

Synopsys' Saber simulator is a

Overview

Advances in automotive design have led to an exponential increase in vehicle electronics. Today's vehicle design improvements are due in large part to the application of electronics to automotive systems. Vehicle functions are divided into systems and sub-systems to provide for passenger entertainment, comfort, and safety, as well as to improve vehicle performance and enhance power-train control.

These systems must communicate with one another over a complex heterogeneous in-vehicle network (IVN). Each network typically contains multiple communication protocols including the industry standard Controller Area Network (CAN), Local Interconnect Network (LIN), and the recently developed FlexRay standard.

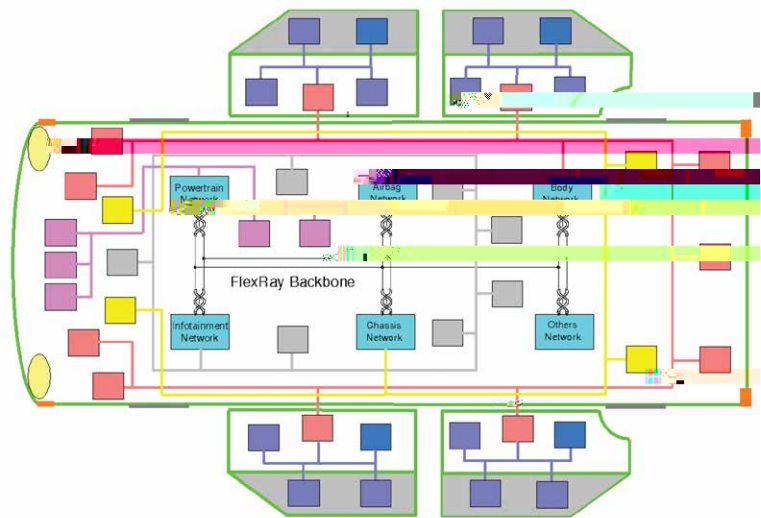


Figure 1: FlexRay communication configuration

Robust Design For Vehicle Power Networks

Controller Area Network

An event driven communication protocol used in applications such as engine management and body electronics. The maximum specified data rate is 1 Mbps, though the practical maximum is 500 Kbps. High-speed CAN is suitable for critical loads such as anti-lock braking systems and cruise control. Low-speed CAN is fault-tolerant and used for loads such as power seats and motorized windows.

Local Interconnect Network

A low-speed master-slave time triggered protocol meant to connect on-off type loads to higher speed networks. Typical loads include door locks, sun roofs, rain sensors, and powered mirrors. A LIN network is used as a low cost alternative if the full functionality of the CAN protocol is not required.

FlexRay

A fault-tolerant high-speed communication protocol targeted toward safety-related applications. The protocol can be operated in single or dual channel mode, where each channel has a maximum data rate of 10 Mbps. Using a dual-channel configuration, a FlexRay network can operate at speeds 20x faster than the maximum CAN bus data rate specification. Along with enabling safety-related applications, a FlexRay network is well suited as a communication backbone connecting heterogeneous networks together.

Physical Layer Design Challenge

Design teams face challenges implementing the electrical network's physical layer - the hardware implementation of the network's architecture, topology, and interconnects. The main objective in analyzing the physical layer is to identify and evaluate signal integrity issues for the network.

Physical layer verification requires that design teams check the transmit and receive waveforms against the system specification. Design teams must ensure the physical layer allows safe and secure transmission of the protocol data. Problems related to the physical layer will impact the entire communication system, slow down the network performance, or cause errors in the control system behavior. If these problems persist, network reliability will be compromised.

Robust Design

Improving IVN reliability requires a systematic development approach that ensures reliability issues are addressed as an integral part of the physical layer design process. Design teams use robust design methodologies to manage complex communication network issues, particularly when verifying the physical layer, taking into account system and environmental variations that affect performance.

Robust design is a proven development methodology that immunizes in-vehicle network performance against variations in system parameters and environmental conditions. The objective is to find the most cost-effective design solution that meets network performance and reliability specifications. Adopting a robust design methodology is essential for ensuring the reliability of the network. The objective is to find the most cost-effective design solution that meets network performance and reliability specifications. Adopting a robust design methodology is essential for ensuring the reliability of the network.

Robust Design for In-Vehicle Networks

- Verify network concepts and topologies early in the development cycle
- Analyze specific network variants (minimum vs. maximum number of ECUs)
- Analyze the impact of topology types and EMC protection elements on signal integrity
- Include wire characteristics in system simulations to analyze possible topology extensions
- Model and characterize in-vehicle network systems using industry standard MAST® and VHDL-AMS languages
- Verify nominal network performance using standard analyses and ensure reliability with advanced sensitivity, statistical, and fault analyses
- Choose from dozens of performance measurements to quickly analyze IVN simulation results
- Increase analysis throughput with distributed simulations across multiple CPUs

Virtual Prototyping

To verify if the vehicle network configuration satisfies system requirements, a prototype is often built to test the system's signal integrity. There are, however, serious limitations with the prototype build and test process. Since variations in manufacturing processes and environmental conditions can affect system performance, the system testing must account for both nominal and variant operation. With the traditional prototyping approach it is impossible to build enough prototypes to adequately test even the most important variations. Therefore, virtual prototyping with simulation is the best solution for verifying data network reliability.

Accurate Models

Beyond basic components such as filter capacitors and termination resistors, in-vehicle network simulation requires specialized models including transceivers, controllers, transmission lines, and chokes. Since the physical layer model will be used to analyze how signal integrity is affected by variations in system parameters, it is important to optimize component models for accuracy and simulation speed. The best approach is to select models that are based on a hardware description language which allows more efficient modeling of important functions. Saber® tools support the MAST and VHDL-AMS modeling language standards. The list of communication network models available for the Saber simulator includes:

- Transceivers: LIN, CAN, FlexRay
- CAN controller including bit timing
- Transmission lines for LIN, CAN, and FlexRay

This library is a set of detailed behavioral models that improve simulation performance without loss of accuracy by using physical effects specific to the network protocol. Important effects to model include signal propagation delay and the nonlinear and parasitic behavior of the drive stage. In addition to the set provided with the Saber environment, many network component manufacturers supply models written in MAST. The Saber solution also supports a suite of modeling tools for characterizing behavioral models, creating standalone models directly from equations, and translating SPICE netlists into Saber simulator equivalents.

