplied by SKF Automotive Division's Drive-by-Wire Business Unit. However, Audi is currently the main car manufacturer today that backs TTP/C. Furthermore, TTP/C has been selected for implementing a cabin pressure control system in Airbus A380 aircraft.

3.8.2 Fault Hypothesis

Apart from the normal communication error types, TTP/C also handles the failure types of a distributed control system well. The following failure types are mentioned in particular:

- Outgoing link failure (a node in a system with multiple nodes cannot send a message)

- Inconsistency (nodes in the system have an inconsistent view of the values of the state variables) including Slightly-off-Specification (as the nodes decide about the correctness of a message according to its appearance within the correct time slot, due to variations in manufacturing etc. processes, the length of the allowed time slot may differ on various nodes and therefore some nodes accept the messages and others don't)

- Spatial proximity failure (replicated units may reside too close to each other enabling a single fault to cause an error in all the replicated units)

- Masquerading (a node in a system takes the identity of another node)

- Babbling idiot (a node in a system occupies the transmission media system by continuous transmissions).

TTP/C protocol is designed to tolerate a single internal physical fault and a TTP/C system can also tolerate a single external fault if it occurs inside a subsystem.

3.8.3 Message Defences

TTP/C is designed for fault operational systems, like steer-by-wire systems (but is, of course, suitable for fail silent systems as well). Therefore, the EN 50159 model is not valid in the context of TTP/C. Consequentially; the list of message defences is not comparable with the other safety buses presented in this report. However, we provide here a similar list of message defences as in context of other protocols (see Table 15). Table 15 clearly shows that the communications threats are mastered by the architecture of the communication system rather than by additional safety layer on top of a standard communication layer. Time-triggered architecture, bus guardian, redundancy and membership service are among the key strategies against the communication threats.

Table 15. TTP/C message defences.

Against	Description
Repetition	Time-triggered architecture. Bus guardian. Replicated star with spatial remoteness.
Deletion	Time-triggered architecture. Redundancy (message, bus and modules). Membership service.
Insertion	Time-triggered architecture. CRC (schedule ID included in the CRC calculation). Bus guardian.
Incorrect sequence	Time-triggered architecture.
Corruption	CRC (16 bits). Redundancy.
Timing errors	Time-triggered architecture.
Masquerade	Time-triggered architecture (static a priori knowledge).
Inconsistency	Membership service. Bus guardians in the star coupler.

3.8.4 Architectural Defences

TTP/C is an example of a communication system design that does not just add some safety features on top of a communication protocol. Instead, the whole communication system architecture (extending to the operating systems of the nodes) is designed for safety-critical systems. The following list presents shortly the architectural properties of TTP/C that contribute to safety:

- Time-triggered architecture (including protocol and RTOS) with static scheduling ⇒ predictability

- Membership supervision

- Fault containment

- Redundancy (messages, transmission media and modules)

- Power supply distribution

- Bus guardian (with spatial diversity, in other words, bus guardian resides in star coupler, not in the node itself)

- Composability is provided.

3.8.5 Off-the-Shelf Availability

Austria Micro Systems as well as NEC provide chips for TTP/C. Tools and prototyping kits for TTP/C development are available from TTTech. Integration of TTTech tools to the Matlab/Simulink/Realtime Workshop environment is also provided. TTTech also supplies a real-time operating system called TTPos, which supports the time-triggered architecture. TTPos is harmonised with the OSEKtime specification, which has been prepared by the OSEK group.

As the TTP/C bus is mainly aimed at X-by-wire systems of cars, no off-the-shelf TTP/C devices, like sensors and actuators, are expected to emerge onto the open market.

3.9 TTCAN

In the emergence of X-by-wire systems, CAN is being fortified to support time-triggered communications by defining a “session layer” above the standard CAN protocol. The main goal in providing time-triggered communication is to provide deterministic communication. Determinism is needed to implement hard real-time safety-relevant systems of which the X-by-wire systems (steer-by-wire, brake-by-wire, etc.) are the most familiar.

The time-triggered CAN is called TTCAN and is being standardised in an ISO work group (ISO TC22/SC3/WG1/TF6). The standard will be added to the ISO 11898 family of CAN standards and will be labelled as ISO 11898-4. The standardisation is supported, for example, by Bosch and NEC.

TTCAN does not change the CAN frame structure nor the basic protocol, but defines a mechanism to schedule messages in regard to a synchronisation message (which is a normal CAN message that may even contain application data). The synchronisation message (called reference message) is sent by a time master, which has been appointed by the system integrator. The application messages are transmitted to the bus in their respective time slots, which are predefined by the system integrator. The predefined “time marks” (that reflect the assigned time slots) are loaded onto the CAN interface at start-up to trigger the transfer of the time-triggered messages.

TTCAN also enables the transfer of event-triggered messages. In order to support mixing of the two transmission modes, two time windows for the communication are defined: an exclusive time window for the time synchronised messages and an arbitrating time window for the event spontaneous messages (see Figure 24). Apart from these two main windows a free window can be allocated to support the addition of future nodes or functions.

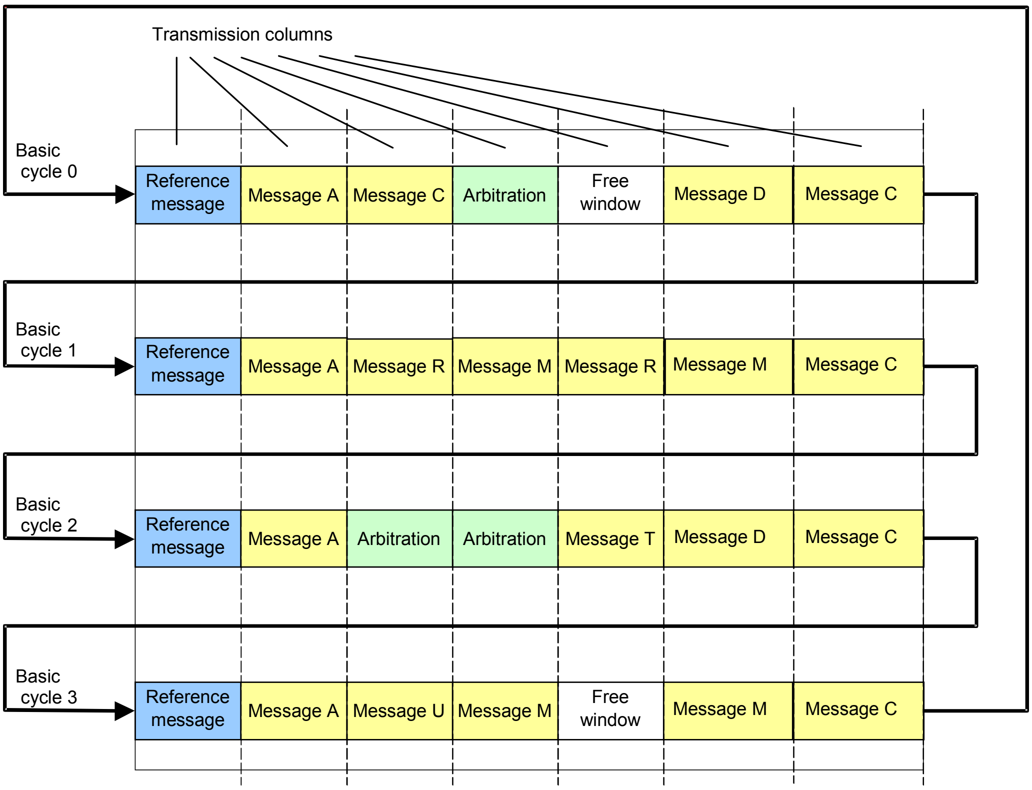


Figure 24. Composition TTCAN matrix cycle [25]. Note: the number of basic cycles within the matrix cycle must be an integer power of two.

In all the windows, the normal arbitration procedure during the CAN identifier takes place. Therefore, if there is a slight error in the timing of the time-triggered messages causing two time-triggered messages to collide, the CAN protocol will resolve the situation so that the higher priority message gets access to the bus. However, it is difficult to assess here whether this is actually a desirable feature, because such a situation indicates a design fault, and it affects the determinism of the bus communication (which is the primary goal of TTCAN). Determinism is lost because, compared to a normal situation, the event ordering of the messages is changed if the colliding messages have been scheduled during the design phase so that the lower priority message is sent first; in the overlap situation described above, the higher priority message is sent first.

It is not necessary to transfer the same time synchronised message in every communication cycle, but a time slot can be shared by multiple messages with the same length but with a different identifier (see Figure 24). Therefore, the transmitter has to know the number of the cycle (counted from the first cycle) when to do the first transmission and the number of cycles to wait between the transmissions thereafter. These two parameters (the base mark and the cycle count) along with the time slot number (or actually the time unit count after the cycle start) form the time mark information for a single message.

Retransmission of frames is prohibited in the case of corrupted CAN messages and even in the case of lost arbitration in order to maintain determinism. Therefore, special precautions for corrupted (and thus lost) messages must be provided, for example, by message redundancy (replication of messages) or by allowing the retransmission of a periodic message in an arbitrating time window.

To facilitate time-triggered communication, every node has to provide a clock tick for the communication scheduling. TTCAN defines two levels of global time accuracy:

- Level 1, which does not support global time negotiation or drift compensation, the reference messages are the only way to set the clock (to zero).

- Level 2, which support both global time algorithm and drift compensation. The level 2 implementation is based on a local clock, which is continuously adjusted to prevent drifting from the master clock speed.

In level 1, introducing margins on the time windows may compensate for the drift. The required margin will increase until the end of a communication cycle; in other words, the last message needs more margin than the first message. Level 1 realisation can be implemented by software (requiring a 16 bit counter), but in practice, for level 2, hardware is needed.

In level 2, the reference message contains the value of the global clock, which is recorded by the time master at the point of sampling of the start of frame (SOF) bit. The "slaves" also measure the time at SOF sampling points and record the time interval between two reference messages. The slave then calculates the same time interval as seen by the master by calculating the time difference of the time values related by the master in the two successive reference measurements. The local clock of the time slave is then adjusted by a factor, which is the quotient of these two values of the interval.

The frame formats of the reference messages in level 1 and level 2 are presented in the following tables (Table 16 and Table 17).

Table 16. Reference message in level 1 (only data bytes are shown). Note: the message may be – 8 bytes in length depending on the usage of the free bytes.

Control byte	Free for appl. use	Free for appl. use	Free for appl. use	Free for appl. use	Free for appl. use	Free for appl. use	Free for appl. use

Table 17. Reference message in level 2 (only data bytes are shown). Note: the message may be 4–8 bytes in length depending on the usage of the free bytes.

Control byte	NTU_RES	MRM LSB	MRM MSB	Free for appl. use	Free for appl. use	Free for appl. use	Free for appl. use

The control byte consists of cycle count (bits 0 - 5), one reserved bit (bit 6) and a Next_is_Gap -bit (bit 7), which informs that instead of the next reference message, there will be a gap. Resynchronisation (new reference message) will take place after the gap.

The MRM (MSB), MRM (LSB) and NTU_RES bytes contain the master's view of the global time in 16 bit + at least 3 bits to add resolution, 7 resolution bits are available in maximum. The bit 0 in NTU_RES is called Disc_bit and informs that discontinuity in the global time is expected by the master, for example, due to external clock correction at the master (phase correction).

To support fault tolerant global time, more than one potential time masters may exist on the bus. If the primary time master disappears, the potential time masters try to become the new time master. The one that has the highest priority (in other words, the lowest CAN identifier) will become the new time master (until the original time master makes a comeback with its highest priority time reference message). For all the time masters, their CAN identifiers are the same except for the last 3 bits. In total, eight potential time masters may exist on the bus. If the reference message is corrupted, it is retransmitted immediately.

As a special feature, TTCAN also provides the synchronisation of reference messages to events, in other words, the time beat is resynchronised by an external event.

TTCAN is not the prefect solution for the future safety-relevant X-by-wire systems because of its relatively low speed and lack of specifications about how to build redundant systems. Therefore, TTCAN is suitable for low-end safety-critical systems like X-by-wire with mechanical or hydraulic backup [18].

CAN chips that include TTCAN support will be soon available, for example, from NEC. Bosch provides a TTCAN IP-module (in VHDL) and an evaluation chip.

3.9.1 Considerations

TTCAN is not comparable to the safety buses presented in this report due to the fact that it only introduces a technique to make CAN communications time-triggered. Therefore, fault hypothesis, list of message and architecture defences are not relevant here.

The following list supplies some notions on TTCAN:

- As retransmissions are prohibited (in general), the intrinsic robustness of the communications is questionable in regard to the original CAN protocol.

- TTCAN does not address redundancy issues.

- TTCAN allows event-triggered messages and sometimes even the retransmission of messages. This makes the communication unpredictable and thus undeterministic and non-composable (see the definition of composability on page 60).

- CANopen also provides a specification for time-triggered transfer of messages with the difference from the TTCAN being that in CANopen the message appears on the bus after the reference message (SYNC message) in arbitration order rather than in time slot order.

Because of the issues above, TTCAN may be suitable only for a small amount of safety-relevant networking applications in work machines, but may be a solution to control applications requiring practically jitterless communications.

3.10 FlexRay

3.10.1 Description

FlexRay is another candidate for automotive X-by-wire systems besides TTP/C and TT-CAN. FlexRay is developed by BMW, DaimlerChrysler, Motorola and Philips. These companies founded a consortium called FlexRay-Group on the 30th of September 2000. Bosch, GM and Volkswagen joined the core group later. Toyota, Honda, Fiat, Renault and several others have joined the Flexray consortium as premium associate members. Besides the automotive companies, a remarkable number of automotive electronics and electrics companies and communication bus tool vendors have joint the consortium. The automotive manufacturers are quite unanimously selected FlexRay since it is the most promising safety-relevant communication bus candidate for cars.

FlexRay provides a deterministic protocol with a time-triggered architecture, but also allows event-triggered communications. The time-triggered part is similar to that of TTP/C and the event-triggered (or asynchronous) part is based on BMW's ByteFlight protocol. The time-triggered communication needs a global time, which is realised with the help of a fault tolerant midpoint algorithm presented in [35]. FlexRay also provides redundancy and bus guardian strategy to detect and tolerate bus faults. FlexRay is specified to support bit rates 5 - 10 Mbit/s. FlexRay supports both optical and copper cables.

FlexRay is not in the production phase yet. Motorola will provide communication controllers and Philips will supply transceivers. Series-production of FlexRay transceivers (with bus guardian) and protocol chips is expected to start at the end of 2004.

Table 18 presents the commonly known facts about FlexRay.

Table 18. FlexRay facts.

Attribute	Description
Bus speed	5 Mbit/s, gross data rate 10 Mbit/s
Cable length	24 m max.
Protocol	Time Division Multiple Access including both static (time-triggered) and dynamic (event-triggered) transferring of messages
Topology	Bus or star; cominations of bus and star topologies
Number of nodes	22 in bus topology
Media	Optical and electrical physical layers supported
Other notes	Deterministic, support of redundant channels, support of fault tolerant global time, bus guardian protects against babbling idiots

Figure 25 illustrates the FlexRay frame format.

Res	PPI	NFI	SFI	SFi	Frame ID (11 bits)	Payload length (7 bit)	Header CRC (11 bit)	Cycle count (6 bit)	Payload data (0 ... 254 bytes)	Frame CRC (24 bits)

Figure 25. FlexRay frame format. Res = Reserved bit, PPI = Payload Preamble indicator bit, NFI = Null frame indicator bit; SFI = Sync frame indicator bit; SFi = Startup frame indicator bit. [17]

The communication cycle is comprised of a static segment for time-triggered messages, a dynamic segment for event-triggered messages, a symbol window for network control symbol (like wake-up) and a network idle time for granting spare time for nodes to perform the correction of their local clocks (see Figure 26).

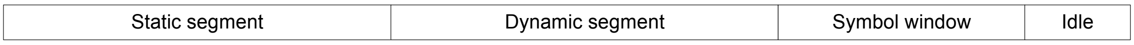


Figure 26. FlexRay communication cycle.

3.10.2 Fault Hypothesis

The FlexRay specification [17] does not describe the fault hypothesis of FlexRay. However, the FlexRay requirements specification [9] sets the following requirements for error handling (the text is an excerpt from the particular document, © BMW AG, DaimlerChrysler AG, Robert Bosch GmbH, Pages 43-44 of 52 Version 2.0.2, 9.04.2002 General Motors/Opel AG):

- *The error management shall follow the "never-give-up" philosophy. Comment: This means that the communication protocol has to support proper operation until a certain critical error state is reached.*

- *The non-arrival of periodic messages shall not be unrecognised. Comment: It is okay, if, e.g. one, periodic message is missed, but this has to be detected. The fact that a periodic message was missed should be signalled to the host.*

- *If a periodic message was missed, no random data shall be given to the host.*

- *Data content of messages, (periodic and spontaneous) must not be changed by the communication protocol.*

- *The change of data content shall be signalled to the host.*

- *After an error was detected at a communication partner in the network, the functionality of the other communication partners shall not be influenced. Comment: The correct function may not depend from the correct function of a certain host, of a certain communication controller or of a certain power supply.*

The communication controller shall detect the following list of errors:

- *Synchronisation error. The communication controller is not any more synchronised to the global time on the bus.*

- *The communication network must offer diagnosis information to the host computer with respect to the bus (channel), incoming/outgoing link failures.*

- *The communication network must offer diagnosis information to the host computer within a defined maximum delay after the occurrence of the failure of the diagnosis element.*

- *The communication network is not required to provide consistent and agreed diagnosis information to the host computer.*

Hardware units

The following faults have at least to be detected by the communication controller:

- *Defect time source (e.g. broken crystal).*

- *Low voltage.*

The following faults have to be recognised by the bus driver as errors:

- *Faulty communication signals caused by e.g. any faulty transmission media (e.g. a broken line, short circuit to ground). Incorrect communication with the host, e.g. communication via the data interface.*

- *Incorrect communications with the communication controller, e.g. busblocking transmit signals.*

- *Deactivated branch.*

Interfaces

- *Status information on detected errors must be provided. Additionally it is required that maskable interrupts for certain detected errors can be requested by the host.*

3.10.3 Message Defences

FlexRay is designed for fault operational systems, like steer-by-wire systems (but is, of course, suitable for fail silent systems as well). Therefore, the EN 50159 model is not valid in context of FlexRay. Consequentially, the list of message defences is not comparable with the other safety buses presented in this report. However, we provide here a similar list of message defences as in the context of other protocols (see Table 19). The table clearly shows that the communication threats are mastered by the architecture of the communication system rather than by additional safety layer on top of a standard communication layer. Time-triggered architecture, bus guardian, redundancy and membership service are among the key strategies used against communication threats.

Table 19. FlexRay message defences.

Against	Description
Repetition	Time-triggered architecture. Bus guardian. Cycle count.
Deletion	Time-triggered architecture. Redundancy.
Insertion	Time-triggered architecture. Bus guardian.
Incorrect sequence	Time-triggered architecture. Cycle count.
Corruption	CRC (separate for the header and for the whole frame). Redundancy. Length mismatch check.
Timing errors	Time-triggered architecture.
Masquerade	Not known.
Inconsistency	Not known.

3.10.4 Architectural Defences

FlexRay is an example of a communication system design that does not just add some safety features on top of a communication protocol. Instead, the whole communication system architecture is designed for safety-critical systems. The architectural defences of FlexRay are based upon:

- Time-triggered architecture
- Support for redundant channels.
- The transceiver includes a bus guardian to hinder babbling-idiot fault mode
- Three degradation models: normal active, normal passive and halt.

3.10.5 Off-the-Shelf Availability

As the FlexRay bus is mainly aimed at car X-by-wire systems, no off-the-shelf FlexRay devices, like sensors and actuators, are expected to emerge onto the open market. Furthermore, at the time of writing, FlexRay is in its development phase and therefore the FlexRay protocol chip availability is poor. However, tools and prototyping systems are available (http://www.flexray.com/products.php).

3.11 SAFELOC

3.11.1 Description

Swedish research institute IVF has designed and constructed a safety PLC (programmable logic controller) called SAFELOC, which is using a CAN bus for data transfer. SAFELOC is designed for collecting safety information from switches, interlocking devices, light curtains and other safety devices that are used in machinery (for example, robots). Therefore it is not only a bus system, but also a programmable logic controller (PLC). SAFELOC is made mainly for demonstration purposes and it is not commercially available. However, the technical file is available.

SAFELOC is constructed from two CPU units, two (or more) output units, input modules and reset modules. Each module has its own CAN controller. The system uses an ordinary CAN bus, which has got some safety features, but most of the safety features are located on a higher level. [10]

The system was made in 1997 and it was validated in the middle of 90's against standard proposal EN 954-1 category 4 (about the same as current EN 954-1). The CPU and output modules are duplicated and there is also cross monitoring between the sister modules. Input modules send a message to both CPUs and both CPUs have to send their messages to both output modules. Timing is also a very important factor since the modules must be in the same phase at all times. The CPU units do not have the module addresses in their memories, but they calculate the values every time they are needed. All the addresses are chosen so that their Hamming distance from each other is as large as possible. All the messages from CPU, input and output modules must come within the defined time (watchdog control). The reset module sends messages only when needed. It is not considered to be as safety-related as other modules. [10]

3.11.2 Fault Hypothesis

Safeloc applies the fault hypothesis and principles of category 4, which was defined in an EN 954-1 standard proposal. This proposal resembles the current EN 954-1 standard [15]. The hypothesis includes the complete safety system with communication, CPUs, sensors, switches and relays. In practice, all critical functions (SW and HW) are duplicated and diversity principles are also applied in many ways; for example, every other message is inverted. If any error is detected, the system enters a stop state, which needs to be reset before start-up.

3.11.3 Message Defences

Table 20 provides the defence methods of SAFELOC against the standard set of message errors.

Table 20. Message defences in the context of SAFELOC. [10]

Against	Description
Repetition	Every other message is inverted and therefore, the same messages are detected. Since the CPU and output modules are duplicated and the sister modules have different addresses the other module may detect the repetition if it receives messages correctly.
Deletion	Since every other message is inverted a missing message is detected. Also the duplicated module may detect the error if it detects its message correctly. If the message does not come in time, watchdog timer detects it.
Insertion	Since every other message is inverted, a message that comes in the wrong phase is detected. The duplicated module may also detect the failure. Because the addresses are not permanently in the memory of the CPU unit, the right addresses hardly appear accidentally.
Incorrect sequence	Since every other message is inverted, an incorrect sequence is detected. Also the duplicated module may detect the error if it receives the messages in the correct order. All the modules are in the same network and they receive the message immediately, and therefore an incorrect sequence seems improbable.
Corruption	In a standard CAN-bus there is a 15 + 1 bit cyclic redundancy check (CRC). Since the modules are duplicated the sister module may detect the failure. All the messages are coded so that the Hamming distance between any messages is as large as possible. The messages are repeated periodically (watchdog timer monitoring), and therefore a faulty message is overwritten quickly. The output of the output modules is a relay, which may need some messages to react.
Timing errors	Each message has to come within a time window of 4 ms (watchdog timer monitoring). Each delay automatically causes a stop function.
Masquerade	There are only predefined safety modules in the network. There is only one receiver for each message. The addresses are not in the memory but the addresses are calculated each time they are needed. Also the duplicated module may detect an error if it receives correct messages.
Inconsistency	CPU units control the network and if they do not get the expected message the system enters a stop state. Also if the CPU units do not operate at the same phase the system crashes, output modules enter a stop state and a reset function is needed to start-up the system again.

3.11.4 Architectural Defences

The system has also a lot of safety features related to software and hardware. For example, it is confirmed that a safety-related signal proceeds from CAN controller to output relay, an input signal proceeds from switch to CAN controller, and the two CPU units operate simultaneously and in the same way. Figure 27 shows the general structure of SAFELOC.

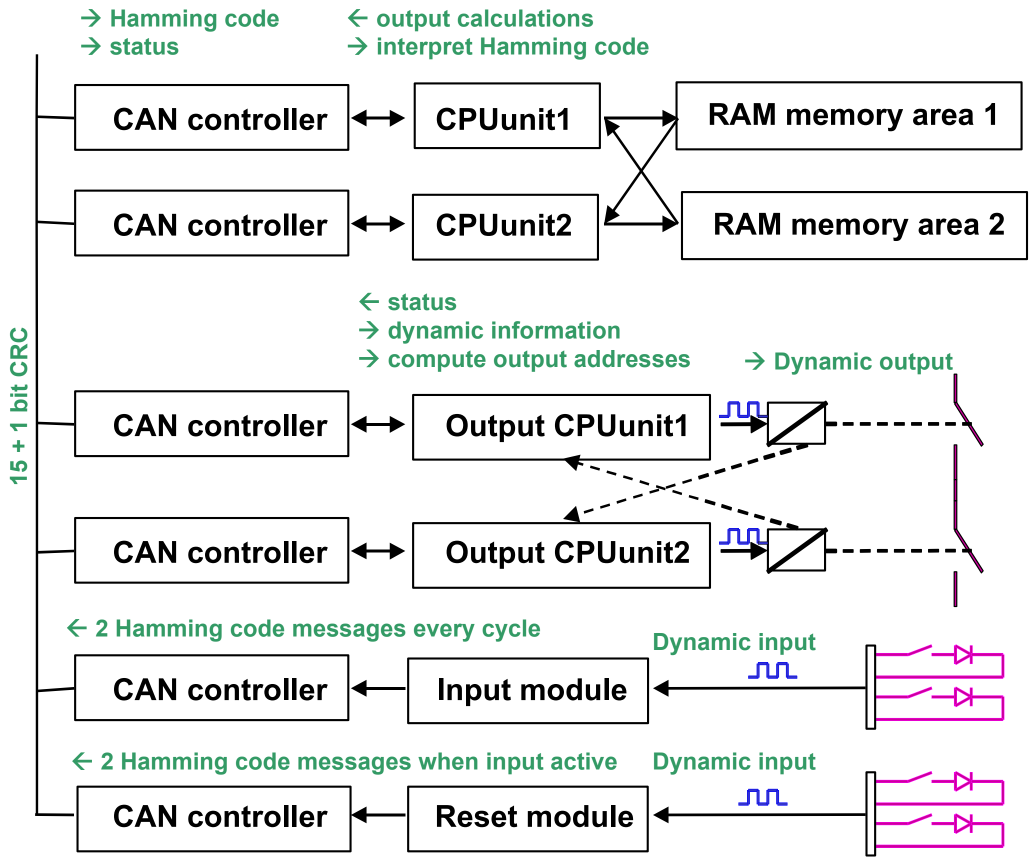


Figure 27. The figure shows the general structure and some safety features of SAFELOC. [21]

3.11.5 Off-the-Shelf Availability

The SAFELOC system is not a commercial system and is not ment to be a commercial system. However, all the technical documents are available for research purposes from IVF Sweden.

3.12 SafeEthernet

3.12.1 Description

SafeEthernet offers an open communication and safe transmission of data via Ethernet. The communication is independent of the transmission medium. HIMA has developed SafeEthernet and it is certified by the German TÜV. SafeEthernet is based on standard ethernet technology (IEEE 802.3). SafeEthernet can be applied to applications with safety requirements up to SIL3 according to IEC 61508 and AK6 according to DIN 19250. [20]

Contrary to a standard Ethernet, HIMA SafeEthernet is deterministic, provided that it is used in its own safety domain (segment). A special protocol mechanism allows SafeEthernet to guarantee a deterministic behavior even in the case of lost or additional nodes or segments. A safety domain can cover up to 32 safety nodes (incl. HIMA OPC server). The PES itself supports up to 64 safety nodes [19].

3.12.2 Fault Hypothesis

SafeEthernet detects and handles failures such as distortion of the transmitted data (double or lost bits, changes of bits, etc.), wrong addressing of messages (sender, receiver), wrong order of data (for example, repetition, loss, exchange, etc.) and wrong timing (inadmissible delay, echo, etc.). [19]

3.12.3 Message Defences

Enough information was not found to make a table of message defences.

3.12.4 Architectural Defences

SafeEthernet supports a fully redundant network for applications with high availability. In case of failure, all components can be replaced on-line during operation [19].

3.12.5 Off-the-Shelf Availability

HIMA presents the product serie called HIMatrix (http://www.hima.com/). The compact and modular safety-related HIMatrix range of controllers has been developed specifically for the time-critical requirements of factory automation. The safety-related networking of the HIMatrix systems takes place on SafeEthernet, which is based on standard Ethernet technology. SafeEthernet speeds up the transmission of safety-related data to 100 Mbit/s and supports the use of all Ethernet functions, even for the configuration of safety-related networks. The performance of the HIMatrix series, which is certified to IEC 61508, EN 954-1 and DIN V 19250, enables the integration of relay functions into the PES, thus increasing flexibility.

Allen-Bradley also introduces PLC products that have the ability to communicate on a Safe Ethernet communications network. The network is TÜV certified for use in safety applications up to category 4 of EN 954-1 and SIL 3 of IEC 61508.

4. Wireless and Safety

In principle, the safety analysis frameworks described in Chapter 1 and in the BIA guidelines (as well as in the EN 50159 set of standards) are fully compatible and applicable for wireless communications as well. This is due to the fact that wireless communications do not bring any new communication error types. The common error types (repetition, deletion, insertion, incorrect sequence, corruption, delay, too early, jitter, masquerade and inconsistency), cover all the communication errors of wireless systems as well. The only difference is that the probabilities of the various error types are different in wireless systems, for example, a corruption error is more probable due to higher bit error ratios, or short-term deletions occur more often due to poor connections. It can also be anticipated that a masquerade error is more relevant, because the free air connection is more vulnerable to intentional or unintentional intruders. Therefore, authentication and cryptographic techniques are more likely to be needed if a wireless communications is applied.

In Europe, the road map (see Figure 28) set forth by the machine directive leads us to EN 60204-1 [14] standard, which recommends in its clause 9.2.7.4 to apply IEC 60870-5-1 [22] frame types and inherent error detection methods for wireless communications. The same clause also requires that the error detection algorithm should ensure a Hamming distance of at least four; in other words, three corrupted bits shall always be detected.

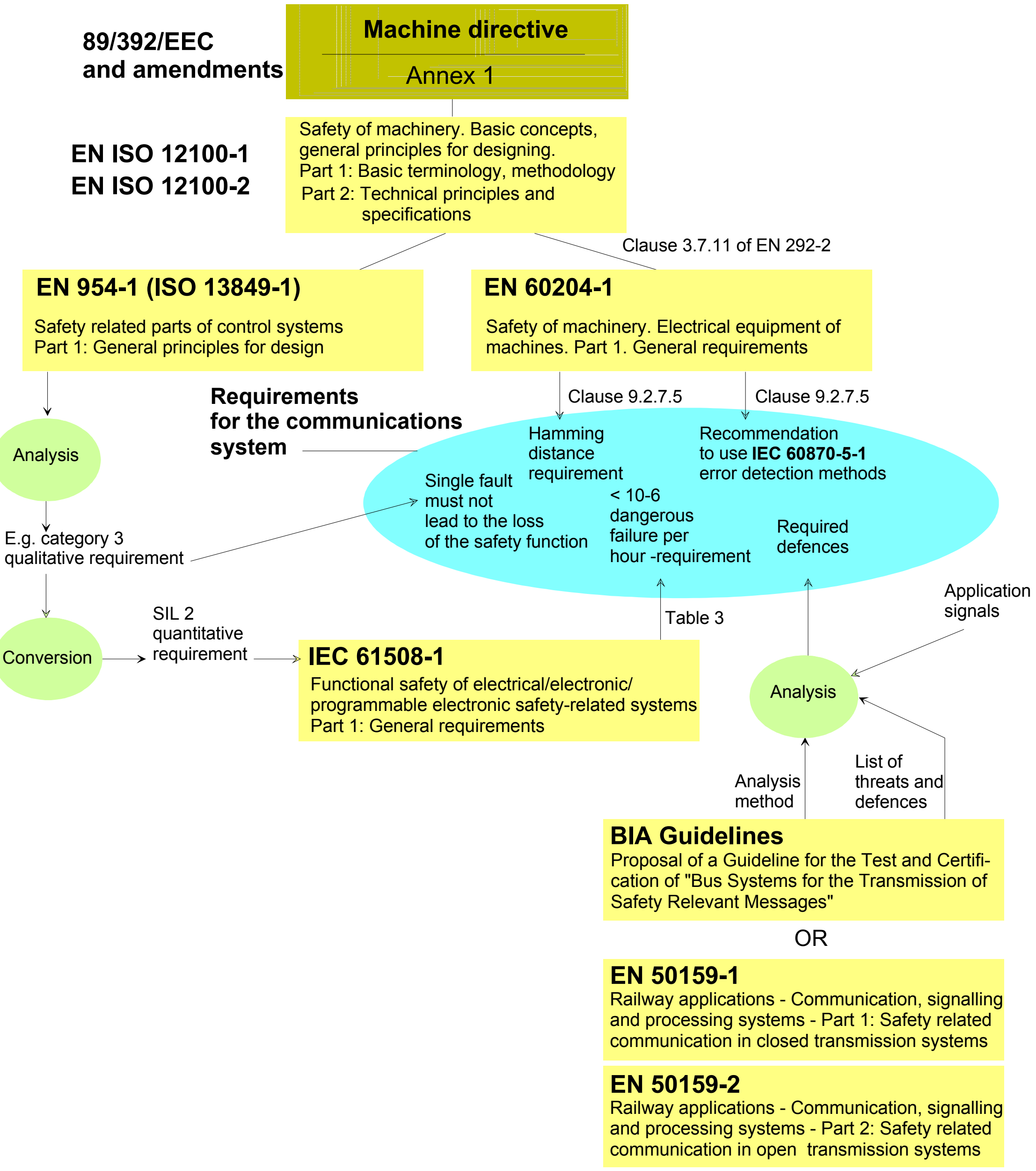

89/392/EEC
and amendments
Machine directive
Annex 1
EN ISO 12100-1
EN ISO 12100-2
Safety of machinery. Basic concepts, general principles for designing.
Part 1: Basic terminology, methodology
Part 2: Technical principles and specifications
Clause 3.7.11 of EN 292-2
EN 954-1 (ISO 13849-1)
Safety related parts of control systems
Part 1: General principles for design
EN 60204-1
Safety of machinery. Electrical equipment of machines. Part 1. General requirements
Clause 9.2.7.5
Clause 9.2.7.5
Requirements
for the communications
system
Analysis
Hamming distance requirement
Recommendation to use IEC 60870-5-1 error detection methods
Single fault must not lead to the loss of the safety function
< 10-6 dangerous failure per hour -requirement
Required defences
E.g. category 3
qualitative requirement
Conversion
SIL 2
quantitative
requirement
Table 3
Application signals
IEC 61508-1
Functional safety of electrical/electronic/
programmable electronic safety-related systems
Part 1: General requirements
Analysis
Analysis method
List of threats and defences
BIA Guidelines
Proposal of a Guideline for the Test and Certification of "Bus Systems for the Transmission of Safety Relevant Messages"
OR
EN 50159-1
Railway applications - Communication, signalling and processing systems - Part 1: Safety related communication in closed transmission systems
EN 50159-2
Railway applications - Communication, signalling and processing systems - Part 2: Safety related communication in open transmission systems

Figure 28. Application of European machine directive and consequent safety standards; wireless communications depicted as a special case.

The Figure 28 presents two requirements that are specific to wireless systems: the recommendation to use IEC 60870-5-1 and the requirement about the Hamming distance set for the wireless communications. Otherwise, the flow of applying standards is generic.

5. Modified HAZOP Analysis with the Help of a Database Tool

A modified HAZOP analysis method and a supporting tool was developed and applied during the KETU-project to assess the applicability of the signal-based analysis model discussed in Chapter 1. The modified analysis method was tested in two case studies. The first one involved a work machine with a CAN-based distributed control system. The second one was a new control system concept to be applied in work machines.

The rationale for the modified HAZOP analysis instead of a general HAZOP analysis was that the number of signals in the work machines under analysis was excessive (about 500). To make a systematic HAZOP analysis would require thousands of deviations to be analysed. Furthermore, a normal HAZOP analysis form makes it difficult to perceive the different levels of defence methods against the deviations (see the different defence levels in Figure 4). In normal HAZOP analysis, the analysis form includes the following columns:

- Deviation
- Causes
- Consequences
- Risk class (or severity and probability columns)
- Detection and safeguards
- Recommended corrective actions.

The detection and safeguards -column is used to record all the defence methods against the particular deviation. In this particular study, the analysis was made with the help of a general-purpose database tool in order to support larger number of columns, which are represented as 'fields' in the database context. In addition, detection of the deviation was divided into two fields: one for detection by the control system itself and one for detection by humans, like the operator of the machine. Furthermore, the probability value was divided into two parts: the probability that the message error occurs (threat probability) and the probability that the top event (the accident) occurs. The different levels of detection and defences are also illustrated in Figure 29.

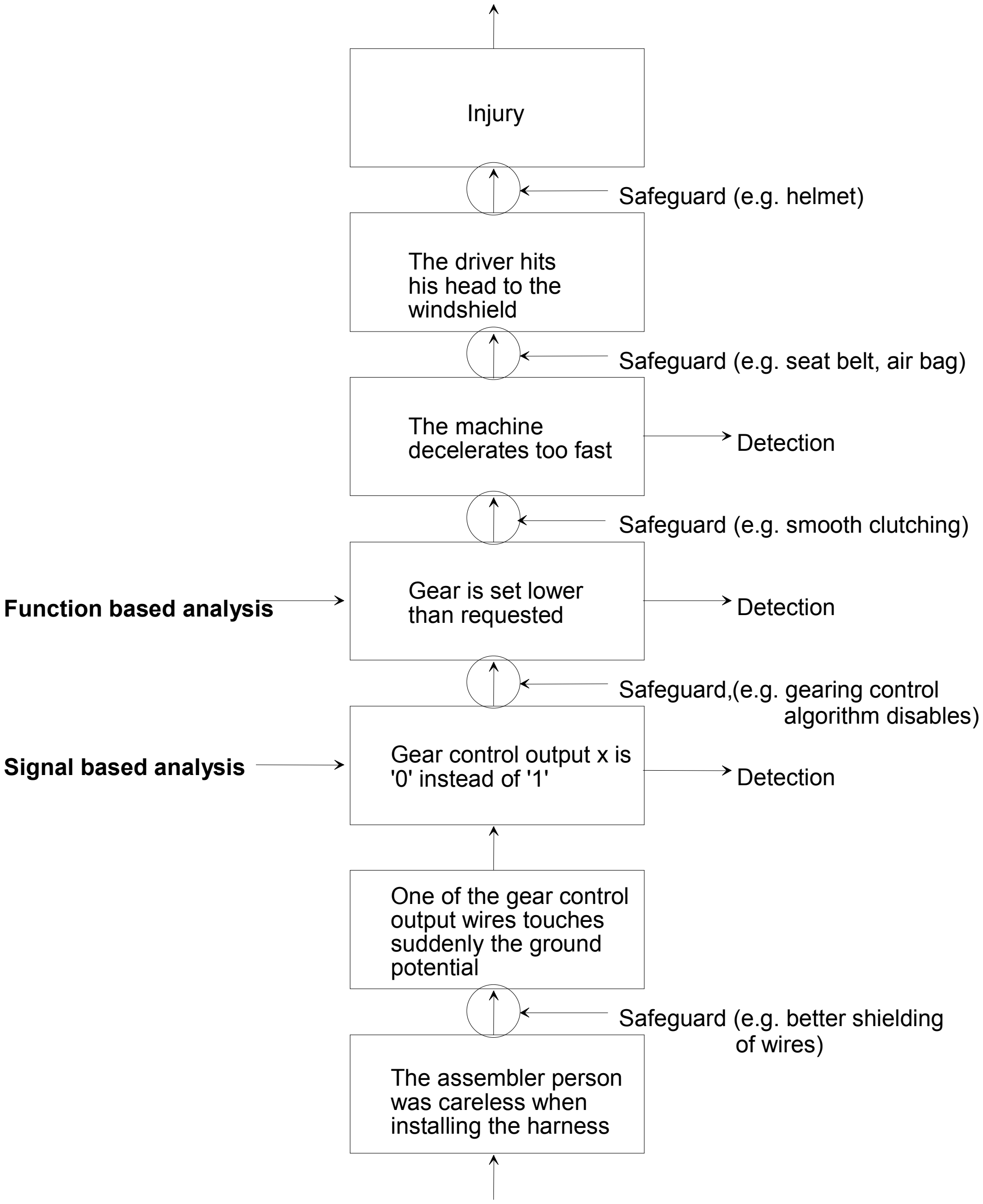

Injury
Safeguard (e.g. helmet)
The driver hits his head to the windshield
Safeguard (e.g. seat belt, air bag)
The machine decelerates too fast
Detection
Safeguard (e.g. smooth clutching)
Function based analysis
Gear is set lower than requested
Detection
Safeguard,(e.g. gearing control algorithm disables)
Signal based analysis
Gear control output x is '0' instead of '1'
Detection
One of the gear control output wires touches suddenly the ground potential
Safeguard (e.g. better shielding of wires)
The assembler person was careless when installing the harness

Figure 29. A fictitious cause-consequence scenario of a hazard; a hazard can be detected and defended against at various levels.

For each of the vertical arrows in the figure, a certain probability is attached. Furthermore, a certain probability is attached to each of the safeguards (the circles around the vertical arrows). It is difficult to determine each of the probabilities and therefore it was decided to use the two combined probabilities as explained above. Five fixed probability levels were allowed: 0.0001; 0.001; 0.01; 0.1 and 1. In this case, a fixed probability for a message error was selected: 0.0001, which is the smallest of the available probabilities, but is still considered pessimistic. This selection was based on the experience of CAN distributed systems. However, this value was only the initial estimation and was later analysed more carefully. This further analysis only took place in the case of a corruption error type for those signals for which the corruption error posed a significant risk. For the top event probability, the full scale of probability levels was used. Normally, the probability is assessed in qualitative terms, like: 'A' means very low probability, 'B' means low probability, 'C' means medium probability and 'D' means high probability (in other words, 'A' means very rare and 'D' quite common failure mode). However, it was noticed during the analysis, that using more quantitative measures like 0.0001, 0.001, 0.01, 0.1 and 1 proved out to be a more practical approach, as it was easier to place questions like: "If this Front pedal signal is corrupted, what will be the probability that the recorded top event (accident) occurs, is it usually one out of ten, one out of hundred...?" Using this method, it was rather easy to find a confident probability level for the accident probability (if the communication error occurs). The selection of the probability level was biased towards pessimistic levels – in other words, if the probability level was considered "little bit poorer than one out of ten", the probability was determined to be one.

The resulting database fields that were shown in the analysis form are listed below:

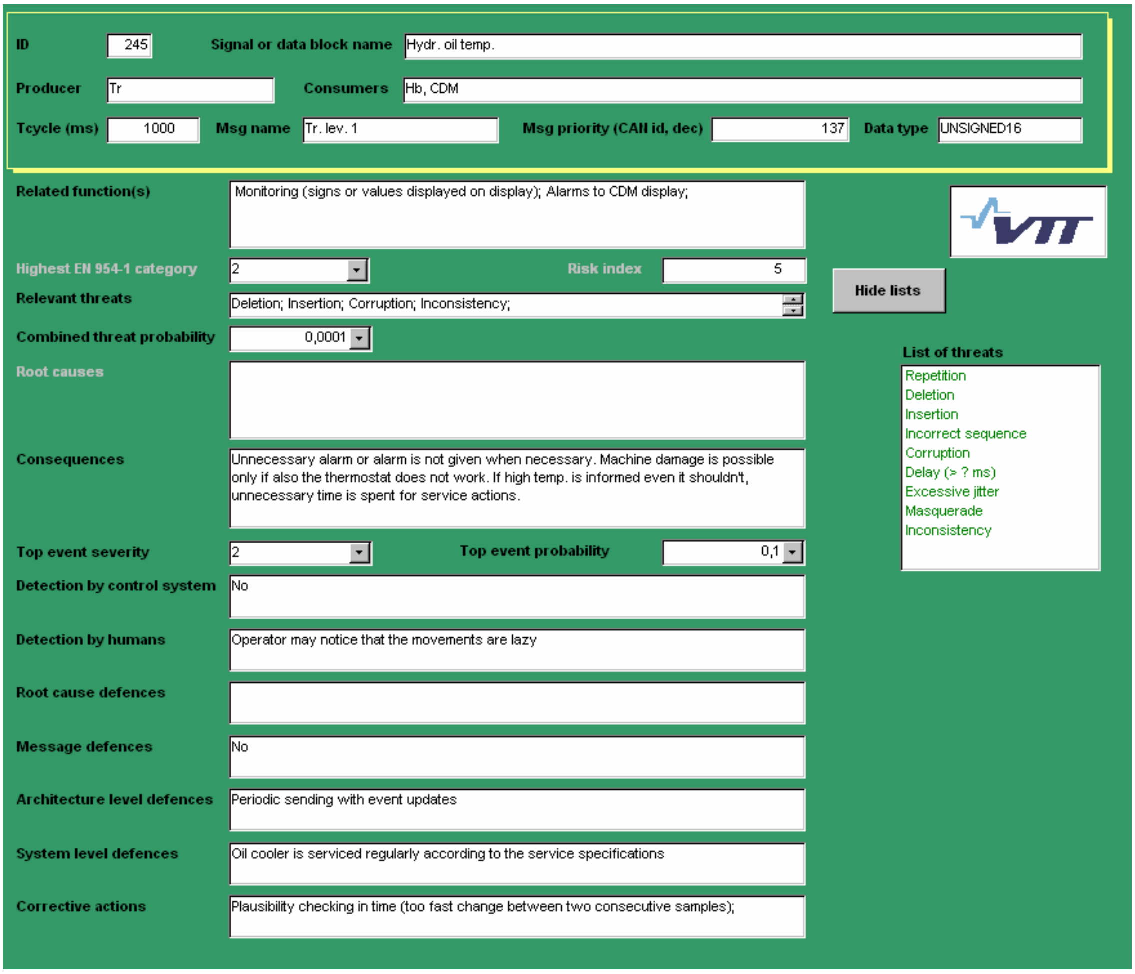
- Related functions
- Required EN 954-1 category
- Relevant threats (deviations)
- Top event severity
- Root causes
- Consequences
- Threat probability
- Top event probability
- Risk index
- Detection by control system
- Detection by humans
- Root cause defences

- Message defences
- Architecture level defences
- System level defences
- Corrective actions.

Furthermore, an additional four fields were supplied for follow-up of the corrective action recommendations:

- Actual actions
- Date of action
- Person responsible (for the corrective action)
- Rationale for the actual actions.

The analysis form is illustrated in Figure 30.



ID: 245
Signal or data block name: Hydr. oil temp.
Producer: Tr
Consumers: Hb, CDM
Tcycle (ms): 1000
Msg name: Tr. lev. 1
Msg priority (CAN id, dec): 137
Data type: UNSIGNED16

Related function(s): Monitoring (signs or values displayed on display); Alarms to CDM display;
Highest EN 954-1 category: 2
Risk index: 5
Relevant threats: Deletion; Insertion; Corruption; Inconsistency;
Combined threat probability: 0,0001
Root causes:
Consequences: Unnecessary alarm or alarm is not given when necessary. Machine damage is possible only if also the thermostat does not work. If high temp. is informed even it shouldn't, unnecessary time is spent for service actions.
Top event severity: 2
Top event probability: 0,1
Detection by control system: No
Detection by humans: Operator may notice that the movements are lazy
Root cause defences:
Message defences: No
Architecture level defences: Periodic sending with event updates
System level defences: Oil cooler is serviced regularly according to the service specifications
Corrective actions: Plausibility checking in time (too fast change between two consecutive samples);

List of threats
Repetition
Deletion
Insertion
Incorrect sequence
Corruption
Delay (> ? ms)
Excessive jitter
Masquerade
Inconsistency

Figure 30. The analysis form; one signal per page is viewed; the fields for the follow-up are not shown.

During the analysis sessions, the following fields were not filled in: Required EN 954-1 category for the particular function (was determined separately by analysis specialists), Root causes and the follow-up fields: Actual actions, Date of action, Person responsible and Rationale (these are filled by the machine manufacturer when reacting to corrective actions recommendations).

Of these, omitting the root causes -field needs more detailed explanation: The common root causes for communication errors among others are, for example, the following: wires breaking, cabling errors, HW random failures, HW ageing and EMI. It was considered that it is sufficient to determine only the consequential threats for each signal because the root causes are mostly generic and can be fixed as is done in EN 50159-2. For example for the error type 'corruption', EN 50159-2 defines twenty-two typical root causes, EMI, cross-talk, etc. Furthermore, the defences against root causes are generic (and not signal specific) and therefore, are better suited to be covered by the dependability programme of the machine manufacturer or its sub-contractors. In some cases, however, it may be necessary to consider also the root causes and related defences. For example, if a large bit error ratio (BER) causes an intolerable residual error rate for a certain signal, it may be necessary to consider better shielding of cables, if the communication protocol leaves no possibility to append additional CRC checking procedures. Nevertheless, the basic thinking behind the analysis and consequential corrective actions is that any of the typical communication error types may occur. Therefore, the defences are designed onto the message, architectural and application levels rather than against the root causes.

For each signal, the following properties were shown: signal name, producer, consumers, cycle time, message name that carries the signal, message priority (CAN id) and data type of the signal.

The goal of the modified HAZOP analysis was to alleviate the amount of work to be done. So far, this has not been the case. In order to speed up the HAZOP sessions, the following procedure was followed:

1. Initially, a list of all signals was prepared.

2. From the list, a quick selection of non-safety-critical signals was done. If there was any hesitation about the criticality of the signal, the signal was marked as safety-critical.

3. For the signals that were marked as safety-critical, a detailed analysis was made. All the deviations (repetition, deletion, insertion, incorrect sequence, corruption, delay, masquerade and inconsistency) were considered at the same time (in normal HAZOP analysis the deviations are analysed separately). The first field to be filled in was the 'Top event severity' field. If the signal was now noticed as non-safety-critical or the level of criticality was 'production loss only', the analysis of the signal was discontinued and the analysis continued with the next signal.

4. For a signal that was marked safety-critical affecting human safety, the following items were determined: Related function(s), Consequences, Top event severity, Top event probability, Detection by control system, Detection by humans, Message defences, Architecture level defences, System level defences, Corrective actions (recommendations).

The ‘Top event severity’ could be assigned the following four levels:

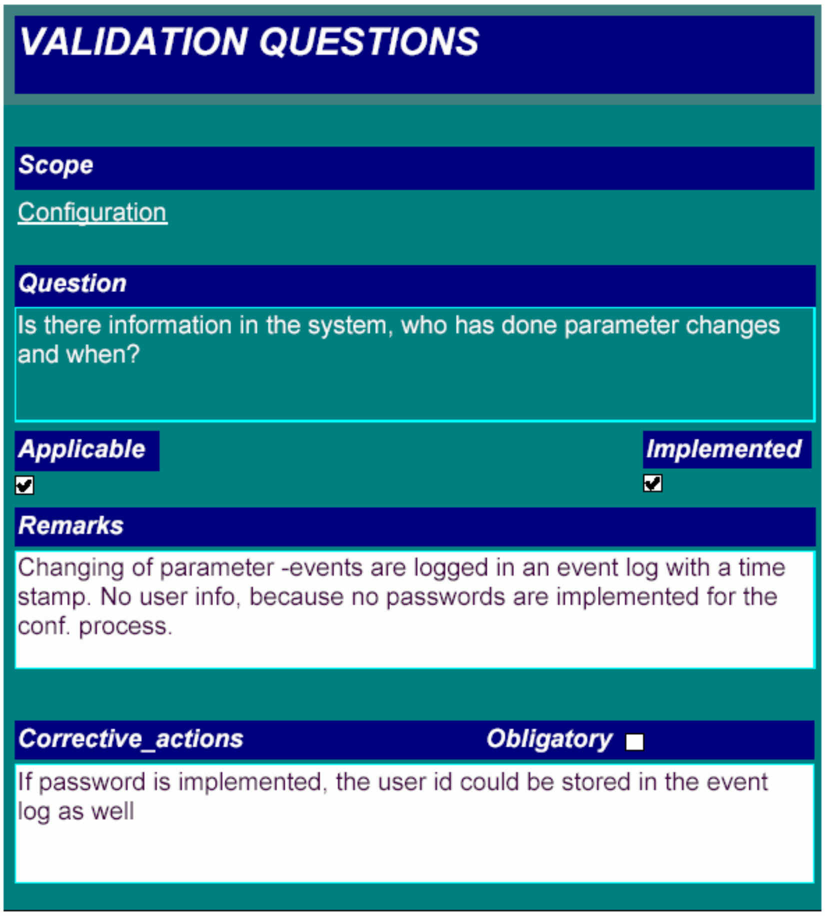
1. Production loss.

2. Machine wear-out, very small injuries perhaps.

3. Machine damage, environmental accident, and small injuries.

4. Machine damage, environmental accident, serious injuries, and deaths.

Apart from supporting the modified HAZOP analysis, the database tool was used to maintain a list of validation questions concerning distributed control systems (see Figure 31). In this case study, a list of validation questions from the Swedish Palbus-project (http://www.sp.se/electronics/RnD/palbus/) was applied. The list was extended with a set of questions concerning the configuration and documentation processes. The following categories were covered by the validation questions: membership agreement, network management, system level, configuration and documentation. The validation questions process also included a follow-up form to record the actual corrective actions, rationale behind the actions, data of action and person responsible for the action.


VALIDATION QUESTIONS

Scope

Configuration

Question

Is there information in the system, who has done parameter changes and when?

Applicable ☑ Implemented ☑

Remarks

Changing of parameter -events are logged in an event log with a time stamp. No user info, because no passwords are implemented for the conf. process.

Corrective_actions Obligatory □

If password is implemented, the user id could be stored in the event log as well

Figure 31. An example record of a validation question; follow-up fields are not shown.

With the database tool it was easy to generate appropriate analysis reports. The following reports were provided:

- List of safety-critical CAN signals
- Signals with fear of corruption
- Signals with fear of delay
- Signals with fear of deletion
- Signals with fear of excessive jitter
- Signals with fear of inconsistency
- Signals with fear of incorrect sequence
- Signals with fear of insertion
- Signals with fear of repetition
- Signals with fear of masquerade
- Corrective actions based on HAZOP analysis
- Corrective actions based on validation questions
- Corrective actions based on other questions
- Corrective actions based on detailed analysis of message threats

The utilisation of a database tool for collecting information and analysing safety-critical signals brings several advantages. Previously, the interfaces (input and output signals, communication messages and signals) of a control system were documented using general word processing or spreadsheet tools. In addition, the safety analysis, like HAZOP, is usually carried out using word processing software. Using this approach, a database tool is used instead of word processing and spreadsheet tools. This provides the following advantages:

- Signals are documented in only one database from which different reports (for example, I/O lists and communications application layer documentation) can be generated for different purposes.

- Information on the bus communication signals can be transferred from the database to a bus analyser tool. Therefore, the analyser tool can interpret the messages to reflect the application-specific message and signal names.

- Safety analysis is well supported by implementing active input forms that provide a list of possible selections, for example, of the possible communication error types. This makes the terminology consistent throughout the analysis. The lists can be extended during the analysis.

- The results of the safety analysis, for example, the suggestions for corrective actions detected during the analysis, can be printed out as reports so that they don't have to be collected manually from the analysis tables created using normal text processing software. The layout and contents of the reports are easily modified.

- The database may also include other information related to safety analysis, for example, lists of validation questions or lists of issues to be resolved, for example, due to poor documentation. The minutes of the analysis meetings can also be stored into the database.

- Residual error rate analysis as well as schedulability analysis can be incorporated into the database (was not done during the case studies).

The communication subsystem can be conveniently analysed signal by signal by using the presented tool. The tool helps to report the causes of the threats, consequences of the threats, detection methods, message and other defences, and possible corrective actions. The input for the analysis tool can be fed from, for example, I/O lists or other communication system data lists. All the I/O signals of a distributed system including the signals carried by the communication system could actually be documented in one database. Therefore, many different kinds of document sources are not needed to ensure that the safety analysis is done for up-to-date signal lists.

In general, the signal-based safety analysis framework described in Chapter 1 was found to provide an effective method to uncover communication-related risks. The main drawback is the significant amount of time needed to perform the analysis. In the first test case, the number of signals carried by the communication subsystems was nearly 500, of which about 70 were found to be safety-critical. For the safety-critical signals, the analysis time was, on average, about half an hour per signal. The whole analysis process took about eight session days. This time includes the time that was needed to do the initial rapid screening process to skip the signals that are clearly not safety-critical. The time does not include the preparatory work, like updating the communications documents and making a list of signals with their properties, nor does it include the time consumed to write the minutes of the session meetings and to make the analysis report with conclusions about the corrective actions.

Some parts of the analysis process still need to be developed. One of these is the determination of the required EN 954-1 categories for different system functions. This is, however, considered as a general level problem, which needs to be discussed and which is not only related to this type of analysis.

6. Conclusions

This report gives basic information on the safety buses available on the market as well as information on the standards to be considered in the design of a bus system especially for the safety-related applications. In addition to this, a documentation and analysis tool developed within the KETU project to support the safety analysis of bus-based communication systems at signal level is presented.

The use of special safety bus solutions is increasing all the time, partly due to the new standards and safety requirements. Each safety bus solution presented in this report has certain advantages and disadvantages, and each supplier aims to highlight the advantages of his own product (for instance, is the product a separate system or integrated into another system?). Some users consider the integrated system as an advantage, but in very safety-critical applications the normal bus and safety bus solutions are recommended to be separate. In the modification phase of a whole system or if the system changes frequently, it is wise to confine the safety relevant tasks of the system in a separate subsystem. This is done because of the great amount of validation work needed to re-analyse the whole system after a change in the non-safety-relevant system, if the safety-relevant and non-safety-relevant tasks intermingle.

The application of the new documentation and analysis tool in the signal level analysis during the design of a new bus-based application brings many benefits, as described in Chapter 0. In spite of the fact that the amount of hours is high when carrying out the signal level analysis work, the new approach presented in this report makes it possible to store all the data produced during the analysis process into the same database, making data management easier.

The use of different wireless control systems is also expanding all the time. It was noticed that the safety analysis frameworks, as described in Chapter 1, BIA guidelines and EN 50159 standard family, are applicable for wireless communications as well. This is due to the fact that wireless communications do not bring any new communication error types to messages. The common error types as described in Chapter 1 cover all the communication errors of wireless systems as well.

References

1. Anderson, S. & Górski, J. Achieving safety in distributed systems. European Workshop on Industrial Computer Systems Technical Committee 7 (EWICS TC7). Working Paper 1025. September 1997. 35 p.

2. Anon. Interbus Safety basics. Interbus Club Deutschland e.V. April 2004. 12 p.

3. Anon. SAFELOC - Computer based personal safety system with distributed I/O. 1997. IVF Research Publication 97822.

4. Anon. Safety and the fieldbus. Industrial Networking and Open Control. Vol. 7, Issue 5, 2004. http://www.industrialnetworking.co.uk/mag/v7-5/f_safety.html

5. Anon. SafetyBUS p® Safe Bus System. P. 165–202. http://www.pilzsupport.co.uk/Downloads/gtpss/Safe_Bus_Systems.pdf

6. Anon. White Paper Interbus Safety. Interbus Club Deutschland e.V. 31 p.

7. ASI. AS-Interface Safety at Work – Safety now included. AS-International Association. 2000. 19 p. http://www.as-interface.com/asi/safetyenglish.ppt

8. Barthel, H., Dönges, E., Gräff, U., Hannen, H.-T., Kühn, T., Lausberg, G., Laux, T. & Stripf, W. Profibus-DP/PA. Profisafe, Profile for Fail-safe Technology, V1.0. 1999. 56 p. http://www.itk.ntnu.no/fag/fordypning/SIE3092/SIE30AB/PDF/ProfiSafe-Profil-100e.pdf

9. Belschner, R. et al. FlexRay Requirements Specification. Version 2.0.2, 9th of April, 2002 by BMW AG, Daimler-Chrysler AG, Robert Bosch GmbH, General Motors/Opel AG.

10. Carlsson, H., Cserepes, B., Danielsen, L., Gårdman, B., Jacobson, J., Karlsson, T. & Pettersson, I. Safeloc. Computer based personal safety system with distributed I/O. Mölndal. IVF-skrift 97822. 1997. 195 p. + app. 10 p.

11. Edler, H., Eriksson, J. E., Hedberg, J. & Sjöström, H. Definitions. Version 2.0. PALBUS Task 10.1. 1 April 2001. http://www.sp.se/electronics/rnd/palbus/

12. EN 50159-1. Railway applications. Communication, signalling and processing systems. Part 1: Safety-related communication in closed transmission systems. Brussels, European Committee for Electrotechnical Standardization. 1.10.2001. 16 p.

13. EN 50159-2. Railway applications. Communication, signalling and processing systems. Part 2: Safety-related communication in open transmission systems. Brussels, European Committee for Electrotechnical Standardization. 1.10.2001. 44 p.

14. EN 60204-1. Safety of machinery. Electrical equipment of machines. Part 1: General requirements. Brussels, European Committee for Electrotechnical Standardization. 21.12.1998. 181 p.

15. EN 954-1. 1996 Safety of machinery. Safety related parts of control systems. General principles for design. Brussels, European Committee for Electrotechnical Standardization.

16. FAET; FAEM III, BIA, Prüfung und Zertifierung von "Bussysteme für die Übertragung sicherheitsrelevanter Nachrichten", Stand 28.05.2000 (English version available: Proposal of a Guideline for the Test and Certification of "Bus Systems for the Transmission of Safety Relevant Messages"; Fachausschuss Elektrotechnik, Gustav-Heinemann-Ufer 130, 50698 Köln).

17. FlexRay Consortium. FlexRay Communications System Protocol Specification 2.0. 2004. 224 s.

18. Führer, Th., Müller, B., Dieterle, W., Hartwich, F., Hugel, R. & Walther, M. Time Triggered Communication on CAN (Time Triggered CAN - TTCAN). Robert Bosch GmbH; Proceedings 7th International CAN Conference (iCC 2000), Amsterdam 24.–25.10.2000. 7 p. http://www.can.bosch.com/docu/CiA2000Paper_1.pdf

19. Hablawetz, D. Safe and redundant networking – Ethernet as a basis for fail-safe and fault-tolerant automation. PLC Symposium, Cologne, May 3–4, 2000. 9 p.

20. Handermann, F. Communication with SafeEthernet. Praxis Profiline – Industrial Ethernet. April 2002. 1 p. http://www.iceweb.com.au/sis/Hima/Safe%20Ethernet.pdf

21. Hérard, J., Sjöström, H., Olsen, O., Stålhane, T., Juul Wedde, K., Løken, T., Söderberg, A., Malm, T. & Hietikko, M. “Round Robin Tests” of Safety-related Control System for Machinery - comparison of validation results. Nordtest Project 1504-00. 2000. 31 p.

22. IEC 60870-5-1. Telecontrol equipment and systems. Part 5: Transmission protocols - Section One: Transmission frame formats. International Electrotechnical Commission. February 1990. 88 p.

23. IEC 61508-2. Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 2: Requirements for electrical/electronic/programmable electronic safety-related systems. International Electrotechnical Commission. May 2000. 72 p.

24. IEC 61882. Hazard and operability studies (HAZOP studies) – Application guide. International Electrotechnical Commission. May 2001. 113 p.

25. ISO/CD 11898-4. Road vehicles – Controller area network (CAN) – Part 4: Time triggered communication.

26. ISO/DIS 15998.2. Draft. Earth-moving machinery – Machine-control systems (MCS) using electronic components – Performance criteria and tests.

27. Malm, T., Hietikko, M. & Alanen, J. Distributed systems for safety-related applications in machinery. 4th International Conference on Machine Automation ICMA'02. Tampere, 11–13 Sept. 2002. IEEE Robotics and Automation Society; IMEKO; MET; TEKES; eTampere (2002), s. 395–402.

28. Meyer-Gräfe, K. Solutions for safety technology using Interbus safety. Control Engineering Europe. June/July 2002, p. 36–39.
http://www.manufacturing.net/ctl/article/CA224031

29. Pilz. Safe Integrated Automation. Control and Monitoring Technology, Safety Technology, Operator Terminals, Industrial Computers. Ostfildern, Pilz Gmbh & Co. June 2001.

30. Redmill, F. J. Dependability of Critical Computer Systems 2. London, Elsevier Science Publishers Ltd. 1989. 286 p.

31. Reinert, D. & Schaefer, M. Sichere Bussysteme für die Automation. Heidelberg, Hütlig. 209 p.

32. Schmersal. Safety Switching Devices with Integral Bus interface. 2 p. Available: http://www.schmersal.de/kasbase/bilddata/broschue/p-info/b_asbpp2.pdf

33. Sivencrona, H., Hedberg, J. & Röcklinger, H. Comparative Analysis of Dependability Properties of Communication Protocols in Distributed Control Systems. Pålbus project SP. 2001. 39 p.
http://www.sp.se/electronics/RnD/palbus/Reports/PALBUS_10_2.pdf

34. Vasko, D. A., Vandesteeg, K. W. & Lenner, J. A. Introduction to DeviceNet Safety. 7th international CAN Conference, Amsterdam, Netherlands, 24.–25.10.2000. Erlangen: CAN in Automation. Pp. 08-07–08-16.

35. Welch, J. L. & Lynch, N. A. A New Fault-Tolerant Algorithm for Clock Synchronization. Information and Computation, Vol. 77, No. 1, April 1998. Pp. 1– 36.

36. X-by-wire team, X-by-wire Safety-related Fault Tolerant Systems in Vehicles. Final report of Brite-Euram project No. BE 95/1329. 1998. 69 p.
http://www.vmars.tuwien.ac.at/projects/xbywire/docs/final.doc

Appendix A: Useful links

http://www.sp.se/electronics/RnD/palbus/

http://www.odva.org

http://www.safetybus.com/

http://www.vmars.tuwien.ac.at/projects/xbywire

http://www.vmars.tuwien.ac.at/projects/tta

http://www.tttech.com/

http://www.ttagroup.org/index.htm

http://www.flexray.com/products.php

http://www.hima.com/

http://www.interbusclub.com/en/news/Fly_IB_Basics_Safety_gb.pdf

http://www.schmersal.se/Finland/News_fi/Einfo_fi/Esalan_fi.html

http://www.pilzsupport.co.uk/Downloads/gtpss/Safe_Bus_Systems.pdf

Published by

Series title, number and report code of publication

VTT Research Notes 2265
VTT–TIED–2265

Author(s)
Alanen, Jarmo, Hietikko, Marita & Malm, Timo

Title

Safety of Digital Communications in Machines

Abstract

The utilisation of digital communications in safety-related machine control systems has been widely extended during the last ten years. This new technology brings about an additional safety engineering challenge compared to a single controller case where only simple wired communication is needed to execute safety-related functions.

The scope of this report is safety-related serial communications in machine automation. Standards and guidelines that include information dealing with safety-related communications and the design of safety-related communication systems are introduced. The typical message error types or threats relating to serial mode transmission as well as defence methods against these threats are introduced.

There are several safety buses available for safety-related machine and automation applications. The basic information on these safety buses is given in this report. This information includes methods against possible transmission errors. Most of the safety bus solutions are commercially available from several suppliers. Some safety bus solutions that are not commercially available are also described.

A documentation and analysis tool to support the safety analysis of bus-based communication systems at signal level is presented. The tool is based on database software, and the analysis method is based on Hazard and Operability study (HAZOP). This tool was developed within this project and tested with two case studies, which were distributed control systems of machine automation applications. The advantages of using this tool are presented.

A serial mode wireless communication is also increasing in safety-related machine applications, and therefore the wireless message transmission possibility is also considered. It was noticed that the safety analysis framework described in this report is applicable for wireless communication as well. Wireless communication does not bring any new message error types. Therefore, the same defence methods against message errors are true for wireless systems as well.

Keywords
serial mode transmission, safety-related communications, safety bus, machines, HAZOP

Activity unit
VTT Industrial Systems, Tekniikankatu 1, P.O.Box 1307, FIN–33101 TAMPERE, Finland

ISBN 951–38–6502–9 (soft back ed.) 951–38–6503–7 (URL: http://www.vtt.fi/inf/pdf/)	Project number G1SU00477

Date November 2004	Language English	Pages 93 p. + app. 1 p.	Price B

Name of project The implementation and maintenance of safety-related functions in machine control systems based on field bus and wireless communication (KETU project)	Commissioned by Finnish Work Environment Fund, Sandvik Tamrock, Timberjack, Metso Minerals, Kalmar Industries, Kone Cranes
Series title and ISSN VTT Tiedotteita – Research Notes 1235–0605 (soft back edition) 1455–0865 (URL: http://www.vtt.fi/inf/pdf/)	Sold by VTT Information Service P.O.Box 2000, FIN–02044 VTT, Finland Phone internat. +358 9 456 4404 Fax +358 9 456 4374

VTT RESEARCH NOTES 2265 Safety of Digital Communications in Machines

The utilisation of digital communications in safety-related machine control systems has been widely extended, bringing about additional safety engineering challenges. This report deals with safety-related serial communication that can be applied in machine automation. Standards and guidelines that include guidance for the design and implementation of safety-related communication systems are introduced. The typical message error types or threats relating to serial mode transmission as well as defence methods that can be used against these threats are described. Wireless communication is also kept in view. The basic information on commercially available safety buses is given, including defences against possible transmission errors. A documentation and analysis tool to support the safety analysis of bus-based communication systems at signal level is introduced. The tool is based on database software and the analysis method is based on Hazard and Operability study (HAZOP). The analysis and documentation tool was developed and tested within this study with two machine automation applications. The advantages of using it are discussed.

Tätä julkaisua myy

VTT TIETOPALVELU
PL 2000
02044 VTT
Puh. (09) 456 4404
Faksi (09) 456 4374

Denna publikation säljs av

VTT INFORMATIONSTJÄNST
PB 2000
02044 VTT
Tel. (09) 456 4404
Fax (09) 456 4374

This publication is available from

VTT INFORMATION SERVICE
P.O.Box 2000
FIN–02044 VTT, Finland
Phone internat. + 358 9 456 4404
Fax + 358 9 456 4374

ISBN 951–38–6502–9 (soft back ed.)
ISSN 1235–0605 (soft back ed.)

ISBN 951–38–6503–7 (URL: http://www.vtt.fi/inf/pdf/)
ISSN 1455–0865 (URL: http://www.vtt.fi/inf/pdf/)