100 RTL Design and Verification Projects

A Curated Project Book for RTL/Verification Entry

Prepared by **Kittu K Patel**
VeriCore Initiative

Empowering Students Freshers into VLSI Design

May 18, 2025

Contents

Introduction

The world of VLSI design demands a strong grip on both **RTL Design** and **Verification**. To gain expertise, real-world project-based learning is the most effective path. This document provides 100 hands-on projects to help you build a strong foundation in digital design using Verilog/SystemVerilog, testbenches, and assertions.

Pro Tip

Start from beginner-level FSMs, then progress to memory controllers, protocol implementations, and bus verifications. Try to simulate every design with timing and waveform analysis.

1 Project List – RTL Design & Verification

Legend

Domain: [Design / Verification / Both]
Complexity: [Beginner / Intermediate / Advanced]

No.	Project Title	Description
1	4-bit Ripple Carry Adder	RTL design of a basic ripple carry adder using structural modeling. Verify using testbench and waveform analysis.
2	4-bit Carry Lookahead Adder	Implements fast addition using carry lookahead logic. Verification involves random testcases and delay checks.
3	8-bit Up/Down Counter	Synchronous counter with control signal to toggle up/down. Verify with directed stimulus.
4	Traffic Light Controller	FSM-based design for 4-way traffic signals. Verification includes timing sequence validation.
5	Sequence Detector (1011)	FSM design to detect binary sequence. Includes both Mealy and Moore implementation.
6	UART Protocol (Tx + Rx)	Universal Asynchronous Receiver Transmitter core. Includes frame structure, start/stop bits.
7	SPI Master-Slave	Serial Peripheral Interface between master and slave. Verification includes burst mode testing.
8	I2C Controller	Implements start, stop, ACK, and data transfer conditions. Use self-checking testbench.
9	ALU with 8 Operations	Arithmetic and logic operations controlled by opcode. Add assertion-based checks.
10	Memory Controller	Read/Write logic, address decoder and enable signals. Test memory access latency and correctness.

Pro Tip

While working on protocol projects like UART, SPI, or AXI, always create separate modules for transmitter, receiver, and control. This helps in modular testbench development too.

No.	Project Title	Description
11	AXI4-Lite Master Interface	RTL design of AXI4-lite master with address, write, and read channels. Verification includes burst and handshake testing.
12	AXI4-Slave Protocol Checker	SV-based monitor and scoreboard to verify AXI4-slave response under back-pressure.
13	APB to AXI Bridge	Design protocol converter between APB and AXI. Validate timing translation with assertions.
14	Round Robin Arbiter	Arbitration logic for 4 masters. Verify fair grant distribution using randomized testcases.
15	LFSR-based Random Number Generator	Linear Feedback Shift Register for pseudo-random bit generation. Verify period and seed behavior.
16	Priority Encoder (8:3)	RTL design with enable and valid signals. Write testbench with edge case priority tests.
17	Wallace Tree Multiplier	High-speed multiplier using tree structure. Simulation and gate-delay analysis recommended.
18	Pipelined Multiplier	3-stage pipelined architecture with stall and flush conditions. Verify latency vs throughput.
19	Booth Multiplier	Signed multiplier using Booth's algorithm. Functional correctness via scoreboard.
20	ALU with Flag Outputs	ALU design with zero, carry, negative, and overflow flags. Assertion checks for edge cases.
21	DMA Controller (2 channels)	Direct Memory Access controller with source and destination handshaking. Use FSM-based design.
22	AXI Write Channel UVM Sequence	Create a reusable write sequence with constraints and callbacks for AXI verification.

No.	Project Title	Description
23	SPI UVM Testbench	Develop agent, monitor, and scoreboard for SPI. Include reset, corner, and burst scenarios.
24	FSM Lock-Unlock Mechanism	Secure lock mechanism using FSM. Include test scenarios for wrong inputs and resets.
25	Parity Generator-Checker	Generate and validate even/odd parity bits. Use assertions to catch errors.
26	UART Frame Checker	Create checker that validates start/stop, data and parity bits using assertions.
27	Gray Code Counter	Design and verify binary to Gray and Gray to binary logic. Focus on glitch-free transitions.
28	Clock Divider	Divide input clock by N. Testbench with pulse counting and frequency verification.
29	Watchdog Timer	Reset system if no activity in defined cycles. Use assertions to check timeouts.
30	Configurable FIFO (Sync)	RTL of synchronous FIFO with parameterizable width and depth. Verify overflow and underflow conditions.

Pro Tip

Always parameterize your designs like counters, FIFOs, and ALUs. It allows you to write scalable verification environments and reuse testbenches effectively.

No.	Project Title	Description
51	Configurable Counter with Load and Hold	RTL of a universal counter with programmable load, hold, reset. Verify using corner case tests.
52	Traffic Light Controller with Pedestrian Override	FSM-based system supporting auto/manual pedestrian signals. Add reset and flash modes.
53	UART VIP with Protocol Scoreboard	Create reusable VIP for UART. Include golden model and scoreboard comparisons for verification.
54	Time-Multiplexed Display Driver	RTL design for 7-segment LED driver using time-division multiplexing. Verify using waveform checking.

No.	Project Title	Description
55	Coverage-Driven Verification Plan for SPI	Write a coverage plan and implement coverage bins for SPI protocol items.
56	UVM Factory Override Demonstration	Show base-to-derived class override in a testbench using UVM factory methods.
57	Reset Synchronizer Design	RTL for synchronizing async reset into the system clock domain. Verify using metastability simulations.
58	Formal Property Verification for FSM	Write SVA to check illegal transitions in FSM design using cover and assume-assert properties.
59	UVM RAL Model for Control Register Set	Build Register Abstraction Layer (RAL) model for DUT with read/write mirror checking.
60	AMBA AHB-Lite Verification IP	Develop lightweight VIP for AHB-Lite including assertion-based protocol checker.
61	SystemVerilog Queue-Based FIFO Monitor	Use dynamic queues to track push/pop behavior and validate ordering.
62	OTP Generation Logic	RTL for generating and validating One-Time Passwords. Use SVA to check uniqueness and validity.
63	Handshake Protocol Checker (Ready/Valid)	Create assertions to check timing and dependency of ready-valid handshake-based design.
64	Clock Gating Cell Verification	Verify functionality of RTL clock gating circuit using coverage and assertions.
65	Bus Arbiter Coverage Plan	Build functional coverage bins to track access patterns and fairness in bus arbitration logic.
66	Sequence Layered Protocol Generator	UVM sequences layered with headers, payloads, CRC fields. Include constrained random generation.
67	Interrupt Controller (RTL + Testbench)	Design and verify priority-based interrupt handler with masking, vector table.
68	Bus Matrix Switch (4x4)	Crossbar switch design for routing inputs to outputs. Validate conflict and simultaneous access.
69	BCD to Binary Converter RTL	RTL of BCD to Binary converter. Include input filtering and verification through assertions.

No.	Project Title	Description
70	Dual Clock FIFO	Design FIFO operating across two asynchronous clock domains. Verify using testbench and metastability checks.

Pro Tip

To stand out, show at least one project where you've applied SystemVerilog Assertions (SVA) or written a complete UVM agent. These are recruiter favorites!

No.	Project Title	Description
71	Low Power Finite State Machine	Design FSM with clock gating and low power techniques. Validate with SDF-aware simulation.
72	Watchdog Timer with Windowing Feature	RTL of watchdog with reset, timeout, and configurable windowing. Include coverage and negative tests.
73	Reusable Delay Line Module	Build a parameterized delay element useful in SoC interfacing or bit alignment.
74	Latch and Flop Identification Checker	Verify latch vs flop behavior using SVA to flag latches in unintended logic.
75	Configurable Priority Encoder	RTL module with variable input width and one-hot encoded output.
76	Arbiter Using Round Robin + Priority Modes	Dual-mode arbiter RTL with configurable arbitration policy. Include UVM coverage for fairness.
77	Clock Divider by N RTL + Checker	Parameterized divider with clean duty cycle output. Add assertion-based glitch detection.
78	Basic TLM Interface Modeling	Implement TLM abstraction for memory transaction verification in UVM.
79	Memory BIST Controller (Simplified)	RTL of Built-In Self Test controller with memory access patterns. Include testbench for stuck-at testing.
80	Static Hazard Detector for Logic Circuit	RTL module that identifies hazards. Verify correctness using waveform snapshots.
81	Multicycle Path Annotator (Timing RTL)	Annotate timing-critical path RTL. Run testbench and report slacks using delays.

No.	Project Title	Description
82	FSM with Illegal State Detector	Include illegal state detector logic inside FSM. Verify entry into valid states only.
83	Reset Generator Logic (POR)	Power-On-Reset generation circuit with delay counter. Simulate power glitch scenarios.
84	Power-Aware Verification Example	Simulate power shut-down of block and assert inactive interface using SVA.
85	Protocol Monitor with Error Injection	Create UVM monitor for packet protocol and inject CRC/sequence errors to check resilience.
86	Dual Clock Synchronizer Checker	Assertions to validate safe crossing between async clock domains using synchronizer stages.
87	SVA for Bus Deadlock Detection	Use SVA to monitor for deadlock on shared bus (e.g., no grant for extended cycles).
88	Lint + CDC Report Analyzer Script	Build a Python + SV interface to parse CDC and lint reports and generate dashboards.
89	Universal Pattern Generator for Testbench	Build parameterized stimulus generator (PRBS, toggle, ramp) for reuse in testbenches.
90	Bus Functional Model (BFM) for APB	Model BFM interface to drive and respond to APB transactions.
91	Interrupt Latency Checker	Build RTL logic and assertions to measure cycles from interrupt request to acknowledgment.
92	Built-in Loopback Test Logic	Add loopback logic to DUT and self-check using testbench or formal.
93	Memory Map Decoder RTL + Test	Address decoder logic with support for block enable, error region, and mirroring.
94	Spec-to-Code Flow Verification	Build project showing complete flow from spec to RTL to testbench to coverage to report.
95	Watchpoint Generator in UVM Monitor	Extend monitor to trigger flags when specific sequence or data seen. Add callback or log.
96	CRC Generator RTL + Polynomial Config	Configurable CRC RTL supporting various standard polynomials. Include testbench with golden CRC check.
97	Command/Response Packet Tracker	Track packet sequences in UVM monitor and assert correct response timing.

No.	Project Title	Description
98	Parametrized N-bit Adder/Subtractor	RTL supporting both addition/subtraction based on select line. Verify overflow, underflow.
99	Multi-Threaded Testbench Framework	Use fork-join, mailboxes to create multi-threaded testbench for load testing.
100	Final Project: Mini UVM SoC Verification Environment	Integrate mini SoC (UART + Memory + Timer). Build complete UVM env with scoreboard, coverage, assertions.

Pro Tip

Final Tip: Group your top 5–10 projects into a portfolio PDF and publish it on LinkedIn or your GitHub. Use QR codes to link to code repositories or waveforms.

Thank You for Reading!

Vision Ahead

This document is not just a list — it is your launchpad into the world of **digital design and functional verification**. Whether you are a student, fresher, or professional, mastering even 50 of these projects will sharpen your RTL and UVM skills like never before.

Words from Kittu K Patel

I hope this curated roadmap fuels your learning journey and inspires you to build more. Remember: *Every great chip starts with a small line of RTL code.*
For more guidance, connect with me on LinkedIn or explore content on my technical platform. Keep learning, keep verifying!