AXI4 Verification IP

Comprehensive User Guide

Version 2.8

Generated: September 01, 2025

Complete Documentation with Hyperlinked Navigation

Table of Contents

1. Introduction 3
2. Getting Started 5
3. Architecture Overview 7
4. Configuration Guide 10
5. Test Development 13
6. Sequence Library 16
7. Master Sequences 19
8. Slave Sequences 22
9. Virtual Sequences 24
10. Coverage Models 26
11. Scoreboard 28
12. Error Injection 30
13. Performance Testing 32
14. Debug Features 34
15. Command Reference 36
16. Troubleshooting 38
17. Best Practices 40
18. API Reference 41
19. Release Notes 42

1. Introduction

Welcome to the AXI4 Verification IP (VIP) User Guide. This comprehensive documentation provides everything you need to effectively use the AXI4 VIP in your verification environment.

1.1 Overview

The AXI4 VIP is a complete SystemVerilog/UVM-based verification solution for ARM AMBA AXI4 protocol. It supports all AXI4 features including burst transfers, outstanding transactions, exclusive access, and QoS signaling.

1.2 Key Features

- Full AXI4 protocol compliance
- Configurable master and slave agents
- 100+ pre-built sequences
- Advanced scoreboard
- Protocol coverage
- Error injection
- Performance metrics
- Bus matrix support
- QoS and AxPROT support
- Multi-clock domain support

1.3 Supported Configurations

Parameter	Range	Default
Data Width	8-1024 bits	64
Address Width	32/64 bits	32
ID Width	1-16 bits	4
Masters	1-10	4
Slaves	1-10	4

1.4 Directory Structure

```
tim_axi4_vip/
■■■ agent/           # Agent BFMs
■■■ assertions/      # Protocol assertions
■■■ env/            # Environment
■■■ master/         # Master agent
■■■ seq/            # Sequences
■■■ slave/          # Slave agent
■■■ test/           # Tests
■■■ virtual_seq/    # Virtual sequences
```

1.5 Prerequisites

- SystemVerilog simulator
- UVM 1.2 or later
- Python 3.6+
- Linux OS

2. Getting Started

Quick start guide for running your first test.

2.1 Installation

```
# Extract and setup
tar -xzf axi4_vip_v2.8.tar.gz
cd tim_axi4_vip
export AXI4_VIP_HOME=$PWD
make compile
```

2.2 Running Tests

```
# Basic test
make test=axi4_write_read_test

# With options
make test=axi4_write_read_test seed=12345 gui=1
```

2.3 Make Targets

Target	Description
compile	Compile VIP
test	Run single test
regression	Run regression
coverage	Generate coverage
clean	Clean files

3. Architecture Overview

VIP architecture follows UVM methodology.

3.1 Components

• Test Layer

• Environment

• Master Agent

• Slave Agent

• Scoreboard

• Coverage

• Virtual Sequencer

3.2 Architecture Details

3.3 Architecture Details

4. Configuration Guide

Configure the VIP for your needs.

4.2 Configuration Details

4.3 Configuration Details

5. Test Development

Develop custom tests.

5.2 Test Details

5.3 Test Details

6. Sequence Library

Pre-built sequence library.

6.2 Sequence Details

6.3 Sequence Details

7. Master Sequences

Master agent sequences.

7.2 Master Sequence Details

7.3 Master Sequence Details

8. Slave Sequences

Slave agent sequences.

8.2 Slave Response Sequences

9. Virtual Sequences

Coordinate multiple agents.

9.2 Virtual Sequence Examples

10. Coverage Models

Coverage collection and analysis.

10.2 Coverage Goals

11. Scoreboard

Data integrity checking.

11.2 Scoreboard Architecture

12. Error Injection

Error injection framework.

12.2 Error Types

13. Performance Testing

Performance metrics and optimization.

13.2 Performance Metrics

14. Debug Features

Debug capabilities.

14.2 Debug Techniques

15. Command Reference

Complete command reference.

15.2 Runtime Options

16. Troubleshooting

Common issues and solutions.

16.2 Debug Tips

17. Best Practices

Recommended practices.

18. API Reference

API documentation.

19. Release Notes

Version 2.8 Release Notes

New Features:

• Error injection framework

• Enhanced bus matrix support

• Python regression scripts

• Performance metrics

For support: support@axi4vip.com