AXI4 VIP Architecture Document

Version 2.8 Update - September 2025

Version 2.8 - Major Architecture Updates

• Test Infrastructure Enhancements:
- Fixed UVM factory registration for 6 critical tests
- Corrected base class inheritance for error injection tests
- Resolved sequencer coordination issues preventing test hangs

• Architectural Improvements:
- Enhanced error injection framework with slave-side X-injection
- Improved reset test architecture with assertion-based verification
- Strengthened exception handling mechanisms

• Regression Infrastructure:
- Fixed quote preservation in regression scripts
- Enhanced command_add parameter handling
- Improved LSF job submission reliability

• Test Coverage Expansion:
- Added 6 new tests (4 slave injection, 2 exception tests)
- Fixed 14 previously failing tests
- Achieved 100% pass rate for all 196 tests

• Bus Matrix Support:
- Verified all fixes across NONE, BASE (4x4), and ENHANCED (10x10) modes
- Ensured consistent behavior across all configurations
- Added bus mode override support for X-injection tests

Key Technical Fixes

AXI4 Verification IP Architecture Document

Version 2.6

August 2025

Production Release

1. Executive Summary

The AXI4 Verification IP (VIP) is a comprehensive SystemVerilog/UVM-based verification solution for ARM AMBA AXI4 protocol. This document describes the architecture, design decisions, and implementation details of the VIP.

1.1 Key Capabilities

- Full AXI4 protocol compliance (IHI0022D specification)
- Scalable bus matrix support (4x4 to 64x64 and beyond)
- Enhanced 10x10 configuration with QoS and USER signal support
- 100% regression pass rate with 140+ test cases
- Production-tested in multiple SoC verification environments

2. Architecture Overview

2.1 High-Level Architecture

The VIP follows a layered architecture:

1. Test Layer: Contains test cases, virtual sequences, and configuration
2. Environment Layer: Integrates scoreboard, coverage, and bus matrix reference model
3. Agent Layer: Master and slave agents with drivers, monitors, and sequencers
4. BFM Layer: Bus Functional Models implementing AXI4 protocol
5. Interface Layer: DUT interface with AXI4 protocol signals

2.2 Component Hierarchy

- Test Layer: All test cases and virtual sequences
- Environment: Agents, scoreboard, and reference model integration
- Master Agent: Driver BFM, monitor, sequencer, sequences
- Slave Agent: Driver BFM, monitor, sequencer, memory model
- Coverage: Functional and code coverage collectors

3. Scalable Bus Matrix Architecture

3.1 Dynamic Configuration System

The VIP uses compile-time configuration for seamless scaling:

- NUM_MASTERS: Configurable from 1 to 64
- NUM_SLAVES: Configurable from 1 to 64
- AXI_ADDR_WIDTH: 32 to 64 bits
- AXI_DATA_WIDTH: 32 to 512 bits
- AXI_ID_WIDTH: 1 to 16 bits

3.2 Bus Matrix Modes

1. NONE Mode: Direct 1:1 connections without bus matrix
2. BASE_BUS_MATRIX: 4x4 configuration for standard testing
3. BUS_ENHANCED_MATRIX: 10x10 configuration with advanced features

3.3 Address Mapping Strategy

Base Address Calculation:
- Slave[i] Base = 0x0100_0000_0000 + (i * 0x1000_0000)
- Each slave allocated 256MB address space
- Supports up to 64 slaves without address overlap

3.4 ID Mapping Architecture

The VIP implements sophisticated ID mapping:
- Mapped_ID = {master_index[3:0], original_id[11:0]}
- Prevents ID collision between masters
- Proper response routing to originating master
- Full support for out-of-order transactions

4. Agent Architecture

4.1 Master Agent

Components:
- Driver BFM: Drives AXI4 transactions on bus
- Monitor: Captures bus activity for checking
- Sequencer: Manages sequence execution
- Coverage Collector: Gathers functional coverage

Key Features:
- Configurable outstanding transactions
- All burst types (FIXED, INCR, WRAP)
- QoS priority levels (0-15)
- USER signal passthrough

4.2 Slave Agent

Components:
- Driver BFM: Responds to master transactions
- Monitor: Tracks slave responses
- Memory Model: Built-in memory for data checking
- Response Generator: Configurable response types

Key Features:
- Configurable response delays
- Error injection capability
- Memory backdoor access
- USER signal verification

5. Enhanced Features (v2.6)

5.1 QoS (Quality of Service) Support

Implementation:
- 16 priority levels (0-15)
- Priority-based arbitration in bus matrix
- Starvation prevention mechanisms
- Dynamic priority adjustment

Test Coverage:
- Priority ordering verification
- Fairness testing at equal priorities
- Stress testing under saturation
- USER signal priority boost

5.2 USER Signal Architecture

Signal Widths:
- AXI_WUSER_WIDTH: 32 bits (Write data user)
- AXI_ARUSER_WIDTH: 32 bits (Read address user)
- AXI_AWUSER_WIDTH: 32 bits (Write address user)
- AXI_RUSER_WIDTH: 16 bits (Read data user)
- AXI_BUSER_WIDTH: 16 bits (Write response user)

Applications:
- Security tagging
- Parity protection
- Transaction tracing
- Custom sideband data

6. Scoreboard and Coverage Architecture

6.1 Transaction Comparison

Multi-level checking implementation:
1. Protocol Checking: Validates AXI4 protocol rules
2. Data Integrity: Compares write/read data
3. Response Checking: Validates response types
4. USER Signal Verification: Checks signal propagation

6.2 Functional Coverage

Coverage Groups:
- Transaction types (READ/WRITE)
- Burst types and lengths
- Response types
- Address alignment
- Outstanding transactions
- QoS levels
- USER signal patterns

Cross Coverage:
- Burst type x Size
- QoS level x Response type
- Master x Slave combinations
- USER signal x Transaction type

7. Test Architecture

7.1 Test Categories

1. Basic Tests: Fundamental read/write operations
2. Burst Tests: All burst types and sizes
3. Boundary Tests: Address boundary conditions (TC046-TC058)
4. Concurrent Tests: Multi-master scenarios (TC001-TC005)
5. QoS Tests: Priority and arbitration (14 tests)
6. USER Tests: Signal verification (4 tests)
7. Error Tests: Protocol violations
8. Stress Tests: High-load scenarios

7.2 Test Configuration System

Automatic bus matrix selection based on test category:

- ENHANCED_MATRIX_TESTS -> BUS_ENHANCED_MATRIX
- BOUNDARY_ACCESS_TESTS -> NONE mode
- DEFAULT_TESTS -> BASE_BUS_MATRIX

Total: 123 test cases with 100% pass rate

8. Performance Optimizations

8.1 Simulation Performance

- Optimized BFM implementation
- Efficient memory modeling
- Smart coverage collection
- Parallel test execution support

8.2 Regression Performance Metrics

- Average test time: 7.9 seconds
- Parallel jobs: Up to 16
- Coverage merge: < 30 seconds
- Memory usage: ~2GB per test
- Total regression time: < 15 minutes with LSF

9. Version History

9.1 Version 2.6 (Current)

- Fixed concurrent test UVM_FATAL errors for Enhanced configuration
- Resolved exhaustive reads timeout (8.3s vs 2min)
- Extended BFM USER signals to 32-bit
- Updated documentation and test matrix

9.2 Version 2.5

- Fixed TC046/TC047 boundary test failures
- Resolved bus matrix mode mismatch issues
- Changed BOUNDARY_ACCESS_TESTS to NONE mode

9.3 Version 2.4

- Fixed QoS/USER test failures (12 tests)
- Resolved address mapping misalignment
- Updated base address calculations

9.4 Version 2.3

- Added comprehensive QoS test suite (14 tests)
- Implemented USER signal features
- Enhanced scoreboard for USER signals

10. File Organization

```
tim_axi4_vip/
  agent/          # Master and slave agents
  bfm/            # Bus functional models
  bm/             # Bus matrix reference model
  env/            # Environment and scoreboard
  include/        # Configuration headers
  seq/            # Sequences library
  test/           # Test cases
  top/            # Top-level modules
  virtual_seq/    # Virtual sequences
```

11. Configuration Parameters

Parameter	Default	Range	Description
NUM_MASTERS	4	1-64	Number of masters
NUM_SLAVES	4	1-64	Number of slaves
AXI_ADDR_WIDTH	32	32-64	Address bus width
AXI_DATA_WIDTH	64	32-512	Data bus width
AXI_ID_WIDTH	4	1-16	ID field width
AXI_WUSER_WIDTH	32	1-128	Write user width
AXI_RUSER_WIDTH	16	1-128	Read user width
AXI_BUSER_WIDTH	16	1-128	Write resp user width
AXI_ARUSER_WIDTH	32	1-128	Read addr user width
AXI_AWUSER_WIDTH	32	1-128	Write addr user width

12. Test Summary

Category	Count	Description
Basic	5	Fundamental operations
Burst	8	Burst transfers
Boundary	13	Address boundaries (TC046-058)
Concurrent	5	Multi-master (TC001-005)
QoS	14	Quality of Service
USER	4	USER signals
Error	9	Error injection
Stress	3	High load
Matrix	2	Bus matrix tests
Data Width	12	Various data widths
Outstanding	3	Outstanding transactions
Others	45	Miscellaneous tests
TOTAL	123	All test cases

13. Support and Maintenance

Bug Reporting:
Email: axi4_vip_support@company.com

Documentation:
- User Guide: AXI4_VIP_User_Guide.html
- Quick Start: AXI4_VIP_Quick_Start.md
- Test Matrix: testcase_matrix.csv

Repository:
GitHub: https://github.com/moonslide/tim_axi4_vip