EE4DSA - Coursework 3

James Reed

Submitted: March 10, 2013

1 Overview

My decoded and commented hex is in decoded_commented_hex.txt, with my unlocking code in a separate text file in this folder. My assembler and disassembler are in the .zip with instructions on how to run them if you wish. My reports are in the reports/ directory as directed and my unlocking code, decoded hex and assembler/disassembler are in the extra/ folder.

2 Reports discussion

My resource utilisation summary for my execution unit is as shown here:

- inferred 9 Adder/Subtractor(s).
- inferred 186 D-type flip-flop(s).
- inferred 13 Comparator(s).
- inferred 407 Multiplexer(s).
- inferred 8 Combinational logic shifter(s).
- inferred 1 Finite State Machine(s).
- Unit `<execution_unit>` synthesized.

My resource utilisation summary for my ALU unit is as shown here:

- inferred 2 Adder/Subtractor(s).
- inferred 4 Multiplexer(s).
- Unit `<alu>` synthesized.

Overall device utilisation is shown here:

- Selected Device : 6slx16csg324-3
- Slice Logic Utilization:
- Number of Slice Registers: 606 out of 18224 3%
- Number of Slice LUTs: 2912 out of 9112 31%
- Number used as Logic: 2912 out of 9112 31%

- Slice Logic Distribution:
- Number of LUT Flip Flop pairs used: 2939
- Number with an unused Flip Flop: 2333 out of 2939 79%
- Number with an unused LUT: 27 out of 2939 0%
- Number of fully used LUT-FF pairs: 579 out of 2939 19%
- Number of unique control sets: 10
- IO Utilization:
- Number of IOs: 38
- Number of bonded IOBs: 38 out of 232 16%
- Specific Feature Utilization:
- Number of Block RAM/FIFO: 10 out of 32 31%
- Number using Block RAM only: 10
- Number of BUFG/BUFGCTRLs: 1 out of 16 6%

With regards to the conversation we had about the difference in clock speeds between mine and Oliver's solutions, Oliver has 100 more flipflops in his execution unit resource utilisation and only a few more multipliers, though overall resource usage seems to be about the same for both of us. Therefore, the only way mine was the 7MHz faster could be that my delay paths were a lot shorter than his, due to the modularity in his design, which although did make for a slower design, meant that his is a lot easier to extend and maintain.

With this final design, about three cores can be fit onto the FPGA, although this is not synthesised for resource efficiency, but for speed. My youtube video to provide proof of the design working is available at: `http://youtu.be/Djtp4h828_A`