Simulation based Timing Analysis of FlexRay Communication at System Level

Stefan Buschmann Till Steinbach
Franz Korf Thomas C. Schmidt

Hamburg University of Applied Sciences
stefan.buschmann@haw-hamburg.de
{till.steinbach, korf, schmidt}@informatik.haw-hamburg.de

6th International Workshop on OMNeT++
March 5th, 2013

CoRE iNET

Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

SPONSORED BY THE
Federal Ministry of Education and Research

Agenda

1 Introduction

2 Background & Requirements

3 Concept

4 Results & Evaluation

5 Conclusion & Outlook

Motivation

Why simulate FlexRay on system level?

- State-of-the-art automotive fieldbus
- Simulation of complex networks
- Important in automotive development

Goals

- Configurable FlexRay simulation
- Compliant to FlexRay specification
- Compatible with other simulation models
    - CAN, Real-time Ethernet, Ethernet-AVB

Agenda

1 Introduction

2 Background & Requirements

3 Concept

4 Results & Evaluation

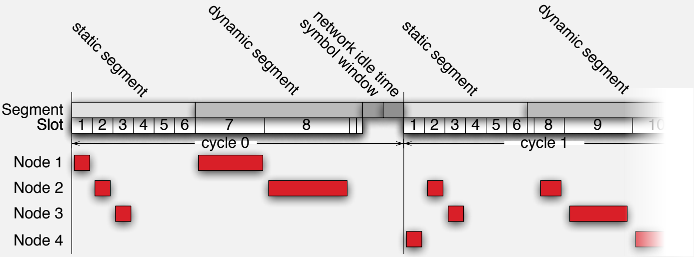
5 Conclusion & Outlook

Technical features

- Layer 1 and 2 in the OSI model
- Communication over two channels
  - Redundant transmission
  - Different data per channel
- 10 MBit/s per channel
- Synchronised time base
- Event- and time-triggered communication

FlexRay

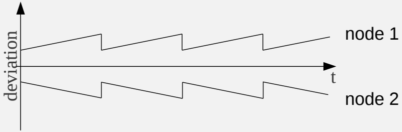
Communication cycle



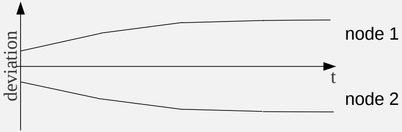
- Time-triggered communication
- Event-triggered communication

FlexRay
Synchronisation

- Time measurement with synchronisation messages in the static segment
- Combination of two synchronisation methods
  - Offset correction



  - Rate correction



Requirements

For the FlexRay model

- Layer 2 in the OSI model
- FlexRay functions
  - Communication
  - Synchronisation
- Implementation of a model of an oscillator
- Configuration of the network structure and the parameters
- System level error detection

Agenda

1 Introduction

2 Background & Requirements

3 Concept

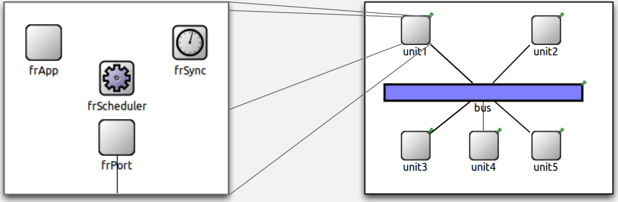
4 Results & Evaluation

5 Conclusion & Outlook

Concept

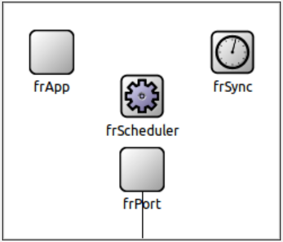
Architecture

- FlexRay nodes as modules
- Bus topology as module



Concept

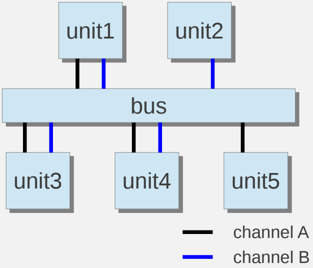
Node



- Several submodules
- Connection to the bus module
- Independent configuration

Concept
Topology

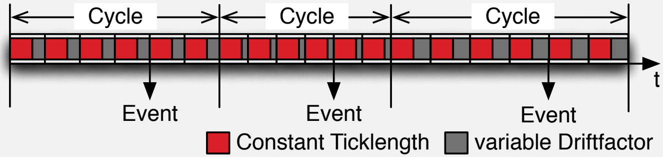
- OMNeT++ only provides point to point communication
- Bus topology
- Realised as module
- Provides a maximum of two connections for each node
- Distribution of incoming messages



Concept

Model of the oscillator

- Very accurate model would simulate every tick
    - Huge amount of events
- Our approach for the clock drift
    - Only one drift value per cycle
    - Reducing the number of events



Agenda

1 Introduction

2 Background & Requirements

3 Concept

4 Results & Evaluation

5 Conclusion & Outlook

Protocol conformance & error detection

Protocol conformance:

- Requirements are fulfilled

Typical error detection:

- Configuration problems
  - Too many sync nodes
  - Frames in the same slot
- Timing errors
  - Frames in wrong slot

Latency Analysis for the dyn. segment

Simulation parameter

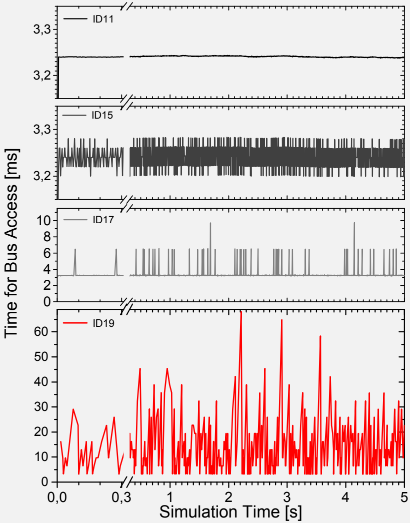
- 4 nodes
- 10 minislots
- Transmission points distributed over dynamic segment
- Dynamic frames require 1 to 3 minislots

dynamic segment

	1				2		3		4	
minislot ID	11	12	13	14	15	16	17	18	19	20

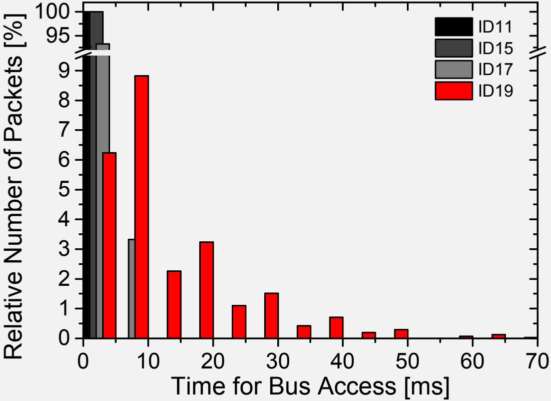
Latency Analysis for the dyn. segment

Latency of frames with different IDs



Latency Analysis for the dyn. segment

Latency distribution



Performance test

- Several networks of various size
- Only messages in the static segment
- Further parameters identical

number of nodes	channels	t_sim/t_real [s]
10	single	∼0.96
20	single	∼0.58
30	single	∼0.45
10	dual	∼0.62
20	dual	∼0.32

- Nearly worst case scenario
- Timing parameter and configuration have a large influence

Evaluation against CANoe

Comparison of latency results

- CANoe
  - Commercial network simulator
  - Variety of automobile communication protocols
- Two equal networks
  - Three nodes
  - Same parameters
- Same behaviour in both networks
  - Amount and timing of messages
  - Repressed dynamic frames
- Difference of approximately 100 ns

Agenda

1 Introduction

2 Background & Requirements

3 Concept

4 Results & Evaluation

5 Conclusion & Outlook

Conclusion

- System level simulation
- Support of different applications
- Evaluation against CANoe
- Can be used for simulation of complete communication-matrices

Outlook

- Extension of the simulation
  - Active star topology
  - Startup procedure and node integration during operation
- Gateway between FlexRay and other communication models
- Simulation of complex real communication-matrices

Thank you!

Thank you for your attention!

- Website of CoRE research group: http://www.haw-hamburg.de/core

SPONSORED BY THE

Federal Ministry
of Education
and Research