ASMD Multiplier Test Plan

Module Name/Internal ID: ASMD_MULT

Engineer(s):
Oluwaseyifunmi Williams
Jonathan Defo

1. Revision

Rev	Comments	Date	Approval: Team/Date
1.0	Initial Creation	12/09/2024	

2. Required Documents

ID	Name	Revision
	ASMD_MUTIPLIER Logic Requirements Document	8/7/2023

3. Supplemental Documents

ID	Name	Revision
	https://en.wikipedia.org/wiki/Algorithmic_state_machine	

4. Acronyms

Abbreviation	Definition
ASMD	Algorithmic state machine with DataPath

Table Of Contents

5. Scope

This document describes in detail, the test plan and test results for the XXX module.

6. Functional Verification

ASMD_MULT_TP001: Verify that the module implements the inputs and asserts ready bit according to given condition:

Verifies: ASMD_MULT_F001

Methods: visual inspection, compilation warnings, synthesis.

Results:

Port	Direction	Width	Description
clk	input	1’b1	Input Clock Signal
rst	input	1’b1	Input Synchronous Reset Signal
start	Input	1’b1	Input module start signal
word0	input	[word-lenght]’b1	Input word 0
word1	input	[word-lenght]’b1	Input word 1
ready	Output	1’b1	Module ready output signal

produce	Output	[word-lenght*2]'b1	Module multiplication product output

ASMD_MULT_TP002: Verify that module implements wordlength with a default value of 4:

Verifies: ASMD_MULT_F002

Methods: visual inspection, compilation warnings.

Results:

Name	Default Value	Description
word_length	4	Input bit length

ASMD_MULT_TP003:

Verifies: ASMD_MULT_F003.

Initialize Testbench:

- Set up the clock signal (clk).
- Initialize the reset signal to 0.

Assert Reset Signal:

- Set reset to 1 for a brief period to simulate the assertion of the reset.

Monitor Outputs:

- Check the ready signal to verify it is asserted (1).
- Check the product signal to verify it is set to zeros.

- **De-assert Reset Signal**:
- Set reset back to 0 and verify that the module transitions out of the reset state correctly.

Methods: simulation.

Results:

Signal	Expected Value	Description
`ready`	`1'b1`	Ready signal should be asserted upon reset.
`product`	`2*word_length` bits of `0`	Product output should be reset to all zeros.

ASMD_MULT_TP004:

Verifies: ASMD_MULT_F004.

1. Set `word0` (binary) and `word1` = (binary).
2. Initially, Set start = 0,
3. Set ready = 1
4. Set Start to 1 for a brief period to simulate the deassertion of the reset.
5. Check the `product` after the multiplication.
6. Ensure that `ready` is asserted again once the multiplication is complete.

Methods: simulation.
Results: After the multiplication is complete, `product` should be word0 * word1