Getting FlexRay Under Control (Part 1) – Physical Layer Basics and Model Generation

This article describes a simulation-based methodology for FlexRay topology within the physical layer. Part 1 focuses on the basics of the FlexRay physical layer and signal integrity. The latter is essential for model generation and analysis of topology quality criteria. Part 2 describes the analysis methodology for efficient evaluation of virtual FlexRay vehicle topologies.

Thorsten Gerke and Thomas Suermann

Due to the rapid development and standardization of FlexRay, it has already been deployed in a variety of production vehicles. However, most automotive companies have not yet gained much practical experience with FlexRay, particularly with its physical layer. For this reason, development cycles are slow and partly based on trial and error. FlexRay and CAN physical layers may be similar from a theoretical point of view, but differ significantly in practical terms. The knowledge of the more familiar CAN physical layer cannot simply be applied to the deployment of FlexRay topologies. The FlexRay Consortium addresses this issue in part by providing development support through specification criteria, guidelines and application notes. However, it is the system developer’s responsibility to individually check each topology for proper functionality. Checking the FlexRay physical layer specification criteria requires analyzing the target topology, which is not available during the concept phase. As a result, system developers are forced to rely on their intuition and experience. Utilizing simulation-based design methodologies, however, allows developers to realize FlexRay’s deterministic concept and accurately predict the final results. A simulation-based methodology enables the evaluation of FlexRay criteria during the early phases of development using virtual topologies. This improved design process means significant value by decreasing overall design time.

Basics of FlexRay Networking Systems

FlexRay is a flexible networking solution that supports a high bandwidth and deterministic communication scheme. In addition, features such as a second communication channel support safety critical aspects of networking systems. An example of the FlexRay communication cycle structure with four segments is shown in figure 1:

- Static
- Dynamic
- Symbol
- Idle

The slot length within the static segment is fixed by configuration. Within the dynamic segment, the slot length may vary, depending on the required length of the message frames. Figure 2 shows the structure of FlexRay frames. Each message frame starts with a Transmission Start Sequence (TSS), which consists of a user-defined number of logical low bits (0). The TSS is used to inform the network about the start of communication to ensure that the gates of active stars are opened for communication. The actual user data, also known as payload, is then mapped to a configurable number of bytes within the message frame. Each byte starts with a so-called Byte Start Sequence (BSS). The trailing edge of the BSS is used for synchronizing the local clock of each FlexRay node with the global clock. Timing tolerances within the system are thereby balanced. FlexRay frames are terminated with a Frame End Sequence (FES). In the dynamic segment, the FES is followed by a Dynamic Trailing Sequence (DTS) that is used for
back-filling the time until the next minislot. Detailed information regarding the communication process can be found in [1].

**FlexRay Physical Layer**

For logical data exchange between FlexRay nodes, a physical layer is required in addition to the actual communication protocol, which transmits the physical signals. The Electrical Physical Layer Specification (EPLS) [2] of the FlexRay Consortium describes the hardware requirements and possible topology types. Because of FlexRay's flexible definitions for topology types, a variety of types are possible based on the OEM's networking philosophy:

- linear
- point-to-point
- passive star
- active star
- hybrid

The application notes included with the specification provide additional information about the use of filter elements, such as common-mode inductors and ESD protection elements. There are no guidelines for specific components, however, and the specification and application notes do not define how to assemble a robust working system. To begin with, there is no information about specific topology types and dedicated component mounting. However, the EPLS does define the required constraints for physical signal parameters. System developers must carefully check these constraints for each topology implementation. The most important evaluation criteria of the EPLS are subsequently explained.

**Propagation Delay**

Due to energy storage within the system, signal transmission speed is not unlimited. The bits generated by the transmitter need a certain amount of time until they arrive at the receiver nodes. This propagation delay is important, since it determines the level of synchronization precision between FlexRay nodes. Defined as the time difference between a trailing edge at the transmitter (TXD) and receiver (RXD), the propagation delay specification allows for a maximum value of 2500ns.

**Truncation of TSS**

The Transmission Start Sequence (TSS) will be truncated on its way to the receiver (see fig. 2), because active stars and transceivers need some time to detect bus activity. The receiver’s communication controller must be able to sample at least one low-bit in order to successfully detect a valid TSS. The specification allows for a TSS truncation range of 100-3500ns. Hence, system developers must ensure a sufficient TSS length.

**Asymmetrical Delay**

Every networking system must accommodate tolerances due to manufacturing or temperature variation. For example, the signal delays through FlexRay transceivers differ slightly for rising and falling edges. In addition, there are varying delay values on different paths through the transceiver (i.e. for TxD→Bus and Bus→Rxd). As a result, the bit lengths at the transmitter and the receiver may differ. A single bit may be extended or shortened, and possible asymmetries within the CC-BD interface must be taken into account too. The manipulation of bit length has to be limited. Otherwise, correct sampling of bit values by the FlexRay communication controller cannot be guaranteed [3].
Active-Idle Transition

The transition phase, Active-Idle, is a critical aspect of the physical layer, which is thoroughly addressed in the next release of the EPLS. The Active-Idle transition happens when the TxEn signal switches from low to high at the FlexRay transceiver. The transceiver’s driver unit switches from the low- to high-impedance state. This may cause unwanted oscillations of the bus signal within the complete network, requiring careful attention from the network developer. This process follows immediately after the FES or the DTS, when the communication controllers of the receiver nodes are monitoring the bus for a bundle of 11 successive high bits. The detection of this Channel Idle Recognition Point (CHIRP) prepares the bus for the following message. Because of the ringing after switching off the transmitter stage, additional low bits may be sampled until the transient oscillations are getting too small to cross the receiver thresholds, which further delays the Channel Idle Recognition Point (CHIRP). In the worst case, the CHIRP would move to the slot of the following message, which in turn would cause a so-called “slot boundary violation”. Future specifications are expected to consider a maximum value for the Idle Reaction Time.

Eye Diagram

In order to evaluate the topology’s analog behavior, as well as the discrete criteria just described, the eye diagram is a useful tool for communication system analysis and for investigating the FlexRay physical layer. When using the eye diagram for evaluation, a reference eye must be defined as an evaluation criterion. If the measured eye matches the reference eye, sufficient signal integrity has been achieved. On the other hand, violating the eye diagram does not necessarily mean the topology will not work in terms of signal integrity. Anomalies in the form of ringing effects, which may cause violations of the eye diagram, are usually eliminated by the FlexRay transceiver’s low-pass behavior. In addition to the conservative eye diagram approach, it is also advisable to evaluate the extent to which the eye diagram has been violated by precisely observing the RxD transceiver output signal.

Model Generation and Simulation of Physical Layer

Analyzing the signal integrity of a FlexRay topology requires investigating the previously described criteria for all network nodes. To determine optimization strategies, it is necessary to analyze the worst case scenario, as well as the nominal case. Yet in the early concept phase, as well as with an existing physical topology implementation, it would be unduly cumbersome to perform such tests and optimizations through in-lab measurements. Furthermore, specific worst-case scenarios cannot be performed with a real topology assembly. Instead, a model-based approach is proposed in order to meet the challenges of the physical layer. Implementing this approach requires two important capabilities: high-quality simulation models and methodologies that enable them to work with the virtual topology in an efficient and optimized way ([4], [6] and [8]). To highlight the modeling requirements, we again examine the FlexRay topology structure and the interfaces of the control units. From a system point of view, the topology type has to be considered first (e.g. whether it is an active star or a simple linear bus). The FlexRay cable, which has significant impact on the analog signal behavior, must be mapped to a corresponding transmission line model. From the control unit’s point of view, the physical bus interface is being taken into account, which consists of (see fig. 6):
- FlexRay transceiver (e.g. TJA1080A)
- common-mode inductor
- ESD protection
- termination (high- or low-impedance)

Additional components that are relevant for signal integrity should be considered as well. The various components and their abstraction level during model generation are explained here in more detail.

**FlexRay Transceiver and Active Star**

The FlexRay transceiver is an important component to consider when modeling the networking topology. To enable efficient system simulation, the transceiver’s functional behavior must be captured in a behavioral model that is an abstraction of the details around the semiconductor implementation. For signal integrity analysis, the following transceiver functions are important:

- behavior of transmitter and receiver (TxD, RxD)
- switching on/off the driver stage (TxEn)
- propagation delays
- threshold and delay asymmetries
- supply voltage (Vcc)

One of the most important aspects of transceiver modeling is the hysteresis of the receiver stage. It avoids permanent state switching during the ringing phase of the digital RxD signal. Due to the low-pass behavior of the receiver stage, high-frequency glitches are eliminated, increasing the stability of the RxD signal. NXP has integrated these functionalities when modeling the FlexRay transceiver family TJA108x [9]. In addition to modeling and simulating the nominal behavior, it is also important to consider worst-case scenarios due to device tolerances and temperature variations when validating a vehicle’s topology. Simulating under worst-case conditions allows one to easily detect weak points in the system [7]. Fig. 7 illustrates this relationship using the receiver stage as an example. The nominal threshold values are ±225mV and are exactly symmetrical to each other in the ideal case. Due to manufacturing tolerances and temperature variations, the threshold values are within a tolerance range that is subject to a statistical distribution. Also, in reality, the threshold values are not symmetrical. The difference between the absolute values can be as high as 10 percent. The transceiver’s filter behavior is affected significantly by these statistical tolerances, and the RxD signal may show different timing behavior compared to the nominal case even if the test patterns are identical. For signal integrity, the worst case uses maximum receive thresholds and minimum output amplitude (i.e. when the signal-to-noise ratio is as small as possible).

Fortunately, the tremendous effort around completely modeling the transceiver’s statistical behavior is unnecessary for system examination. From a system point of view, the so-called corner cases represent the worst-case scenario. NXP has modeled these corner cases – verifying them through comprehensive measurements with the real IC. NXP developed the transceiver’s simulation model in Saber using the IEEE standard VHDL-AMS and will provide this model to customers on request. For IP protection, the model has been encrypted according to the IEEE standard 1076. The active star device TJA1085 is in principle a combination of four single transceivers and an additional signal router. The router forwards the arriving signal from one network segment to the other enabled segments. The existing TJA1080A model can be used for modeling the internal transceiver blocks. The signal router is then modeled as a digital state machine within Saber’s State AMS tool, which automatically generates the IEEE compliant VHDL-AMS code.

![Fig. 7: Single Chip Transceiver and active Star](image-url)
The Common-Mode Inductor

There are several possibilities for common-mode inductor modeling. A common method is to model the inductor based on an RLC network. There are some issues with this approach, since the corresponding equivalent circuit diagram has to be determined and parameterized. Modeling the parasitic effects is particularly difficult, but as an alternative, data sheet information about the scattering parameters can be used directly. Such details are provided by the vendor and represent a tool-independent standard enabling simple model exchange between the device manufacturer and their customer. Representing the model, such scattering parameters can be read into Saber through an interface. The tool supports the user by providing an intuitive graphical process. As a result, there is no need to develop and parameterize an equivalent circuit diagram.

The Transmission Line

Modeling the cable as a transmission line is an issue of critical importance that justifies careful analysis of the relevant requirements. The cable model should allow the user to define a variable cable length. By adapting the cable length it is relatively simple to analyze different topology configurations with a single model. In addition, the cable model is essential for correctly describing physical behavior in ranges around hundred megahertz. Investigations performed by the FlexRay Consortium show that it is needed to account for the skin effect (i.e. the frequency dependent losses of the transmission line). Modeling a constant loss factor only would lead to inaccurate results, since the attenuation of harmonics would be too small and those of steady-state characteristics would be too high. As a solution to this problem, Saber offers an implementation of the W-element, which is used as a standard for transmission line simulation (the result of research activities at the University of California, Berkeley, the W-element has been continuously improved and refined over the past few years).

The simulation models described above, together with the FlexRay EPLS criteria, represent the starting point for developing FlexRay systems using virtual topologies. Part 2 of this article series will expand on this approach by detailing an analysis and evaluation methodology demonstrated with an actual automotive FlexRay system.

Literature:


The authors:

Thorsten Gerke studied Mechanical Engineering/Mechatronics at the University of Duisburg and started his career as support and applications engineer at Avanti Systems. In 2002 he joined Synopsys, where he was responsible for automotive solutions in the areas of applications and presales. Since 2006 he is leading the technical marketing for the Saber product family in Europe. He qualifies product requirements and works on future strategies for the Saber product portfolios. thorsten.gerke@synopsys.com

Thomas Suermann studied Physics at the University of Paderborn. He started his career as a systems engineer at Philips Semiconductors. At the Automotive Innovation Center in Hamburg he was responsible for product definition of CAN transceivers, before he took care of the successful implementation and start of production of the first FlexRay conform TJA1080A transceiver. During this time he was also working on the development of the FlexRay EPLS within the FlexRay consortium. Currently he is the main contact person in the area of system simulation of vehicle networks at NXP Semiconductors. thomas.suermann@nxp.com