Lab 3 - Microprocessor Pipeline and Memory Caching 1: The Basics

Name: Jason Xie
NSID: xix277
Student Number: 11255431

2-Stage Pipelining

microprocessor_slow_rom

	Resource	Usage
1	Estimated Total logic elements	1,188
2		
3	Total combinational functions	1167
4	▾ Logic element usage by number of LUT inputs	
1	-- 4 input functions	107
2	-- 3 input functions	36
3	-- <=2 input functions	1024
5		
6	▾ Logic elements by mode	
1	-- normal mode	1139
2	-- arithmetic mode	28
7		
8	▾ Total registers	50
1	-- Dedicated logic registers	50
2	-- I/O registers	0
9		
10	I/O pins	30
11	Total memory bits	2112
12		
13	Embedded Multiplier 9-bit elements	0
14		
15	Maximum fan-out node	clk~input
16	Maximum fan-out	62
17	Total fan-out	1879
18	Average fan-out	1.46

Figure 1: Microprocessor Slow Rom Resources

	Fmax	Restricted Fmax	Clock Name	Note
1	16.47 MHz	16.47 MHz	clk	

Figure 2: Fmax at 85C

	Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
1	-29.859	program_memory:pr...orta_address_reg0	Instruction_decoder:instr_decoder|ir[5]	clk	clk	0.500	-0.521	29.836
2	-29.849	program_memory:pr...orta_address_reg0	Instruction_decoder:instr_decoder|ir[6]	clk	clk	0.500	-0.521	29.826

Figure 3: Critical Path at 85C

	Fmax	Restricted Fmax	Clock Name	Note
1	17.84 MHz	17.84 MHz	clk	

Figure 4: Fmax at 0C

Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
-27.524	program_memory:prog_mem|altsyncram:al...rated|ram_block1a0~porta_address_reg0	Instruction_decoder:instr_decoder|ir[5]	clk	clk	0.500	-0.450	27.573
-27.520	program_memory:prog_mem|altsyncram:al...rated|ram_block1a0~porta_address_reg0	Instruction_decoder:instr_decoder|ir[6]	clk	clk	0.500	-0.450	27.569
-27.196	program_memory:prog_mem|altsyncram:al...rated|ram_block1a0~porta_address_reg0	Instruction_decoder:instr_decoder|ir[4]	clk	clk	0.500	-0.450	27.245

Figure 5: Critical Path at 0C

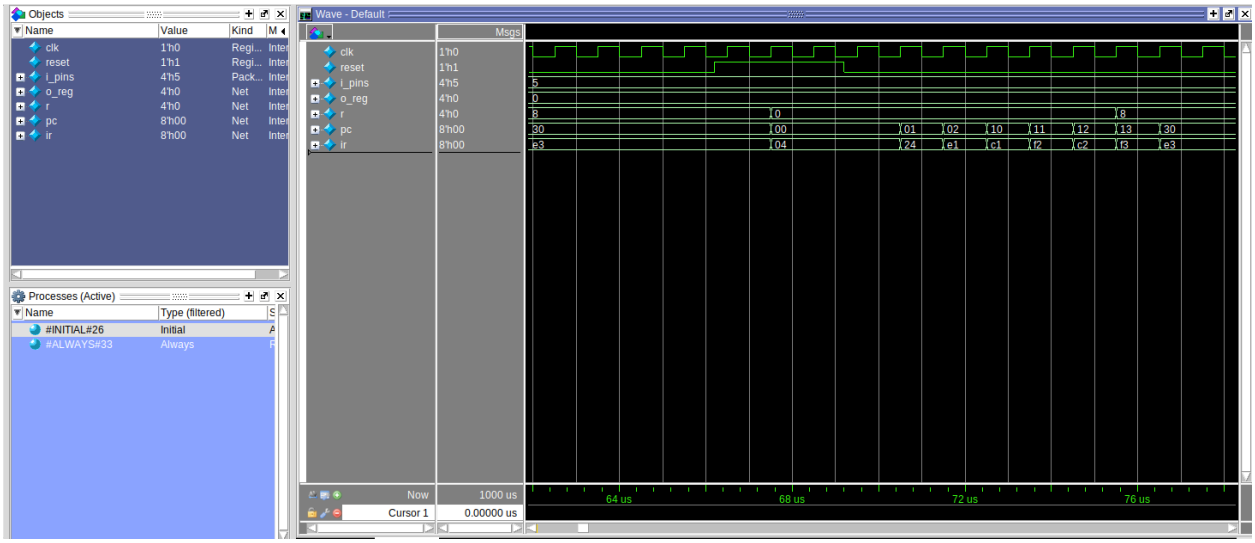

Figure 6: Microprocessor Slow Rom Simulation

microprocessor_srom_pipe1

	Resource	Usage
1	Estimated Total logic elements	1,203
2		
3	Total combinational functions	1174
4	▾ Logic element usage by number of LUT inputs	
1	-- 4 input functions	112
2	-- 3 input functions	42
3	-- <=2 input functions	1020
5		
6	▾ Logic elements by mode	
1	-- normal mode	1146
2	-- arithmetic mode	28
7		
8	▾ Total registers	58
1	-- Dedicated logic registers	58
2	-- I/O registers	0
9		
10	I/O pins	47
11	Total memory bits	2112
12		
13	Embedded Multiplier 9-bit elements	0
14		
15	Maximum fan-out node	clk~input
16	Maximum fan-out	70
17	Total fan-out	1959
18	Average fan-out	1.46

Figure 7: Microprocessor SROM Pipeline 1 Resources

	Fmax	Restricted Fmax	Clock Name	Note
1	17.33 MHz	17.33 MHz	clk	

Figure 8: Fmax at 85C

	Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
1	-29.174	data_pipe[2]~reg0	Instruction_decoder:instr_decoder|ir[2]	clk	clk	1.000	0.044	30.216
2	-28.775	data_pipe[6]~reg0	Instruction_decoder:instr_decoder|ir[6]	clk	clk	1.000	-0.376	29.397

Figure 9: Critical Path at 85C

	Fmax	Restricted Fmax	Clock Name	Note
1	18.74 MHz	18.74 MHz	clk	

Figure 10: Fmax at 0C

	Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
1	-26.938	data_pipe[2]~reg0	Instruction_decoder:instr_decoder|ir[2]	clk	clk	1.000	0.055	27.992
2	-26.581	data_pipe[6]~reg0	Instruction_decoder:instr_decoder|ir[6]	clk	clk	1.000	-0.332	27.248

Figure 11: Critical Path at 0C

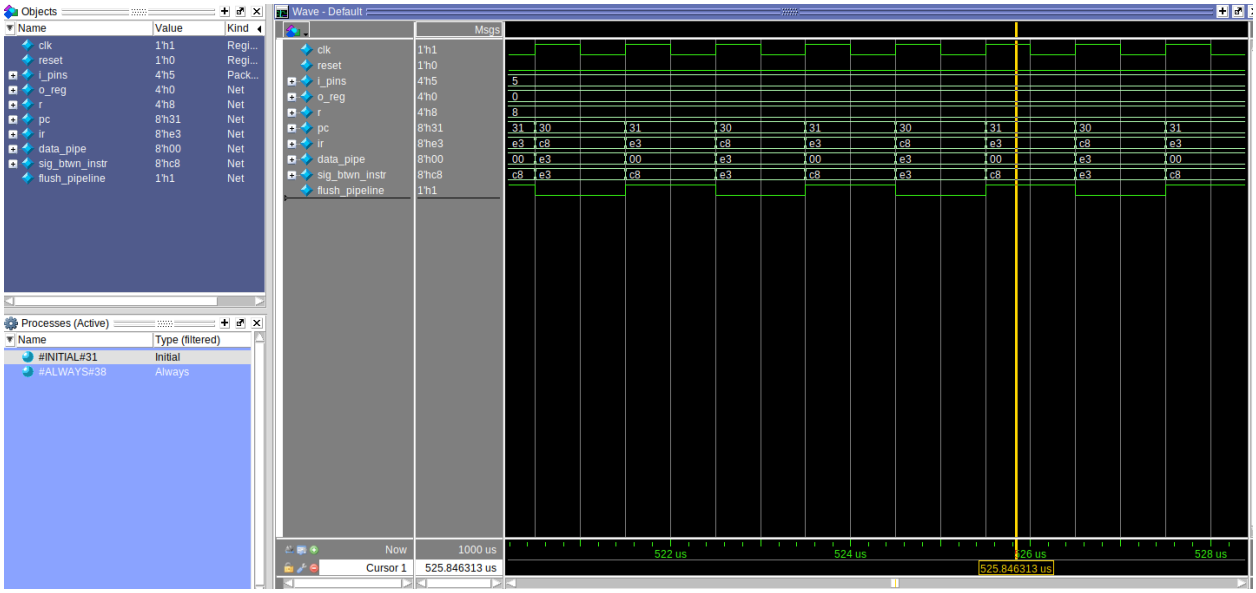

Figure 12: Microprocessor SROM Pipe 1 Simulation

microprocessor_srom_pipe2

	Resource	Usage
1	Estimated Total logic elements	1,211
2		
3	Total combinational functions	1182
4	▾ Logic element usage by number of LUT inputs	
1	-- 4 input functions	106
2	-- 3 input functions	49
3	-- <=2 input functions	1027
5		
6	▾ Logic elements by mode	
1	-- normal mode	1154
2	-- arithmetic mode	28
7		
8	▾ Total registers	59
1	-- Dedicated logic registers	59
2	-- I/O registers	0
9		
10	I/O pins	56
11	Total memory bits	2112
12		
13	Embedded Multiplier 9-bit elements	0
14		
15	Maximum fan-out node	clk~input
16	Maximum fan-out	71
17	Total fan-out	1986
18	Average fan-out	1.45

Figure 13: Microprocessor SROM Pipe 2 Resources

	Fmax	Restricted Fmax	Clock Name	Note
1	18.75 MHz	18.75 MHz	clk	

Figure 14: Fmax at 85C

Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
-28.247	data_pipe[6]~reg0	Instruction_decoder:instr_decoder|ir[6]	clk	clk	1.000	-0.308	28.937
-28.157	data_pipe[1]~reg0	Instruction_decoder:instr_decoder|ir[1]	clk	clk	1.000	-0.304	28.851

Figure 15: Critical Path at 85C

	Fmax	Restricted Fmax	Clock Name	Note
1	20.23 MHz	20.23 MHz	clk	

Figure 16: Fmax at 0C

	Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
1	-26.043	data_pipe[6]~reg0	Instruction_decoder:instr_decoder|ir[6]	clk	clk	1.000	-0.262	26.780
2	-25.921	data_pipe[0]~reg0	Instruction_decoder:instr_decoder|ir[0]	clk	clk	1.000	-0.259	26.661

Figure 17: Critical Path at 0C

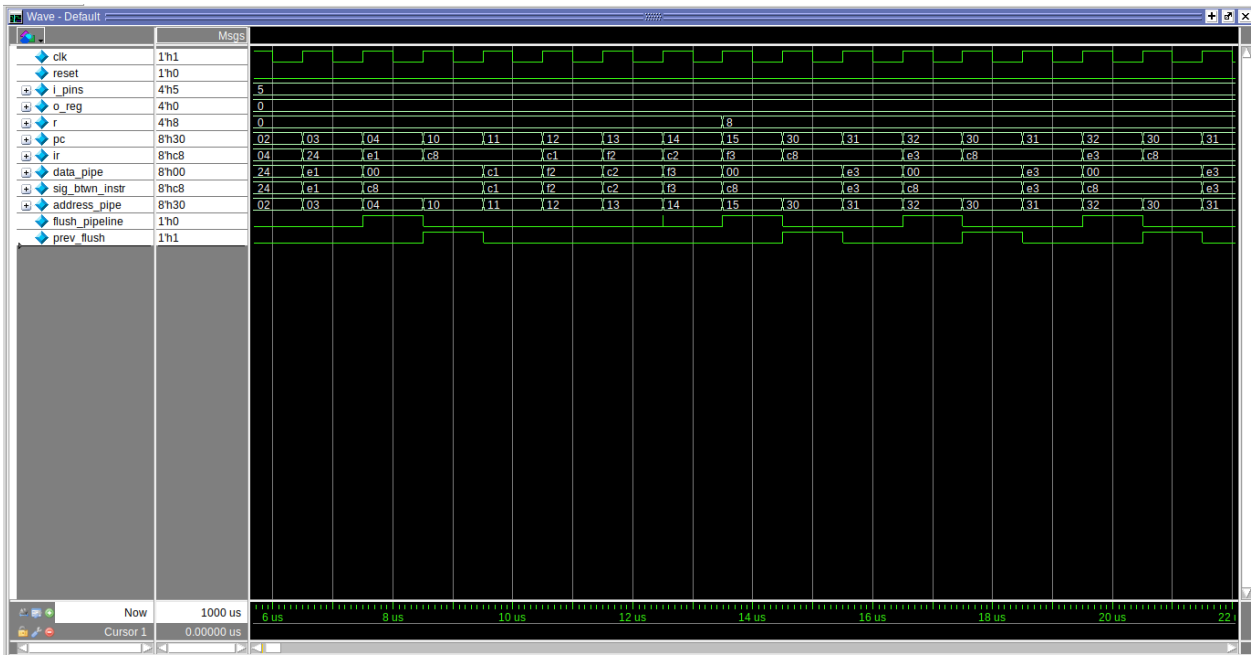

Figure 18: Microprocessor SROM Pipe 2 Simulation

See above figures for hardware utilization, max frequency, and critical path results.

The trade-off between hardware utilization and maximum frequency comes down to area vs speed. From the 3 versions of the microprocessor, adding in pipelining increased the number of logic elements used but also increased the speed of the microprocessor. For example, comparing the Slow ROM to the SROM Pipe 2, the Pipe 2 microprocessor frequency increased by about 2 MHz while also increasing the number of logic elements used by 23.

The advantages of adding pipelining is that it increases the speed of the microprocessor making it more efficient and also adds in the aspect of parallelism meaning that multiple instructions can be in different stages. The disadvantages however, is that with the introduction of pipelining, more resources are added. This increases the area of the microprocessor and also increases the power consumption due to more logic elements

Suspending the Microprocessor

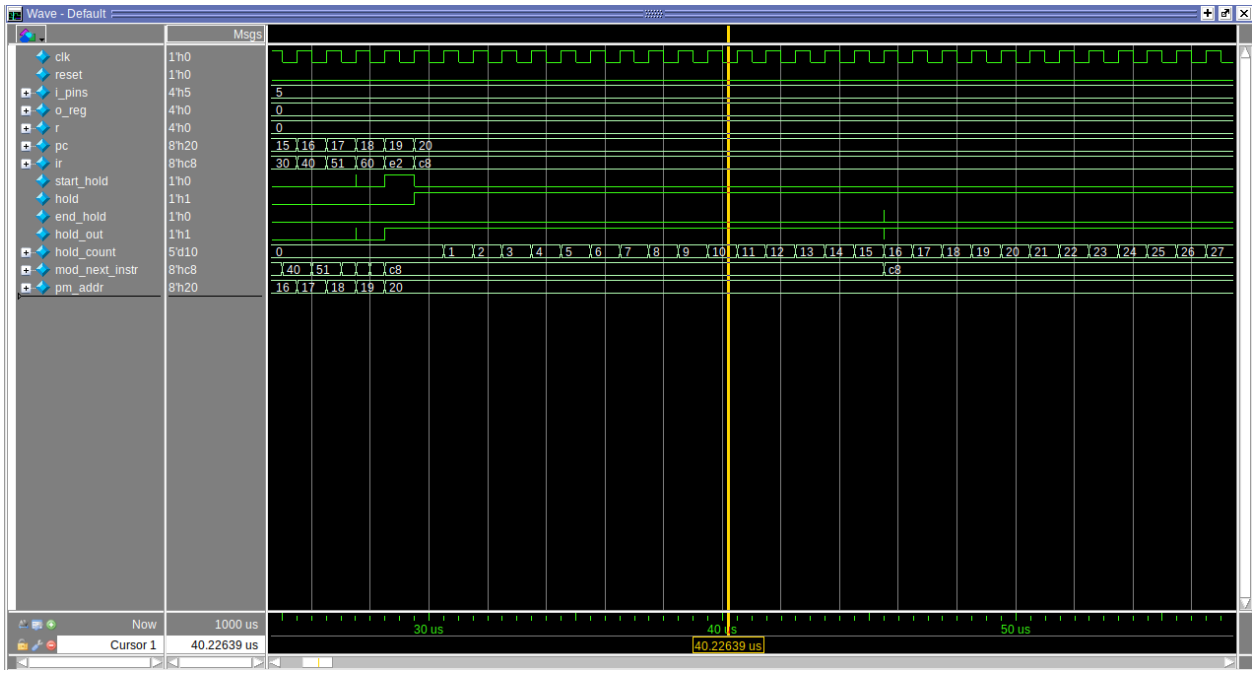

Figure 20: Microprocessor Suspension - Partial Waveform

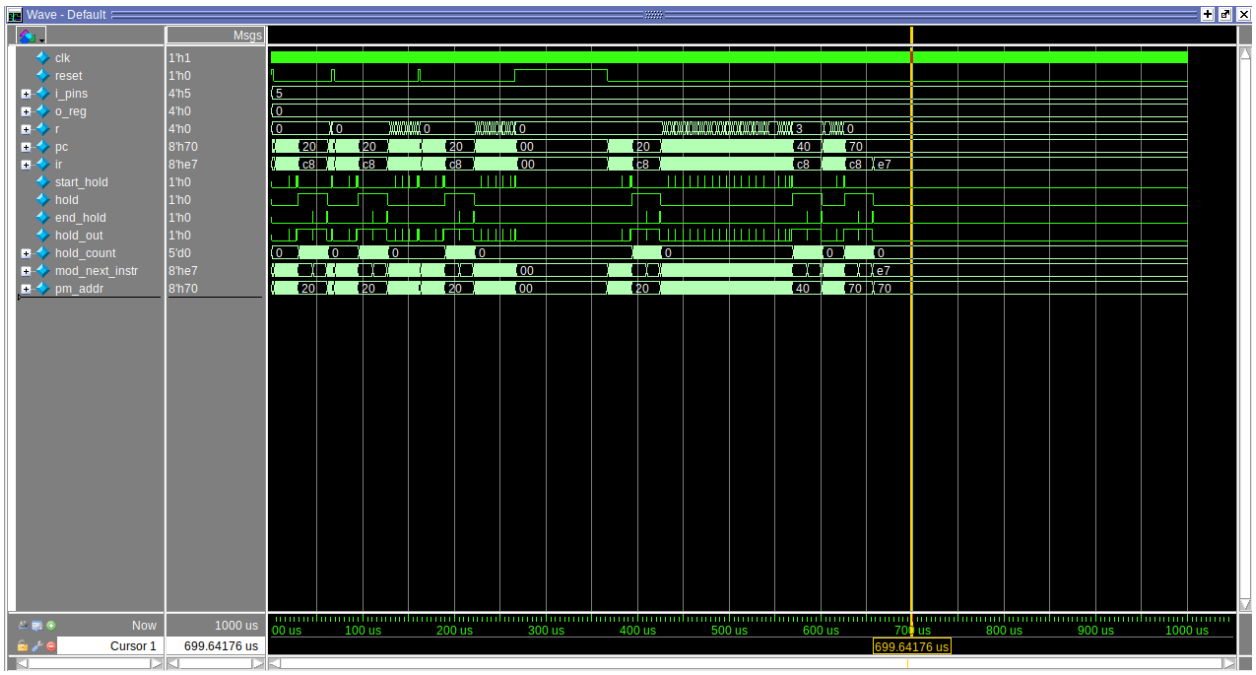

Figure 21: Microprocessor Suspension - Full Waveform

From the computational test file, the microprocessor should suspend when the 3 MSB's of pc and pm_addr do not match, and this happens when the instruction is a jump (E2). From the

partial waveform, it can be seen that the microprocessor does suspend when the jump instruction is hit.

The cache size should be 256 bits or 32 bytes since you suspend the 8-bit microprocessor for 32 cycles. Therefore the cache size should be 32 cycles * 8-bits/cycle = 256 bits or 32 bytes.