Midterm Design Report
ECE 437

By

Xiao Xiao

Zhuofan Li

TA: Nitin Rathi, Alec Funke

Due date: March 3rd 2017

Overview

The purpose of this report is to provide a comparison of the single cycle and pipeline processors. The performance of each design will be determined by the max possible frequency, the average instructions per clock cycle, the latency of one instruction, total execution time of a specific program as well as FPGA resources required for each design. All this information will be collected from the log files generated along with the mapped designs. The benefits of the single cycle design include lower instruction latency, simpler design and lower power consumption due to less number of combination logics compared with pipelined design. The benefits of the pipelined design are that it can run at a higher frequency with a higher throughput and the latency for each stage is much lower than in single cycle.

The program selected to test both designs is mergesort.asm which covers a wide variety of all MIPS instructions and contains a fair amount of dependencies between instructions. As a conclusion, pipelined design improves the throughput and allows larger number of instructions to be executed at one time. The pipelined design also has a shorter critical path time and shorter execution time. Overall, the pipelined design works better on the merge sort program than the single cycle does.

Processor Design

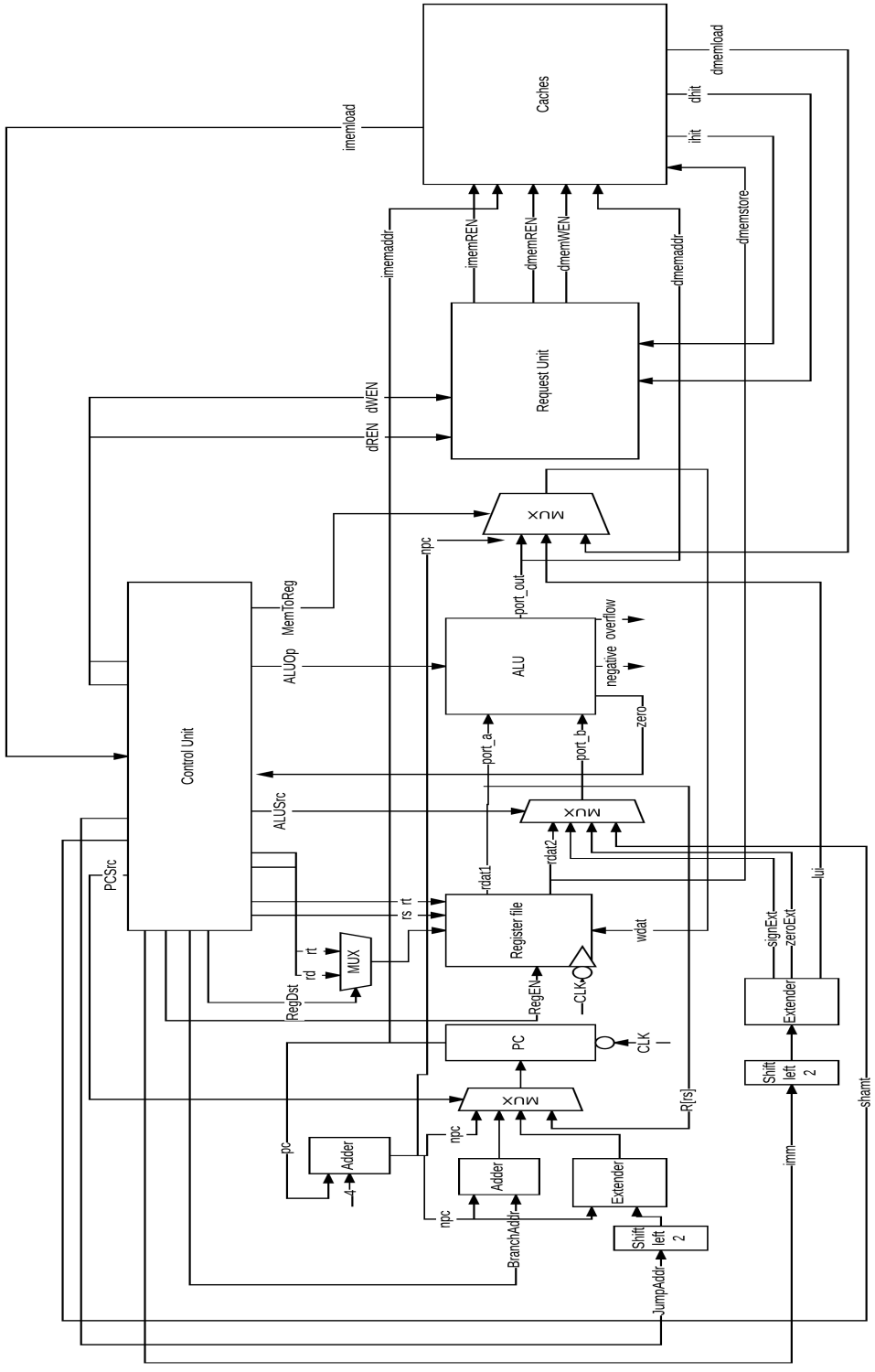

Figure 1: Single Cycle design

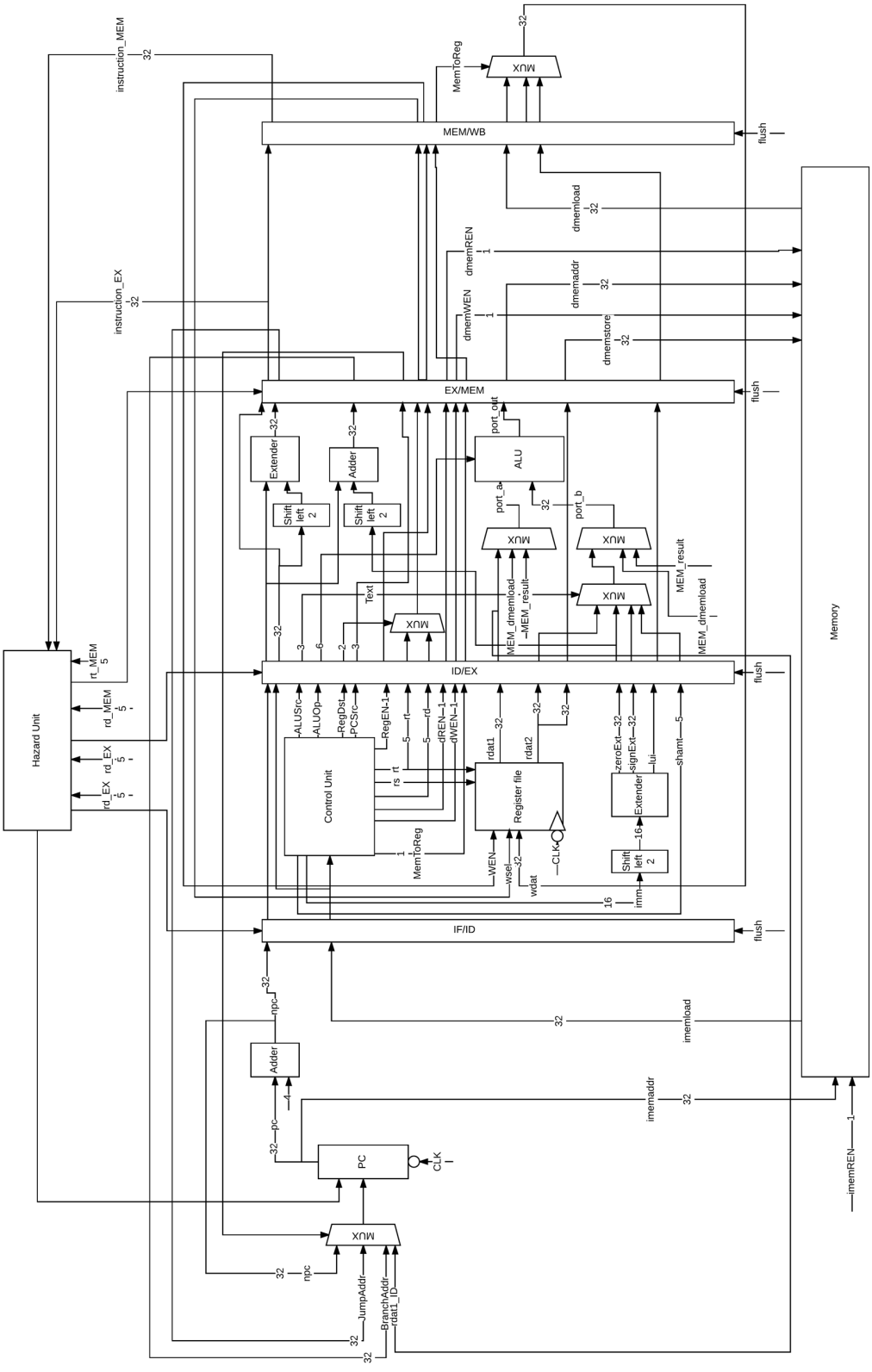

Figure 2: Pipelined design

Result

	Single Cycle	Pipeline	Calculation
CLK Frequency	34.94 MHz	61.21 MHz	Fmax
CLK Period	28.620ns	16.337ns	1/Fmax
Length of critical path	52.242ns	27.675ns	period - slack
Latency	28.620ns	81.685ns	stage*period
Execution time of merge sort	394698ns	310027ns	cycles*period
Average instruction per cycle	0.3914872018	0.2845922724	instructions/cycles
FPGA resources			
Total logic elements	3,229 / 114,480 (3%)	4,209 / 114,480 (4%)	
Total combinational functions	2,896 / 114,480 (3%)	3,857 / 114,480 (3%)	
Dedicated logic registers	1,278 / 114,480 (1%)	2,092 / 114,480 (2%)	
Total registers	1279	2093	
Total pins	102 / 529 (19%)	102 / 529 (19%)	
Total memory bits	524,288 / 3,981,312 (13%)	524,288 / 3,981,312 (13%)	

Table1. Comparison metrics

The frequency of the designs obtained from the max possible frequency from logs generated along with the mapped version with gate timing. The average instructions per clock cycle obtained by the total number of instructions obtained from running ‘sim’ and total number of cycles by running ‘system.sim’. The latency of one instruction is calculated by the stage number times the period of the design. Total execution time of the program is calculated by the total number of cycles times the period of each design. FPGA resources required are obtained from the Fitter Summary section in the log along with the synthesized version of design.

Conclusions

Overall, the pipelined design is better than the single cycle design in a few data we collected. The max frequency of pipelined is almost doubled than the single cycle version. Since the pipelined design is separated into several stages, the critical path is shorter. The execution time for merge sort in faster in the pipelined design.
However, the instruction latency of the pipeline design is longer since it takes five clock cycles for an instruction to traverse through the entire pipeline. The pipelined design needs slightly more FPGA resources than single cycle since pipelined design has hazard unit, forwarding and prediction which all take resources.

Contributions

Xiao Xiao

- Drew and edited the block diagram for the pipelined design
- Helped design and implement the pipeline design
- Debugged the pipelined design
- Finished the report and collecting results

Zhuofan Li

- Implemented the pipelined design
- Designed and implemented the hazard unit along with forwarding
- Implemented the 2bits predictor
- Finished the report and collecting results