

ECE371 Design of Digital Circuits and Systems  
  
Winter 2019

**Lab 2 - Memory Blocks in FPGA**

Student: Jeff Josephsen  
Professor: Dr R. Hussein

1-24-2019

**Abstract:**

This lab is an investigation into methods of creating memory blocks in on an FPGA device. The creation of these blocks of memory follow followed a few general steps. These steps included creating block diagrams with appropriate inputs, internal characteristics, and outputs. These block diagrams were followed by the crafting of SystemVerilog line coding of modules that would instantiate memory modules and by use of system tools provided for that same purpose. The instantiation of these memory modules and their testing were done both in software on Quartus II with ModelSim and in hardware on the Altera DE1-SOC with the testing by SignalTap II Logic Analyzer software.

**Introduction:**

Nearly every computer system will require the use of a block of memory. The skill of creating a block of memory on an FPGA is a fundamental building block for most embedded firmware engineers. In this lab, in 4 different task sections, memory blocks were created and tested. In the first task a RAM module was created from system tools containing 32 words wherein each word held 4 bits. The RAM was instantiated in software and tested with software tools. In the second task, that same RAM and modules were instantiated in Hardware and tested by hand with buttons and switches to write in and read out data in the memory. In the third task, modules that would instantiate memory were designed and written by hand. They were then instantiated in software and test by software means. Lastly, in the fourth task, a dual port memory module and required top level modules assuring correct connections was instantiated in hardware and tested by use of the SignalTap II Logic Analyzer software.

**Procedure:**

*What were you testing or designing? What did you actually do? Give good in-depth discussion*

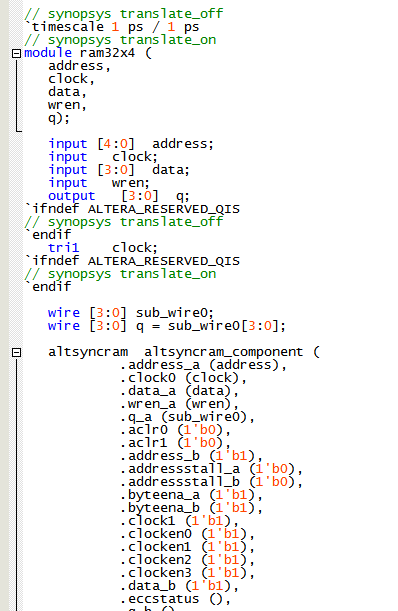
*of your design and your code, and describe your process. List any equations you used or*

*techniques you employed. This will be what takes up the majority of pages in your report.*

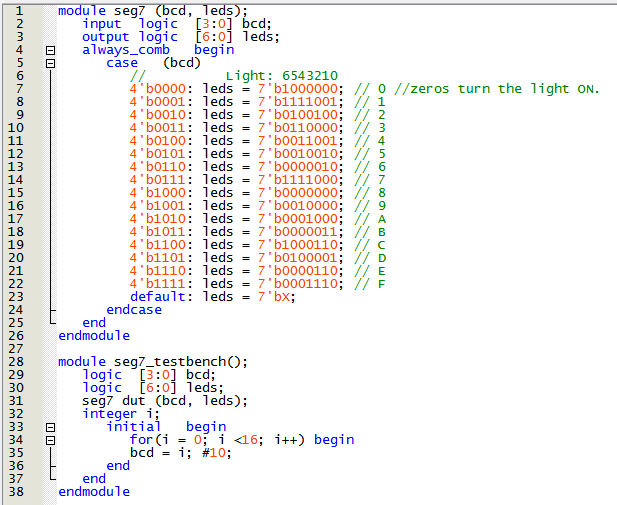
*Include any relevant graphics or pictures to help make topics clear.*

Task 1 and 2:

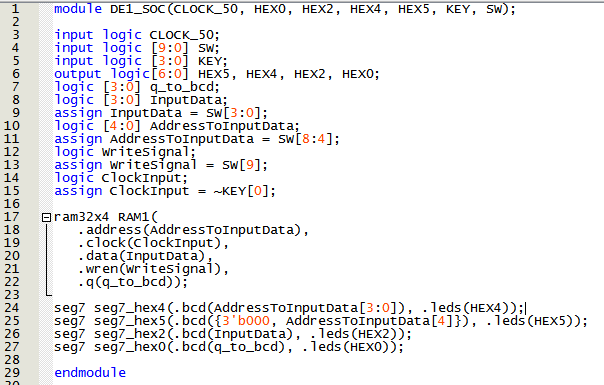
In the first task a RAM module was created from system tools containing 32 words wherein each word held 4 bits. The RAM was instantiated in the QUARTUS Prime Lit Edition and tested with the use of the ModelSim Intel FPGA Starter Edition 10.5b software. The following images shows part of the module used created by system tools to instantiate RAM of size 32x4.



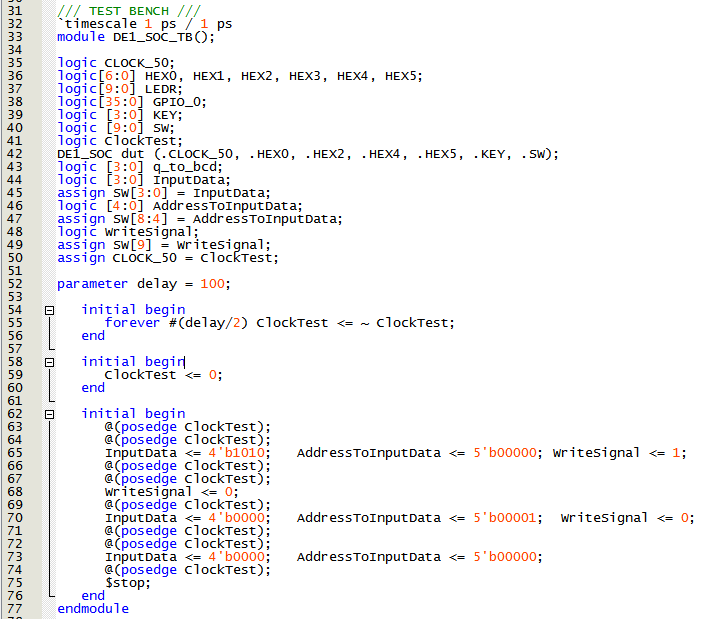
The following module to help run a seven segment display was included in the project. This was done because the second task would require that the RAM be instantiated in hardware. The following image shows the module for a seven segment display augmented for display of hexi-decimal characters from 0 to F.



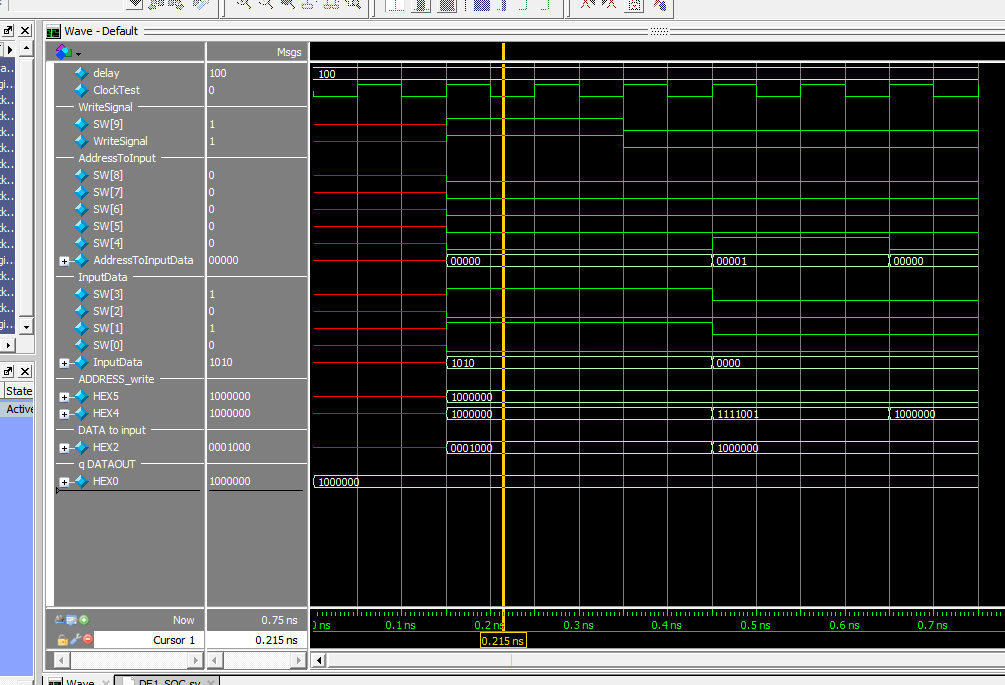
The RAM module and seven segment display module were instantiated in a top level module which also connected correct signals to appropriate lines of the Altera DE1-SOC hardware. This following image shows this top-level module.



The following test bench was written to test and verify the correct working of the top-level test bench.



The test bench was simulated in ModelSim software. The resulting waveforms can be seen in the following image.

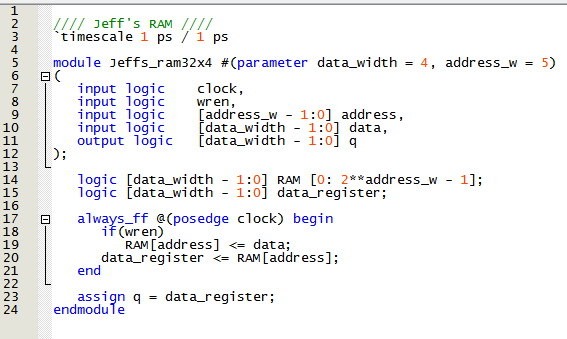


The memory module was found to work as expected.

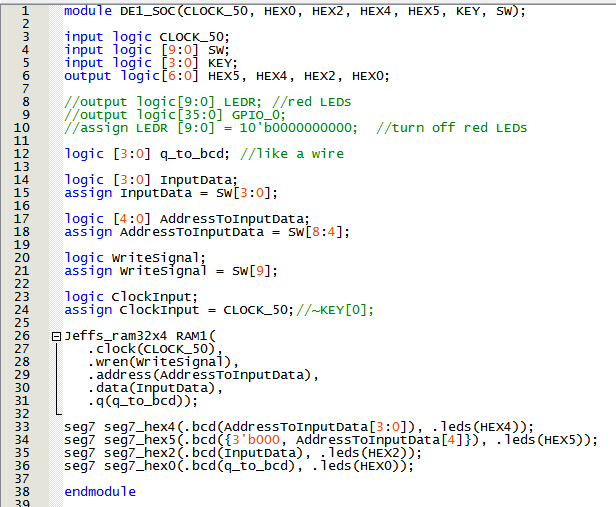
Task 3:

In the third task, modules that would instantiate memory were designed and written by hand. They were then instantiated in software and test by software means.

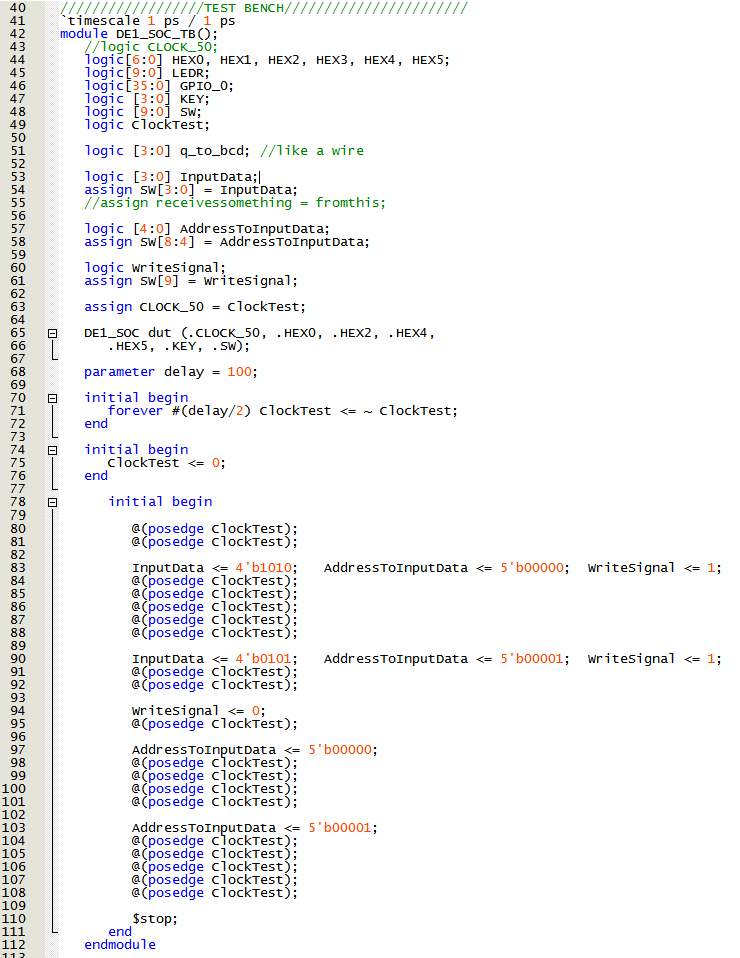
The following is a copy of the 32x4 bit RAM custom written by the student.



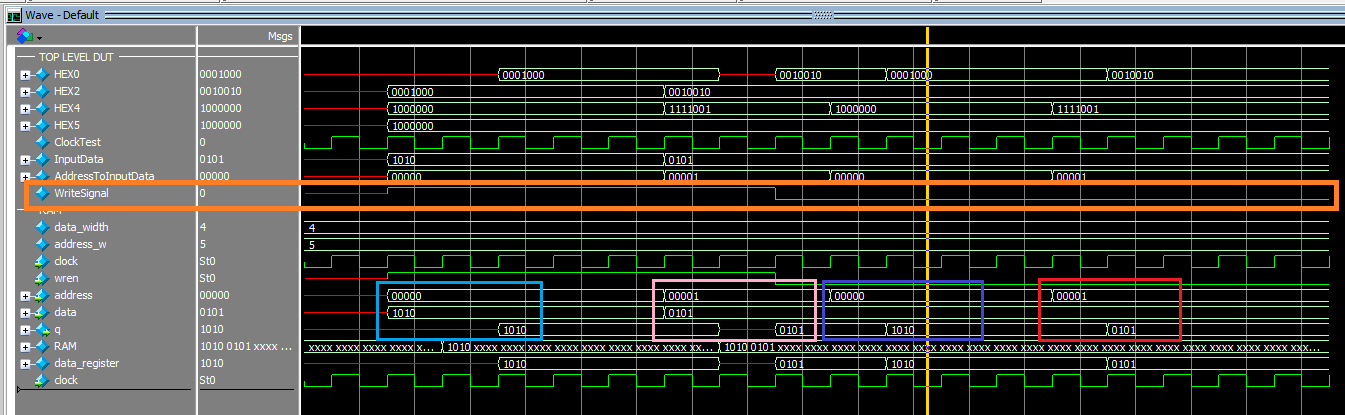
The custom build RAM was instantiated in software and hardware in the following top level module.



That top level module was tested with the following test bench.



The test bench waveforms were simulated in ModelSim software and the results can be seen in the following image.

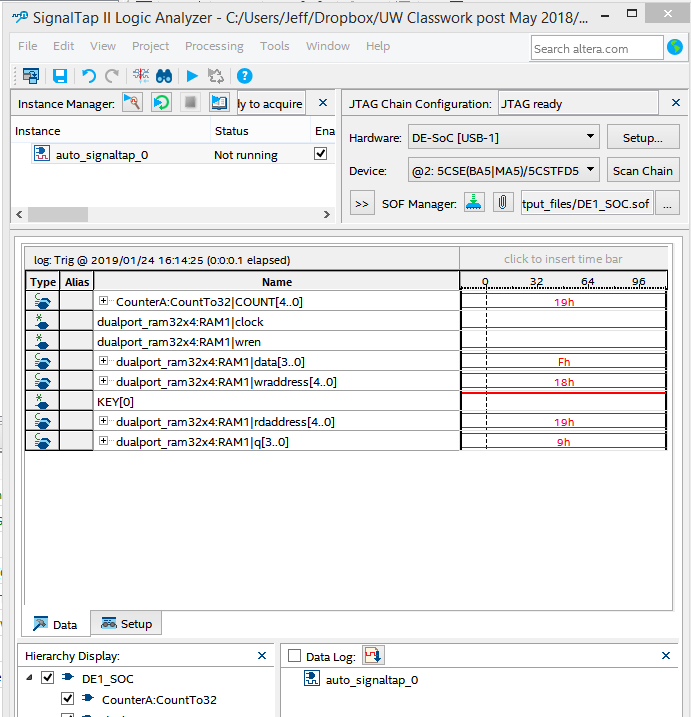


In the above image it can be seen in the light blue and pink boxes where diffreent data is written to two different locations in memory, and as the write enable signal is disabled in orange, the data can be seen as it is recalled by address in the blue and red squares.

Task 4:

Lastly, in the fourth task, a dual port memory module and required top level modules assuring correct connections was instantiated in hardware and tested by use of the SignalTap II Logic Analyzer software.

The dual port ram module was created with system tools in the same fashion as Task 1. These modules were again instantiated with use of a top level module and implemented in hardware. The goal of this task was to investigate the values of signals inside the hardware by way of the SignalTap II Logic Analyzer. The following image shows the appropriately working findings of signals in the hardware.



**Results:**

Here, present and discuss your findings. Explain them, and how (or if) they differed from the

expected. Use graphics to help elucidate your results. Answer any questions raised in the lab

report. Also use this section to talk about any known errors in your lab, or any challenges you

faced.

The modules and implementation in software and hardware were produced as required by the instruction of the lab. The results came together as expected. There was an issue with the triggers in SignalTap. The finding of signals without triggers was not a problems as the signals were received by continuous measurement.

**Conclusion:**

This lab was an excellent lab in teaching the principles of RAM creation in an FPGA. This seems to be just one of a number of tools that will become the tools of my own tool belt for the creation of Application Specific Integrated Circuits on an FPGA lattice.