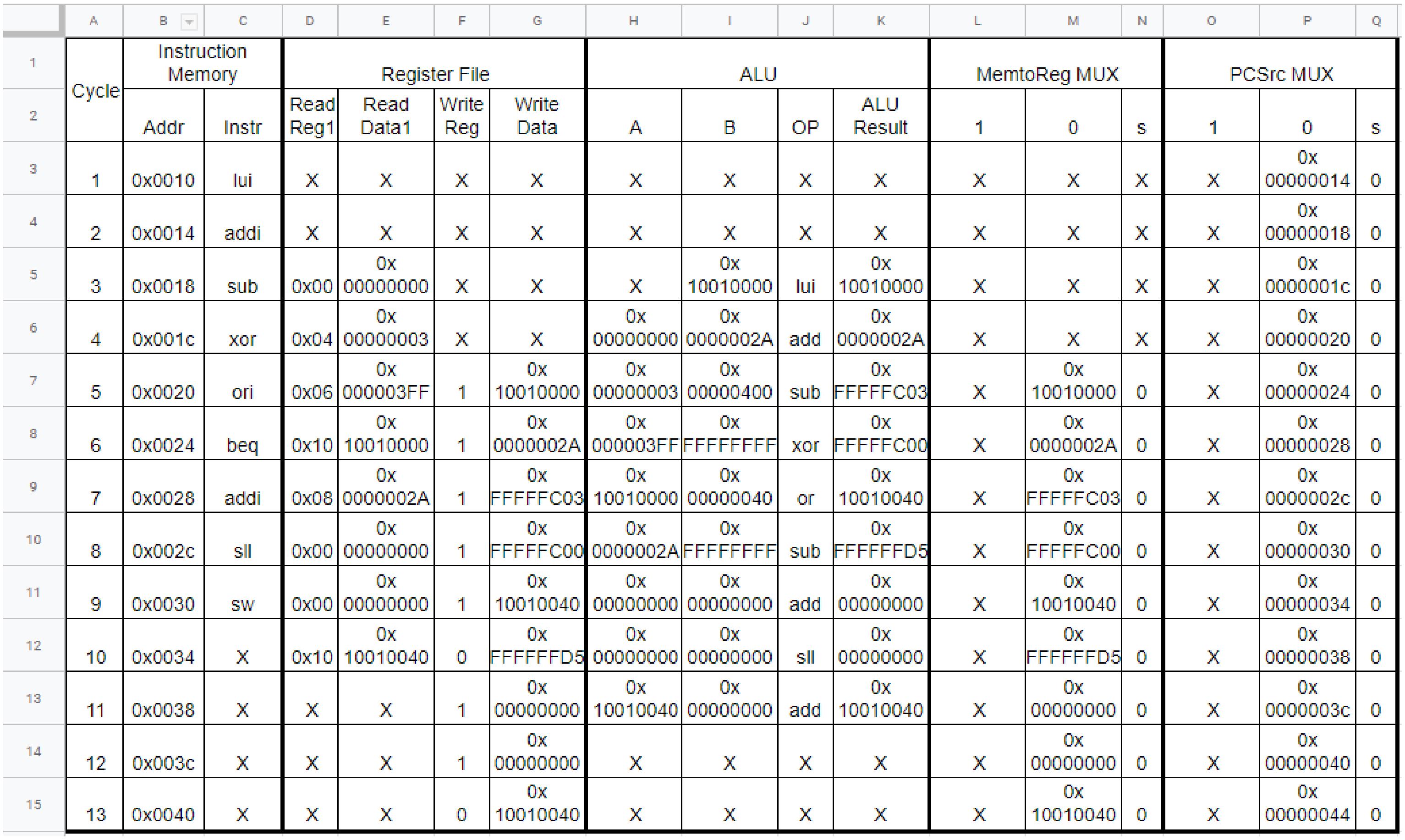
Sean Gordon
Sgordon4

1)

The cycle time for the processor will be 200ps, as the section that takes the most time includes I-Mem and a Mux, needing 200ps and 20ps respectively. When the section is used, I-Mem will take 200ps to output correctly, and with the mux operating in parallel its own time-cost will be far overshadowed by I-Mem, resulting in a total cost of 200ps. This section sets the cycle time because if the cycle time was any shorter than the maximum of necessary times of all sections, some of the sections are not guaranteed to output correct values.

Sean Gordon
Sgordon4

2)



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Cycle	Instruction Memory		Register File				ALU				MemtoReg MUX			PCSrc MUX		
2		Addr	Instr	Read Reg1	Read Data1	Write Reg	Write Data	A	B	OP	ALU Result	1	0	s	1	0	s
3	1	0x0010	lui	X	X	X	X	X	X	X	X	X	X	X	X	0x 00000014	0
4	2	0x0014	addi	X	X	X	X	X	X	X	X	X	X	X	X	0x 00000018	0
5	3	0x0018	sub	0x00	0x 00000000	X	X	X	0x 10010000	lui	0x 10010000	X	X	X	X	0x 0000001c	0
6	4	0x001c	xor	0x04	0x 00000003	X	X	0x 00000000	0x 0000002A	add	0x 0000002A	X	X	X	X	0x 00000020	0
7	5	0x0020	ori	0x06	0x 000003FF	1	0x 10010000	0x 00000003	0x 00000400	sub	0x FFFFFC03	X	0x 10010000	0	X	0x 00000024	0
8	6	0x0024	beq	0x10	0x 10010000	1	0x 0000002A	0x 000003FF	0x FFFFFFFF	xor	0x FFFFFC00	X	0x 0000002A	0	X	0x 00000028	0
9	7	0x0028	addi	0x08	0x 0000002A	1	0x FFFFFC03	0x 10010000	0x 00000040	or	0x 10010040	X	0x FFFFFC03	0	X	0x 0000002c	0
10	8	0x002c	sll	0x00	0x 00000000	1	0x FFFFFC00	0x 0000002A	0x FFFFFFFF	sub	0x FFFFFFD5	X	0x FFFFFC00	0	X	0x 00000030	0
11	9	0x0030	sw	0x00	0x 00000000	1	0x 10010040	0x 00000000	0x 00000000	add	0x 00000000	X	0x 10010040	0	X	0x 00000034	0
12	10	0x0034	X	0x10	0x 10010040	0	0x FFFFFFD5	0x 00000000	0x 00000000	sll	0x 00000000	X	0x FFFFFFD5	0	X	0x 00000038	0
13	11	0x0038	X	X	X	1	0x 00000000	0x 10010040	0x 00000000	add	0x 10010040	X	0x 00000000	0	X	0x 0000003c	0
14	12	0x003c	X	X	X	1	0x 00000000	X	X	X	X	X	0x 00000000	0	X	0x 00000040	0
15	13	0x0040	X	X	X	0	0x 10010040	X	X	X	X	X	0x 10010040	0	X	0x 00000044	0

^^

As D-Mem stage left out, MemtoReg MUX appears skewed by one stage

3)
Concerning the MIPS Single Cycle Processor, it is commonly stated that the ‘lw’ instruction takes the critical path. One of your friends in CprE 381 was out on an important spy mission the week it was explained, and asks you to help them understand what they missed:

a) What is a “critical path” in the context of CprE 381, and what does it have to do with ‘lw’?

b) Outline how a critical path is determined.

c) Describe one thing that could be done to reduce the impact of, or even change the critical path.

Answer:

a) A critical path is the path between the input and the output of the circuit with the maximum delay. Because each component in the MIPS processor carries with it some amount of propagation delay, there will always be delay when a signal attempts to propagate across a circuit. The path with the most delay is termed the ‘critical path’, and requires the clock period be at least that long, lest there be invalid outputs. The critical path in the MIPS Single Cycle Processor happens to occur as a result of the lw instruction.

b) A critical path can be found by starting at the input of the circuit in question (in the case of the MIPS processor, the PC register output), and tracing each path through the circuit, recording the amount of delay encountered per path. The path with the most delay is named the ‘critical path’.

c) As the critical path is dependant on the delays of the components that it passes through, one such method would be designing a faster version of a particular component. In the case of the lw instruction, designing a faster version of ‘I-Mem’ or ‘D-Mem’ would decrease the time needed to complete the path.

Sean Gordon
Sgordon4

4)
https://www.comsol.com/blogs/havent-cpu-clock-speeds-increased-last-years/
This article by, the Comsol blog, answers a question about why clock speeds in market cpu's are no longer steadily increasing over the years. Pär Persson Mattsson relates the halt in clock speed increase to Dennard Scaling, and the actual size of the circuitry themselves. Speaking about how as the transistor size has gotten smaller, the power required to run them has not. Packing more and more transistors into a small space has greatly increased power consumption, and as a result heat. The transistors themselves, with such minimal compositions, lack structural integrity, adding another constraint to the heap.

This article relates to the terms 'execution time' and 'instruction' when the author mentions a solution to the problem: multi-core processors. Mattsson notes that these processors could speed up execution time by distributing the instruction load across multiple processors. He also notes that as the transistors decrease in size, the 'power' they consume does not. As we have been learning about the constraints of pipelining a processor, this illustrates that pipelining alone isn't the end of the line for processors.