EECE 3324: Computer Architecture and Organization

Project Part 2

Alexander Wise

002907473

June 17, 2024

Summary and Approach:

The purpose of this second part of the project was to take the parts developed in the previous part of the project and combine them all together to make a fully pipelined, 3 stage CPU with registers and all stages working in conjunction with one another thus, allowing for more complex operations and instructions to be completed.

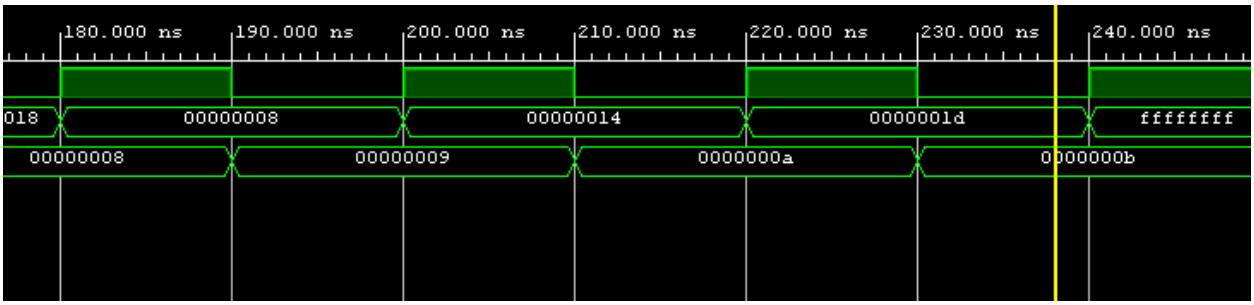
My approach to this was similar to that of Part 1: work in stages to create the final working processor. To do so, I simply looked at each additional component that we needed to design and took time to analyze what the necessary components were to make it work.

Top Level Module:

To design the top level module, I took into account each input at each stage. For the first stage, there were 6 inputs, for the second stage there were 6 inputs, and for the final stage, there were 2 inputs. With this in mind, I created a register for each input where the input was the corresponding value from the previous stage and the output was the corresponding value for the next stage. This was replicated for all 3 stages with the creation of new variables for each new input and output.

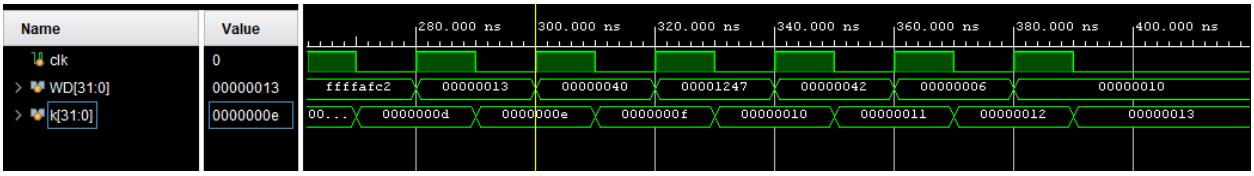
Running:

Test scenarios 6, 7, and 8 correspond to AND, ANDI, and ADD operations respectively. The results of these operations don't appear in the waveform until k = 9, a, and b. These k values are 3 higher than the test scenario indices. This is due to the 3 stage system in the pipelined processor. The answer is not written back to the register file until 3 clock cycles have passes and then the result is written to the register file on the rising edge of the clock.



As seen in the image above, we can see that the result of operation 6 is 00000008 with a corresponding k value of 9. Similarly, we can see that the value is written to the register on the rising edge of the clock as seen by the change in the value on the rising edge of the clock in the top line.

Additional Scenarios:



Above is the waveform output of the 5 additional scenarios. The scenarios completed above are as follows:

- SLL R4 R1 R5
- ADDI R18 R2 #1234
- ADD R6 R4 R1
- ANDI R9 R6 #00FF
- ADD R5 R3 R7

When run all together, the console returns the following output signaling the successful completion of all test case scenarios:

```
   ibus=xxxxxx xxxxx xxxxx xxxxx xxxxxxxxxxx for instruction         18
 Time=     380.0ns
   clk=1
 Time=     390.0ns
   clk=0
   Testing output operand for instruction          16
     Your WD =    00000000000000000000000000010000
     Correct WD = 00000000000000000000000000010000
 ---------YOU DID IT!! SIMULATION SUCCESFULLY FINISHED----------
relaunch_sim: Time (s): cpu = 00:00:00 ; elapsed = 00:00:08 . Memory (MB): peak = 1671.078 ; gain = 0.000
```

Conclusion

The simplified 3 stage pipelined processor is a simple design that allows for simple operations to be computed by the processor. The overall design of the pipelined processor allows for easy instruction handling and operations with multiple instructions running through the pipeline at any given time. Overall, the efficient design of these processors is the reason why they are so coveted to this day. While the design here is simple and lacks some other components such as instruction memory and an instruction counter, the basic design serves as a fundamental building block for the main processor.