Design and Verification of a Low-Power, Noise-Robust SPI Communication Interface Using Verilog and eSim

FOSSEE eSim Semester Long Internship — Autumn 2026
Yamini Pasala
Department of Electronics and Communication Engineering
Rajiv Gandhi University of Knowledge and Technologies, Nuzvid

1. Abstract

This report presents the design, RTL implementation and mixed-signal verification of a low-power, noise-robust Serial Peripheral Interface (SPI) communication block, developed entirely in Verilog HDL and co-simulated in eSim. The design implements a single-master, single-slave, full-duplex SPI link operating in SPI Mode 0 (CPOL = 0, CPHA = 0, MSB-first) with 8-bit data transfer. The complete functionality — master control FSM, transmit/receive shift registers, bit counter, chip-select control, a controlled digital bit-error injection path for robustness testing, an error-detection block, and an RTL-level switching-activity (low-power) control block — is integrated into a single top-level module, spi_communication. The design was compiled and simulated inside eSim using ADC/DAC bridge components to interface the digital Verilog core with the Ngspice analog simulation engine, and its correct full-duplex, mode-0 behaviour was confirmed through waveform-level verification of the SPI bus signals (SCLK, MOSI, MISO, CS) as well as the internal master and slave shift-register contents.

2. Introduction

The Serial Peripheral Interface (SPI) is a widely used synchronous, full-duplex serial communication protocol for short-distance, chip-to-chip data exchange, valued for its simplicity and high throughput compared to protocols such as I2C or UART. A conventional SPI implementation, however, does little to address two practical concerns that matter in real deployments: susceptibility of the serial data path to bit-level noise/errors, and unnecessary switching activity (and hence dynamic power) when the bus is idle.

This project targets both concerns while keeping the design fully synthesizable and simulation-verifiable in an open-source flow. The RTL adds (i) a controlled, RTL-level digital bit-error injection path on the MOSI line together with a dedicated error-detection block that flags mismatches between transmitted and received data, and (ii) an activity-control block that keeps the SPI logic idle — reducing RTL-level switching activity — whenever no transfer is in progress. The complete design is captured in Verilog HDL and verified functionally with a self-checking testbench, and separately co-simulated inside eSim 2.x using the digital-to-analog (DAC) and analog-to-digital (ADC) bridge components that let a compiled Verilog block participate in an Ngspice transient simulation.

3. SPI Configuration

The communication interface is configured as follows:

- Protocol: Serial Peripheral Interface (SPI), single master – single slave
- Data width: 8 bits per transfer, MSB first
- Duplexing: Full-duplex — simultaneous Master→Slave (MOSI) and Slave→Master (MISO) transfer
- SPI mode: Mode 0 (CPOL = 0, CPHA = 0) — SCLK idle LOW, data sampled on the rising edge and updated on the falling edge
- Bus signals: SCLK (serial clock, master-driven), CS (chip select, master-driven), MOSI, MISO

4. RTL Architecture

The design is implemented as a single top-level Verilog module, spi_communication, internally partitioned into the following functional blocks:

- System Clock / SPI Clock Divider — generates the internal SPI timing reference
- SPI Master FSM / Control Logic — sequences the transfer and drives the datapath and CS/SCLK generation
- Master TX Shift Register and Master RX Shift Register — 8-bit serializer/deserializer pair on the master side
- SPI Bit Counter — tracks bit position within the current 8-bit frame
- Chip Select Control — generates the CS signal that frames each transaction
- Slave RX Shift Register and Slave TX Shift Register — 8-bit serializer/deserializer pair on the slave side
- Controlled Digital Bit-Error Injection — an RTL-level fault-injection stage on the MOSI path, driven by Noise Enable and Error Mask
- Error Detection — compares Master RX Data against Slave TX Data and Slave RX Data against Master TX Data, producing an Error Detected flag
- Low-Power Activity Control — RTL-level switching-activity reduction; the SPI logic is active only while a transfer is in progress and idle otherwise
- Verification / Testbench — a self-checking environment exercising data integrity, timing and error-injection scenarios

4.1 Circuit Block Diagram

Figure 1 shows the block-level architecture of the design. The master-side datapath (left, blue) drives MOSI through the noise-robustness path (red) into the slave (green); the slave returns data on MISO back to the Master RX Shift Register, giving genuine full-duplex operation. SCLK and CS are generated by the master and drive the slave in parallel with the data lines. The error-detection block (right) and the low-power activity-control and verification blocks (bottom) surround the core datapath, and the eSim ADC/DAC bridge boundary used for mixed-signal co-simulation is shown along the bottom of the figure.

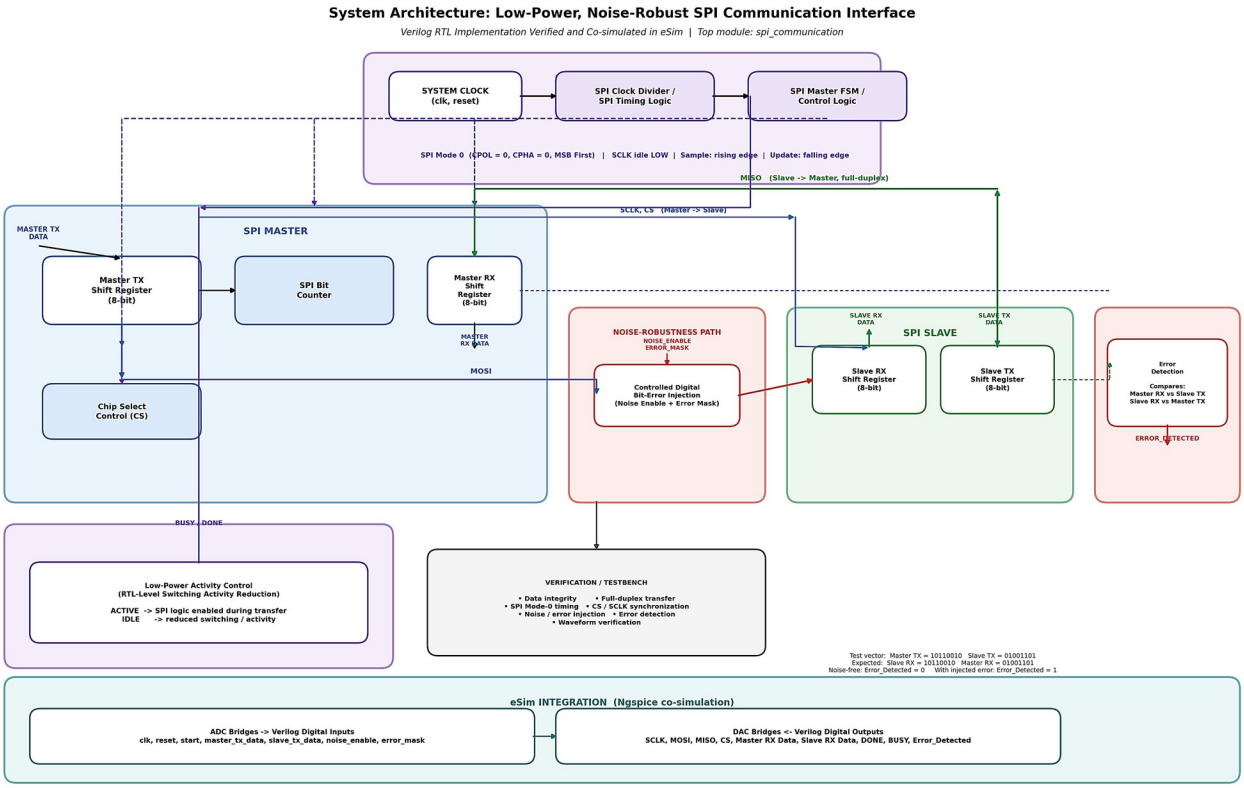

Figure 1: Block-level architecture of the proposed low-power, noise-robust SPI communication interface implemented using Verilog HDL and integrated with eSim for circuit-level simulation and verification.

5. Circuit Schematic in eSim

The compiled Verilog top module (spi_communication) was placed as a digital block inside the eSim schematic editor and connected to the Ngspice analog domain through ADC and DAC bridge components. Digital stimulus for clk, reset, start and the parallel input data/control buses (master_tx_data, slave_tx_data, noise_enable, error_mask) is applied through ADC bridges fed by pulse and constant digital vector sources; the digital outputs of the design — SCLK, MOSI, MISO, CS, the master/slave RX data buses, BUSY, DONE and Error Detected — are converted back through DAC bridges (each output bit terminated into a 1 kΩ load resistor to ground) so that the transient behaviour can be observed as analog waveforms in Ngspice. Figure 2 shows the complete eSim schematic used for this co-simulation.

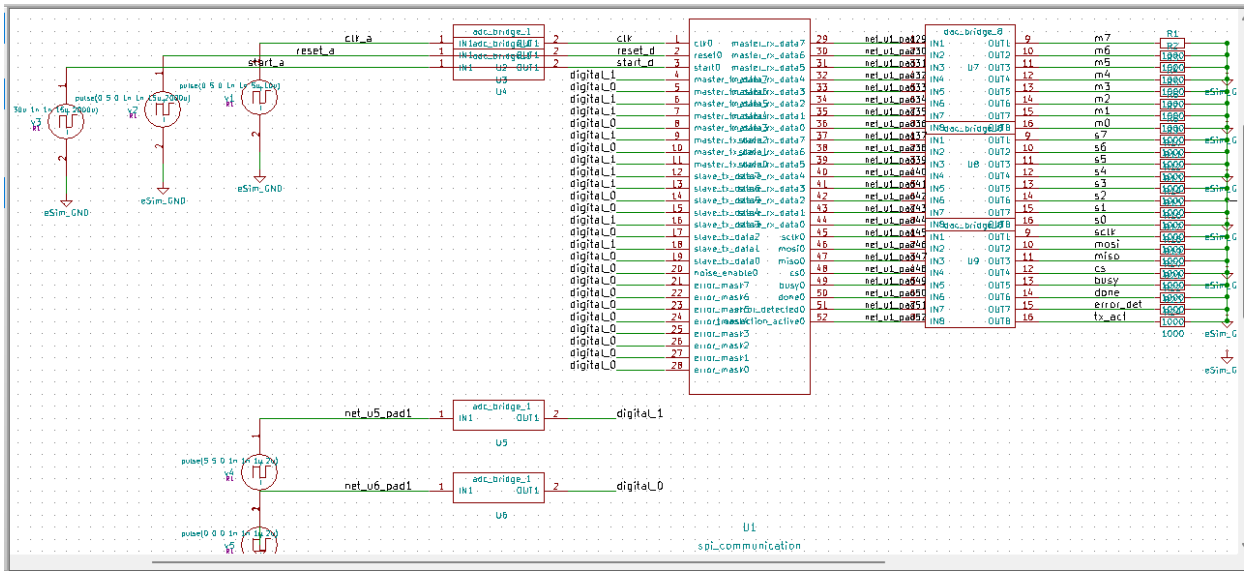

Figure 2: eSim schematic of the spi_communication design showing ADC/DAC bridge interfacing to the Ngspice simulation environment.

5.1 Schematic Components

Component	Role in the schematic
spi_communication (Ngveri/digital block)	Compiled Verilog top module — the SPI master/slave RTL core
adc_bridge_1	Converts pulse / constant digital vector sources into Verilog digital inputs (clk, reset, start, data and control buses)
dac_bridge_8	Converts 8-bit groups of Verilog digital outputs into analog nodes for Ngspice observation
pulse (eSim sources)	Digital stimulus sources for clock, reset and start signals
Resistor (1 kΩ)	Load resistors terminating each DAC bridge output to form valid analog nets

6. Top-Module Interface

The table below summarises the top-level port list of spi_communication as exercised in the eSim testbench. (The full Verilog source is maintained separately in the project repository / Ngveri source file and is not reproduced here.)

Signal	Direction	Description
clk	Input	System clock
reset	Input	Synchronous/asynchronous system reset
start	Input	Initiates an SPI transaction
master_tx_data [7:0]	Input	Parallel data loaded into the Master TX Shift Register
slave_tx_data [7:0]	Input	Parallel data loaded into the Slave TX Shift Register
noise_enable	Input	Enables the controlled digital bit-error injection path
error_mask [7:0]	Input	Bit mask applied by the error-injection block when noise_enable is asserted
sclk	Output	SPI serial clock (Mode 0 — idle LOW)
mosi	Output	Master-out / Slave-in serial data
miso	Output	Master-in / Slave-out serial data
cs	Output	Active chip-select / slave-select
master_rx_data [7:0]	Output	Data received by the Master RX Shift Register
slave_rx_data [7:0]	Output	Data received by the Slave RX Shift Register
busy	Output	Asserted while a transaction is in progress
done	Output	Pulses when a transaction completes
error_detected	Output	Asserted when a mismatch is found by the Error Detection block

7. Simulation Methodology

The design was verified in two complementary ways. First, a Verilog testbench exercises the RTL directly (data integrity across repeated transfers, SPI Mode-0 timing, full-duplex transfer, CS/SCLK synchronisation, and both noise-free and error-injected transactions). Second, the same compiled module was co-simulated inside eSim: a transient (.tran) analysis was run in Ngspice with the digital stimulus described in Section 5, and the resulting bus and internal-register waveforms were captured over the simulation window for visual verification.

- Primary bus signals observed: SCLK, MOSI, MISO, CS
- Master-side internal state observed: Master TX/RX shift-register bits m0–m7
- Slave-side internal state observed: Slave TX/RX shift-register bits s0–s7
- Status/handshake signals observed: BUSY, DONE, TX_ACTIVE (low-power activity flag), ERROR_DETECTED

7.1 Representative Test Case

A representative transfer used for functional verification is summarised below:

- Master TX = 10110010, Slave TX = 01001101
- Expected result: Slave RX = 10110010, Master RX = 01001101
- Noise-free transfer: Error Detected = 0
- With controlled bit-error injection active: Error Detected = 1

8. Simulation Results — eSim Waveforms

Figures 3–6 show the Ngspice transient-analysis waveforms captured from the eSim co-simulation, covering the SPI bus signals, the master- and slave-side shift-register bits, and the status/error signals. Each figure is a combined waveform plot — the related signals for that group are overlaid on a single time axis so their relative timing and transitions can be compared directly.

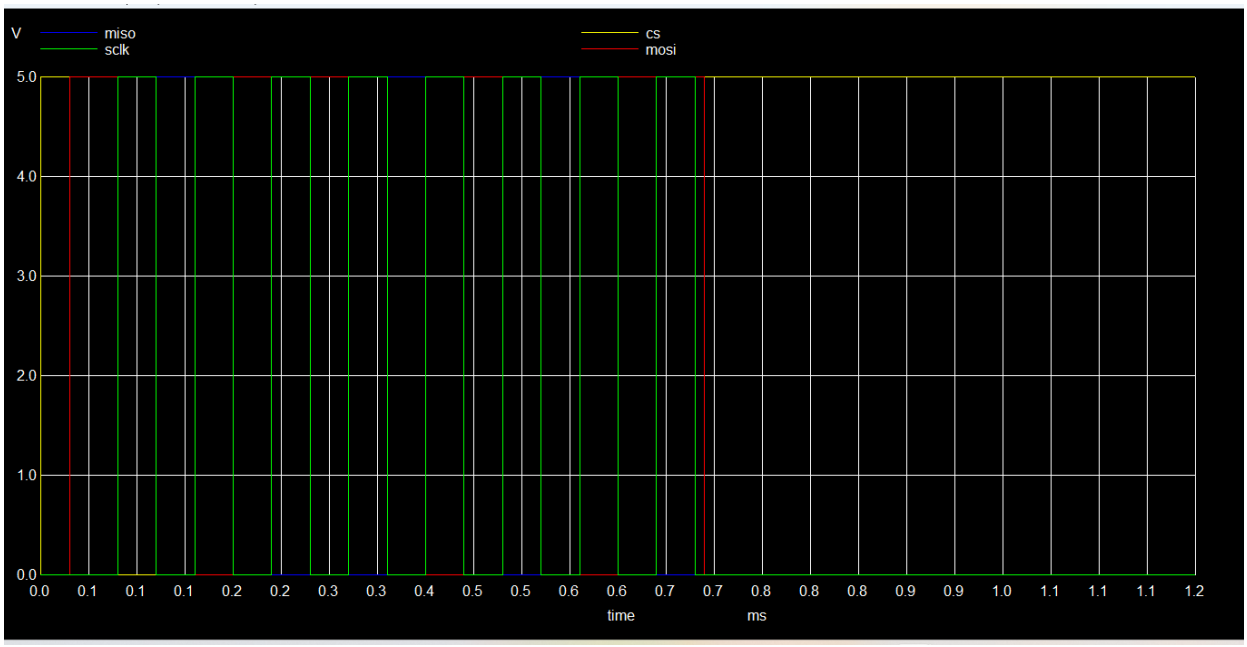

Figure 3: SPI bus signals — MISO, SCLK, CS and MOSI over the simulated transaction window.

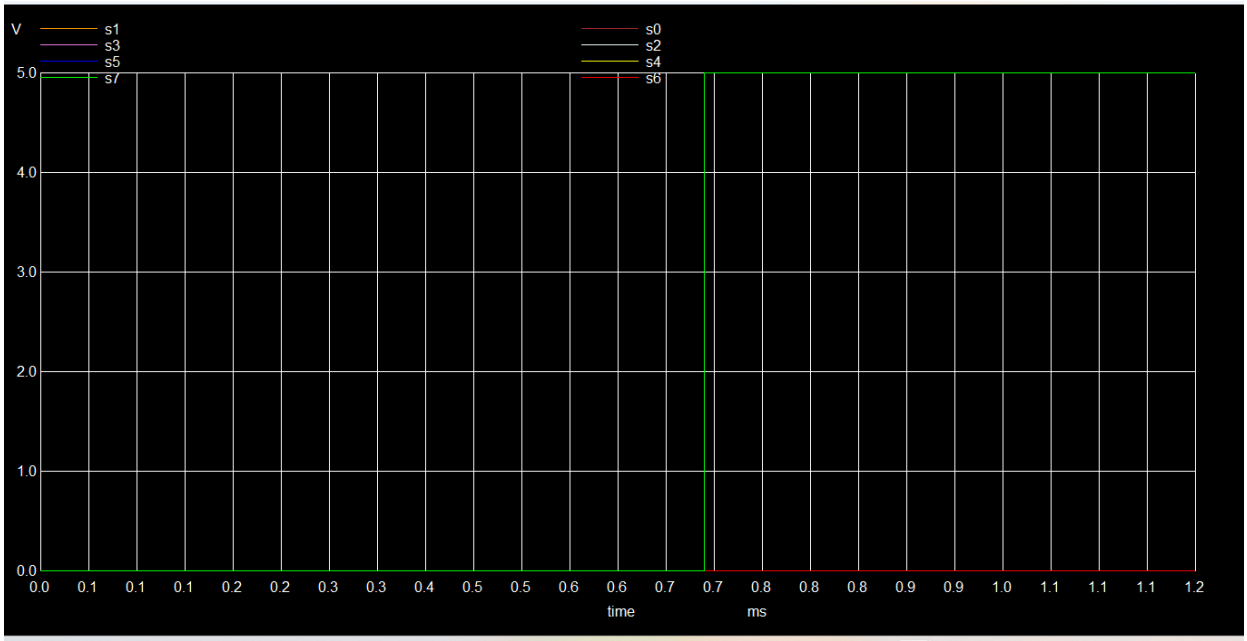

Figure 4: Slave shift-register bits s0–s7.

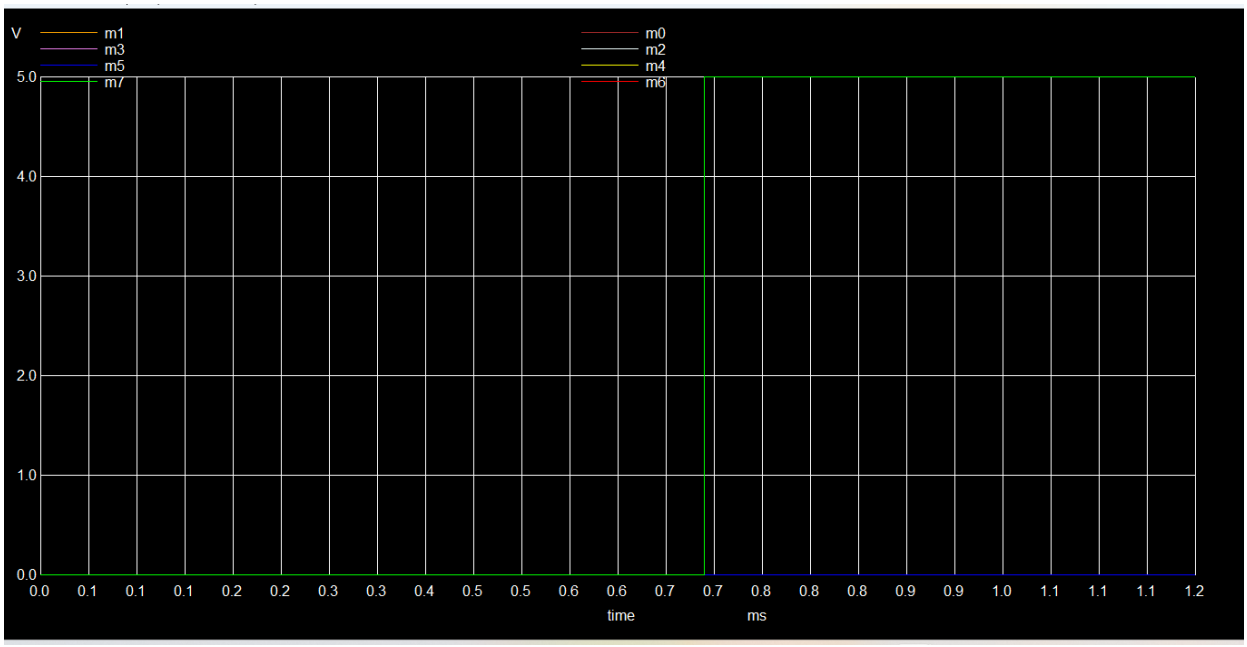

Figure 5: Master shift-register bits m0–m7.

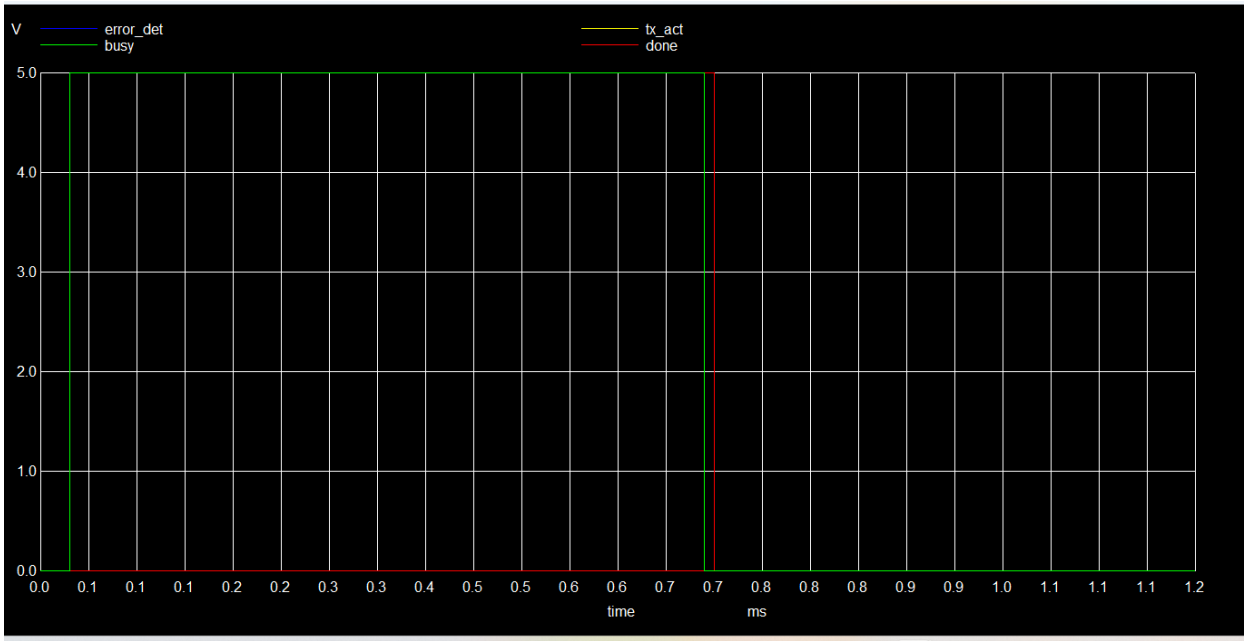

Figure 6: Status and error-handling signals — error_det, busy, tx_act (low-power activity) and done.

8.1 Observations

- The SPI clock (SCLK) and chip-select (CS) are generated correctly by the master and frame the serial data exchange on MOSI and MISO, consistent with Mode-0 (CPOL = 0, CPHA = 0) operation.
- The master and slave shift-register bit waveforms settle to the expected steady bit patterns once the active transaction completes, confirming correct parallel-to-serial and serial-to-parallel operation on both sides of the link.
- The busy/tx_act signal is asserted only for the duration of the active transfer and returns to its idle state afterward, consistent with the intended RTL-level switching-activity reduction during IDLE periods.
- The done and error_det status outputs transition at the end of the transaction window, indicating completion and the noise/error-detection outcome for that transfer.

9. Conclusion

A single-master, single-slave, full-duplex SPI communication interface was designed in Verilog HDL with built-in noise-robustness (controlled digital bit-error injection and error detection) and an RTL-level low-power activity-control mechanism, all integrated within one top-level module, spi_communication. The design was verified functionally through a dedicated testbench and additionally co-simulated in eSim using ADC/DAC bridge components to interface the digital core with the Ngspice transient-simulation environment. The captured waveforms confirm correct SPI Mode-0, full-duplex behaviour and correct operation of the error-injection and low-power activity paths, meeting the objectives of the FOSSEE eSim Autumn 2026 internship project.

10. References

- FOSSEE Team, IIT Bombay, eSim User Manual. [Online]. Available: https://esim.fossee.in
- Ngspice Reference Manual. [Online]. Available: http://ngspice.sourceforge.net/docs.html
- Motorola/Freescale, SPI Block Guide — Serial Peripheral Interface protocol reference.

- G. H. Makar, F. J. Badenas, R. G. Simone, A. Furfaro, K. S. Stevens, and R. Suaya, "Low Power SPI Design Based on Relative Timing Techniques," in Proc. 2019 26th IEEE Int. Conf. on Electronics, Circuits and Systems (ICECS), Genoa, Italy, 2019, pp. 166–169. [Online]. Available: https://ieeexplore.ieee.org/document/8965199