Single Cycle CPU Design

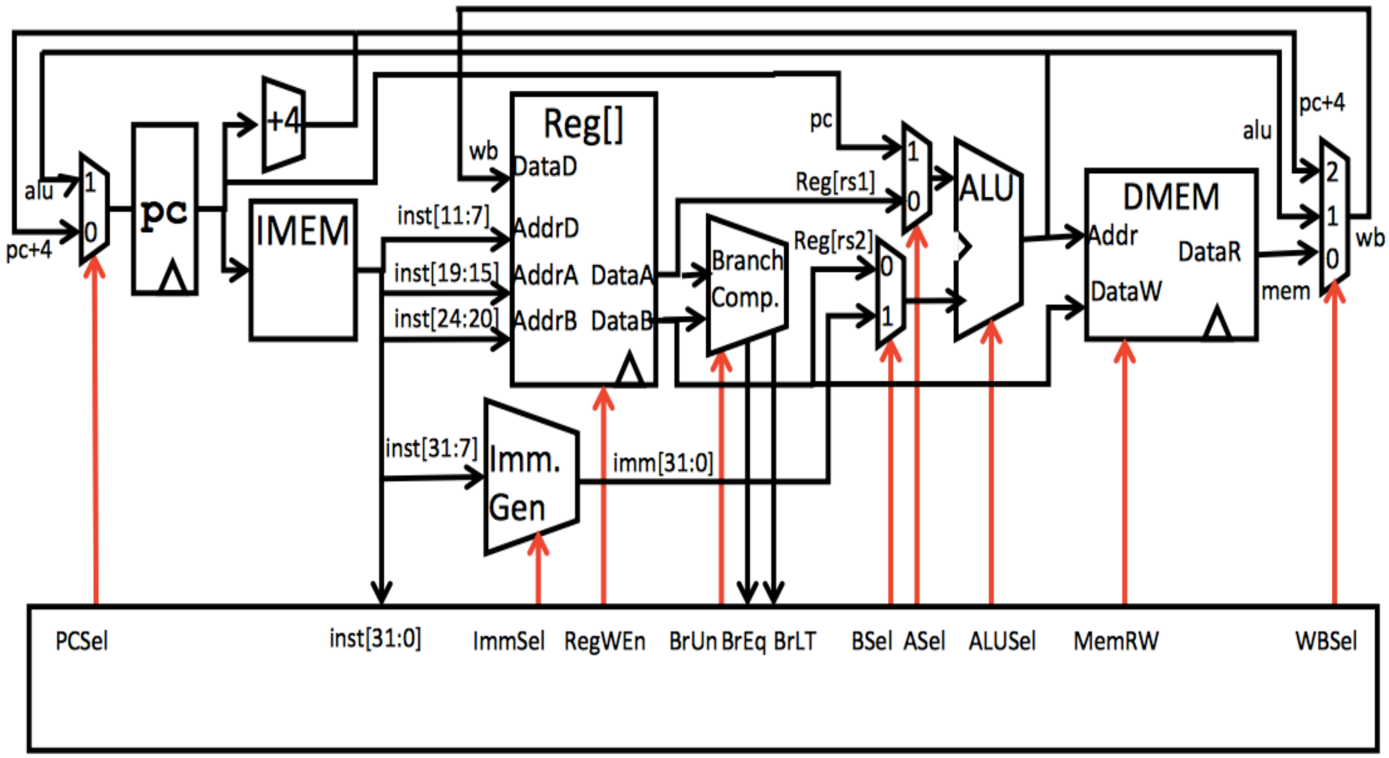
Here we have a single cycle CPU diagram. Answer the following questions:

1. Name each component.

2. Name each datapath stage and explain its functionality.

Stage	Functionality
Instruction Fetch	Send an address to the instruction memory Read the instruction (IMEM[PC])
Decode / Register Read	Generate the control signal values using the opcode & funct fields Read the register values with the relevant fields and generate the immediate
Execute	Perform arithmetic / logical operations and branch comparison
Memory	Read from / write to the data memory (DMEM)
Register Write	Write back the ALU result / the memory load / PC + 4 to the register file

3. Provide data inputs and control signals to the next PC logic.



Single Cycle CPU Control Logic

Fill out the values for the control signals from the previous CPU diagram.

Instrs.	Control Signals									
	BrEq	PCSel	ImmSel	BrUn	ASel	BSel	ALUSel	MemRW	RegWEn	WBSel
`add`	X	+4	X	X	Reg	Reg	Add	Read	1	ALU
`ori`	X	+4	I	X	Reg	Imm	Or	Read	1	ALU
`lw`	X	+4	I	X	Reg	Imm	Add	Read	1	MEM
`sw`	X	+4	S	X	Reg	Imm	Add	Write	0	X
`beq`	1	ALU	SB	X	PC	Imm	Add	Read	0	X
`jal`	X	ALU	UJ	X	PC	imm	Add	Read	1	PC + 4

(Put an X if the signal doesn't matter)

Clocking Methodology

- The input signal to each state element must stabilize before each rising edge.
- Critical path: Longest delay path between state elements in the circuit.
- If we place registers in the critical path, we can shorten the period by reducing the amount of logic between registers.

Single Cycle CPU Performance Analysis

The delays of circuit elements are given as follows:

Stage	IF	ID	EX	MEM	WB
Delay	200 ps	100 ps	200 ps	200 ps	100 ps

1. Mark the stages the following instructions use and calculate the time to execute.

Instruction	IF	ID	EX	MEM	WB	Total
add	X	X	X		X	600 ps
lw	X	X	X	X	X	800 ps
sw	X	X	X	X		700 ps
xori	X	X	X		X	600 ps
beq	X	X	X			500 ps
jal	X	X	X		X	600 ps

2. Which instruction(s) exercises the critical path?

Load word (lw), as it both reads from memory and writes to register, which most instructions don't use both memory and writing to register.

3. What is the fastest you could clock this single-cycle datapath? (1 ps = 10^{-12} s)

1/(800 ps) = 1/(800 * 10^{-12}s) = 10^{12} / 800 = 1,250,000,000 = 1.25 GHz

4. Why is a single cycle CPU inefficient?

Not all instructions exercise the critical path, but we have to clock to the slowest instruction. It is not parallelized, but we can use each component concurrently

5. How can you improve its performance? What is the purpose of pipelining?

We can use pipelining, which involves putting registers between two stages → reducing the clock time for each stage (covered next week!)