ASMD\_MUTIPLIER Logic Requirements Document

Module Name/Internal ID: ASMD\_**MULT**

Engineer(s):

# Revision

| Rev | Comments | Date | Approval: Team/Date |
| --- | --- | --- | --- |
| 1.0 | Initial Creation | 8/7/2023 |  |

# Required Documents

|  |  |  |
| --- | --- | --- |
| ID | Name | Revision |
|  |  |  |

# Supplemental Documents

|  |  |  |
| --- | --- | --- |
| ID | Name | Revision |
| ASM | Algorithmic\_state\_machine (Wikipedia) | https://en.wikipedia.org/wiki/Algorithmic\_state\_machine |

# Acronyms

|  |  |
| --- | --- |
| Abbreviation | Definition |
| ASMD | Algorithmic state machine with DataPath |

Table Of Contents

[1. Revision 1](#_Toc144373190)

[2. Required Documents 1](#_Toc144373191)

[3. Supplemental Documents 1](#_Toc144373192)

[4. Acronyms 1](#_Toc144373193)

[5. Module Scope 2](#_Toc144373194)

[6. Functional Requirements 2](#_Toc144373195)

[1. Interfaces 2](#_Toc144373196)

[2. Parameters 3](#_Toc144373197)

[3. Logic requirements 3](#_Toc144373198)

[7. Verification/Testing 3](#_Toc144373199)

# Module Scope

This module accepts two words as inputs and outputs thier product.

# Functional Requirements

## Interfaces

ASMD\_MULT\_F001: Module shall implement the interfaces described in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| 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 |

## Parameters

ASMD\_MULT\_F002: module shall implement the parameters described in the following table:

|  |  |  |
| --- | --- | --- |
| Name | Default Value | Description |
| word\_length | 4 | Input word size in bits |

## Logic requirements

* + - 1. ASMD\_MULT\_F003: Module shall, upon assertion of **rese**t:
         1. assert **ready**.
         2. Assign **product** to zeros.
      2. ASMD\_MULT\_F004: Module shall, upon assertion of **start**:
         1. deassert **ready**.
         2. Calculate the product of **word0** \* **word1**.
         3. Assert **ready** only when the multiplication computation is complete.
         4. Ouput the the multiplcation computation on the **product** output port.