6th IFAC Symposium Advances in Automotive Control
Munich, Germany, July 12-14, 2010

A FlexRay parameter calculation methodology based on the electric/electronic architecture of vehicles

Matthias Heinz, Martin Hillenbrand, Patrick von Brunn, Klaus D. Mueller-Glaser

Karlsruhe Institute of Technology,
Insitute for information processing technology, Karlsruhe, Germany,
phone: +49 721 608-2501, fax: +49 721 608-2511
(e-mail: {heinz, hillenbrand, vonbrunn, klaus.mueller-glaser} @kit.edu)

Abstract:
The well established FlexRay bus system allows the adjustment to many configurations and applications. This flexibility comes for the price of a huge set of parameters which have to be mandatory correct,to operate the bus system robust and reliable. In this paper we provide a methodology to determine the necessary settings directly from the electric and electronic architecture description, which is developed in the design phase of a vehicle. Our approach includes the calculation of all global and local parameters, frame packing and scheduling. The implementation of the methodology was tested with various modeling data sets. The calculated parameters could be exported in a predefined data structure which was used to seamlessly configure target FlexRay controllers, successfully running a real hardware network.

Keywords: FlexRay, parameter calculation, automotive, electric/electronic architecture modeling, frame packing, scheduling, optimization, system architecture

1. INTRODUCTION

The FlexRay bus system is established by many car manufacturers. The great flexibility of the bus system perfectly fits into the different communication needs of most engineering manufacturers. However, this flexibility leads to a huge amount of adjustable parameters. The adjustment of more than 70 FlexRay parameters is not an easy task. Setting up an optimum set of FlexRay parameters to a working configuration can ideally be supported by tools. The calculation of an optimum configuration set can be based on the topology and signal transmissions on the bus.

Our approach shows an efficient method to determine an ideal set of parameters for the wide range of usage of FlexRay networks. This is achieved by using the data of an electric and electronic architecture (EEA) model, which is developed in the design phase of a vehicle. The automatic calculation allows for rapidly exploring different bus architectures and their corresponding configurations. The calculation is done for the static as well as the dynamic segment of the FlexRay protocol.

The paper is organized in five sections. Chapter 2 presents related work on this topic. A short introduction to FlexRay is given in chapter 3. The next chapter presents our approach to solve the calculation dependencies between the parameters, frame packing, message scheduling and the overall configuration of the FlexRay protocol parameters. Chapter 5 describes the verification and test of the approach on hardware FlexRay controllers followed by conclusions and outlook in section 6.

2. RELATED WORK

Tools like DaVinci Network Designer FlexRay [Vector Informatik GmbH (2009)], TTX Plan [TTTech Automotive GmbH (2009)] or FlexConfig [Eberspächer Electronics GmbH & Co. KG (2009)] allow for in detail configuration of FlexRay Network Parameters. Although there is a consistency checker for the values implemented, the user has to have in-depth knowledge about the correlation of parameters, to set them right corresponding to the applied bus topology and the communicated signals.

In literature there are several methods presented to solve the frame packing and schedule optimization problem. [Lukasiewycz et al. (2009)], [Pop et al. (2006)], [Wandeler and Thiele (2006)] and [Schmidt and Schmidt (2003b)] give an idea about the scheduling of the static segment on the FlexRay bus. Some methods, for example given in [Schmidt and Schmidt (2003b)] are not compliant with todays FlexRay controllers, because they use an arbitrary selectable cycle mask. Papers [Schmidt and Schmidt (2003a)] and [Pop et al. (2007)] additionally deal with the configuration of the dynamic segment. Papers [Hamann and Ernst (2005)], [Ding et al. (2005)] use evolutionary search algorithms for solving the scheduling problem.
However, these papers do not give an answer on how to calculate the full set of FlexRay parameters for the different communication nodes. Also there is no information where the necessary data, on which the calculations are made, is based on. Our approach shows the complete procedure from the input of topology information and signals till the configuration of physical FlexRay controllers.

3. FLEXRAY

A FlexRay communication node consists of a host controller running the application software and a FlexRay communication controller (CC) , which is responsible for the protocol operation (figure 1). Physical access to the bus is accomplished using FlexRay transceivers which implement the electrical potentials on the bus.

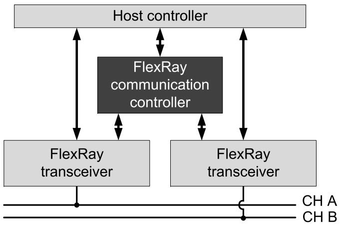

Fig. 1. FlexRay communication node

Communication on the bus is arranged in 64 equally long, repeating cycles, numbered from 0 to 63 (figure 2). A communication cycle consists of static segment, dynamic segment, symbol window and dynamic slot idle phase. The static segment holds a configurable number of slots, arranged in a time division multiple access (TDMA) schema. Each FlexRay node is assigned to a number of slots for transmitting it's data, based on the real-time-enabled TDMA structure. The optional dynamic segment can be used to send low priority data or event triggered messages. The likewise optional symbol window can be used to send special symbols and the network idle time is used by the CCs for clock synchronization.

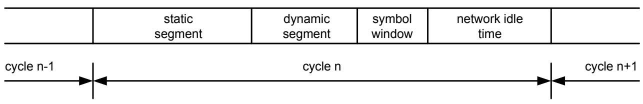

Fig. 2. Communication cycle and segments

3.1 Protocol Parameters

The configuration of the whole network is done by setting the specified FlexRay parameters [FlexRay Consortium (2005)]. There are two types of FlexRay parameters, global parameters are being set for the entire FlexRay Cluster and hence are identical for all nodes and identified with a "g" in the name, local Parameters are set for each node individually and usually vary for different controllers. They are identified using a "p". Parameters using the prefix "d" describe a timing duration. Constants, marked by a "c" are fixed by the protocol specification and can not be modified.

Most of the global parameters are used to directly configure the CC. A certain set forms the basis for calculating other local controller parameters. For example *gMacroPerCycle* is set globally and influences the calculation of *pMicroPerCycle*, which is set individually for each node.

3.2 FIBEX cluster description

The FIBEX Standard holds all automotive network configuration data in a XML structure file ASAM e.V. (2008). Besides FlexRay, also the description of other communication buses, for example controller area network (CAN) can be included. The file holds all information about network cluster, channels, electronic control units (ECUs), gateways, frames and signals. Topology depended data is available as well as communication related data like frames, timings, senders and receivers.

3.3 Controller Host Interface (CHI)

A FlexRay Controller internally holds a controller host interface which establishes the connection between external application controller and FlexRay controller. Therefore a CHI file is needed to configure the internal interface of a specific FlexRay controller. The interface file is dependent on the controller type and the model series. It must be generated using the global settings for the FlexRay cluster as well as the local parameters for the controller, like frames, assigned slots and message buffers.

4. FLEXRAY PARAMETER CONFIGURATION

Our methodology automatically determines a valid set of parameters for a given EEA. This architecture is given in form of an XML file including ECU specific data, signal transmissions between ECUs, cluster specific data like usage of channels and additional components like active star couplers, used for connecting ECUs in star topology. All this data is available in an EEA modeling tool like PREEvision aquintos GmbH (2009).

To calculate the whole set of FlexRay parameters, an initial set of values is necessary. This set is given by the information available in PREEvision. The determination of relevant data and the meta model for the data structure is part of our ongoing research and will be published in another paper. The XML file, which is based on the meta model structure holds all necessary data for the parameter calculations.

The next sections are organized as follows. First the dependencies of the parameter calculation and the starting set is described. In section 4.2 an example for the calculation of parameters is given. Section 4.3 describes the frame packing, followed by three methods for message scheduling in section 4.4. A short overview about the programmed PC application is finally given in 4.5.

4.1 Dependencies and calculation sequence

The whole set of parameters can be split into two groups. The first set consists of initial input values which have been imported from the EEA model. The second set contains all values which are calculated based on the first set. The dependencies between the parameters, define the calculation order shown in figure 3. Horizontal lines separate sets of parameters which can be calculated concurrently.

The uppermost level in the diagram specifies the set of initial values which are the starting point for the calculations. One of the first parameters is *gdSampleClockPeriod* and specifies the data rate of the bus system. The Parameter *gClusterDriftDamping* is used for the frequency correction inside the FlexRay Controller and heavily depends on the application. So it must already be set during the EEA