Group 2- Patrick Balcombe, Harry Austin, Devashish Bishnoi

**ECS615U-** **Laboratory Session 5**

# **Abstract**

**Method**

The first stage to create the devices was taken in ISE project navigator in which the VHDL code was written for the actual devices and their test benches. Through this we then also simulated the devices in ISim to gain timing diagrams for our test benches to confirm that the devices were working as expected according to the derived truth table. The constraints files were then created using a standardised UCF file created in a previous lab which were then altered for the I/O names of the specific devices in ISE. These constraints files map the inputs out outputs of the top level devise to the FPGM board that we later uploaded to the devices. We configure the FPGM board and upload the top level devises bit files, created in ISE, using iPACT after which the devices actual inputs and outputs where then tested.

The devices higher up in the design such as the Modulo-m counters use the functionality of the devises lower down in the design such as the half adder and d-flip flop to create a hierarchical design. Devices that are further down are port mapped, defining their connections within the device, into the higher level device so that it can be passed the signals necessary to carry out its function.

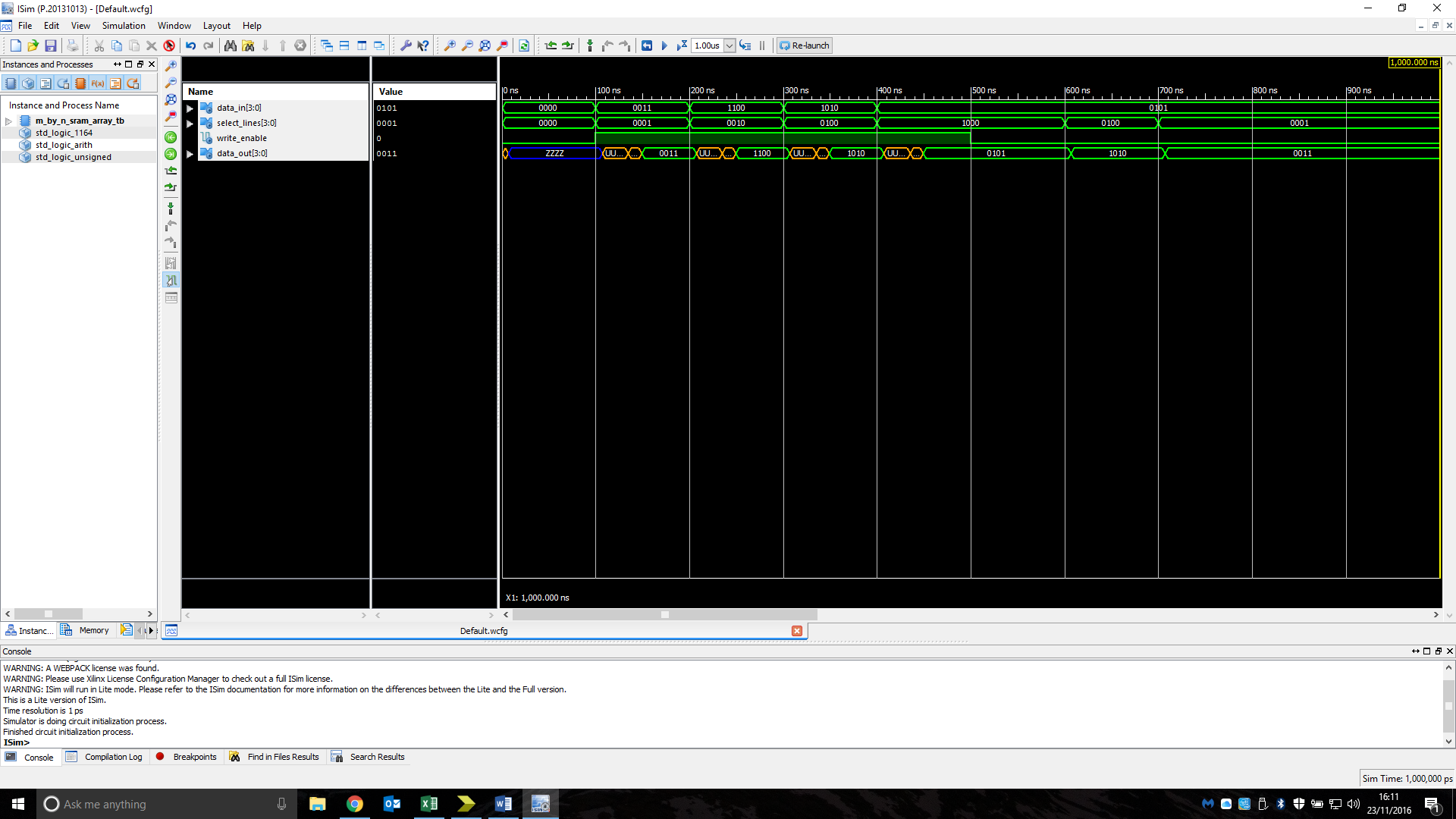
As in this lab clocks were now required for the devices a signal had to be supplied to the FPGA to act as the clock. We used the variable clock that is available on the FPGA board and connected it with wire connecting it from its pin to one of the board’s general IO pins.

In this lab the NI Elvis development board did not have enough switches to cover all the inputs to the top level devises so virtual switches provided by the NI software had to be used. A set of virtual switches were opened on the computer and then connected to the FPGA by connecting pins 1-7 of general purpose I/O (GPIO) to pins 1-7 of digital I/O (DIO).

# Simulation and FPGA Testing

In this section we individually run the test bench files for each device in Modelsim and on the physical FPGM board to check the timing diagram as well as actual results of the I/O matches its truth table. The Truth tables lists possible combinations of inputs and the relevant outputs which are compared to the results gained from the simulation and tests on the board.

**m x n-bit Static RAM cell array**

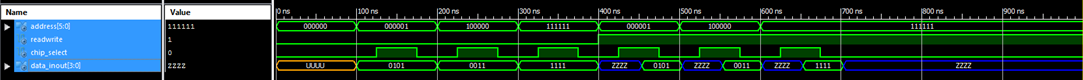
Truth table:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA\_IN | | | | SELECT\_LINES | | | | WRITE\_ENABLE | DATA\_OUT | | | |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| X | X | X | X | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| X | X | X | X | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| X | X | X | X | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| X | X | X | X | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Figure1: Simulation timing diagram of an m x n-bit Static RAM cell array showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**64 x 4-bit Static RAM**

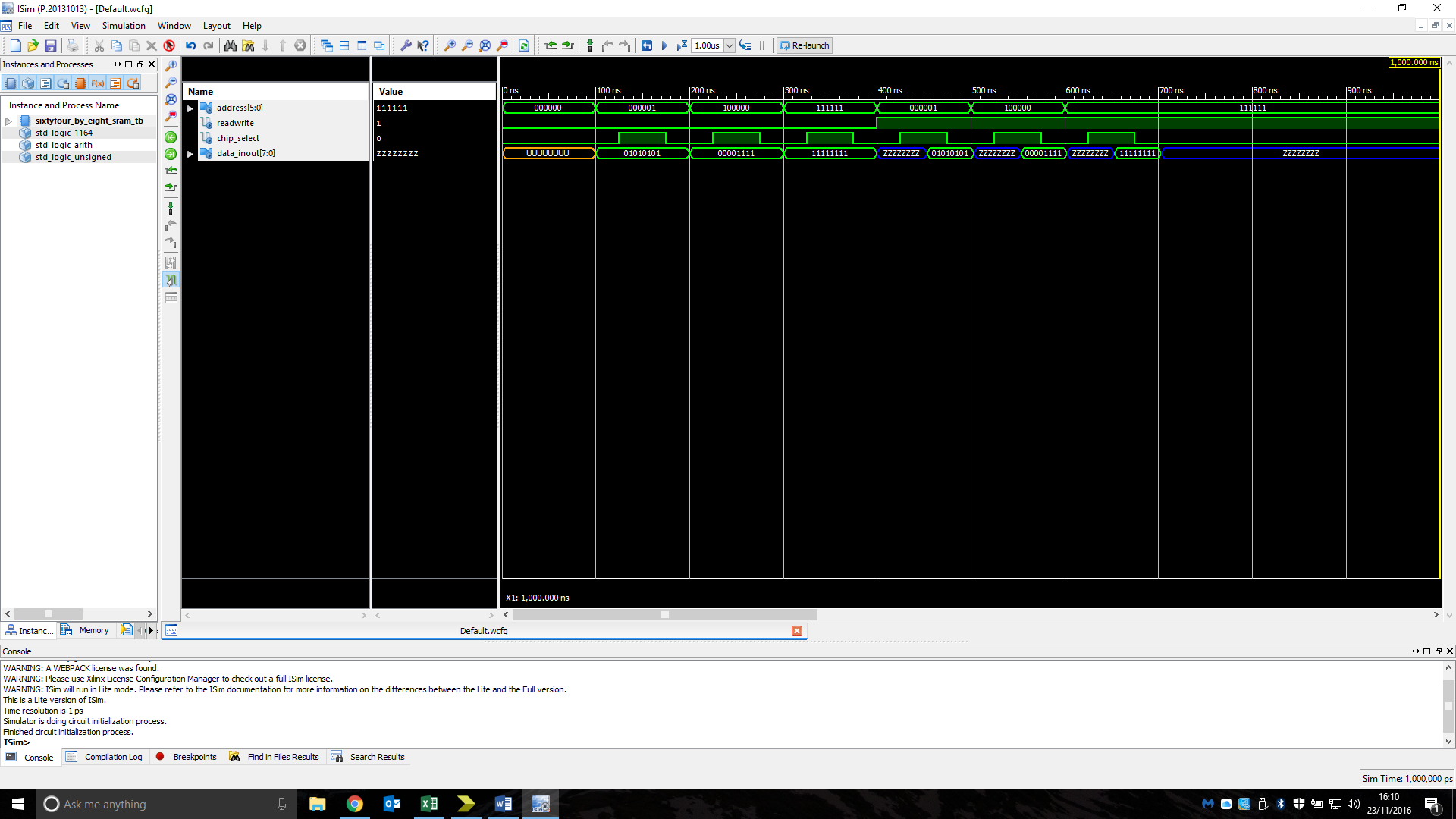
Truth table:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ADDRESS | | | | | | READ/'WRITE | CHIP\_SELECT | DATA\_INOUT | | | |
| X | X | X | X | X | X | X | 0 | X | X | X | X |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Figure2: Simulation timing diagram of an n-bit asynchronous counter showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**64 x 8-bit Static RAM**

Truth table:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ADDRESS | | | | | | READ/'WRITE | CHIP\_SELECT | DATA\_INOUT | | | |
| X | X | X | X | X | X | X | 0 | X | X | X | X |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Figure3: Simulation timing diagram of a 64 x 4-bit Static RAM showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**Static RAM Cell**

Truth table:

|  |  |  |  |
| --- | --- | --- | --- |
| Data\_in | Cell\_select | Write\_enable | Data\_out |
| 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| x | 0 | x | z |

Figure4: Simulation timing diagram of a Static RAM Cell showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**D-Latch**

Truth table:

|  |  |  |  |
| --- | --- | --- | --- |
| c | D | Q | Q-bar |
| 0 | 0 | last Q | last Q-bar |
| 0 | 1 | last Q | last Q-bar |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Figure5: Simulation timing diagram of a D-Latch showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**Register File Cell (RFC)**

Truth table:

Figure6: Simulation timing diagram of an Register File Cell showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**n-bit RFC register**

Truth table:

Figure7: Simulation timing diagram of an n-bit RFC register showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**8 x n-bit Register File**

Truth table:

Figure8: Simulation timing diagram of an 8 x n-bit Register File showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for both the simulation and test on FPGM board.

**16 x 7-bit ROM seven-segment display decoder**

Truth table:

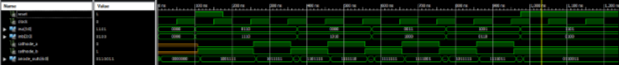
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | | | | Segment | | | | | | |
| A3 | A2 | A1 | A0 | A | B | C | D | E | F | G |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |

Figure9: Simulation timing diagram of an 16 x 7-bit ROM seven-segment display decoder showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for both the simulation and test on FPGM board.

**TDM (Time-Division Multiplex) display circuit** Truth table:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Value | reset | clock | InA | InB | Cathode A | Cathode B | Anode\_outs |
| 1 | 0 | ↑ | 0110 | xxxx | 1 | 0 | 1 0 0 1 1 1 1 |
| 2 | 0 | ↑ | xxxx | 1110 | 0 | 1 | 1 0 1 1 1 1 1 |
| 3 | 0 | ↑ | 0000 | xxxx | 1 | 0 | 1 1 0 1 1 1 1 |
| 4 | 0 | ↑ | xxxx | 1010 | 0 | 1 | 1 1 1 1 1 1 0 |
| 5 | 0 | ↑ | 0011 | xxxx | 1 | 0 | 1 1 1 1 1 1 1 |
| 6 | 0 | ↑ | xxxx | 1000 | 0 | 1 | 1 1 1 1 0 0 1 |
| 7 | 0 | ↑ | 1001 | xxxx | 1 | 0 | 1 0 1 1 1 1 1 |
| 8 | 0 | ↑ | xxxx | 0110 | 0 | 1 | 1 1 1 1 0 1 1 |
| 9 | 1 | x | 1101 | xxxx | 1 | 0 | 0 1 1 0 0 1 1 |
| 10 | 1 | x | 1101 | 0100 | 1 | 0 | 0 1 1 1 1 0 1 |

Figure10: Simulation timing diagram of an TDM display circuit showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for both the simulation and test on FPGM board.

**Additional Question**

**512 x 32 bit RAM Device**

Design:

Truth table:

Figure11: Simulation timing diagram of a 512 x 32 bit RAM Device showing its behaviour when tested with a range of possible inputs.

Analysis: The output of the device is as expected for the simulation.

**Discussion**

All devices that were tested matched their truth tables for all stimuluses’ input. In the case of the N-bit synchronous counter with enable it was able to be simulated once an inconsistency in the name of the incrementer given by the engineer and that provided code was rectified. In future labs further efforts should be made to check for possible inconsistencies when defining the name of the files at the beginning of the implementation process. The two mod-m counters both count up from zero while the enable is high until they reach the same level as the m\_value and revert to “0000”. As soon as the asynchronous counter reaches the value of m\_value it resets however the synchronous counter waits till the next clock pulse this is because the asynchronous counters comparator is connected directly to the reset input which is not clock dependant unlike the sync counter that has it connected to the load/count input.

# Conclusion