Performance Optimization of EE Architecture For Software Define Vehicles

[S.Venkatesh, P.Sai Chaithanya, G.Chethan Sree Royal

Supervisor: Dr. S Karthick

Abstract

This project focuses on optimizing the Electronic and Electrical (EE) architecture in Software Defined Vehicles (SDVs) by enhancing key communication protocols such as I2C, SPI, and CAN.

I2C and SPI are used for short-distance communication within systems, balancing speed, complexity, and power consumption, while the CAN protocol ensures real-time, reliable communication between critical vehicle subsystems. The goal is to identify performance and propose improvements to increase data transfer rates, reduce latency, and improve system reliability. The findings aim to support the scalability and efficiency of future SDVs by refining their communication infrastructure.

Background

I2C Protocol: I2C stands for Inter-Integrated Circuit. It uses only 2 bi-directional open-drain lines for data communication called SDA and SCL. Both these lines are pulled high. Serial Data (SDA) – Transfer of data takes place through this pin. Serial Clock (SCL) – It carries the clock signal. I2C operates in 2 modes.

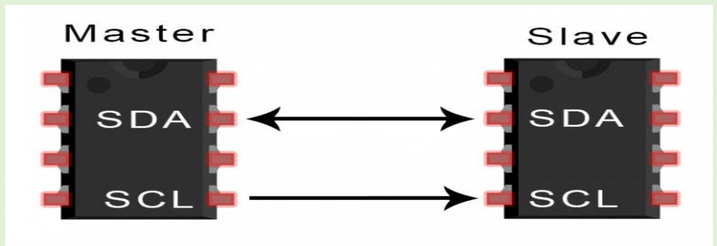
SPI Protocol: Devices communicating via SPI are in a master-slave relationship. The master is the controlling device, while the slave (usually a sensor, display, or memory chip) takes instruction from the master. The simplest configuration of SPI is a single master, single slave system, but one master can control more than one slave.

CAN Protocol:Controller Area Network, in short CAN protocol is a serial communication protocol which provides efficient support to mainly automotive real time control systems with a very high level of security, error detection and correction. CAN data frame can be defined as in the fig below. It consists of start of frame (SOF), arbitration field, control field, data field, CRC field, ACK field and end of frame(EOF).

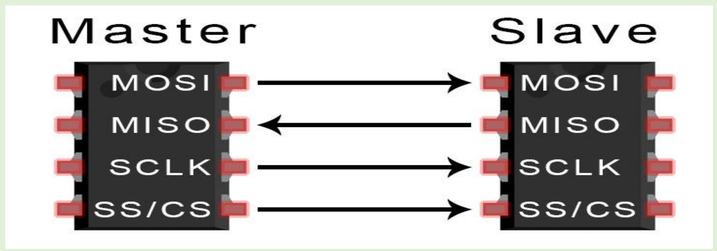
Methods

Design and Implementation: To design and implement the CAN, I2C, and SPI communication protocols in Verilog, ensuring each protocol meets industry standards for data transfer, signal integrity, and timing.

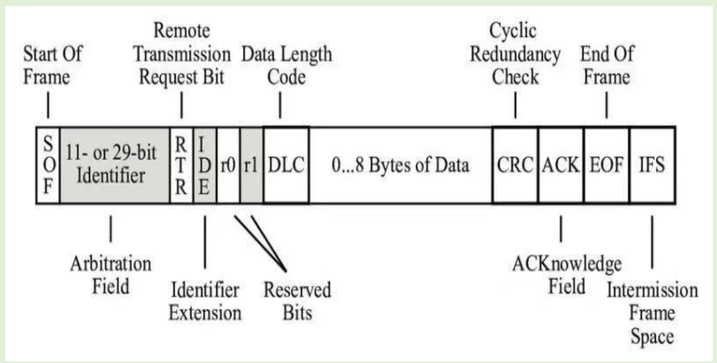
I2C Protocol:



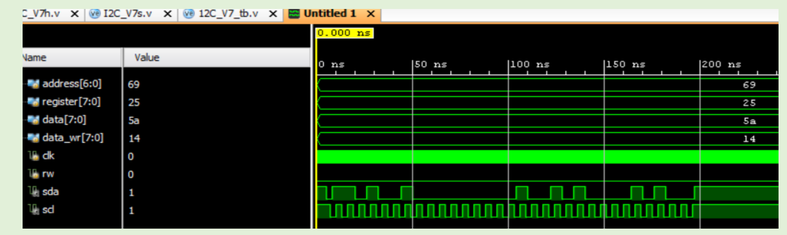
SPI Protocol:

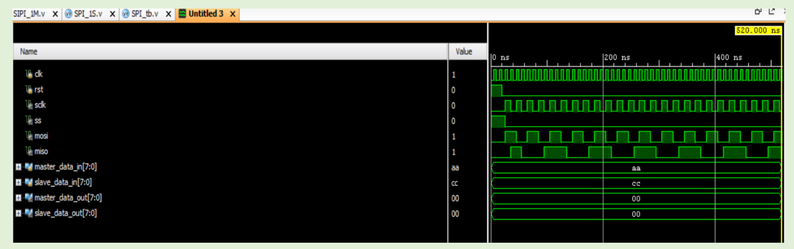


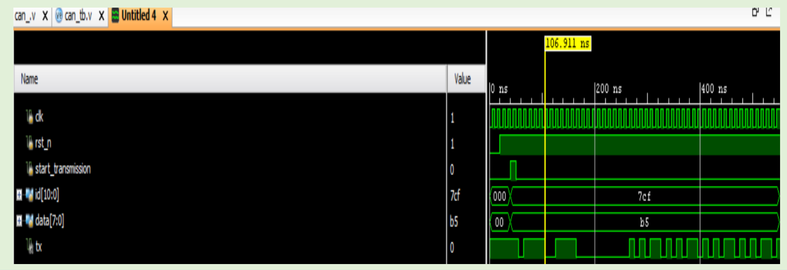
CAN Frame:



Results







Conclusion

The project successfully implemented and simulated the CAN, I2C, and SPI communication protocols in Verilog, showcasing a understanding of embedded systems. Each protocol was designed and rigorously tested to meet industry standards, focusing on signal integrity, timing accuracy, and reliable data transfer. Extensive simulations and waveform analysis were conducted to verify the real-time performance and practical applicability of the protocols.

Future Perspectives

- **Hardware Implementation**: The next phase of this project involves implementing the CAN, I2C, and SPI protocols on a Xilinx FPGA development kit. This will allow real-world testing and validation beyond simulations.
- **Real-Time Testing:.** By deploying the protocols on the Xilinx kit, we can evaluate their real-time performance in handling data transfers between connected devices such as sensors, memory units, and controllers.

Impact on Society

The project aims to accurately model of the protocols, ensuring they meet industry standards for performance and functionality. Through simulation and waveform analysis, the protocols will be tested for their real-time capabilities and error-handling mechanisms. The ultimate goal is to create a foundation for future hardware implementation on FPGA platforms, where these protocols can be tested and optimized in real-world scenarios.