Memory 16x32 Verification Plan

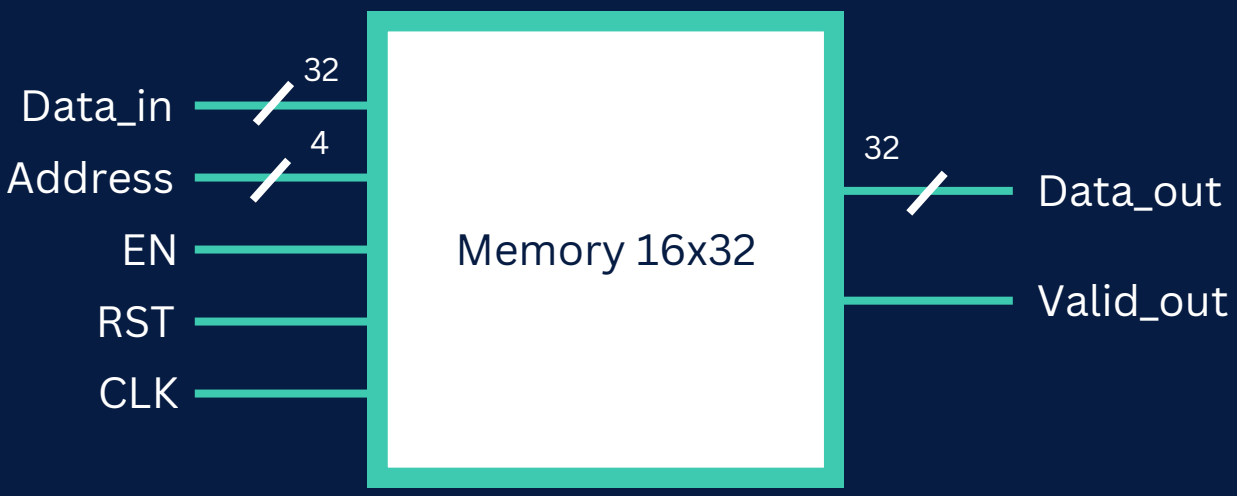
By: Ziad Abdelfattah

Memory 16x32 Verification Plan

Contents

Introduction

This document includes all details regarding the design details and specifications, the signals and ports as well as the test plan and cases to cover. The module is a simplistic 16x32 Memory with data, address, and enable ports as inputs, data and valid ports as outputs.



IP Design Details

a. Signals I/O’s

Signal	Direction	Width	Description
Data_in	Input	32	Data Bus for input in memory
Address	Input	4	The address to be specified in memory for read or write operations
EN	Input	1	Enable signal: low for read and high for write
CLK	Input	1	Clock Signal
RST	Input	1	Positive edge asynchronous reset signal
Data_out	Output	32	Output data bus for read operations, stays on last read value till address is changed
Valid_out	Output	1	Valid signal to validate output when operation is read

b. Internal Signals

1. memory: Reg type for implementing the physical memory in the module

2. i: Integer for memory reset loop variable

c. Design Functionality

The design is a 16x32 bit simple memory with two main inputs and three control signals. It has two modes of operation “Read” and “Write” controlled by the “EN” enable signal; low operates in “Read” and high operates in “Write”.

IP Design Details

c. Design Functionality

i. Read Operation

When the “EN” is set to low, at the positive edge of the clock “CLK” the output “Data_out” takes the value of the memory with the specified address from the input “Address” and the output “Valid_out” is set to high, unless reset “RST” is high.

ii. Write Operation

When the “EN” is set to high, at the positive edge of the clock “CLK” the memory takes the value of “Data_in” with the specified address from the “Address” registering it in the memory and setting “Valid_out” to low and leaving “Data_out” with last read value, unless reset “RST” is high.

Verification Strategy

Verification Plan

Test ID	Tested Feature	Steps/Input order	Expected Output
BT00 - Random Tests "Boundary Testing / Group Testing"			
BT01	Initial Write after reset	1. Reset/Initialize Memory 2. Set EN high, Set Data_in (32'd10), Set Address (4'd1) 3. Set EN low, Set Data_in (32'd0), Set Address (4'd1) 4. Read Data_out and Valid_out	Data_out = 32'd10 Valid_out = 1'b1
BT02	Write max value	1. Set EN high, Set Data_in (MAX), Set Address (4'd2) 2. Set EN low, Set Data_in (32'd0), Set Address (4'd10) 3. Read Data_out and Valid_out	Data_out = 32'hFFFFFFFF Valid_out = 1'b1
BT03	Overwrite with min	1. Set EN high, Set Data_in (32'd30), Set Address (4'd3) 2. Set EN low, Set Data_in (32'd0), Set Address (4'd3) 3. Set EN high, Set Data_in (32'd0), Set Address (4'd3) 4. Set EN low, Set Data_in (32'd0), Set Address (4'd3) 5. Read Data_out and Valid_out	Data_out = 32'd30 Valid_out = 1'b1
BT04	Overwrite with value	1. Set EN high, Set Data_in (32'd20), Set Address (4'd4) 2. Set EN low, Set Data_in (32'd0), Set Address (4'd4) 3. Set EN high, Set Data_in (32'd40), Set Address (4'd4) 4. Set EN low, Set Data_in (32'd0), Set Address (4'd4) 5. Read Data_out and Valid_out	Data_out = 32'd40 Valid_out = 1'b1
EXA00 - Address Exhaustive Testing "Loop on all bits"			
EXA01	Address Bits - Write then Read every address	1. Set EN high 2. Write in memory 3. Read data to check 4. Increment Address	Golden Model Implementation
RST00 - Reset Function			
RST01	Reset memory and Output	1. Fill Memory "EN high, Data_in non-zero" 2. EN low 3. Trigger Reset 4. Read data of whole memory	1. Valid_Out should be low for 1 CLK cycle then high 2. All Data_out outputs must be low

Exit Criteria

When all nodes are hit, all bits are toggled, all features are tested, and all tests are successful.