IP Verification using Cocotb

Test Plan

This document defines the test specification and plans for the verifying “sumofN” module.

Revision History

Date	Version	Author	Description
04-05-2021	1.0v	Vineet Jain	Initial test Specification
08-05-2021	1.1v	Vineet Jain	Added figures and point of contact

Table of Contents

1. Overview

1.1. Purpose

The purpose of this document is to:

- Plan out the required verification methodology needed for the task.
- Lay out the required strategy the followed in the process.
- Frame a timeline describing deliverables in the project duration.
- Explore various aspects of verification using Cocotb.

1.2. Scope

This document details the testing that will be performed as a part of the coding test. It defines the overall testing and planning view of the project. Few objective pointers are as follows:-

- The “sumofN” module will be tested
- The test will be performed using Cocotb
- The resources required are:
    - IP design specification
    - Cocotb documentation and examples
    - Git – to commit the results

1.3. Scope of Testing

1.3.1. In scope

Verification of “sumofN” module using Cooctb. Any two assertion’s coverage is compulsory.

1.3.2. Out of scope

Change of design specification, change in Cocotb documentation after this documentation design date.

2. Test Strategy

2.1. Test strategy & Approach

The procedure to implemented/conduct verification of given IP module will be as follows:

a) Add the function **$dumpfile** and **$dumpvars** to the sumofN.v Verilog File, so VCD dump can be generated to be used by the tools like GKTWave.

b) Create a Makefile with the target as:
   a. **TOPLEVEL_LANG** = Verilog
   b. **SIM** = Icarus

c) The name of MODULE (in python for verification in Cocotb) being ***rtltb***(register transfer level testbench)

d) Create "**rtltb.py**" Python file and import all the necessary dependencies.

e) Create a main Class, "**Sum2NTB**", which will instantiate the input data drivers and Monitors (both input and output) transactions.

f) Create a separate monitoring class for both **DataBusIn** (input) and **DataBusOut** (output) transaction, inherited from BusMonitor base class of Cocotb.

g) Create a separate class for **DataBusDriver**, which drives the random stimulus to the input.

h) We are creating the main @cocotb.test() module,"**rtltb,**" which stitches all the class module instances and implementing at least two assertions.

i) Assertions are:-

   a. *assert (dut output value == testbench expected value)*
   b. *assert (N(configured data) == length of testbench expected array)*

2.2. Data and resource provision plan

2.2.1. Test environment

For the test environment, the requirements are:-

1) *Linux OS*
2) *Python*
3) *gtk-wave (viewing waveforms)*
4) *icarus (simulation for verilog)*
5) *Git (to commit changes)*

2.2.2. Data Requirements

The IP Design specification sheet for the "sumofN" module is required.

2.2.3. Resources & Skills

Define the types of resources required during the testing window. For example:

- Cocotb documentation and examples to understand its usage and implementation.
- Resources for Python programming and syntax understanding.

2.3. Testing Tools:

The following tools will be used for testing:

Process	Tool
Text-Editor	VS-Code
Test case planning	Microsoft Excel
Test Plan document	Microsoft Word

2.4. Diagram:

The following architecture is used for creating the cocotb model to verify the model:

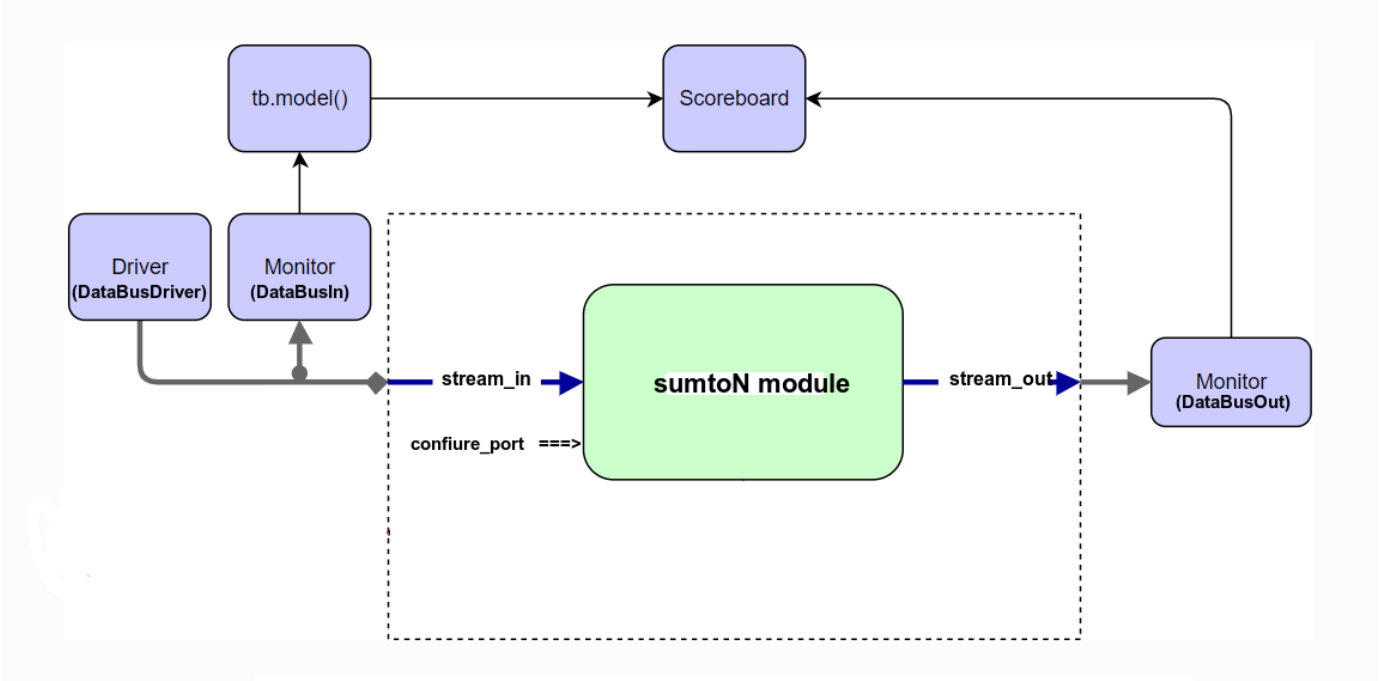

Fig-1: sumtoN module's testbench design with this structure

3. Timeline Plan

3.1. Timeline Details

Task	Requirements	Responsibility	Start Date	End Date
Setting up environment	We are completing all the data and resource provision requirements.	To setup IDE and any dependencies needed	04-05-21	
Creating makefile for Cocotb	Creating makefile with all required arguments like- a) TOPLEVEL_LANG, SIM, MODULE, DUT, the path of rtl file, and include cocotb-config b) Add all: to make sim target c) Add clean: to remove the sim_build, VCD, etc. files		04-05-21	
Modifying the "sumofN" module	Inside the initial block, add functions $dumpfile and $dumpvars to create a VCD dump waveform, which can be analyzed by gtk-wave		04-05-21	
Creating "rtltb.py" main python file	a) Import necessary cocotb libraries b) Define parameter N, for configure_data(sumofN) c) While referring to examples in cocotb documentation, make the "Sum2NTB" class, which instantiates drivers and monitors.		05-05-21	
Making demo implementation by replicating waveform	Creating demo a cocotb. Test ()by replicating the waveform (given as task) to make sure not errors(typos) are present in either waveform and IP specs. This task will also help in learning and getting familiar/comfortable with the tool.		05-05-21	
Making sumofN python model	Create tb_model()to operate on input data monitor, which will be given to Scoreboard.		05-05-21	
We are creating Custom monitors and driver classes with IP-specific names and overriding with required behavior.	a) Make a custom monitor class (DataBusOut), inherited from BusMonitor, with IP specification specific signal names and conditions (i.e., En_out_get, out_get). b) Repeat the above process for (DataBusIn), with IP specification's signal names (i.e., EN_in_put, in_put) c) Make custom bus driver (DataBusDriver) inherited from ValidatedBusDriver, with IP's specs signal names. d) For DataBusDriver, take the Avalon bus driver implementation as a reference to overriding any implementation or required parameter.		06-05-21 to 07-05-21	
Creating Main @cocotb.test() module and testing	Connecting all the required block (as shown in fig-1) and providing random stimulus in input and verifying the sumofN design for any corner cases.		07-05-21	

4. Approval and Point of Contact

4.1. Approvals

The following persons are responsible for the critical aspects of testing:

Task	Responsible Person	Signature
Reviewer	Vijayvithal jahagirdar	

4.1. Points of contact

The following people can be contacted about this document

Primary Contact	
Name	Vineet Jain
Title/Organisation	Student, B-Tech (Final-year) from LNMIIT, Jaipur
Phone	+91-7597291103
Email	vineetajm1999@gmail.com