EEC 180 Lab 6

Bradley Manzo

916953788

Objective and Introduction: The objective of this lab is to construct the hardware necessary to multiply two 8 x 8 matrices, and then successively accelerate the hardware through the implementation of additional MAC modules and multiport RAM inputs and outputs.

Prelab:

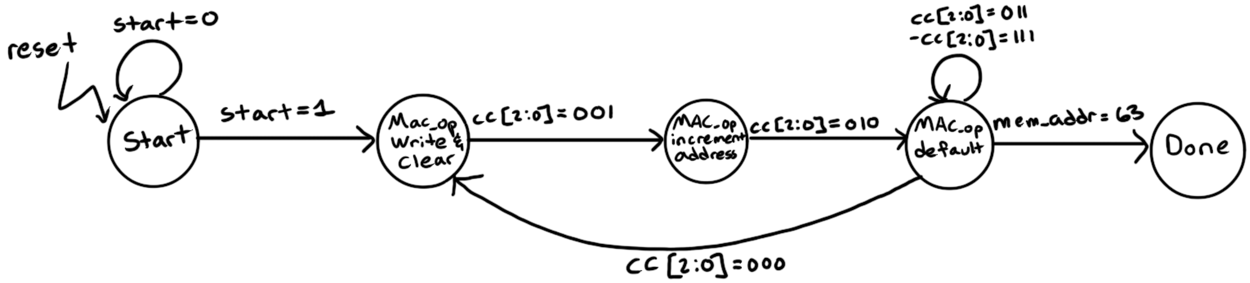

Figure 1: Task 2 State Diagram

Results:

Task1:

Estimate the resources usage:

number of registers:	19
number of logic elements:	20
number of memory bits:	0
number of embedded multipliers:	1
number of embedded adders:	1

Task2:

Estimate the resources usage:

number of registers:	22
number of logic elements:	33
number of memory bits:	.28 kB
number of embedded multipliers:	1
number of embedded adders:	1

Estimate the performance of your design:

number of clock cycles:	514

- bottleneck: every element has to be read 8 times due to the single MAC and single port input and output RAM

Task3:

Estimate the resources usage:

- number of registers: 15
- number of logic elements: 19
- number of memory bits: .28 kB
- number of embedded multipliers: 2
- number of embedded adders: 2

Estimate the performance of your design:

- number of clock cycles: 261
- bottleneck: every element only has to be read 4 times now, but the Single port output RAM necessitates a writing delay

Conclusion:

After concluding this lab I feel I have greatly solidified my understanding of finite state machines, as well as my understanding of how to instantiate multiport RAM. Initially I approached this lab using convoluted conditional statements as if I were writing in C++, but I found that my code fluency increased exponentially when I adapted the logic into the style of an FSM. Not only is the separation of next state and combinational logic more stable and intuitive, but it also brings the structure closer to the actual hardware. In task 4 I chose to octo-port my input ram so that my only bottleneck is my single ported matrix B which is left to read each element a single time. I suppose that it would be possible to have wire 16 MACs together to divide each row in half and theoretically bring the cycle count down to ~32 however this would drastically change the schedule of my FSM and I am limited by the deadline.