Institutionen för systemteknik
Department of Electrical Engineering

Examensarbete

Evaluation of Communication Interfaces for Electronic Control Units in Heavy-duty Vehicles

Examensarbete utfört i fordonsteknik
vid Tekniska högskolan vid Linköpings universitet
av

Henrik Johansson

LiTH-ISY-EX--12/4580--SE

Linköping 2012

Linköpings universitet
TEKNISKA HÖGSKOLAN

Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

Linköpings tekniska högskola
Linköpings universitet
581 83 Linköping

Evaluation of Communication Interfaces for Electronic Control Units in Heavy-duty Vehicles

Examensarbete utfört i fordonsteknik
vid Tekniska högskolan vid Linköpings universitet
av

Henrik Johansson

LiTH-ISY-EX--12/4580--SE

Handledare: **Mats Halvarsson**
RECU, Scania CV AB
Andreas Thomasson
ISY, Linköpings universitet

Examinator: **Per Öberg**
ISY, Linköpings universitet

Linköping, 1 juni 2012

Avdelning, Institution
Division, Department

Division of Vehicular Systems
Department of Electrical Engineering
SE-581 83 Linköping

Datum
Date

2012-06-01

Språk
Language

☐ Svenska/Swedish
☒ Engelska/English
☐ ______________

Rapporttyp
Report category

☐ Licentiatavhandling
☒ Examensarbete
☐ C-uppsats
☐ D-uppsats
☐ Övrig rapport
☐ ______________

ISBN
—

ISRN
LiTH-ISY-EX--12/4580--SE

Serietitel och serienummer
Title of series, numbering

ISSN
—

URL för elektronisk version

http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-78665

Titel
Title

Utvärdering av kommunikationsgränssnitt för styrenheter i tunga fordon

Evaluation of Communication Interfaces for Electronic Control Units in Heavy-duty Vehicles

Författare
Author

Henrik Johansson

Sammanfattning
Abstract

The number of electronic control units in heavy-duty vehicles has grown dramatically over the last few decades. This has led to the use of communication buses to reduce the complexity and weight of the networks. There are reasons to believe that the de facto standard communication interface in the automotive industry, the Controller Area Network, is obsolete in some areas. Hence an evaluation of available communication interfaces is needed.

This study focuses on lower levels of the Open Systems Interconnect (OSI) model. Initially a theoretical study is presented in order to give an overview of automotive embedded systems in general and different communication interfaces in particular. Ethernet and FlexRay are identified as two interfaces of interest for future use in Scanias vehicles. The former is new in automotive applications but is believed to become popular over the years to come. A possible use of this interface could be as a backbone to take the load off other interfaces. The use of FlexRay in Scanias vehicles is limited because of the modular system used and the static scheduling needed. It could however be used between mandatory ECUs where the nodes and the messages are all known beforehand.

The report also contains the result from emission measurements on a number of interfaces performed using a stripline antenna in a shielded enclosure. Strong conclusions can not be drawn since it's hard to tell what the transceivers, circuit boards and interfaces contributed to in the spectra with the method used. The FlexRay hardware is worse than for the other interfaces. Similarities can be seen between low-speed and high-speed CAN but it could be characteristics of the transceivers used rather than the interface itself.

Nyckelord
Keywords

Embedded Systems, Automotive

Abstract

The number of electronic control units in heavy-duty vehicles has grown dramatically over the last few decades. This has led to the use of communication buses to reduce the complexity and weight of the networks. There are reasons to believe that the de facto standard communication interface in the automotive industry, the Controller Area Network, is obsolete in some areas. Hence an evaluation of available communication interfaces is needed.

This study focuses on lower levels of the Open Systems Interconnect (OSI) model. Initially a theoretical study is presented in order to give an overview of automotive embedded systems in general and different communication interfaces in particular. Ethernet and FlexRay are identified as two interfaces of interest for future use in Scanias vehicles. The former is new in automotive applications but is believed to become popular over the years to come. A possible use of this interface could be as a backbone to take the load off other interfaces. The use of FlexRay in Scanias vehicles is limited because of the modular system used and the static scheduling needed. It could however be used between mandatory ECUs where the nodes and the messages are all known beforehand.

The report also contains the result from emission measurements on a number of interfaces performed using a stripline antenna in a shielded enclosure. Strong conclusions can not be drawn since it's hard to tell what the transceivers, circuit boards and interfaces contributed to in the spectra with the method used. The FlexRay hardware is worse than for the other interfaces. Similarities can be seen between low-speed and high-speed CAN but it could be characteristics of the transceivers used rather than the interface itself.

Acknowledgments

First of all I want to thank Scania for giving me the opportunity to write this master thesis. I also want to thank my supervisor Mats Halvarsson for his support, Mikael Johansson for sharing his knowledge and all other colleagues at RECU, Scania.

Andreas Thomasson at the department of Electrical Engineering at Linköping University has been helpful all the way, always answered my questions and guided me when needed. My examiner at the same department, Per Öberg, has also supported me and made sure that my work has progressed.

I also want to thank Irène Wahlqvist at UTYI for borrowing litterature for my thesis from libraries all over Europe. This has been an invaluable resource during my work!

Last but not least I want to thank Sune for being there for me.

Linköping, Juni 2012

This page is unintentionally left blank.

Notation

Abbreviations

Abbreviation	Meaning
CAN	Controller Area Network
CRC	Cyclic Redundancy Check
CSMA	Carrier Sense Multiple Access
CSMA/CA	CSMA with Collision Avoidance
CSMA/CD	CSMA with Collision Detection
CSMA/CR	CSMA with Collision Resolution
D2B	Domestic Digital Bus
DUT	Device Under Test
ECU	Electronic Control Unit
EMC	Electromagnetic Compatibility
FTDMA	Flexible Time Division Multiple Access
HMI	Human-Machine Interface
IEEE	Institute of Electrical and Electronics Engineers
I^2C	Inter-Integrated Circuit
LIN	Local Interconnect Network
MOST	Media Oriented Systems Transport
OSI	Open Systems Interconnect
SAE	Society of Automotive Engineers
TDMA	Time Division Multiple Access
TTA	Time-Triggered Architecture
TTCAN	Time-Triggered Control Area Network
TTP/A	Time-Triggered Protocol class A
TTP/C	Time-Triggered Protocol class C
VAN	Vehicle Area Network

Contents

1 Introduction

This section will give the reader an understanding to why and how this study has been conducted (in section 1.1-1.2 and 1.4 respectively). What the study should result in is presented in section 1.3 and its delimitations in section 1.5.

1.1 Background and Problem Definition

Today's heavy-duty vehicles contain a large number of electronic control units (ECUs) connected to sensors and actuators. The number of ECUs has increased over the years because of added functionality and technological advances. This development has made the use of communication buses necessary to decrease the point-to-point interconnections between the nodes. Not only has it made the systems less complex but it has also increased reliability and lowered cost and weight.

Scania uses J1939, a higher level Controller Area Network (CAN) protocol for inter-ECU communication in their vehicles. There are reasons to believe that the development of the embedded systems have made the use of CAN obsolete in some areas. Hence there's a need to evaluate other interfaces.

The development within the automotive industry imposes a number of requirements on the communication interfaces used. The amount of data sent between ECUs has increased rapidly which makes bandwidth an important aspect. CAN is relatively limited when it comes to bandwidth compared to many other interfaces. Electromagnetic compatibility (EMC) is another important factor and there's a lack in knowledge in this area at Scania today. The electromagnetic radiation the bus causes is an area of concern since it can affect other electronic devices in the vehicle. Some customers are also asking for a low radiation environment, especially in vehicles for military use. The physical distance for cables between ECUs can in some vehicles (mainly in articulated buses) reach the limit for the interface. This makes the maximum communication bus length for reliable information transfer of interest to Scania.

Some studies have been conducted in this area earlier at Scania. These have however all been interface specific and covering higher levels of the Open Systems Interconnection model (OSI) described in section 2.2. A broad evaluation of available interfaces is therefore needed with a hardware focus on the ones most viable for implementation i Scanias vehicles.

1.2 Purpose

The purpose of this report is to increase the knowledge about communication interfaces suited for implementation in Scanias vehicles. The theoretical study is meant to give comparative results between the different interfaces from the perspective of lower layers in the OSI model. The practical tests are conducted to further examine the differences and similarities between them.

1.3 Goals

This study should result in a brief description of communication interfaces available on the market today. The ones suitable for implementation in Scanias vehicles as a replacement for CAN will be closer studied. The test results will be presented together with a conclusion on how the different interfaces compare to each other from a hardware perspective, especially with a focus on emission. The test method used will also be evaluated.

1.4 Method

Initially a study of the ECUs in Scanias was carried out. The purpose of this was to get an overview of the units used in the vehicles and the requirements on the communication interfaces used. Internal Scania documents and personal communication was used for this. Parallel to this a literature servey was performed in order to get an overview of available interfaces. Four of them (apart from CAN), FlexRay, Ethernet, TTP/C (Time-Triggered Protocol class C) and TTCAN (Time-Triggered CAN) were selected for a closer study because of their characteristics.

Although Ethernet emerged as one of the most reasonable alternative to CAN it was never examined closer in a lab environment. The requirements in Scanias vehicles, the development of vehicle buses and the time horizon for implementation all spoke in favor of this interface. The equipment available however didn't make it possible to conduct any tests with clear results. Instead comparative tests were conducted on low and high speed CAN, FlexRay and LIN (Local Interconnect Network) based on emission levels from the buses. Evaluation boards were used to create a network for measurements in the stripline antenna. A Scania ECU was also used to see how it compares to the other tested interfaces.

The literature used is mainly books and articles in the area of embedded systems.

1.5 Delimitations

This study is focused on wired serial communication in the automotive domain. Interfaces developed for use in avionics, trains or automation like OPENcan and Multifunction Vehicle Bus are not studied. Protocols developed for diagnostics like K-line will also not be covered. The study focuses on lower levels of the OSI-model, particularly the physical layer and the data link layer. Higher levels are covered when appropriate.

Although immunity is an important part of electromagnetic compatibility it is not examined closer in this study. Instead the report focuses on emission levels from the different buses.

1.6 Outline of the Report

Chapter 1, Introduction: The background to why this study has been conducted is presented along with its goals the method used.

Chapter 2, Automotive Embedded Systems: The reader is presented with an introduction to embedded systems in automotive applications.

Chapter 3, Available Communication Interfaces: A number of communication interfaces used within the automotive industry are described here from the perspective of lower levels in the OSI model.

Chapter 4, Theoretical Results: This chapter contains the results from the theoretical study based on chapter 2 and 3.

Chapter 5, Measurements: The measurements and results from the tests conducted in the lab environment.

Chapter 6, Results and Conclusion: The conclusions based on the theoretical study and the tests are presented here.

Chapter 7, Future Work: Ideas and suggestions for the future and studies to come.

2

Automotive Embedded Systems

This section starts with a brief description of the background to the use of embedded systems within the automotive industry (section 2.1). A more general description of the OSI model used to define communication protocols and available network topologies can then be found in section 2.2 and 2.3. What is required of an embedded system in a vehicle (described in section 2.4) is highly dependant on where in the vehicle it operates. This has led to the definition of a number of functional domains (described in section 2.5) and a division of the communication interfaces into separate classes (section 2.6). There are two basic control paradigms for how information is exchanged between ECUs. A description of this can be found in section 2.7 followed by different ways of handling errors in 2.8. Electromagnetic compatibility will be given some attention (2.9) as well as an overview to the CAN network used in Scanias vehicles today (section 2.10).

2.1 Background

The number of electronic systems in vehicles has grown drastically since they were first introduced in the 1970s. These systems have made it possible to improve both safety and comfort and at the same time add functionality. Antilock breaking system (ABS), active suspensions, engine control and multimedia applications are just a few examples of this [Navet and Simonot-Lion, 2009a]. Legislation regulating exhaust emissions is another factor behind the use of embedded systems in vehicles [Mayer, 2008, Simonot-Lion and Trinquet, 2009]. Initially every function was implemented in a stand-alone ECU but subsequently the functionality was distributed to multiple units. This made it possible to increase the functionality but soon proved to be too complex and expensive because of the interconnections. The systems also grew large and added a lot of extra weight to

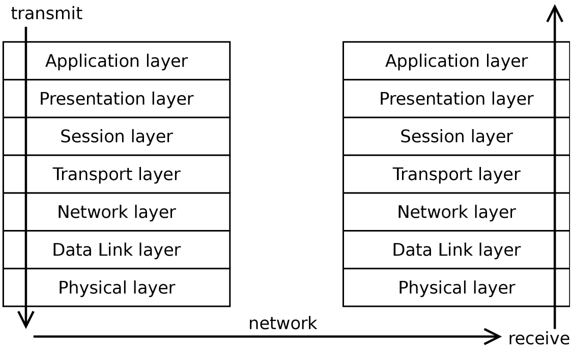


Figure 2.1: *The Open Systems Interconnect (OSI) model. The seven hierarchical levels can be clearly seen with the application layer closest to the end user. The transmission and reception of a message can also be seen in the figure.*

the vehicles. This development lead to the use of other network topologies, such as the bus, in the vehicles [Mayer, 2008, Navet and Simonot-Lion, 2009a].

2.2 OSI Model

The Open Systems Interconnection Model (OSI) is a way of defining communication protocols. It consists of seven abstract layers independent from hardware or software implementation. The layers are in a strict hierarchical order (as seen in figure 2.1) with related functions grouped together. The OSI model was not defined with real-time systems, field buses or embedded systems in mind which has to be considered. Most of today's fieldbus only are only defined in the lower levels of the OSI model [Zucker and Dietrich, 2001]. The upper three layers are independent of the medium used to transfer the message. The transport layer separates them from the network dependent lower three layers [Nolte, 2006].

Layer 1: Physical Layer: The physical layer is the lowest level in the OSI model. It defines actual medium connecting nodes together and its electrical and mechanical characteristics [Zucker and Dietrich, 2001]. This layer specifies bit representation and synchronization, electrical and optical levels, cable specifications, hubs and line termination [Paret and Riesco, 2007].

Layer 2: Data link Layer: The second layer, the data link layer, includes message framing, arbitration and the handling of acknowledgements [Paret and Riesco, 2007]. It is responsible for an error free transfer between nodes in a network and so error detection and error signalling is defined here as well [Paret and Riesco, 2007, Zucker and Dietrich, 2001]. The data link layer is also responsible for the addressing of messages in a point-to-point interconnection [Zucker and Dietrich, 2001].

Layer 3: Network Layer: The network layer is responsible for routing of messages in a network, from the source to the destination. It adds additional addressing unrelated to the addresses on the data link layer. It is also responsible for establishing, reestablishing and terminating network connections. [Zucker and Dietrich, 2001]

Layer 4: Transport Layer: This layer controls the data flow between two end users and guarantees that a message reaches its endpoint. It also assigns logical addresses to the physical addresses in the network layer. [Zucker and Dietrich, 2001]

Layer 5: Session Layer: The session layer defines how end users starts and terminates a session as well how data-exchange is established. [Zucker and Dietrich, 2001]

Layer 6: Presentation Layer: How the received information is to be interpreted is handled by the presentation layer. [Zucker and Dietrich, 2001]

Layer 7: Application Layer: All lower layers in the OSI model are accessed through the application layer. It offers an interface that can be used by an application. [Zucker and Dietrich, 2001]

2.3 Network Topologies

This section will present different network topologies along with their advantages and disadvantages. Examples of the physical layouts will be shown in figures to illustrate their characteristics.

2.3.1 Point-to-point interconnections

Point-to-point interconnections was the single most common way to connect ECUs until the beginning of the 1990s. Every single function was added as a standalone unit in the vehicles [Robert Bosch GmbH, 2011, Navet and Simonot-Lion, 2009a]. As the functions were divided over multiple ECUs the need for communication increased. Linking all nodes together increases the number of communication channels in the order of n^2, where n is the number of ECUs. The complexity, cost and weight increased as the networks grew and at the same time the reliability decreased. These issues motivated the use of other network topologies [Navet and Simonot-Lion, 2009a]. Figure 2.2 shows an example of a point-to-point network.

2.3.2 Bus

The bus, also known as a linear bus, is the simplest network topology. It consists of one or more wires without any equipment to amplify or route the signals as seen in figure 2.3a and 2.3b. A large number of connected nodes can affect the performance negatively since only one node at the time can send data [Rausch, 2007]. The bus branches, stubs, are unterminated and can give rise to signal reflections (see section 2.9). This problem can be considered non-existent for lower frequencies but it will be more tangible as the frequency increases [Paret

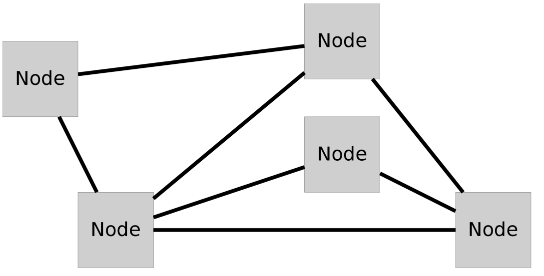
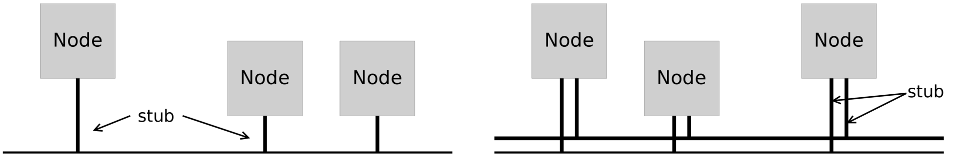


Figure 2.2: Point-to-point interconnections between five nodes. The wiring grows rapidly when the number of nodes increases. This led to the use of other network topologies such as the bus.



(a) 1 channel bus *(b) 2 channel bus*

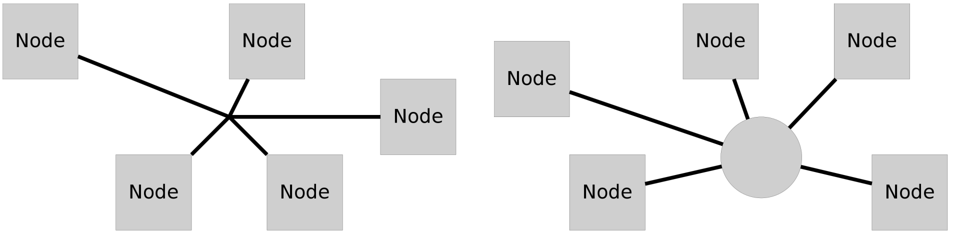
Figure 2.3: Example of bus networks. These are small, cheap and easy to extend by just adding a node to the shared wire. The stubs can give rise to signal reflections and should be kept to a minimum.

and Riesco, 2007]. The physical layer requirements of an interface often sets constraints on their length to minimize this problem. Line termination in the end of the bus is also required to minimize reflections. One major advantage of the bus topology is its simplicity. It is small, cheap and easy to extend [Jarboe et al., 2002]. What differs it from a passive star is the number of splices, a bus always has more than one [Rausch, 2007].

2.3.3 Star

There are two types of star networks, active and passive. The former consists of a number of nodes connected to each other in one central point called an active star. These stars amplify the incoming signal and can broadcast it to all other nodes or route it to the target node. The wires are properly terminated in both ends since the star is an active device [Paret and Riesco, 2007]. Passive stars merely acts as a connection point and does not amplify or route the signals [Jarboe et al., 2002, Rausch, 2007]. It differs from the bus layout because of its single splice [Rausch, 2007]. Star networks are flexible since it's easy to add or remove nodes. It can however require a lot of wiring and the star is a weak spot in the layout [Jarboe et al., 2002]. Examples of an active and a passive star network can be seen in figure 2.4a and 2.4b respectively.

Active star networks can be connected in cascades by adding another active star



(a) Passive star network *(b) Active star network*

Figure 2.4: *The passive star network to the left connects the nodes together in one single point. No amplification or routing is performed here. This can however be done in the star network to the right because of the centrally positioned active star node.*

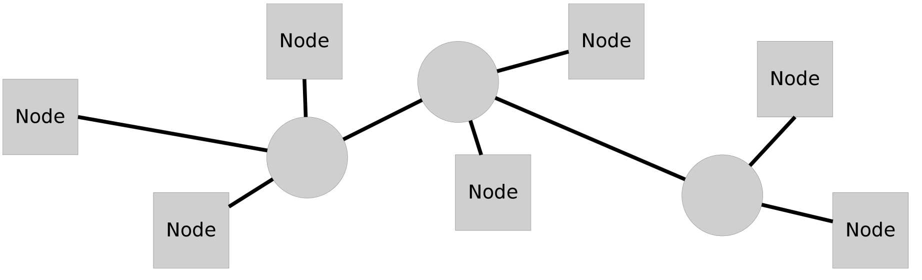


Figure 2.5: *Three cascaded active stars with a total of seven nodes. The active star nodes can act as repeaters over long distances. This does however add extra propagation delay throughout the network.*

where a node would otherwise go [Rausch, 2007, Lim et al., 2011]. ECUs in vehicles are often placed in distinct positions and the distance between them can be long. Clusters of ECUs centered around cascaded active stars can reduce the cabling and increase the determinism [Lim et al., 2011]. The number of active stars in a network is limited because of the extra delay each one of them causes [Cena and Valenzano, 2009]. This type of network can be seen in figure 2.5.

2.3.4 Hybrid

Hybrid networks are simply a combination of other network topologies that does not exhibit the characteristics of a standard network topology [Lim et al., 2011]. An example of a hybrid network can be seen in figure 2.6.

2.3.5 Ring

Every one of the n nodes has one input and one output connecting to its neighbours using a total of $n-1$ point-to-point interconnections as seen in figure 2.7. A node can either be in passive mode and only bypass the data or active mode

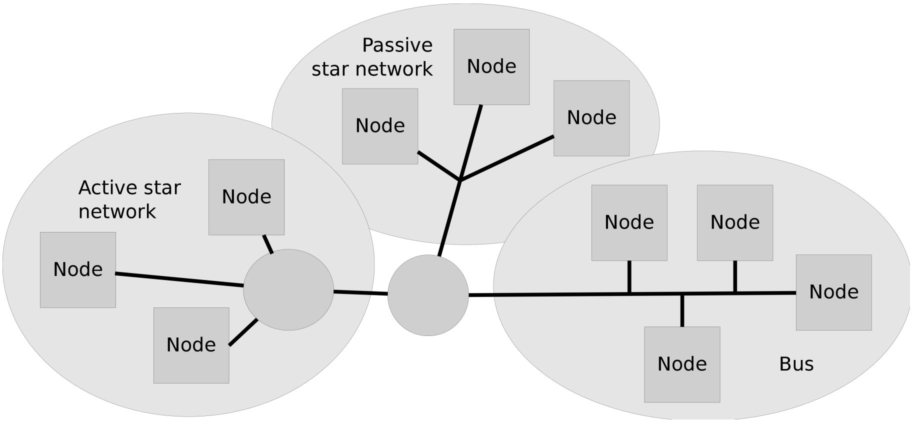


Figure 2.6: *A hybrid network consisting of an active star network, a passive star network and a linear bus. What makes it a hybrid network is that its characteristics differs from that of a standard topology.*

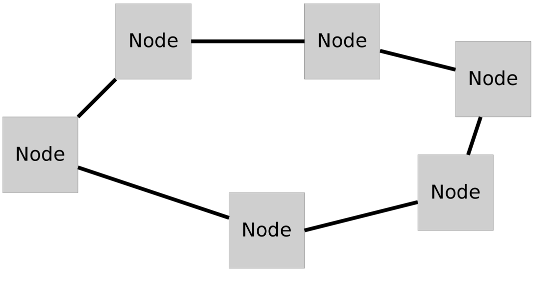


Figure 2.7: *A ring network consisting of six nodes resulting in five interconnections. A node is said to be active if it modifies the received data and passive if it just passes it on to the next node.*

and modify it [Strang and Röckl, 2008].

2.3.6 Daisy Chain

A daisy chain is basically a number of nodes connected in series as seen in figure 2.8. Prioritization can be assigned based on a node's position in the network. [Maxim, 2012]

2.4 Requirements

Embedded systems have become an important way to add functionality to vehicles. It can also replace mechanical or hydraulic systems or even implement functionality that otherwise would be impossible to implement. This requires a lot from the systems used as will be seen in this section. Today's vehicle manu-

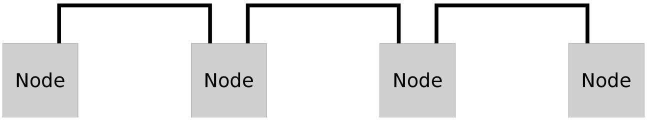


Figure 2.8: *A daisy chain network consisting of four nodes connected in series.*

facturer also have to consider customers needs, legislation and competition on a globalized market when designing their products. Technological progress in both hardware and software has made it possible to meet all of these demands [Keskin, 2009]. The requirements on an embedded systems depends on its use in the vehicle to a high degree. These functional domains are explained in section 2.5 below.

2.4.1 Deterministic Behaviour

Many of the safety-critical systems in automotive applications requires a deterministic behaviour to guarantee the correct reception of a message. Real-time requirements is another important factor since timing and a maximum latency is crucial for some systems [Nolte, 2006, Pimentel et al.]. Static scheduling (described in section 2.7.1) can be used in to meet hard deadlines [Marwedel, 2006].

2.4.2 Reliability

Fault tolerance, error detection and error handling is important in safety-critical systems. A detailed description of this can be found in section 2.8. [Nolte, 2006]

2.4.3 Bandwidth

The required bandwidth for automotive embedded systems has increased over the years as the number of ECUs has grown. Telematics, multimedia and HMI applications have further added to this [Pimentel et al.].

2.4.4 Flexibility

The communication interfaces also have to be flexible in terms of scalability, handling of varying load and ability to handle both synchronous asynchronous events [Keskin, 2009]. Time Division Multiple Access networks (TDMA, described in section 2.7.1) provides the least amount of flexibility because of the off line scheduling. Carrier Sense Multiple Access networks (CSMA, described in section 2.7.2) on the other hand, offers a great deal of flexibility. Some protocols even support both time- and event-triggered communication [Nolte, 2006]. Design, integration and configuration flexibility is important in order to make the development easier. Functional flexibility is a must for the vehicles to support a variety of vehicle functions [Pimentel et al.].

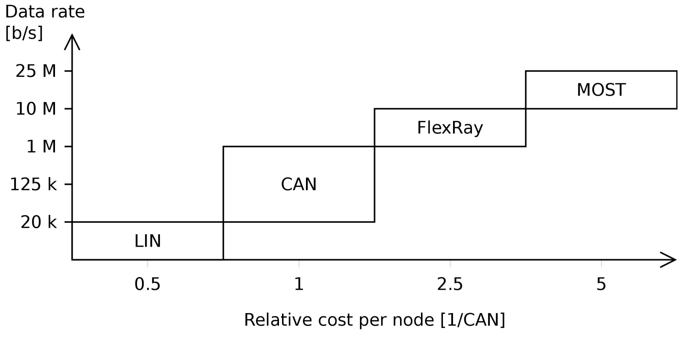


Figure 2.9: *Trade off between cost and bandwidth for LIN, CAN, FlexRay and MOST. Interfaces supporting a higher bandwidth generally cost more. The cost in the figure is set in relation to the cost of CAN for comparative reasons.*

2.4.5 Robustness

The environment in automotive applications can be rough, especially in heavy-duty vehicles. The interfaces have to handle high temperatures, low temperatures, vibrations, electromagnetic interference, long wires and mechanical wear. [Jeffery et al., 2005]

2.4.6 Cost and Market

Designing embedded systems for vehicles is a matter of trade offs between cost, functionality and performance. This is more of an issue for manufacturers of heavy-duty vehicles than car manufacturers because of the comparative low volume of manufactured vehicles [Fröberg et al., 2003]. Another important factor is the availability of the network technology on the market. The systems have to be available for years to come and possible to adapt to the requirements of tomorrow [Keskin, 2009]. How some of the interfaces compare in the trade off between cost and bandwidth can be seen in figure 2.9.

2.5 Functional Domains

The use of embedded systems in vehicles can be divided into different functional domains. A number of them have been defined, among them powertrain, chassis, body, multimedia, telematics, HMI (Human-machine Interface) and X-by-wire [Robert Bosch GmbH, 2011, Nolte, 2006].

2.5.1 Powertrain Domain

The powertrain domain is made up by systems controlling the engine, transmission, drive shaft and differentials, i.e. systems involved in the propulsion of the vehicle. Typically these systems produce and transmit information about vehicle speed, engine rotation speed etc. The ECU in the HMI domain controlling the dashboard then presents this information to the driver. The rough environment and the systems the ECUs control puts a high demand on real time behaviour, bandwidth, reliability and fault-tolerance [Simonot-Lion and Trinquet, 2009, Keskin, 2009]. These requirements makes it possible to use systems with a low degree of flexibility [Keskin, 2009].

2.5.2 Chassis Domain

The chassis domain includes system responsible for the stability, agility and dynamics of the vehicle. This includes systems controlling steering and breaking like ABS and all wheel drive [Robert Bosch GmbH, 2011]. X-by-wire systems can also be placed in this domain because of the similarities in their requirements on the communication interface. The term X-by-wire refers to electronics replacing mechanical or hydraulic systems. An example of this is the steering of the vehicle that today is managed through sensors and actuators. Systems in the chassis domain require flexibility to a greater extent than the ones in the powertrain domain. A high degree of dependability and bandwidth is still needed [Keskin, 2009].

2.5.3 Body Domain

Systems supporting the car's driver like wipers, lighting, airbag, climate control, seats and mirrors are said to belong to the body domain [Simonot-Lion and Trinquet, 2009]. The body domain normally contains a large number of units exchanging small pieces of information. Both higher and lower bandwidth interfaces such as CAN and LIN are used here [Robert Bosch GmbH, 2011]. Reliability is not as important as in the previous mentioned domains because of the absence of safety-critical systems [Keskin, 2009].

2.5.4 Multimedia, telematics, HMI

The multimedia, telematics and HMI domain refers to systems responsible for audio, video, displays, switches, radio and Internet access among others [Simonot-Lion and Trinquet, 2009]. Navigation, driver assistance and fleet management are systems becoming increasingly important in heavy-duty vehicles [Nolte, 2006]. One thing that characterizes many of the systems found in this domain is that they require a high bandwidth [Robert Bosch GmbH, 2011].

2.6 Interface Classes

The communication interfaces can roughly be divided into four categories based on bandwidth and area of use. Only three of them have been formally defined by SAE (Society of Automotive Engineers) but the development of the interfaces has called for the definition of another category. A brief description of the classes can be found below. [Navet and Simonot-Lion, 2009a,b]

Class A: The interfaces found in class A provide a bandwidth of less than 10 kb/s and are used mainly for sensors and actuators. LIN and TTP/A (Time-

Triggered Protocol class A) are examples of class A networks. [Navet and Simonot-Lion, 2009a,b]

Class B: A Class B network has a data rate of 10 - 125 kb/s. It is used mainly for data exchange between ECUs and information sent in the body domain of the vehicle. Both J1850 and low-speed CAN can be found in this category. [Navet and Simonot-Lion, 2009a,b]

Class C: Interfaces in this class operate with a data rate between 125 kb/s and 1 Mb/s. These interfaces are used in the chassis and power train domains and for gateways between subsystems. High-speed CAN is an example of a class C network. [Navet and Simonot-Lion, 2009a,b]

Class D: Although class D hasn't been formally defined yet it's a widely used concept within the automotive industry. The fault-tolerance and high bandwidth provided by modern communication interfaces distinguishes them from older ones. Class D networks have a data rate over 1 Mb/s and are used in the chassis and power train domains but also for multimedia applications. MOST (Media Oriented Systems Transport), FlexRay and TTP/C are examples from this category. [Navet and Simonot-Lion, 2009a,b]

2.7 Media Access Control

Communication interfaces are based on two basic design paradigms, event- and time-triggered control. The main characteristic of an event-triggered system is that it reacts to internal or external events. Messages are passed as soon as possible and priorities are used to avoid collisions. Time-triggered systems rely on static scheduling where every node has its own time slot. A more detailed description of the two designs can be found below. [Keskin, 2009]

2.7.1 Time-Triggered Bus

TDMA

Time-triggered buses are based on static scheduling. Every node has its own timeslot for sending messages based on the Time Division Multiple Access (TDMA) scheme. The predictability of networks using this method makes it easy to discover faulty or missing messages. This deterministic behaviour makes it ideal for safety-critical real time applications because of its dependability. The bandwidth can also be increased since there's no need for arbitration or prioritizing of messages. These systems are however inflexible, the scheduling has to be revised as soon as a node is added or removed or the functionality is changed. Clock synchronization is also important because of the static scheduling. [Keskin, 2009]

TDMA is a network access model based on time slots assigned to the different nodes as seen in figure 2.10. The timeslots doesn't necessarily have to be of the same size but can be adjusted to the amount of data a node has to send. Clock synchronization is important since there's no handshaking prior to sending a mes-

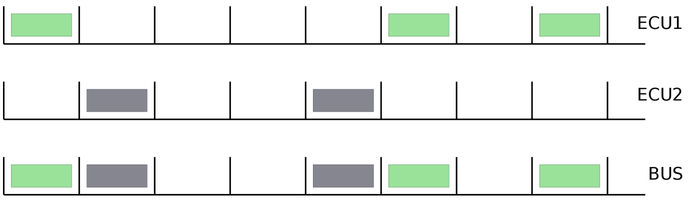


Figure 2.10: *Example of a TDMA network. The nodes are assigned slots in which no other node can transmit to the network. This is illustrated by messages sent by two nodes and the resulting traffic on the network.*

sage. This can be handled by a master node sending a synchronization message prior to every cycle. Another solution is based on decentralized clock synchronization. [Sauter, 2009]

2.7.2 Event-Triggered Bus

Event-triggered messages are sent as a reaction to asynchronous events, internal or external. Arbitration and prioritization are used to determine if a message can be sent or not and to avoid collisions. These systems are highly flexible and easy to extend since no static scheduling is used. [Keskin, 2009]

Different principles of arbitration have been developed to handle the prioritization of messages in event-triggered networks.

CSMA

CSMA is a method for handling arbitration on a shared media network. The basic idea is to detect if the channel is busy before sending a message and the messages are sent as soon as it's possible. Collisions can occur because of the propagation delay on the physical medium. Two nodes can send messages to what appears to be an idle channel before the message from the other node can be detected. A node detecting a collision waits for a random amount of time before sending its message again. Long propagation delays increases the risk for collisions and thereby decreases the performance. This method is called 1-persistent CSMA because it sends the message with a probability of 1 as soon as the channel is idle. [Tanenbaum, 2003]

Zero delay channels can also suffer from collisions. If two nodes are waiting for a third node to finish transmitting they will send their messages simultaneously once it has stopped transmitting resulting in a collision. This can be avoided by not continuously monitoring the channel for a message to end but wait for a random amount of time before checking the channel and then sending. This less greedy approach, non-persistent CSMA, increases the utilization of the network but at the same time suffers from longer delays. [Tanenbaum, 2003]

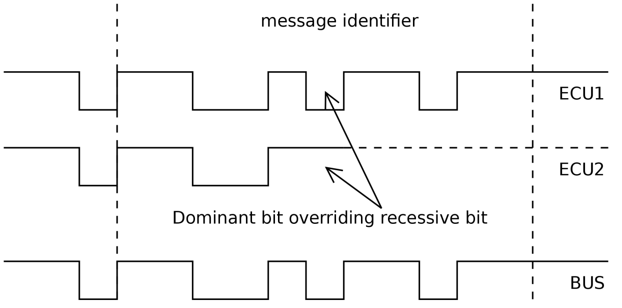


Figure 2.11: *CSMA/CR and CSMA/CA arbitration. ECU 1 has a lower valued message identifier and thereby a higher priority on the network. As soon as one of its dominant bit overrides a bit sent by ECU 2 the latter stops transmitting.*

CSMA/CD

The CSMA/CD (CSMA with Collision Detection) is a method to increase the bandwidth and decrease delays on the channel. If a node detects a collision it immediately stops sending its message. There's a high risk that colliding messages are garbled so there's really no need for sending them once a collision has been detected. The node then waits for a random amount of time before checking the channel for other messages and a possibility to send again. Message collisions are detected by listening to the channel and comparing what it reads back with what it sends. Special encodings might be used to increase the chance of collision detection, two colliding bits represented by 0 V might be hard to detect. [Tanenbaum, 2003, Sauter, 2009]

CSMA/CA

CSMA/CA (CSMA with Collision Avoidance) is another type of media access control method. A node determines if a channel is busy and transmits its message if it's not. If a busy line is detected the message is sent as soon as it's idle again. Collisions are handled by waiting a random amount of time before sending again [Tanenbaum, 2003]. There's also a mechanism determining which message to send if two messages are sent at the same time. The channel is designed to use dominant and recessive bits. If both are sent at the same time the dominant bit overwrites the recessive one on the channel, typically a "1" overwritten by a "0". If the message sent differs from what is read from the network the node stops transmitting. In the end only the node with the lowest identification number (highest priority) sends its message. The physical length of the wiring is because of this limited to guarantee correct prioritization behaviour [Sauter, 2009]. An example of this mechanism can be seen in figure 2.11

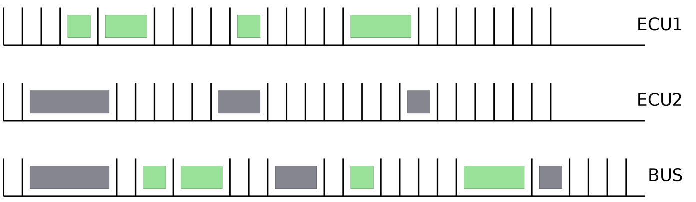


Figure 2.12: *Two nodes sending data to a bus using minislotting. The size of the slots is adjusted to fit the data being sent by one of the nodes. Slots in which no data is sent are not extended.*

CSMA/CR

CSMA/CR (CSMA with Collision Resolution) begins with a arbitration phase in which all nodes with a message to send transmits an identification number. Nodes with a lower priority than other nodes simply keep from transmitting. Collisions are resolved through prioritization [Thiele, 2010]. Similar to CSMA/CA dominant and recessive bits are used to give the node with the lowest identification number (highest priority) access to the channel (see figure 2.11). Messages are never destroyed as with CSMA/CA or CSMA/CD since the prioritization is determined in advance [Sauter, 2009].

2.7.3 Minislotting

Minislotting, also known as Flexible Time Division Multiple Access (FTDMA) divides the communication window into a number of equally sized slots. The size of the slot is expanded to fit any data transmitted, otherwise it keeps its size resulting in a short period of idle time on the network. The minislots are numbered just like the slots in the static segment of FlexRay and the nodes have slots assigned in which they can send data [Paret and Riesco, 2007, Koopman, 2011]. There might not be time for all nodes to send data in the dynamic segment since its size is limited. This gives nodes with the possibility of sending data in a lower numbered frame priority. Any data not being sent has to wait for the dynamic segment in the next FlexRay cycle [Koopman, 2011]. An illustration of how minislotting works can be seen in figure 2.12.

2.7.4 Mixed Time- and Event-triggered Buses

Some interfaces offers both event- and a time-triggered communication. This is usually achieved by dividing the message window into a static and a dynamic part featuring the two paradigms separately. FlexRay is an example of this. [Paret and Riesco, 2007, Rausch, 2007]

2.8 Fault Handling

Errors can cause faults in three different domains, time, space and value. Timing faults are caused by messages being sent, received or computed at the wrong time or not at all. Clock drift is an example of a timing fault. Incorrectly computed, sent or received messages are said to be a part of the value domain. Incorrect physical voltages on a network is an example of this. Faults in the space domain, spatial proximity faults, includes damaged hardware and can affect multiple ECUs. Networks with hardware redundancy are less sensitive to this kind of errors. [Rushby, 2001]

2.8.1 Error Detection

Numerous techniques have been developed to identify erroneous messages on a network.

Bus monitoring: The sending node compares its sent signal with the data on the network. If the two differs an error has occurred and the node act accordingly. [Paret and Riesco, 2007]

Cyclic redundancy check: Some protocols use a cyclic redundancy code to detect errors. The implementation is relatively simple since ordinary shift registers can be used. [Paret and Riesco, 2007]

Parity check: A simple parity check is used by some interfaces. [Motorola Inc., 2001]

Message frame check: Reserved bits in a message frame are set to a specific value as a limiter between different fields. An error has occurred if these bits in a received message doesn't follow the standard. [Paret and Riesco, 2007]

Acknowledgement Acknowledge signals (ack) are sent to confirm that a message has been correctly received. A missing ack is an indication of an erroneous transmission. [Paret and Riesco, 2007]

Error signalling: A node detecting an error can signal other nodes of the faulty transmission. [Paret and Riesco, 2007]

2.8.2 Clock Synchronization

Clock synchronization is crucial in time-triggered networks for all the nodes to keep to their time-slots. This is determined by the quality of the nodes' local oscillators and the synchronization algorithm [Rushby, 2001].

There are two basic synchronization algorithms, averaging and event based. The former measures the skew of a node's clock compared to that of every other. The clock is then set to an average of these values. Fault tolerant averaging algorithms have been developed to deal with faulty clocks. [Rushby, 2001]

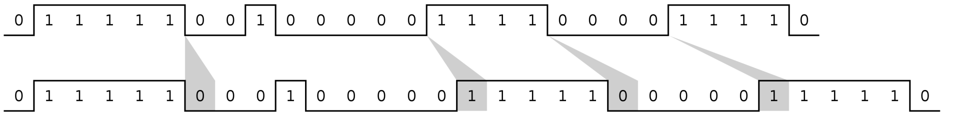


Figure 2.13: *Example of bit stuffing. The lower signal represents the traffic on the bus when sending the sequence on the top line. In this case a complementary bit is inserted after five consecutive bits of the same value.*

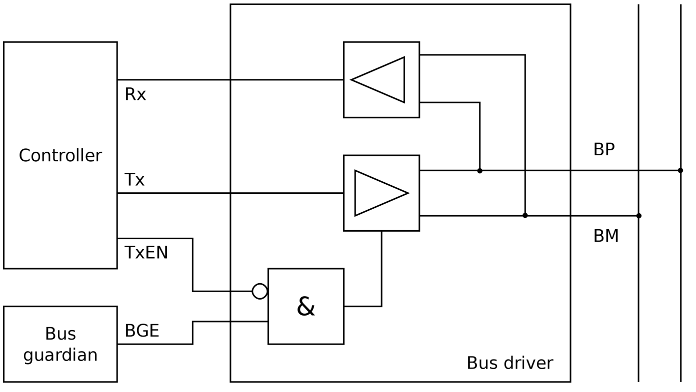


Figure 2.14: *Schematic figure of a local bus guardian. The TX enable signal reaching the transceiver is controlled by the Bus Guardian Enable signal to prevent the node from becoming a "babbling idiot".*

Event based synchronization relies on message passing between nodes. A node sets it clock when it has received a certain number of events from other nodes. [Rushby, 2001]

Some fieldbuses features what is called bit stuffing. After a protocol specific number of consecutive bits with the same value a bit of the complementary is inserted. By doing so synchronization can be kept at the cost of some extra overhead. An example of this can be seen in figure 2.13. [Paret and Riesco, 2007]

2.8.3 Bus Guardian

A bus guardian is a device controlling the access to the network to avoid containment errors. It can prevent the node from sending on the communication channel but can't communicate itself. This is to prevent what is known as "babbling idiots", nodes sending information in an incorrect time slot. The transmission of a message has to be explicitly allowed by the bus guardian (by a Bus Guardian Enable signal, BGE) to guarantee a collision free network. Bus guardians can be implemented locally in every node or centrally guarding an active star. An example of a locally implemented bus guardian can be seen in figure 2.14. [Rausch, 2007]

2.8.4 Never-Give-up Strategy

A never-give-up strategy is a mechanism to bring a faulty system back to a safe state. It must be able to detect software errors during runtime and handle it maintaining the security of the vehicle. [Kopetz et al., 2000]

2.9 Electromagnetic Compatibility, EMC

Electromagnetic compatibility, EMC, is important within the automotive industry for a number of reasons. First of all there are legal requirements on radiation levels that have to be met. Secondly the increased use of electronic systems in vehicles has also increased the problem with systems interfering with each other. Crosstalk due to inductive and capacitive coupling can decrease performance and affect the functionality of the equipment. Customer needs is another important factor to why EMC is important. [Rybak and Steffka, 2004]

Digital signals are one major contributor to the emission from an embedded system. Short rise and fall times can only be achieved using higher frequency components producing higher levels of radiation [Schmitt, 2002].

Communication in an electrical medium between two nodes in a network can be done by single-ended or differential signalling. The former uses a single wire to transfer the signal and a ground wire used as a reference voltage. These systems are more sensitive to external disturbances which can be a problem in automotive applications because of the environment the systems operate in. The ground level can also vary a lot through the network because of resistance and inductance. Differential signalling on the other hand uses two complementary signals and thus requires two wires. It eliminates the requirement for a common ground wire since the signal is represented by a difference in voltage between the wires. Twisting the cables also reduces the emission and risk of crosstalk [Marwedel, 2006]. Common mode chokes can be used to eliminate common mode noise on the differential channel. They consist of a ferrite core with the signal wires wrapped around it. The resulting induced magnetic field from two signal wires with opposite currents is zero because of the cancellation and so differential signals pass. Common mode signals however induces a magnetic field, experiences the high inductance of the coil and are “choked” [Schmitt, 2002].

Signal reflections can arise in cables if lines aren’t properly terminated. The standard line termination for a two wire CAN or FlexRay bus is a resistor at each end of the bus. Split terminations can be used to further increase the EMC. It consists of two resistors in series with a bypass capacitance between them connected to ground. [Rausch, 2007]

Cables can act as both a main source and receiver of electromagnetic interference in an embedded system. Shielding can be used to protect a signal from disturbances and to contain it in a conductor to reduce emission. There are two different mechanisms contributing to this, reflection (most important for low frequencies) and conduction to ground (for higher frequencies) [Schmitt, 2002]. There

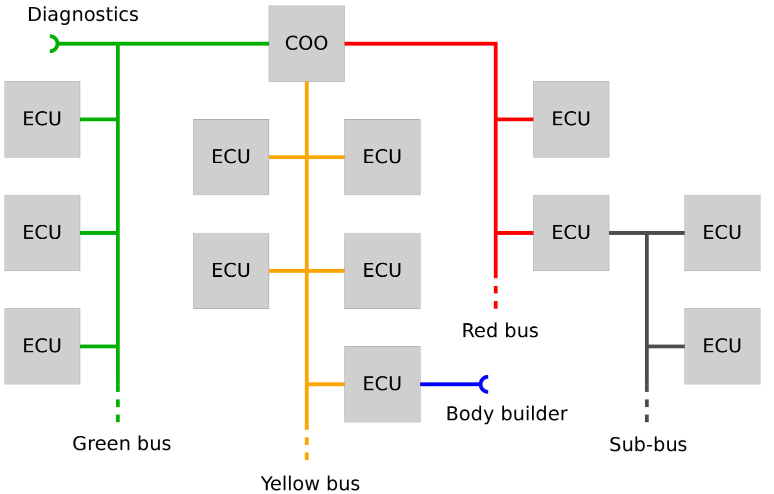


Figure 2.15: *Layout of the J1939 bus network in Scanias vehicles. The buses (red, yellow and green) roughly corresponds to the functional domains defined in automotive applications.*

are two types of shielding, foil and braids. The former covers the conductor with a thin layer of metal foil connected to ground, either directly or through a ground wire. Braids consists of a mesh of copper wires surrounding the conductor without covering it fully because of the gaps in the mesh. Cables are advantageously shielded in one end to avoid ground loops [AlphaWire, 2009].

2.10 Scania CAN Network

The CAN network in Scanias vehicles consists of three buses (green, yellow and red) connected by a gateway called the coordinator (COO). It utilizes SAE J1939 (described in detail in 3.2) at 250 or 500 kb/s. The placement of the ECUs on the different buses roughly follows the domains described above (chapter 2.5). The time critical powertrain ECUs are placed on the red bus and the least time critical ECUs in the body domain are places on the green. The network load is reaching the CAN bus limit on some of the buses which has lead to the creation of sub buses. There are also external body builder and diagnostic interfaces. Figure 2.15 shows the basic network layout. [Gustafsson, 2010]

Scania uses a modular system when constructing their trucks and buses. Some of the ECUs are mandatory since they control essential parts of the vehicle like engine, dashboard and lighting. The modular system allows the vehicles to be customized to meed the customer's need but it also affects the communication interfaces being used. [Gustafsson, 2010]

3
Available Communication Interfaces

This section presents different communication interfaces of all classes described in section 2.6. Apart from hardware and protocol aspects consideration will also be taken to market adaption, cost and area of use in the evaluation of the interfaces.

3.1 Controller Area Network, CAN

Bosch started developing CAN, the Controller Area Network, in the mid 80s to meet the growing need for inter-ECU communication. It has become the de facto standard in automotive embedded systems and it is also used in other areas such as automation [Paret and Riesco, 2007, Navet and Simonot-Lion, 2009a]. It can be used as a class C interface in the powertrain and chassis domains (running at 250 or 500 kb/s) or as a class B network in the body domain (at 125 kb/s). CAN is completely event-triggered and does not offer any static scheduling [Navet and Simonot-Lion, 2009a]. CAN has only been specified in the first two layers of the OSI model, the hardware and data link layers. A number of CAN based application layers have been developed to simplify the design and implementation [Cena and Valenzano, 2009]. One of them, J1939, is currently used in Scanias vehicles. More information on J1939 can be found in 3.2. Other application layers such as openCAN (used in automation) are not covered by this study.

3.1.1 Wiring

Generally there are two types of wiring used in CAN networks, single parallel differential pairs and twisted differential pairs, the latter both with and without screening. The parallel lines can because of the differential signaling on the CAN bus cause electromagnetic interference why twisted wires are to prefer. Shielding

Bit rate	Length
1 Mb/s	40 m
500 kb/s	130 m
250 kb/s	270 m

Table 3.1: *Maximum bus length depending on the bit rate used. The limitation is due to the arbitration process. A signal has to be able to propagate from one end of the network to the other and back in order to guarantee a correct behaviour, otherwise different nodes can have different views of the value on the bus.*

is in theory another good option for eliminating interference but long distances and poor ground return can cause problems. Shielding is also expensive why unscreened twisted pairs often turn out to be the best option when it comes to wiring. [Paret and Riesco, 2007]

The maximum distance between two nodes in a CAN network depends on the bit rate used. This can be seen in table 3.1 [Paret and Riesco, 2007]. Also the stub length is affected by the bit rate. At 1 Mb/s it cannot exceed 30 cm. Connectors are not standardized by the CAN specification but a standard 9 pin DSUB connector is widely used [Cena and Valenzano, 2009].

3.1.2 Network Topology

The by far most common CAN topology is the bus. It typically has a 120 Ω characteristic impedance with a matching line termination in each end of the bus [Paret and Riesco, 2007]. Both single-wire and two-wire buses are supported as well as optical transmission mediums [Cena and Valenzano, 2009].

3.1.3 Signal Representation

The bits on a CAN network are coded using non-return to zero (NRZ). One major drawback with this coding is the risk of losing synchronization after long periods of bits with the same value. CAN uses bit stuffing (described in section 2.8.2) of 5 bits to overcome this problem [Paret and Riesco, 2007]. Theoretically a bit stuffing of 5 bits could cause an encoding efficiency of only 80%. I can be seen however that only 2 to 4 bits are added to a frame normally. Not all of the CAN frame is encoded using bit stuffing. It only applies to the part from the start of frame (SOF) to the cyclic redundancy check (CRC) [Cena and Valenzano, 2009]. Figure 3.1 shows the representation of the signal on the two wires, CAN_L and CAN_H.

3.1.4 Media Access Control

CAN uses a CSMA/CR protocol for accessing the network using dominant and recessive bit representations [Thiele, 2010]. This method is described in section 2.7.2

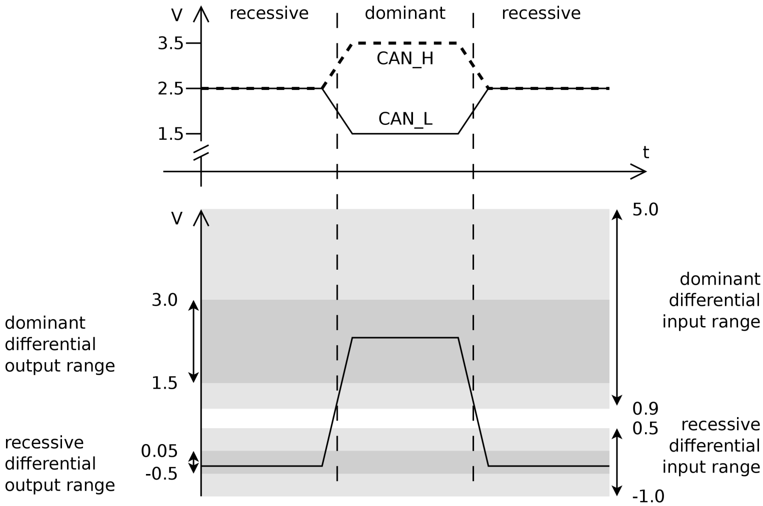


Figure 3.1: *CAN signal showing two recessive and one dominant bit on its two wires, CAN_L and CAN_H. The ranges of the voltage on the output and input of each node can be seen in the lower section.*

3.1.5 Fault Handling

CAN uses many of the error detection methods described in section 2.8.1. [Cena and Valenzano, 2009]

Cyclic redundancy code: CAN uses a 15 bit CRC appended to the message frame to discover errors. It is able to detect 5 erroneous bits or and error bursts up to 15 bits.

Frame check: The CRC, ACK delimiters and EOF fields all have to be recessive on a CAN network, otherwise an error has occurred.

Acknowledge check: The transmitting node verifies whether a dominant ACK bit has been sent. An error in the transmission has occurred if not.

Bit monitoring: The sent bit is compared to the value on the bus and an error is assumed to have occurred if they don't match. No such errors are generated during the arbitration phase or the acknowledgement slot.

Bit stuffing: An error is generated if 6 consecutive bits of the same value are detected between and including the SOF and the CRC.

3.2 J1939

J1939 was developed by SAE for use in heavy-duty vehicles and it quickly became the accepted industry standard within this area. It is a higher-layer protocol based on CAN and shares many of its advantages; it's easy to implement, reliable and it only requires two wires. The J1939 protocol is based on two older SAE standards, J1708 and J1587, both described in section 3.18. It replicates their behaviour and is backward compatible to some extent. A number of protocols based on J1939 have been developed, MilCAN (military applications), ISOBUS (agricultural industry) and NMEA 2000 (marine applications) [Voss, 2008]. J1939 is the main inter-ECU communication standard used in Scanias vehicles today. Section 2.10 describes this in more detail.

3.2.1 Wiring

The physical layer of J1939 specifies both a shielded and an unshielded twisted pair wire as its medium. The cables have, according to the specification, a 40 meter length limit at 250 kb/s (a 500 kb/s version is also defined). A maximum of 30 nodes can be connected to the same bus. These requirements are relatively strict compared to the standard CAN specification allowing longer wires at higher bandwidths (270 meters at 250 kb/s). [Voss, 2008]

3.2.2 Media Access Control

J1939 uses CSMA/CR, the same media access method as CAN uses [Voss, 2008] This method is described in section 2.7.2

3.2.3 Fault Handling

J1939 shares the error handling methods used by CAN. See section 3.1.5 for more information. [Voss, 2008]

3.3 FlexRay

FlexRay was developed by a consortium mainly consisting of car manufacturers BMW Group, Daimler AG, General Motors and Volkswagen AG. Bosch (the company behind CAN), Motorola and Philips can also be found among its core members. FlexRay has been designed to meet the limitations of the current de facto standard within automotive embedded networks, CAN. First of all the bandwidth has been increased from a maximum 1 Mb/s for high speed CAN to 10 Mb/s. It's also based on a time-triggered design philosophy to meet the real-time requirements in many of today's vehicles. This differs a lot from CAN with its probabilistic event-based behaviour. Last but not least it has also been designed to support a number of different network topologies with redundancy on the physical level. This fact makes it usable in X-by-wire applications requiring a high degree of reliability [Paret and Riesco, 2007].

3.3.1 Wiring

An electronic, two wire, differential transmission line is the only transmission medium defined in the FlexRay specifications but optical interconnections can also be used. FlexRay nodes must however be capable of supporting a two channel layout. This increases the redundancy and fault tolerance of the network but it also enables higher data rates. The propagation delay must not exceed 2500 ns according to the specification. It also defines a propagation delay of maximum $10ns/m$ which implies a theoretical maximum wire length of $2500/10 = 250m$. As will be seen later the actual limitations in the different topologies are much lower. Also the difference in delay between the two channels, A and B, should be kept to a minimum [Paret and Riesco, 2007, Rausch, 2007].

The cables in a FlexRay network are not standardized but usually a shielded or unshielded twisted pair cable is used. The characteristic impedance of the wiring should be between 80 and 110 Ω and the attenuation less than $82dB/km$. The lines can be terminated with a single resistor between the wires. A split termination can be used to gain better EMC characteristics. [Rausch, 2007]

3.3.2 Network Topology

There are two supported FlexRay network topologies apart from point-to-point interconnections, the bus and star layouts (and combinations of the two). FlexRay was originally designed as a two channel system but it also supports single channel layouts [Rausch, 2007, Paret and Riesco, 2007]. The cable length between two active nodes must not be longer than 24 meters. This constraint applies to the following distances:

- between an ECU and an active star
- between two arbitrary ECUs on a bus or a passive star
- between two active star nodes

Another constraint is the number of active components between two arbitrary selected ECUs. A maximum of two stars can be cascaded which gives a maximum distance between nodes of 72 meters [Rausch, 2007, Paret and Riesco, 2007]. A more general description of the network topologies can be found in 2.3, this section discusses only the FlexRay specific properties.

Point-to-point link: The point-to-point interconnection consists of a bidirectional, differential channel with a termination in each line end to avoid reflections. The termination resistance has to be between 80 and 110 Ω to match the characteristic impedance of the wires. The maximum wire length is 24 meters according to the FlexRay specification. [Paret and Riesco, 2007]

Passive linear bus: The stubs on a FlexRay bus can give rise to reflections, antinodes and wave clusters on the bus because of the high bandwidth. This can cancel out or amplify the voltage and alter the bits being sent. To avoid this a number of constraints have been on the network. Apart from the maximum wire length of 24 meters between two nodes no more than 22 ECUs can be connected together using this method. The distance between two splices on the bus also has to be greater than 150 mm. [Rausch, 2007, Paret and Riesco, 2007]

Passive star: Just as for a passive linear bus the maximum length between two arbitrary selected nodes is 24 meters. Also no more than 22 ECUs can be

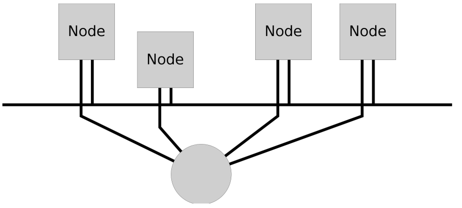


Figure 3.2: *Example of a FlexRay two-channel layout. The channels are connected using a linear bus and an active star network respectively.*

connected [Paret and Riesco, 2007, Rausch, 2007]. The two farthest situated nodes have line terminations in order to avoid reflections [Paret and Riesco, 2007].

Active star and cascaded active stars: The maximum length between two ECUs in an active star network is just as for the other layouts 24 meter. No more than two active stars may be present in a network which gives a maximum distance of $3*24 = 72m$ between two nodes [Paret and Riesco, 2007, Rausch, 2007]. An active star network is electronically active and so all lines are properly terminated at each node per definition [Paret and Riesco, 2007].

Hybrid: All the networks described above can be connected into a hybrid network through active stars as long as the previous mentioned constraints are met. [Paret and Riesco, 2007]

The two channel nature of FlexRay nodes makes it possible to connect a node to two different networks. These doesn't necessarily have to be of the same type (see figure 3.2). Connecting an ECU to more than one network can either be used to increase the bandwidth or to add redundancy if one channel should fail. [Paret and Riesco, 2007]

3.3.3 Signal Representation

The bits in a FlexRay system are encoded using non-return to zero. The nominal voltage is defined according to the equation below. BP and BM are the names of the differential signals used in accordance with the FlexRay specification.

$$\frac{U_{BP} + U_{BM}}{2} = 2500mV \tag{3.1}$$

A positive voltage differential between the two cables represents 1 and a negative differential 0. A channel in which the voltages are equal but non-zero in the two wires is said to be idle. In other words tri-state is supported but actually a fourth state can be defined by the two wires having a 0 V potential. This is called a low powered down mode [Rausch, 2007]. These states can be seen in figure 3.3.

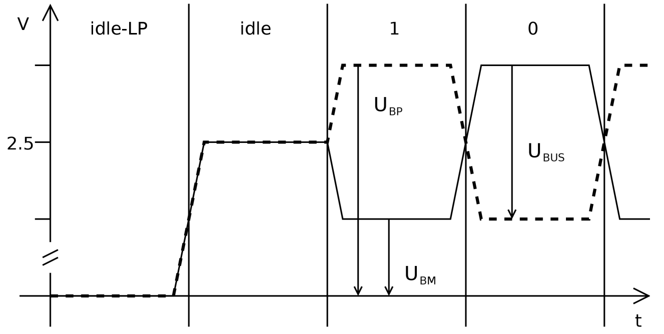


Figure 3.3: *The four different FlexRay states. The logical zero and logical one are represented by a difference in voltage between the two wires. BP and BM are the names of the differential signals used.*

Figure 3.4: *A FlexRay communication cycle consisting of a static part, a dynamic part, a symbol window and network idle time. The first and the last of the four are always present in a FlexRay cycle.*

3.3.4 Media Access Control

The FlexRay protocol is divided into cycles built up by Network Idle Time (NIT) and a static part (both mandatory) as well as a dynamic part and a Symbol Window [Rausch, 2007]. This can be seen in figure 3.4.

The static segment (illustrated in figure 3.5) uses a TDMA design to provide collision free communication and a deterministic behaviour. It's divided into equally sized slots, each of them reserved for a unique node. The bandwidth is known beforehand since only one node at the time can send information. The TDMA principle also guarantees real-time characteristics within this segment because of the static scheduling. [Paret and Riesco, 2007, Rausch, 2007]

The dynamic part of the FlexRay protocol (illustrated in figure 3.6) is based on minislotting (seen in figure 2.12). More information about this can be found in 2.7.3

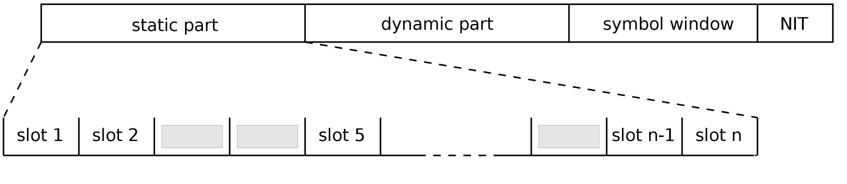


Figure 3.5: *The static part of a FlexRay consists of equally sized slots and uses the TDMA network access model.*

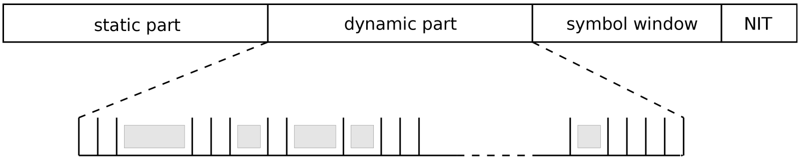


Figure 3.6: The dynamic part of a FlexRay cycle utilizes minislotting a its network access model. Slots in which data is sent are extended to fit the data, otherwise their size are kept to a minimum.

3.3.5 Fault Handling

Cyclic redundancy code: FlexRay includes an 11 bit CRC in its header. It is able to detect 5 erroneous bits or and error bursts. [Rausch, 2007, Paret and Riesco, 2007]

Bus guardian: FlexRay nodes can include a bus guardian to avoid containment errors. [Rausch, 2007, Paret and Riesco, 2007]

3.3.6 Electromagnetic Compatibility

FlexRay only has two dominant states compared with the dominant and recessive states of CAN. The difference in voltage between the states is about 700 mV. This comparative small value only makes a small contribution to the total electromagnetic radiation from a FlexRay network [Paret and Riesco, 2007]. The relatively high data rate can however give rise to a higher degree of emission due to the faster switching [Rausch, 2007].

3.4 Time-Triggered Controller Area Network, TTCAN

TTCAN is a higher-level protocol aimed at bringing time-triggered communication to CAN. It's basically a hybrid TDMA layer on top of the default CAN CSMA/CR allowing both time- and event-triggered traffic on the network [Nolte, 2006, Pimentel et al.]. TTCAN is mainly defined in the session layer of the OSI model. The physical layer and data link layer is identical to that of CAN Paret and Riesco [2007]. It is intended for use in X-by-wire applications but it doesn't offer the same degree of reliability as FlexRay or TTP/C. The bandwidth is also limited to 1 Mb/s just as for CAN which makes it less usable in some applications [Nolte, 2006].

The TTCAN communication is divided into what is called basic cycles consisting of a series of fixed length time windows. The first window in every cycle is a reference message sent by a node called the time master on the network. This message contains timing information and the cycle count. The counter is needed for the frames to know the current cycle to be able to follow the correct schedule. The slave nodes compares the measured duration of a cycle with the time in the reference message to correct for clock drift. A maximum of 64 basic cycles can be

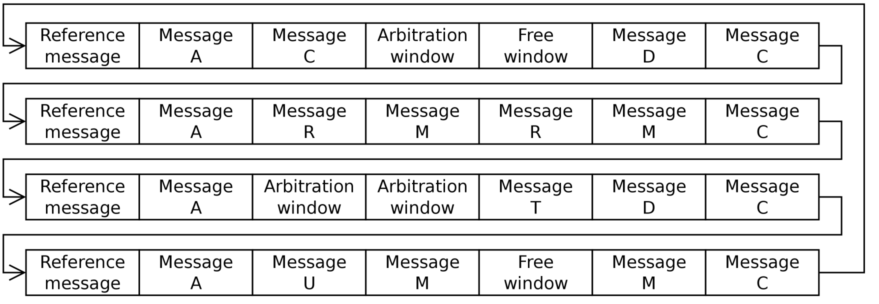


Figure 3.7: *Four TTCAN basic cycles, each one of them starting with a reference message containing timing information and cycle count. The first cycle starts again as soon as the last one finishes.*

combined into what is called a system matrix that is sent repeatedly. The content of a window can vary from cycle to cycle. [Pimentel et al.]. An example system matrix can be seen in figure 3.7

Three different types of messages can be sent in every window apart from the reference message. Resending of erroneous messages are disabled in TTCAN. Bus guardians are not a part of the specification just as for CAN. [Pimentel et al., Nolte, 2006, Obermaisser]

Exclusive time window: A message statically scheduled to a given window in a given basic cycle. This is set off-line during system design.

Arbitrating time window: Asynchronous messages can be sent using the arbitrating mechanism found in CAN (described in section 2.7.2). A message can be sent if it was ready at the beginning of the window.

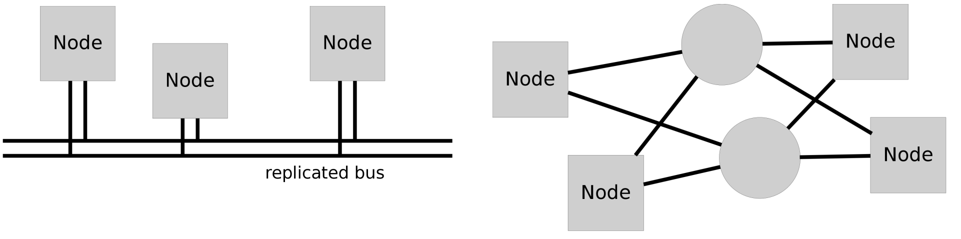
Free time window: Reserved for future extension of the TTCAN protocol.

TTCAN can be run using two different levels. Level 1 using a local cycle time and level 2 offering an external hardware supported clock continuously correcting for clock drift. [Nolte, 2006, Pimentel et al.]

CAN and TTCAN shares many characteristics because of their similarities. This fact makes the transition from CAN easy which is a major advantage to the protocol. [Nolte, 2006]

3.5 TTP/C

TTP/C is a SAE class C network supporting a transfer speed of up to 25 Mb/s [Keskin, 2009]. It's based on the time-triggered architecture, TTA just like TTP/A (described in section 3.8) [Elmenreich and Ipp, 2003].



(a) *TTA bus* *(b)* *TTA active star network*

Figure 3.8: *Example of TTA networks showing the two supported network topologies, the linear bus and the active star network.*

3.5.1 Network Topology

TTA networks supports two network topologies, the active star and the bus (illustrated in figure 3.8a and 3.8b). Every node is connected to replicated channels to provide redundancy if one of the channels should fail. [Elmenreich and Ipp, 2003]

3.5.2 Media Access Control

TTA networks nodes use a TDMA scheme to access the bus offering highly dependable real-time communication with a high level of fault tolerance. Each node has its own sending slot each cycle (round) and has to send data within this slot. Event-triggered data can only be sent within the beforehand scheduled time slots. [Elmenreich and Ipp, 2003]

3.5.3 Fault Handling

TTP/C uses a bus guardian to prevent nodes from sending data in the wrong time slot and thereby eliminating the problem with babbling idiots. [Elmenreich and Ipp, 2003]. Every node on a TTP/C bus network is equipped with its own bus guardian. It's implemented as a separate component, placed a distance from the node with its own power supply and clock to prevent faulty units from affecting the functionality of the bus guardian as well [Elmenreich and Ipp, 2003]. A completely different approach is used in a active star TTP/C network. The bus guardians are in this case implemented centrally in the active star nodes. Because of this only one bus guardian per interconnection has to be implemented instead of one per node. This reduction in the number of units makes it an economically attractive alternative [Elmenreich and Ipp, 2003, Rushby, 2001]. The bus guardians are not as sensitive to spatial faults in each node [Elmenreich and Ipp, 2003] but is a single point of failure if the active star node is faulty [Rushby, 2001].

Each node in a TTP/C network has its own membership list containing all other nodes that it considers correct. A failure leads to the removal of the receiving node from the sending node's list. This implies that the membership lists can vary from node to node and only reflect the local view of the network. The lists are compared to keep a consistency within the network. A membership list that differs from the others is assumed to be incorrect. Acknowledgement replies will only be sent to nodes on the membership list. [Elmenreich and Ipp, 2003]

The TTP/C protocol also features a 24 bit CRC to detect errors. [Elmenreich and Ipp, 2003, Rushby, 2001]

3.6 Ethernet

3.6.1 Wiring

Twisted pair cabling is the most common common medium used for Ethernet applications even though fiber optics also can be used. A number of twisted pair Ethernet cable categories have been developed: [Iniewski, 2009]

CAT3: The category 3 cable is an older Ethernet standard (10BASE-T) designed for data rates of up to 10 Mb/s. This cable type has been replaced today by other Ethernet cables because of its limitations.

CAT5: Category 5 cables are used in Fast Ethernet (100BASE-TX) applications (100 Mb/s).

CAT5e: CAT5e is basically a standard Category 5 cable compliant with Gigabit Ethernet transfers.

CAT6: Category 6 cables are designed for Gigabit Ethernet and can be divided into two subcategories, CAT6a and CAT6b. The former supports transfer frequencies of up to 625 MHz at a 100 meter distance and the latter 500 MHz up to 55 meters.

CAT7: CAT7 cables are designed for data transfer frequencies of up to 600 MHz to meet Gigabit Ethernet requirements.

The category 5 cable is the most widely used of the cables mentioned above. It consists of four pairs of twisted wires and RJ-45 connectors. 100BASE-TX uses two of the pairs, one for transmitting data and one for receiving data. It supports data transfers up to 100 meters and has a characteristic impedance of 100Ω. There's also a number of cables using cable screening and/or individual pair shielding. [Spurgeon, 2000]

Ethernet cables can also be used for providing power, either on unused cable pairs or the ones used for transferring data [Unterdorfer, 2008, Hajjar, 2009]. This is described in detail in section 3.6.6.

3.6.2 Network Topology

There are two network topologies supported by Ethernet, the bus and the star networks. [Iniewski, 2009]

4-b block	5-bit code	description
0000	11110	data 0
0001	01001	data 1
0010	10100	data 2
0011	10101	data 3
0100	01010	data 4
0101	01011	data 5
0110	01110	data 6
0111	01111	data 7
1000	10010	data 8
1001	10011	data 9
1010	10110	data A
1011	10111	data B
1100	11010	data C
1101	11011	data D
1110	11100	data E
1111	11101	data F
...	...	...

Table 3.2: *4B/5B code. Every block of four bits is represented by five bits on the network. The number of consecutive ones or zeroes are kept to a minimum educing the risk of losing clock synchronization.*

Bus: This type of network is based on coaxial cabling with T-connectors used to connect each node. A 50Ω line termination is used to avoid signal reflections. The use of Ethernet bus networks have decreased over the years but can still be found in older networks.

Star: The star network has become the by far most common Ethernet topology. A passive star network is implemented by using a hub as a star node. It only works in the physical level of the OSI model and repeats a received signal to all of its outputs. Active star networks are implemented using a switch operating up to the second or third layer of the OSI model. The network efficiency increases since it only forwards the received data to the targeted node. [Iniewski, 2009]

3.6.3 Signal Representation

100 Mb/s Ethernet (100BASE-TX) signals are encoded using a method called 4B/5B. The data is divided into 4-bit blocks translated into a 5-bit code for transmission over the network according to table 3.2. The codes are designed to spread out the electromagnetic patterns being transmitted and to keep clock synchronization. [Spurgeon, 2000]

The coded 4-bit blocks are sent over the channel using MLT-3 encoding (Multi-Level Threshold 3). It's based on three voltage levels where a change in voltage represents a logical 1 and a logical 0 is represented by a constant signal level

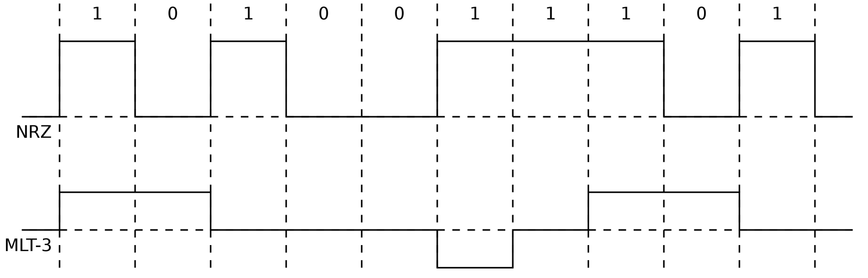


Figure 3.9: *MLT-3 encoding based on three voltage levels. A logical one is represented by a change in voltage and a logical zero by a constant signal level.*

(see figure 3.9. The voltage swing on each wire ranges from approximately 0 to +1 V on the positive wire and 0 to -1 V on the negative wire. The 100BASE-TX data rate implies high frequencies being used even though the MLT-3 encoding reduces the signaling rate. Cables meeting the standard is a must to avoid frame errors and decreased performance. [Spurgeon, 2000]

3.6.4 Media Access Control

Ethernet uses the CSMA/CD method to access the network. [Iniewski, 2009, Spurgeon, 2000]

3.6.5 Electromagnetic Compatibility

Differential signaling over twisted pair cables is a good way to reduce the electromagnetic emission from a cable. The interference can also be kept to a minimum because of the complementary nature of the signals. Adding shielding and screening further improves the EMC characteristics resulting in more expensive wiring [Iniewski, 2009]. The 4B/5B and MLT-3 coding used in 100BASE-TX Ethernet further reduces the signalling rate [Spurgeon, 2000].

3.6.6 Power over Ethernet

Power over Ethernet is a technology used to deliver supply voltage in a standard category 5 or category 6 cable (CAT5/CAT6) along with the data. These cables contain four twisted pair conductors but only two of them are used for data transfer for 10BASE-T and 100BASE-T. [Unterdorfer, 2008, Hajjar, 2009]

Power over Ethernet has been standardized in IEEE802.3af. It identifies two types of devices, Powering Sourcing Equipment (PSE) delivering the power and Powered Devices (PD) using it. Two methods have been developed for feeding power to the destination node through these cables. [Unterdorfer, 2008, Hajjar, 2009]

Mode A combines the voltage feed and the data signal on the same wires. This is called phantom powering and is used for telephones and microphones. This means that only two twisted pairs are used for data and power for 10 and 100

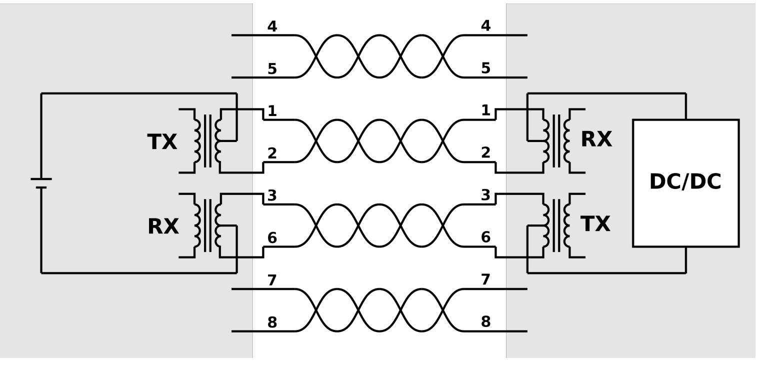


Figure 3.10: *Phantom powering, also known as mode A, in which the voltage is fed over the same wires as the signals. Only two of the four cable pairs in a standard CAT5 or CAT6 cable are used. The PSE can be seen to the left and the PD to the right.*

Class	Power Range [W]
Class 0	0.44-12.95
Class 1	0.44-3.84
Class 2	3.84-6.49
Class 3	6.49-12.95
Class 4	For future use

Table 3.3: *A list of the standardized Power Over Ethernet classes.*

Mb/s transfer rates using this method. All four wire pairs are used for 1000 Mb/s transfers, two for both data and power and two for data exclusively [Unterdorfer, 2008, Hajjar, 2009]. Figure 3.10 shows the wiring of a phantom powering layout.

Mode B (spare-pair power) uses the spare wires in a standard CAT5 or CAT6 cable at 10 or 100 Mb/s data rates. Data and power are transmitted separately and uses all four twisted pairs in the cable. This is unusable for Gigabit Ethernet where all wires are used for data transfer [Unterdorfer, 2008, Hajjar, 2009]. The wiring for spare-pair power transfer can be seen in figure 3.11.

According to the standard (802.3af) a PSE can provide up to 48 volts DC and 15.4 W per port. Because of power loss in the cabling only 12.95 W can be guaranteed to reach the PD. Table 3.3 shows the standardized classes. [Iniewski, 2009]

3.7 Local Interconnect Network, LIN

The LIN consortium was formed in the late 90s to develop a new communication interface for automotive applications. Its members consisted of car manufacturers Daimler AG, Volkswagen, Volvo, Audi and BMW as well as the semiconductor supplier Motorola. The goal was to develop a interface for use in subnets to re-

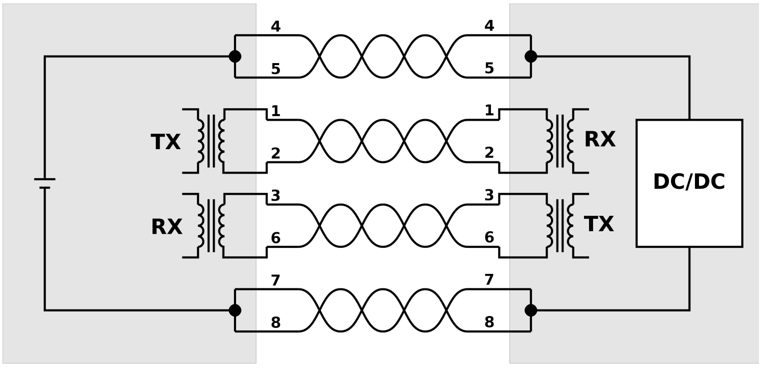


Figure 3.11: *Spare-pair powering using the spare wires in a standard CAT5 or CAT6 cable. This is also known as power over Ethernet mode B. The PSE can be seen to the left and the PD to the right.*

duce the bus load on main buses. It was aimed to be a cheap, logic extension to CAN, scalable and a long term solution. Typical applications are windscreen wipers, light control, rain sensors, mirror adjustment, seat adjustment and door locks. [Motorola Inc., 2001]

3.7.1 Wiring and Network Topology

LIN is based on a single wire bus topology. A LIN bus can theoretically have 63 nodes connected but according to the LIN 2.2A specification it "should not exceed" 16. The line termination should be 1 kΩ for the master node and 20-47 kΩ for the slave nodes. The maximum wire length is 40 meters. [Paret and Riesco, 2007, LIN Consortium, 2010]

3.7.2 Signal Representation

The physical layer of LIN unlike CAN clearly defines the dominant "0" as ground potential and the recessive "1" as the supply voltage used. The bits are encodes using non-return to zero and the bit rate is 20 kb/s. [Paret and Riesco, 2007]

3.7.3 Media Access Control

Because of its single master multiple slave architecture LIN offers a deterministic behaviour without any conflicts on the bus and without any need for arbitration. The communication is asynchronous without any need for clock data distribution but rather synchronization in the beginning of a sent frame. [Paret and Riesco, 2007]

3.7.4 Fault Handling

A LIN frame contains 2 parity bits at the end of every identifier field and a 1 bit CRC at the end of the frame. All transmitter errors are possible to detect but there is no way for a slave to signal an error on the bus. The master can however request diagnostic messages from the other nodes containing this information. The sending node also have a mechanism to compare the value of the bus with the data it sends to detect errors. If errors are temporary or persistent can also be identified and react to the errors accordingly. LIN does not use acknowledgements as other communication interfaces such as CAN. The master node can however retransmit a message if an expected reply from a slave node doesn't appear on the bus. [Paret and Riesco, 2007]

3.7.5 Electromagnetic Compatibility

A LIN circuit for automotive applications should be able to handle an ESD discharge of $\pm 8kV$ at the connectors to the ECU. LIN suffers from emission problems like all other single bus systems (I^2C, one wire CAN etc.). The radiation from a LIN network operating at 20 kb/s is comparable to the radiation caused by a two wire CAN at 500 kb/s in a worst case scenario. [LIN Consortium, 2010]

3.8 TTP/A

TTP/A is a SAE class A network based on the time-triggered architecture, TTA, just like TTP/C (described in section 3.5). It's designed to be a low cost interface used to integrate actuators and sensors into a network. One thing TTP/C and TTP/A has in common is their real-time characteristics and small latency. Every TTP/A network has a master node responsible for the synchronization of the nodes. A shadow master can take over this task in case of a failure. Every round starts with the master broadcasting what is known as a firework byte. This byte defines the type of data sent in the round and initializes synchronization. [Elmenreich and Ipp, 2003]

3.8.1 Network Topology

The network topology is a part of TTA and is described in section 3.5.1

3.8.2 Media Access Control

TTP/A is based on the TTP architecture and uses the same media access method as TTP/C. This is described in section 3.5.2.

3.8.3 Fault Handling

TTP/A uses a classic 8 bit checksum in which the sent bytes are added together to detect transfer errors. The same calculation is performed in the receiving node and the inverse result is added to the received checksum. The result should be 0xFF, otherwise an error has occurred. [LIN Consortium, 2010]

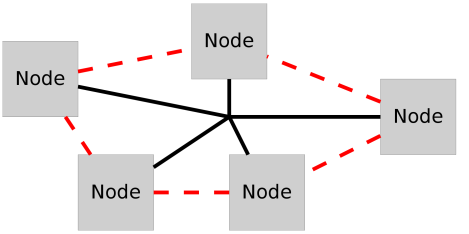


Figure 3.12: *A logical ring network implemented on a physical star network.*

3.9 Media Oriented Systems Transport, MOST

The Media Oriented Systems Transport protocol, MOST, has been developed to meet the requirements of the increased use of equipment in the multimedia, telematics and HMI domains. Radio, CD players, phones, navigational systems and video players are examples of this. Bandwidth is one of the most critical aspects, an audio signal alone can require more bandwidth than a CAN bus can provide. The MOST specification covers all seven layers of the OSI model. There are three MOST generations available today, MOST25 (25 Mb/s), MOST50 (50 Mb/s) and MOST150 (150 Mb/s). [Zimmermann and Schmidgall, 2006]

MOST is based on D2B, a communication interface developed by Daimler and shares some of its characteristics. Like many other interfaces for automotive applications MOST was developed in cooperation between a number of companies, in this case BMW, Daimler and Audi [Strang and Röckl, 2008]. Today it's widely spread within the automotive industry and used by car manufacturers such as Audi, BMW, Volvo, Porsche and Mercedes Benz [Kohler, 2008].

3.9.1 Wiring

The MOST protocol itself is indipendent of the transmission medium but MOST25 defines a plastic optical fiber in its physical layer. MOST50 and MOST150 defines an optical layer [Zimmermann and Schmidgall, 2006] as well as an electrical medium [Kohler, 2008], the former an Unshielded Twisted Pair cable (UTP) and the latter as an extension of 100BASE-T Ethernet [Strang and Röckl, 2008].

3.9.2 Network Topology

MOST uses a ring topology to connect nodes in a network (see section 2.3.5). A logical ring can also be implemented on a physical star network according to figure 3.12. [Strang and Röckl, 2008]

3.9.3 Signal Representation

MOST signals are coded using a differential Manchester encoding seen in figure 3.13. Each bit period is divided into two parts, a clock part and a data part. The clock part always contains a transition from a low to high or high to low. A logical 1 is represented by making the first half of the bit period equal to the

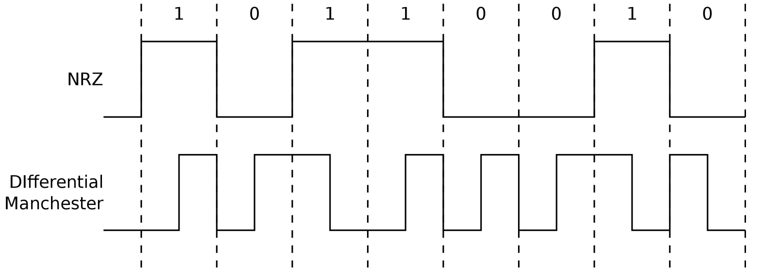


Figure 3.13: Differential Manchester encoding compared to non-return-to-zero encoding. Every bit is represented by a signal level change.

last half of the previous bit period. A logical 0 is represented by setting the first half to the complementary value of the previous value. In other words a logical 1 bit period contains one transaction and a logical 0 period two [Strang and Röckl, 2008]. An advantage of the bits being represented as transactions between levels is that the synchronization is easy to keep since there are no long periods of consecutive ones or zeroes. [Willig and Woesner, 2005, Spurgeon, 2000]

3.10 Safe-by-Wire

Safe-by-wire is a communication bus mainly used for controlling airbags. It was developed because CAN and LIN weren't considered safe enough. Many of its features are taken from CAN but it only supports data rates of up to 150 kb/s [Nolte et al., 2005]. Safe-by-wire was developed to solve the problem of point-to-point connections between airbags. As the number of airbags grew the number of interconnections increased rapidly [Paret and Riesco, 2007]. Due to its specialised area of use and limited bandwidth it's not of any interest to this study.

3.11 Domestic Digital Bus, D2B

The Domestic Digital Bus, D2B can be seen as a predecessor to MOST because of its characteristics [Mercedes Benz, 2004]. It's based on an optical transmission medium (allowing communication at 20 Mb/s) between nodes connected in a ring [Nolte et al., 2005]. The nodes are also connected with electrical wake-up lines in a star network according to figure 3.14. The network has a D2B master controlling the wake-up and sleep of the other nodes. The sleep mode is used because of the high current usage of the fiber optic transceivers. [Mercedes Benz, 2004]

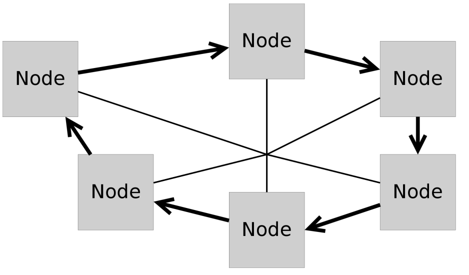


Figure 3.14: D2B ring network with wake-up lines controlling the wake-up and sleep state of the nodes. This is used because of the high current usage of the optic transceivers.

3.12 Vehicle Area Network, VAN

VAN is intended for use in the body domain of a vehicle. The frame format and data rate is similar to CAN but it also has some additional features. VAN was used by Peugeot-Citroën for years but it was abandoned in favor of CAN [Robert Bosch GmbH, 2011, Navet and Simonot-Lion, 2009b]. VAN is of no further interest to this study since it was not adopted by the market.

3.13 Inter-Integrated Circuit, I²C

I²C is a synchronous, multi master bus with two bidirectional communication channels, serial data (SDA) and serial clock (SCL). These are implemented using open-drain I/Os with pull-up resistors to provide a dominant "0" and a recessive "1" for arbitration in the same way as CAN. The master node determines the clock speed and uses the SCL channel is used for synchronization. The protocol specifies a 7 bit address on a bus where every node has its unique address. The original bandwidth was 100 kb/s but today there are versions operating at bandwidths of up to 3.4 Mb/s. [Leens, 2009]

The maximum bus length is limited and the use of this bus is often limited to short distances. Other interfaces are more frequently used why this interface is of no interest for future implementation in Scanias vehicles. [Paret and Riesco, 2007]

3.14 J1850

The SAE J1850 is a class B bus available in a 10.4 kb/s single wire version or a 41.6 kb/s two-wire version. It is designed for use in the body domain to meet real-time requirements. J1850 is expected to be phased out and replaced by other buses such as CAN or LIN [Navet and Simonot-Lion, 2009a]. It is widely used in USA but has not gained the same popularity in Europe [Nolte et al., 2005]. The bandwidth, the limited use in Europe and its loss of popularity doesn't make it fitted for implementation in Scanias vehicles.

3.15 Byteflight

Byteflight is a communication interface intended for use in passive safety-critical applications such as airbags and seat-belt tensioners. Some of its design goals was to provide support for event-triggered traffic and a higher bandwidth than CAN (10 Mb/s) [Nolte et al., 2005]. Byteflight is not suited for x-by-wire systems because of its lack of fault-tolerance [Kopetz et al., 2000]. This was however added in connection to Byteflight's implementation into the FlexRay protocol. There are many similarities between the two protocols because of this, one of them is the FTDMA design [Kopetz et al., 2000]. Because of the development of FlexRay the use of Byteflight has decreased and it will not be studied further.

3.16 Motorola Interconnect

Motorola Interconnect is an interface similar to LIN in many ways. It's a simple, low-cost master/slave network intended for use in the body domain. Seat and mirror adjustment and window lifts are some areas of application. LIN is almost exclusively used instead of Motorola Connect why this won't be studied further. [Nolte, 2006, Nolte et al., 2005]

3.17 Distributed Systems Interface

The Distributed Systems Interface, DSI, was primarily developed as a safety-critical network for use in airbag systems. It can however also be used for connecting ECUs and sensors in general. The development was aimed at creating a sensor bus protocol with improved EMC characteristics and higher bandwidth then other similar protocols. [DSI Consortium, 2011]

Three network topologies are supported, point-to-point interconnections, daisy chain and bus. Every network has a node and up to 15 slave nodes communicating with separate command and response phases using the TDMA method. DSI uses two channels, a forward communication channel from a master to a slave and a reverse communication channel from a slave to a master. The former uses voltage fluctuations and the latter current fluctuations when communicating [DSI Consortium, 2011]. The forward communication channel can communicate at 125 kb/s and the reverse communication channel at 256 kb/s [DSI Consortium, 2010].

The signals on the forward communication channel are coded using Manchester encoding in which a logical 0 is represented by a transaction from low to high and a logical 1 is represented by a transaction from high to low. The reverse communication channel uses three different current level to represent the signals. The data to send is divided into 4-bit blocks (16 possible combinations) represented by three bits (27 possible combinations) on the bus. DSI uses an eight bit CRC to detect errors. [DSI Consortium, 2011]

3.18 J1708/J1587

SAE J1587 defines the communication between ECUs on a network with a physical layer specified by SAE J1708. It's designed for use in heavy-duty vehicles and operates at 9600 baud/s. SAE J1708/J1587 has been replaces by SAE J1939 and is not viable for implementation in Scanias vehicles. [Voss, 2008]

3.19 Serial Peripheral Interface, SPI

Serial Peripheral Interface (SPI) is a single master/slave protocol. The protocol is based on four signal wires [Leens, 2009]:

Clock signal: All signals are synchronous to the clock signal distributed from the master to all slaves on this channel.

Slave select: Used to select which slave to communicate with.

Master Out - Slave In: A channel for data transfer from the master to the slaves.

Master In - Slave Out: Data channel from the slaves to the master.

SPI does not define any addressing, flow control, acknowledgement mechanism or maximum data rate [Leens, 2009]. The lack of these functions and the complicated design compared to other interfaces makes SPI a poor choice for implementation.

4

Theoretical Results

The results presented in this chapter are based on conclusion drawn from the studies in chapter 2 (Automotive Embedded Systems) and chapter 3 (Available Communication Interfaces). The interfaces are presented in accordance with the functional domains described in section 2.5. The interfaces in the powertrain and chassis domains shows similarities in Scanias vehicles today why they both are described in section 4.1. The results from the study of the body domain can be found in section 4.2.

4.1 Powertrain and Chassis Domains

The communication interfaces meeting the requirements in the powertrain and chassis domain are all class C or class D interfaces. Bandwidth is one important factor but reliability is also important because of the safety critical systems found in these domains. There are four interfaces apart from CAN that are of interest for a closer study considering this, Ethernet, FlexRay, TTP/C and TTCAN. The first three all offer a considerably higher bandwidth than CAN which makes them interesting alternatives. The last three offers reliability through static scheduling, something that could prove to be important and need to be considered even though the Scania CAN network doesn't offer it today. One of the major differences between the four interfaces is the network access method used. Three categories can be identified.

- CAN and Ethernet: These interfaces are solely based on one of the CSMA methods providing flexibility but less of real time characteristics.
- TTP/C: This interface is based on a TDMA scheme.

- TTCAN and FlexRay: Offers communication based on TDMA but also support for time-triggered communication to some extent.

Scania has up to this point based all of its inter-ECU communication on the CSMA/CR based J1939 protocol. No need for an interface offering static scheduling has been seen so far. The modular system used in Scanias vehicles can be a reason to this. A great number of possible combinations of the ECUs are possible since only a small number of them are mandatory. This makes static scheduling unpractical since all of the combinations have to be considered during the design process. A mass-produced vehicle with a given setup of ECUs can be designed without the need for modifications from vehicle to vehicle.

One major advantage with CAN (and J1939) is the fact that it's the de facto standard within the automotive industry today. It's been used for a long time and can be considered both mature and stable. It is well suited for event-triggered communication and prioritizing and immediately reacting to important messages like alarms thanks to its arbitration process. This can however also turn into a problem. There is no way of anticipating the asynchronous triggered messages why a repeated high priority message can stop low priority messages from reaching the network. A babbling idiot could completely block other nodes from reaching the bus.

The maximum CAN wire length is somewhat limited with its 40 meters at 1 Mb/s but it is considerably higher for lower bandwidths (see table 3.1). J1939 has a 40 meter limit as well even though it's only defined for 250 and 500 kb/s. There are no apparent differences between the physical layers explaining the different limits and the J1939 protocol doesn't imply any restrictions either. The most probable explanation is that the J1939 specifications are set to be more strict to avoid runtime problems. The limited length can be a problem in Scanias vehicles, especially in articulated buses where the wires can reach a length of over 40 meters. 1 Mb/s CAN could be problematic in these vehicles because of the propagation delay on the wires. A signal has to be able to propagate from one end of the bus to the other and back in order to guarantee a correct arbitration process, otherwise what is read from the bus could be incorrect. This is a physical limitation that cannot be solved by the protocol why CAN always will be limited when it comes to bandwidth. TTCAN suffers from the same limitation on wire lengths and bandwidth but offers time-triggered communication as well.

FlexRay is a relatively new communication interface that is getting more popular in automotive applications. It offers a high bandwidth compared to CAN and a high degree of reliability with its TDMA scheme. This limits the use of FlexRay in Scanias vehicles with the modular system used because of the reasons mentioned earlier. One area of use could however be between mandatory ECUs with a need for high bandwidth and reliability. This would not require the static scheduling to be modified from vehicle to vehicle since the nodes are always present.

Another drawback of using FlexRay is the maximum distances between nodes in a FlexRay network. The longest wires in an articulated bus would require at least one ECU or repeater to meet these requirements which would add an extra delay to the signal. The dynamic segment can just like CAN suffer from high priority nodes blocking nodes with a lower priority to send. This can however avoided using a bus guardian.

The use of TTP/C in Scanias vehicles is also limited because of its time-triggered nature. It does however offer a bandwidth higher than the one supported by CAN and FlexRay.

There are a number of factors indicating that Ethernet will gain popularity within the automotive industry in the years to come. It's a widely used standard within other areas that meets the need for higher bandwidths. It's available off-the-shelf which makes it cost effective compared to many other interfaces. Even though it's not covered by this study, Ethernet can also be used for diagnostics since it's a standardized interface. A possible application for Ethernet is as a backbone in an embedded network. It would provide a high bandwidth interface to take the load off other interfaces used. Another interesting feature is Power Over Ethernet that could be used to transfer power as well as signals on the same wires. Something that could be of interest to examine closer is how the high bandwidth affects the emission spectrum from the bus (see section 7).

4.2 Body Domain

Low-speed CAN (bit rates up to 125 kb/s) used in the body domain basically has the same characteristics as high-speed CAN in the powertrain and chassis domains. It's widely used, stable and reacts quickly to events. The lower bit rate also allows longer wires between the ECUs.

Another interface widely used in this domain is LIN. The biggest difference between it and low-speed CAN can be found in the physical layer. CAN uses differential signalling on two wires whereas LIN is based on a single wire bus. This should theoretically affect the electromagnetic compatibility since differential signalling reduces the emission and increases the immunity to electromagnetic interference. The emission levels from the two interfaces might very well match because of this even though LIN is far more limited when it comes to bit rate.

There are no clear indications on the direction of communication interfaces used in the body domain. From an EMC perspective low-speed CAN is a far better alternative because of its differential nature. What speaks in favor of LIN is its simplicity and hence the low costs associated with it.

5 Measurements

The first section in this chapter presents the tested interfaces and the equipment used in all of the measurements (section 5.1). A description of the EMC lab and the test equpment used can be found in section 5.2. This section also have a few comments on the test method used. The results are presented in 5.3-5.6 along with a few notes on the test setup for the particular interface.

5.1 The Tested Interfaces

The tests were carried out with the domains in section 4 (Theoretical Results) in mind. A SofTec SK-S12XDP512-A RevC evaluation board supporting low-speed CAN (125 kb/s), high-speed CAN (1 Mb/s), LIN and FlexRay (using a Freescale MFR4310FRDC FlexRay daughter card) was used when conducting the tests. For more information about the hardware used, refer to the SofTec SK-S12XDP512 [Microsystems, 2007] and Freescale MFR4310FRDC [Inc., 2007] user manuals.

The interfaces tested in the powertrain and chassis domains were High-speed CAN and FlexRay. FlexRay was chosen because it's a relatively new interface that need to be examined closer. Measurements were also made on an existing ECU using J1939 for comparative purposes. Initially the intention was to test Ethernet but the available test equipment radiated to a degree that made it impossible to determine what was caused by the wiring. Similar problems would later arise in the FlexRay tests as described in section 5.6.

Low-speed CAN and LIN are the tested interfaces suited for use in the body domain. The main purpose was to compare the emission levels between the differential, two wire CAN bus and the single wire, non-differential LIN with its lower bit rate.

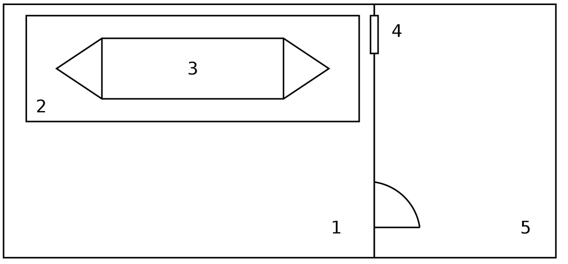


__Figure 5.1:__ The test environment at Scania consisting of two shielded rooms connected by a cable duct. The stripline antenna is located in the left chamber in the figure and the power supply, spectrum analyzer and computers in the right.
(1) Shielded room containing stripline antenna, (2) Ground plane, (3) Stripline antenna, (4) Cable duct, (5) Shielded room for power supply and measuring equipment

5.2 Test Environment

All tests have been conducted in a EMC lab at RECU, Scania. It consists of two shielded rooms connected by a cable duct as seen in figure 5.1. The EMC chamber contains a stripline antenna suited for measurements up to 300 MHz. Power can be provided through the cable duct, filtered to eliminate high frequency components. The filter used, FCP-11-2x30, provides an insertion loss of over 100 dB from 100 kHz to 10 GHz in a 50 Ω system.

The following equipments was used in all of the tests:

- Agilent (Hewlett Packard) E7402A EMC Analyzer
- EMCO 3825/2 LISN (Line Impedance Stabilization Network)
- FCP-11-2x30 power line penetration filter for shielded enclosures

The test is based on the use of a stripline antenna creating an electrical field between two ground planes. The wire is placed between the planes separated from the lower one by 5 cm [International Organization for Standardization, 2002]. This distance represents the clamping and routing of cables on a fixed distance from a structure in a vehicle [United States Department of Defence, 1999]. When twisted pair cables were used (during the FlexRay, low-speed and high-speed CAN and J1939 tests) the wiring consisted of two ⌀1 mm cables with a twist rate (also known as pitch) of 36 times/meter as in Scanias vehicles. Unfortunately no RF absorbing material has been available for use in the test chamber during the tests as suggested by ISO (International Organization for Standardization) 11452-5.

The power supply was placed outside of the EMC chamber in the second shielded enclosure and the power lines were filtered before they were connected to the

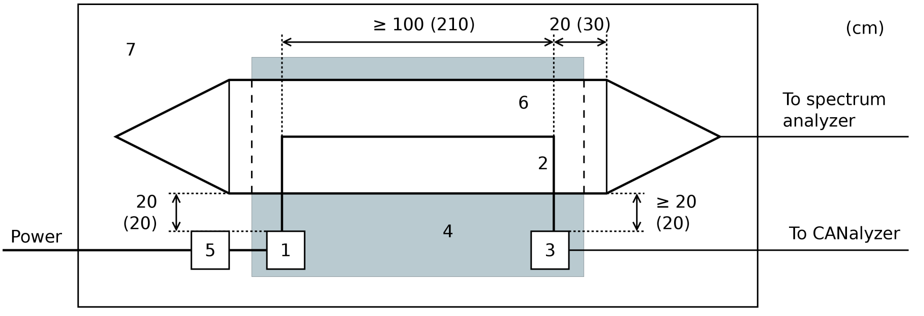


Figure 5.2: *J1939 test setup with distances given by ISO 11452-5. Values within parentheses are the actual distances used during the tests.*
(1) DUT, (2) Wire, (3) Fiber optics receiver, (4) Insulating base, (5) LISN, (6) Stripline antenna, (7) Ground plane

DUT (Device Under Test). A LISN has been used in the test setup as well. This serves a number of purposes, it provides a constant impedance to the DUT and filters the power fed to the DUT among others.

CANalyzer was used when testing units used in Scanias vehicles today. It offers the possibility of customize the traffic on the bus and also provides a lot of statistics like bus load. CodeWarrior was used for the low-speed CAN, high-speed CAN, FlexRay and LIN tests. It is only used to program the SofTec EVB and doesn't offer any statistics or way to control the traffic. The bus load should however be significant since the traffic is based on the analog to digital converter found on the board. The value is sent over the bus as soon as it's available resulting in a heavy bus load.

All measurements has been done after a recalibration of the spectrum analyzer. The figures contain the background noise, the emission from the equipment when power fed and the emission with traffic on the bus for comparative reasons. These are called Ambient, Power and Load respectively in the spectra found in this report.

5.3 Scania ECU (J1939)

The test of the J1939 bus used in Scanias vehicles today has to the greatest extent possible been conducted in accordance to ISO 11452-5, Road Vehicles - Component test methods for electrical disturbances from narrowband radiated electromagnetic energy. The DUT is a Scania ECU designed for use on a 500 kb/s CAN bus. CAN traffic was generated using CANalyzer on a computer outside of the test chamber. The signals were transfered into the chamber using fiber optics. The fiber optic CAN transceiver was power fed using an internal battery. The test setup, the distances given by the ISO 11452-5 and the ones used can be found in figure 5.2. The picture in figure 5.3 is taken in the EMC lab used.

Figure 5.3: *EMC chamber during J1939 test. The LISN can be seen standing on the ground plane to the left, the ECU on the insulating base to the left, the fiber optics receiver on the insulation to the right and the cable duct in the background to the right.*

The following equipment was used during the J1939 tests apart from that described in section 5.2:

- Scania ECU
- SonTec OPTOCAN 2000 fiber optics transceivers
- Windows 7 computer running CANalyzer 7.2.38

Initially two different measurements were conducted on the J1939 bus. The first one with the ECU power fed resulting in a 0.86% bus load and the second one with a maximum bus load generated by CANalyzer. A measurement of the ambient noise was also conducted for comparative purposes and to evaluate the shielding of the EMC chamber.

The maximum bus load reached in the tests was always below 93%, 92.14% in this case. It's unclear to why this number wasn't exceeded since the maximum allowed number of signals were sent from CANalyzer with a short periodicity. It does however give a good picture of how the bus behaves under heavy load.

Figure 5.4 shows an overview of the emission from 1 MHz to 300 MHz. The step in the measurements is due to the change of resolution bandwidth from 9 kHz to 150 kHz. Note the evenly distributed radiation for frequencies lower than roughly 20 MHz and the low level radiation compared to the ambient level for higher frequencies.

The radiation levels from the power fed ECU and the background noise in the lower band are approximately the same apart from additional spikes and the interval between 10 and 20 MHz. The latter can be explained by the CAN traffic and 0.86% bus load. The resonance frequencies from the 500 kb/s bus at 92.14% load can be seen in the repetitive pattern in figure 5.5 where a part of this frequency

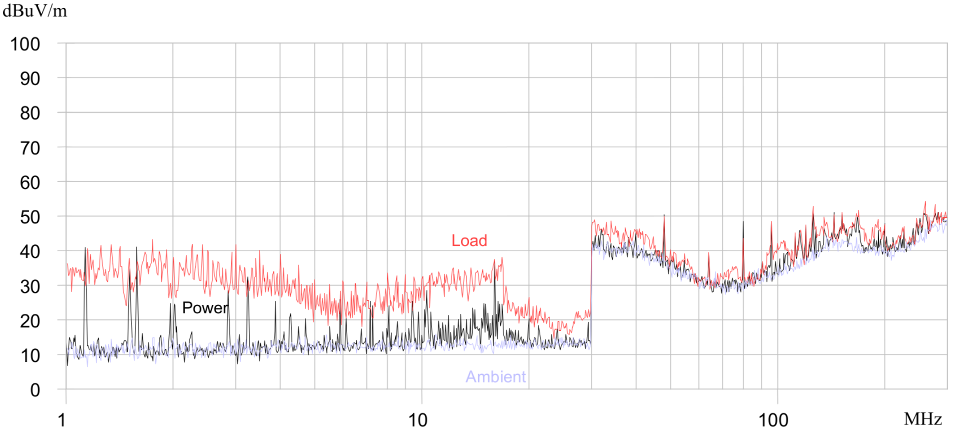


Figure 5.4: *Emission spectrum from a Scania ECU using J1939, 1-300 MHz. The step is due to the change in resolution bandwidth. The increase in emission is relatively evenly distributed for lower frequencies. Power feeding the unit or sending data does not affect the spectra to any greater extent for frequencies over 30 MHz.*

band has been plotted on a linear scale.

5.4 CAN

The CAN (both low and high speed) test setup can be seen in figure 5.6. The only difference between the tests is the transceivers used. The following equipment was used apart from that described in section 5.2:

- SofTec SK-S12XDP512-A RevC evaluation board
- CodeWarrior IDE v5.9.0 running under Windows XP SP3

The spectrum from the high-speed CAN measurements is shown in figure 5.7. Figure 5.8 shows the spectrum from the measurements of low-speed CAN in the interval 1-300 MHz.

Low-speed and high-speed CAN shows some similarities as earlier mentioned. The spikes at frequencies below 10 MHz are present in both figure 5.8 and 5.7 just as the broadband increase in radiation for 10-20 MHz. The latter is hard to associate to the communication interface itself since there's also a significant increase there when power feeding the EVB. There is very little or no resemblance between the two and the J1939 spectrum in figure 5.4 even though the physical layer is more or less the same. This of course can be due to the fact that J1939 has been tested in an existing ECU and not on a EVB.

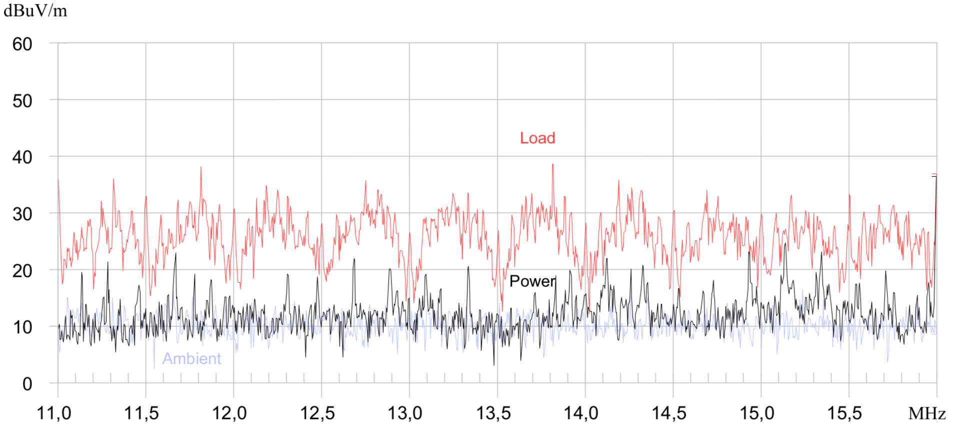


Figure 5.5: *Emission spectrum from a Scania ECU using J1939, 11-16 MHz. The resonance frequencies can be seen in this plot in the pattern repeated every 500 Hz.*

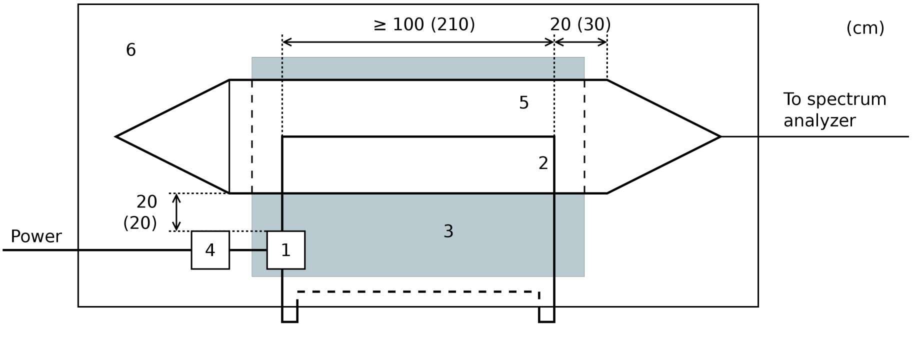


Figure 5.6: *Test setup for CAN and LIN tests. The return wire is routed under the ground plane to prevent it from affecting the measurements.*
(1) Evaluation board, (2) Wire, (3) Insulating base, (4) LISN, (5) Stripline antenna, (6) Ground plane

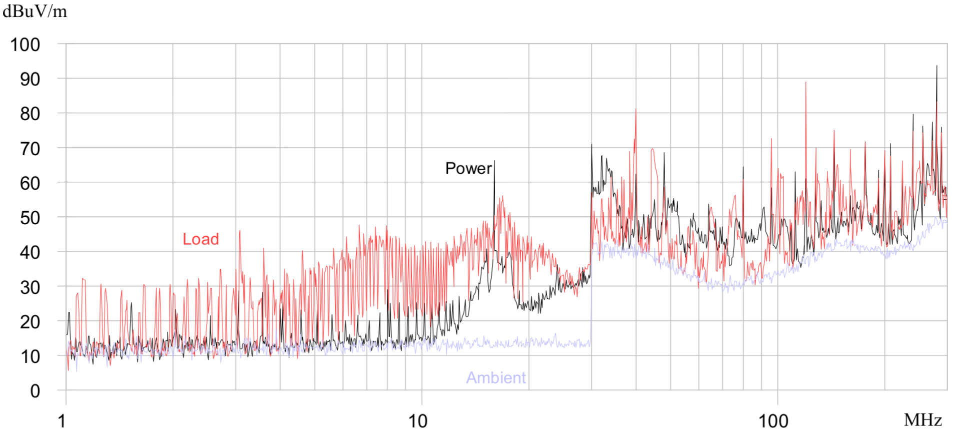


Figure 5.7: *Emission spectrum from high-speed CAN measurements on the SofTec evaluation board, 1-300 MHz. The step is due to the change in resolution bandwidth. The low-speed and high-speed CAN spectra shows some similarities, the spikes at lower frequencies among others.*

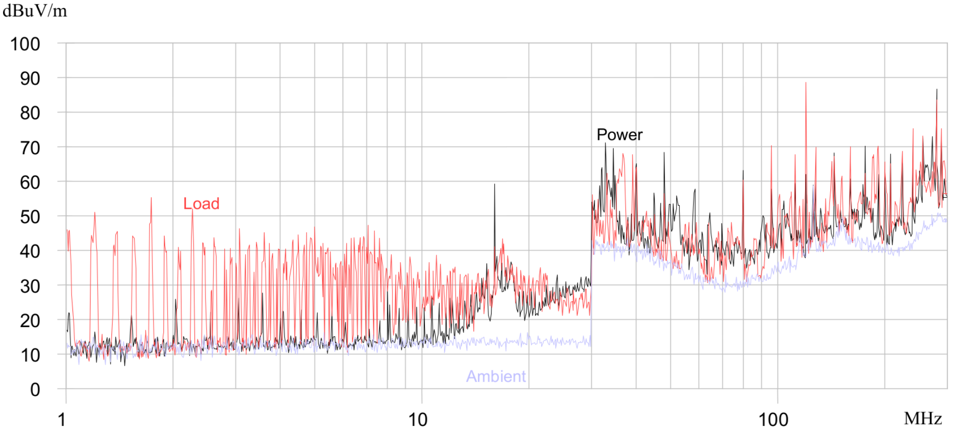


Figure 5.8: *Emission spectrum from low-speed CAN measurements on the SofTec evaluation board, 1-300 MHz. The step is due to the change in resolution bandwidth. The spectrum, especially the spikes at lower frequencies, reassembles that of high-speed CAN except for the signal levels.*

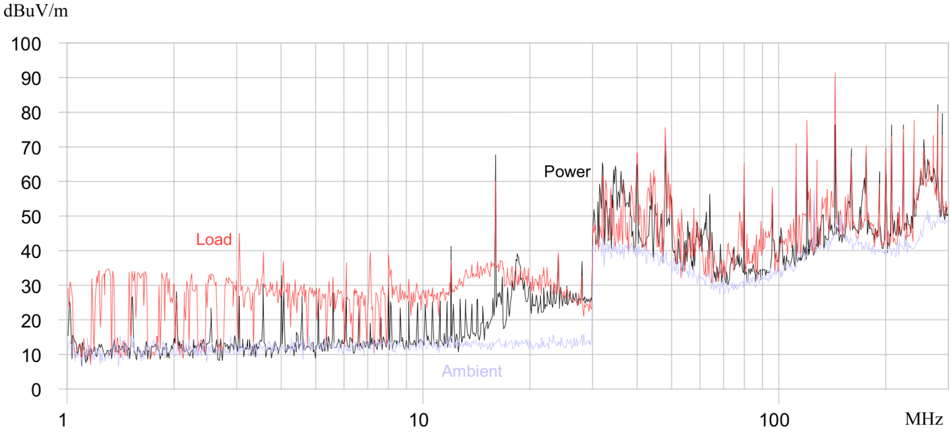


Figure 5.9: Emission spectrum from LIN measurements on the SofTec evaluation board, 1-300 MHz. The step is due to the change in resolution bandwidth. A difference to the tested CAN interfaces can be seen at lower frequencies where the spikes are broader. The emission levels are also more even over the whole spectrum.

5.5 LIN

The following equipment was used during the LIN tests apart from that described in section 5.2:

- SofTec SK-S12XDP512-A RevC evaluation board
- CodeWarrior IDE v5.9.0 running under Windows XP SP3

The setup is identical to the one used on the CAN measurements apart from the single wire used instead of a twisted pair cable and the interfaces used on the evaluation board. The test setup can be seen in figure 5.6.

The result from the measurement of LIN in the frequency span 1-300 MHz can be seen in figure 5.9. The emission levels are more evenly distributed than in the CAN measurements. There are no spikes in lower frequencies as in the CAN spectra but rather broader bands of radiation. It's hard to tell whether this is a characteristic of the interface in general or the particular transceiver used.

5.6 FlexRay

The following equipment was used during the FlexRay tests apart from that described in section 5.2:

- SofTec SK-S12XDP512-A RevC evaluation board
- Freescale MFR4310FRDC FlexRay daughter card

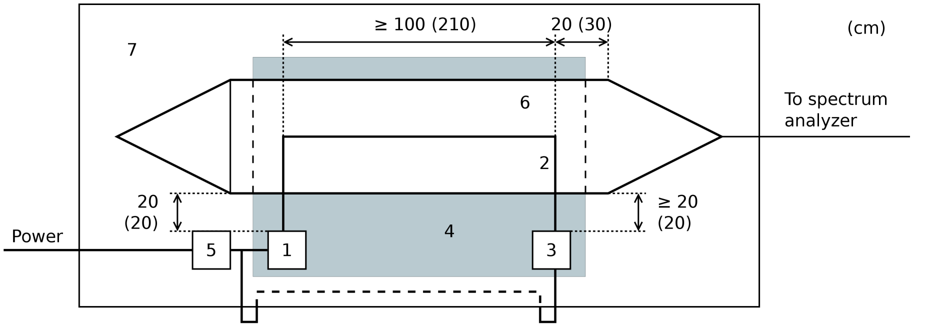


Figure 5.10: *Test setup for FlexRay test. Two evaluation boards has been used in this test, both power fed through the* LISN.
(1) Evaluation board 1, (2) Wire, (3) Evaluation board 2, (4) Insulating base, (5) LISN, *(6) Stripline antenna, (7) Ground plane*

- CodeWarrior IDE v5.9.0 running under Windows XP SP3

The test setup can be seen in figure 5.10.

The FlexRay spectrum in figure 5.11 distinguishes itself from the others in one important way. There's a significant difference between the ambient measurement and when the unit is power fed (marked Power in the figure) at all frequencies. All other interfaces has a low level of radiation (apart from spikes) up to 10 MHz and a much lower level of radiation for higher frequencies. There can be a number of reasons for this. The transceiver naturally contributes but can't explain the wideband nature of the emission. A more likely explanation is the design of the FlexRay card and how it's connected to the evaluation board. Regardless, it's hard to tell what's caused by the transceiver, the evaluation board and the interface which makes any results inconclusive.

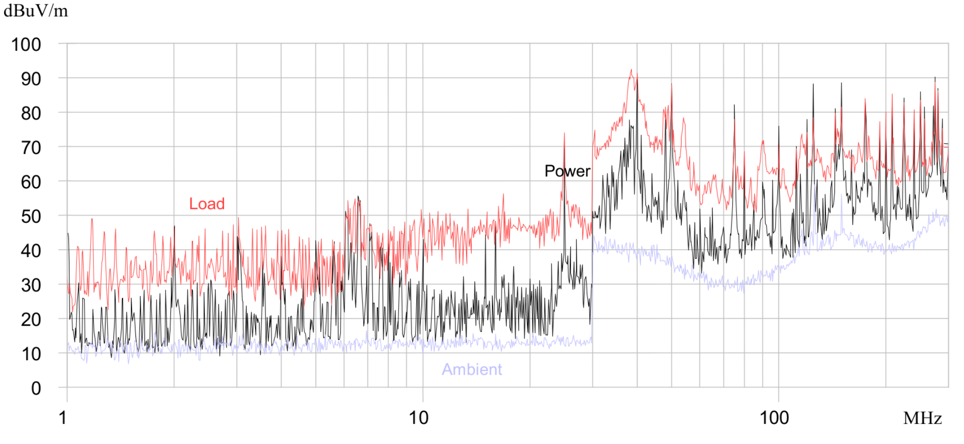


Figure 5.11: *Emission spectrum from FlexRay* EVB*, 1-300 MHz. The step is due to the change in resolution bandwidth. Note the increase in emission levels over the whole spectrum after power feeding the evaluation boards.*

6

Results and Conclusion

6.1 Results

It's hard to draw any far-reaching conclusions from the performed tests as described throughout chapter 5. It's obvious that the emission levels are highly dependant on the equipment used. E.g. there is no apparent reason to why the emission levels from a low-speed CAN transceiver should be that much higher than that of a high-speed CAN transceiver as in the performed tests (see figure 5.8 and 5.7). The design of the evaluation boards has probably affected the test results to an extent where no or very few conclusions are possible to draw as well. An example of this is the FlexRay tests. The radiation from the EVB polluted the spectrum in the whole examined frequency band as seen in figure 5.11 (and described in section 5.6). This makes it even harder to recognize any differences between the interfaces themselves.

The approach to test communication interfaces as performed in this study was insufficient. The evaluation boards available on the market are more suited to test functionality than hardware. A better solution would probably be to test transceivers themselves on a custom-designed circuit board. This would give the designer a possibility to control the layout and hence minimize the radiation caused by the board itself. Testing a larger number of transceivers could probably also make it possible to recognize characteristics of the different interfaces. This is harder when only a single transceiver of each interface is tested. An example of this is the measurements of the LIN and high-speed CAN interfaces on the EVB. The radiation level is roughly the same at lower frequencies (see figure 5.7 and figure 5.9) and this could definitely be caused by differences in the physical layer of the two interfaces. The differential, two wire CAN bus should give rise to a much lower emission level than the single wire LIN bus. The broader emission bands for low frequencies for LIN compared to CAN could also be a characteristic for the interface. It is however hard to confirm when only a single transceiver of each interface has been used.

The spectrum from the measurement of the Scania ECU shows few similarities with the spectra from any of the CAN interfaces on the EVB as described in section 5.4. Again, the rest of the hardware affects the measurements to an extent where no conclusions are possible to draw.

There are interfaces that are of interest for a closer study. Ethernet is becoming increasingly popular within the automotive industry and could very well be used in Scanias vehicles in the future. One possible use could be as a backbone in the network where a high bandwidth is needed. FlexRay will probably not be used in the same way because of the TDMA scheme and the layout of the network in Scanias vehicles. It could however be used between mandatory ECUs where the nodes are known beforehand and static scheduling is not a problem. The modular system used in the vehicles makes it unpractical to use FlexRay in other ways (see section 4.1).

6.2 Conclusion

This study has evaluated a number of communication interfaces used in automotive applications from a hardware and EMC perspective. A theoretical study identified interfaces apart from CAN of interest for a closer study from what is required of them by Scania. The theoretical results can be found in chapter 4 but a brief description can be found below.

Powertrain and chassis domain

- Ethernet is definitely an alternative to the CAN bus used today. It offers high bandwidth, event-triggered communication and it could be use as a backbone in a network. The interface is described closer in section 3.6
- FlexRay is inteded to use in networks requiring high bandwidth and reliability. It's based on static scheduling which makes the use of it limited in Scanias vehicles because of the modular system used. It does however offer event-triggered communication to some extent through minislotting. More information on FlexRay can be found in section 3.3
- TTCAN is a higher level CAN protocol. It has the same limitations as CAN in terms of bandwidth and bus length but offers static scheduling and high reliability. The interface is described in section 3.4.
- TTP/C is based on static scheduling which limits its use in Scanias vehicles. It is described closer in section 3.5.

Body domain

- LIN is an alternative to low-speed CAN in many areas. Its main advantages is its simplicity and low cost. More information on LIN can be found in section 3.7.

The following interfaces were tested in each domain:

Powertrain and chassis domain: High-speed CAN, J1939 (Scania ECU), FlexRay

Body domain: Low-speed CAN, LIN

Few conclusions are possible to draw from the tests with the test method used. The differences between the spectra are hard to trace to the interface itself and not the transceivers used and the layout of the circuit boards.

Future tests should be based on measurements of a number of transceivers of the same interface if conclusions about the interfaces themselves are to be drawn. A custom based circuit board could be used in order to minimize the effect the layout has on the emissions.

7 Future Work

There is an increasing interest for Ethernet as an inter-ECU communication interface in the automotive industry today. There are a number of advantages compared to other interfaces. It's a stable and well used standard, cost effective since it's available off-the-shelf and supports high bandwidths. Ethernet has been studied at Scania earlier but at a higher level perspective in the OSI model. A study of Ethernet based on hardware and EMC characteristics would add to an overall picture of the interface.

FlexRay is another interface that could need a closer study. It's relatively new and there's a lack of knowledge about it at Scania. There are areas in which it could prove useful and examining the hardware and EMC aspects of it could be important.

A few remarks on the test method used in this study and how it could be improved can be found in chapter 6.

Bibliography

AlphaWire. Understanding Shielded Cable. http://www.newark.com/pdfs/techarticles/alphaWire/USC.pdf, 2009. Accessed April 24 2012. Cited on page 21.

Gianluca Cena and Adriano Valenzano. Controller Area Networks for Embedded Systems. In Richard Zurawski, editor, *Embedded Systems Handbook*. CRC Press, Boca Raton, 2nd edition, 2009. Cited on pages 9, 23, 24, and 25.

DSI Consortium. Distributed Systems Interface. http://www.dsiconsortium.org/downloads/DSI3M_20100617.pdf, 2010. Accessed March 15 2012. Cited on page 42.

DSI Consortium. DSI Bus Standard. http://www.dsiconsortium.org/downloads/DSI3_Bus_Standard_r1.00.pdf, 2011. Accessed March 5 2012. Cited on pages 42 and 43.

Wilfried Elmenreich and Richard Ipp. Introduction to TTP/C and TTP/A. Technical report, Institut fr Technische Informatik Vienna University of Technology, TTTech Computertechnik, 2003. Cited on pages 31, 32, 33, and 38.

Joakim Fröberg, Kristian Sandström, Christer Norström, Hans Hansson, Jakob Axelsson, and Björn Villing. Correlating bussines needs and network architectures in automotive applications - a comparative case study. In *Proceedings of the 5th IFAC International Conference on Fieldbus Systems and their Applications (FET)*, pages 219–228. IFAC, July 2003. Cited on page 12.

Magnus Gustafsson. Datakommunication, CAN. Technical report, RESA, Scania, 2010. Cited on page 21.

Tom Hajjar. Power Over Ethernet (PoE). Technical report, L-com Global Connectivity, 2009. Cited on pages 33, 35, and 36.

Freescale Semiconductor Inc. Freescale Generic FlexRay Daughter Card (FRDC) FLXRAYIPMFR4200 & MFR4300FRDC/MFR4310FRDC User Manual. http://cache.freescale.com/files/microcontrollers/doc/

user_guide/MFR4300_UM.pdf, 2007. Accessed May 28 2012. Cited on page 49.

Krzysztof Iniewski. *Internet Networks: Wired, Wireless, and Optical Technologies*. CRC Press, Boca Raton, 2009. Cited on pages 33, 34, 35, and 36.

International Organization for Standardization. *ISO 11452-5 Road Vehicles, Component Test Methods for Electrical Disturbances from Narrowband Radiated Electromagnetic Energy: Part 5, Stripline*. ISO, 2002. Cited on page 50.

Greg Jarboe, Hollis Thomases, Mari Smith, Chris Treadaway, and Dave Evans. *Networking Complete*. Sybex, Hoboken, 3d edition, 2002. Cited on page 8.

Carl Jeffery, Rueben Cutajar, Andrew M. D. Richardson, Steve Prosser, M. Lickess, and Steve Riches. The integration of on-line monitoring and reconfiguration functions into a safety critical automotive electronic control unit. *Journal of Electronic Testing*, 21(4):405–416, 2005. ISSN 0923-8174. Cited on page 12.

Uğur Keskin. In-vehicle Communication Networks: a Litterature Survey. Technical report, Technische Universiteit Eindhoven, 2009. Cited on pages 11, 12, 13, 14, 15, and 31.

Harald Kohler. MOST150 - The Next Generation Automotive Infotainment Backbone. http://www.itu.int/dms_pub/itu-t/oth/06/10/T06100008010001PDFE.pdf, 2008. Accessed April 13 2012. Cited on page 39.

Philip Koopman. The FlexRay Protocol. http://www.ece.cmu.edu/~ece649/lectures/23_flexray.pdf, 2011. Accessed May 21 2012. Cited on page 17.

Hermann Kopetz, Wilfried Elmenreich, and Christoph Mack. A comparison of LIN and TTP/A. In *3rd IEEE International Workshop on Factory Communication Systems (WFCS 2000), 6.-8. September 2000, Porto, Portugal*, pages 99–107, September 2000. Cited on pages 20 and 42.

Frédéric Leens. An Introduction to I²C and SPI Protocols. *IEEE Instrumentation & Measurement Magazine*, 12(1):8–13, 2009. Cited on pages 41 and 43.

Hyung-Taek Lim, Kay Weckemann, and Daniel Herrscher. Performance study of an in-car switched ethernet network without prioritization. In *Proceedings of the Third international conference on Communication technologies for vehicles*, Nets4Cars/Nets4Trains'11, pages 165–175, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 978-3-642-19785-7. Cited on page 9.

LIN Consortium. *LIN Specification Package*. LIN Consortium, Munich, Germany, rev 2.2a edition, 2010. Cited on pages 37 and 38.

P. Marwedel. *Embedded System Design*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 1402076908. Cited on pages 11 and 20.

Maxim. Glossary of Electrical Engineering Terms. http://www.maxim-ic.com/glossary, 2012. Accessed April 23 2012. Cited on page 10.

Eugen Mayer. Serielle Bussysteme im Automobil. Technical report, Vector, 2008. Cited on pages 5 and 6.

Mercedes Benz. Domestic Digital Bus (D2B). http://www.mercedestechstore.com/pdfs/416_Telematics/416HOD2B(CooksonI)03-09-04.pdf, 2004. Accessed April 17 2012. Cited on page 40.

SofTec Microsystems. SK-S12XDP512-A User's Manual. http://www.softecmicro.com/downloads/doc/sk-s12xdp512-a_manual.pdf, 2007. Accessed May 28 2012. Cited on page 49.

Motorola Inc. LIN protocol description. http://www.freescale.com/files/microcontrollers/doc/reports_presentations/LINOVERVIEWPRESENT.ppt, 2001. Accessed January 31 2012. Cited on pages 18 and 37.

Nicolas Navet and Françoise Simonot-Lion. A Review of Embedded Automotive Protocols. In Nicolas Navet and Françoise Simonot-Lion, editors, *Automotive Embedded Systems Handbook*, Industrial Information Technology Series, pages 4.1–4.31. Taylor & Francis / CRC Press, 2009a. ISBN 978-0-8493-8026-6. Cited on pages 5, 6, 7, 13, 14, 23, and 41.

Nicolas Navet and Franoise Simonot-Lion. Trends in Automotive Communication Systems. In Richard Zurawski, editor, *Embedded Systems Handbook*. CRC Press, Boca Raton, 2nd edition, 2009b. Cited on pages 13, 14, and 41.

Thomas Nolte. *Share-Driven Scheduling of Embedded Networks*. PhD thesis, Mälardalen University, May 2006. Cited on pages 6, 11, 12, 13, 30, 31, and 42.

Thomas Nolte, Hans Hansson, and Lucia Lo Bello. Automotive communications - past, current and future. In *Proceedings of 10th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'05)*, pages 985–992 (volume 1). IEEE Industrial Electronics Society, September 2005. Cited on pages 40, 41, and 42.

Roman Obermaisser. Time-Triggered Communication. In Richard Zurawski, editor, *Embedded Systems Handbook*. CRC Press, Boca Raton, 2nd edition. Cited on page 31.

D. Paret and R. Riesco. *Multiplexed networks for embedded systems: CAN, LIN, Flexray, Safe-by-Wire...* Wiley, 2007. ISBN 9780470034163. Cited on pages 6, 7, 8, 17, 18, 19, 23, 24, 26, 27, 28, 29, 30, 37, 38, 40, and 41.

Juan Pimentel, Julian Proenza, Luis Almeida, Guillermo Rodriguez-Navas, and Manuel Barranco Joaquim Ferreira. Dependable Automotive CAN Networks. In Richard Zurawski, editor, *Embedded Systems Handbook*. CRC Press, Boca Raton, 2nd edition. Cited on pages 11, 30, and 31.

Mathias Rausch. *FlexRay: Grundlagen, Funktionsweise, Anwendung*. Hanser Fachbuchverlag, Munich, 2007. Cited on pages 7, 8, 9, 17, 19, 20, 27, 28, 29, and 30.

Robert Bosch GmbH. Introduction to Automotive Bus Systems. `http://nik.uni-obuda.hu/mobil/tantargyak/bir/bsc/intro_farkas_external_2011_10_21.pdf`, 2011. Accessed February 2 2012. Cited on pages 7, 12, 13, and 41.

John M. Rushby. Bus architectures for safety-critical embedded systems. In *Proceedings of the First International Workshop on Embedded Software*, EMSOFT '01, pages 306–323, London, UK, UK, 2001. Springer-Verlag. ISBN 3-540-42673-6. Cited on pages 18, 19, 32, and 33.

Terence Rybak and Mark Steffka. *Automotive Electromagnetic Compatability*. Kluwer Academic Publishers, New York, USA, 2004. Cited on page 20.

Thilo Sauter. Fieldbus Systems: Embedded Networks for Automation. In Richard Zurawski, editor, *Embedded Systems Handbook*. CRC Press, Boca Raton, 2nd edition, 2009. Cited on pages 15, 16, and 17.

Ron Schmitt. *Electromagnetics Explained*. Newnes, 2002. Cited on page 20.

Françoise Simonot-Lion and Yvon Trinquet. Vehicle Functional Domains and Their Requirements. In Nicolas Navet and Françoise Simonot-Lion, editors, *Automotive Embedded Systems Handbook*, Industrial Information Technology Series, pages 1.1–1.22. Taylor & Francis / CRC Press, 2009. ISBN 978-0-8493-8026-6. Cited on pages 5 and 13.

Charles Spurgeon. *Ethernet: The Definitive Guide*. O'Reilly, Sebastopol, Carlifornia, USA, 2000. Cited on pages 33, 34, 35, and 40.

Thomas Strang and Matthias Röckl. Vehicle Networks: Multimedia Protocols. `http://www.sti-innsbruck.at/fileadmin/documents/vn-ws0809/06-VN-MultimediaNetworks.pdf`, 2008. Accessed April 13 2012. Cited on pages 10, 39, and 40.

Andrew Tanenbaum. *Computer Networks*. Prentice Hall PTR, Upper Saddle River, New Jersy, USA, 4th edition, 2003. Cited on pages 15 and 16.

Lothar Thiele. Embedded Systems Communication. `http://www.tik.ee.ethz.ch/education/lectures/ES/slides/8_Communication2.pdf`, 2010. Accessed March 5 2012. Cited on pages 17 and 24.

United States Department of Defence. *Requirements for the Control of Electromagnetc Interference Characteristcs of Subsystems and Equipment*. United States Department of Defence, 1999. Cited on page 50.

Pascal Unterdorfer. Power over Ethernet. Technical report, Hirschmann Automation and Control GmbH, 2008. Cited on pages 33, 35, and 36.

W. Voss. *A Comprehensible Guide to J1939*. Copperhill Media Corporation, 2008. ISBN 9780976511632. Cited on pages 26 and 43.

Andreas Willig and Hagen Woesner. Principles of Lower-Layer Protocols for Data Communications in Industrial Communication Networks. In Richard Zurawski, editor, *The Industrial Communication Technology Handbook*. CRC Press, Boca Raton, 2005. Cited on page 40.

Werner Zimmermann and Ralf Schmidgall. *Bussysteme in der Fahrzeugtechnik, Protokolle und Standards*. Vieweg, Wiesbaden, 2006. Cited on page 39.

Gerhard Zucker and Dietmar Dietrich. ISO/OSI Model. In Gurdeep S. Hura, editor, *Data and Computer Communications*. CRC Press, Boca Raton, 2001. Cited on pages 6 and 7.

Index