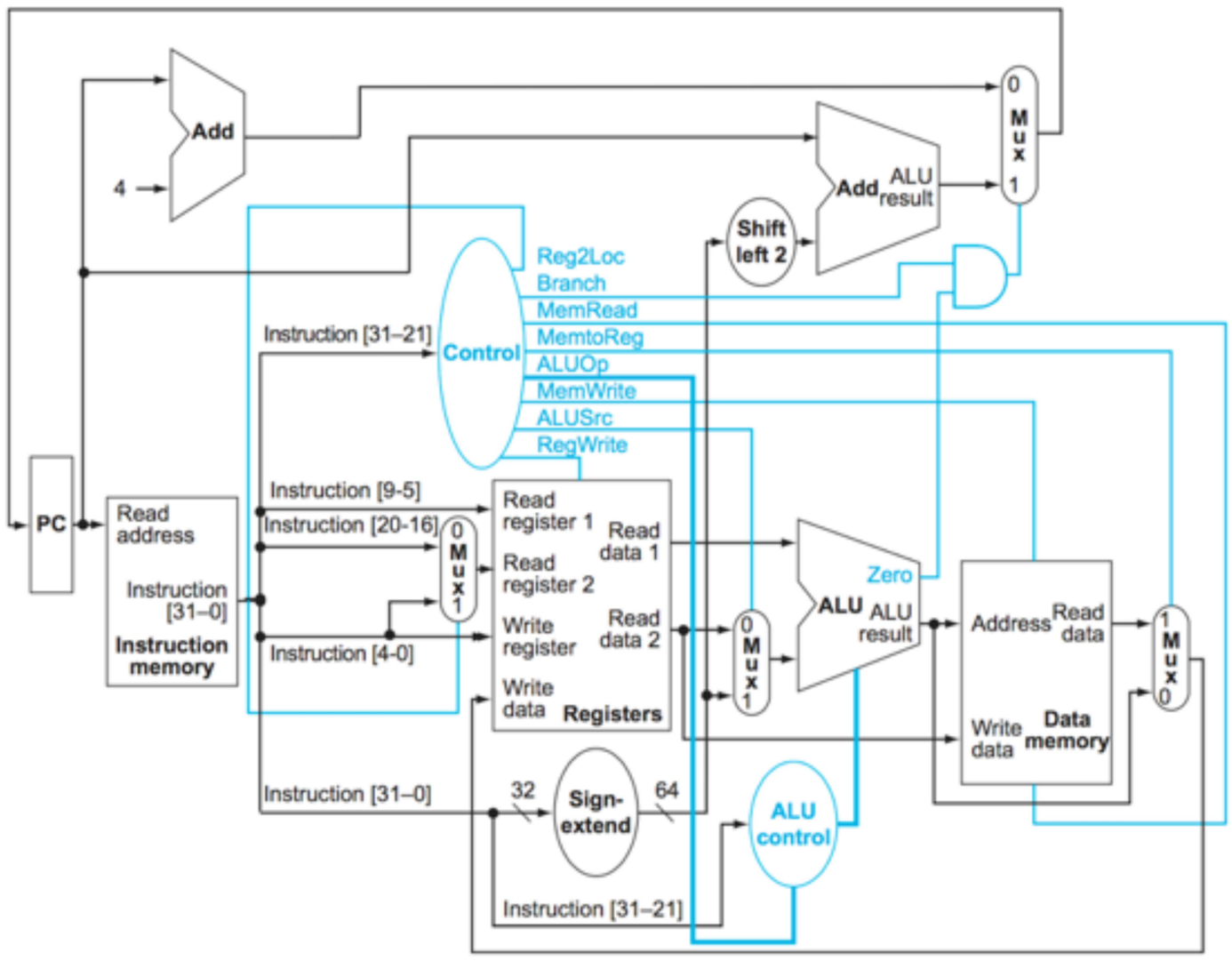
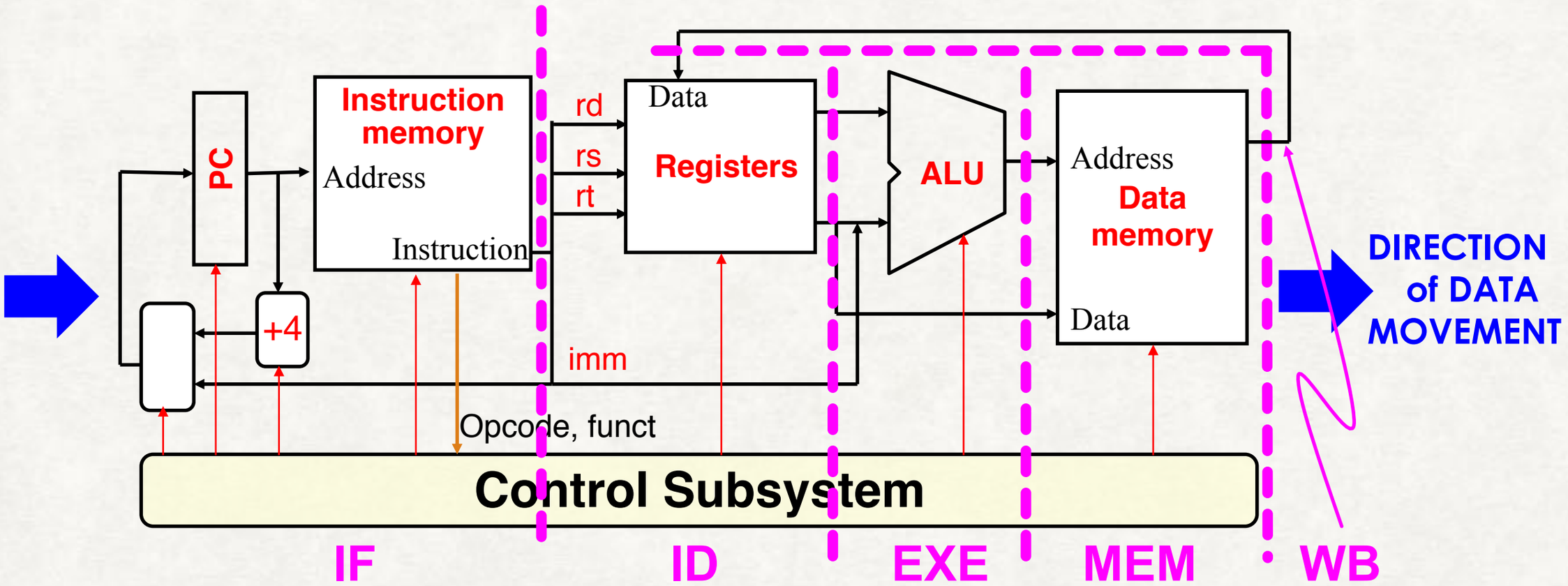


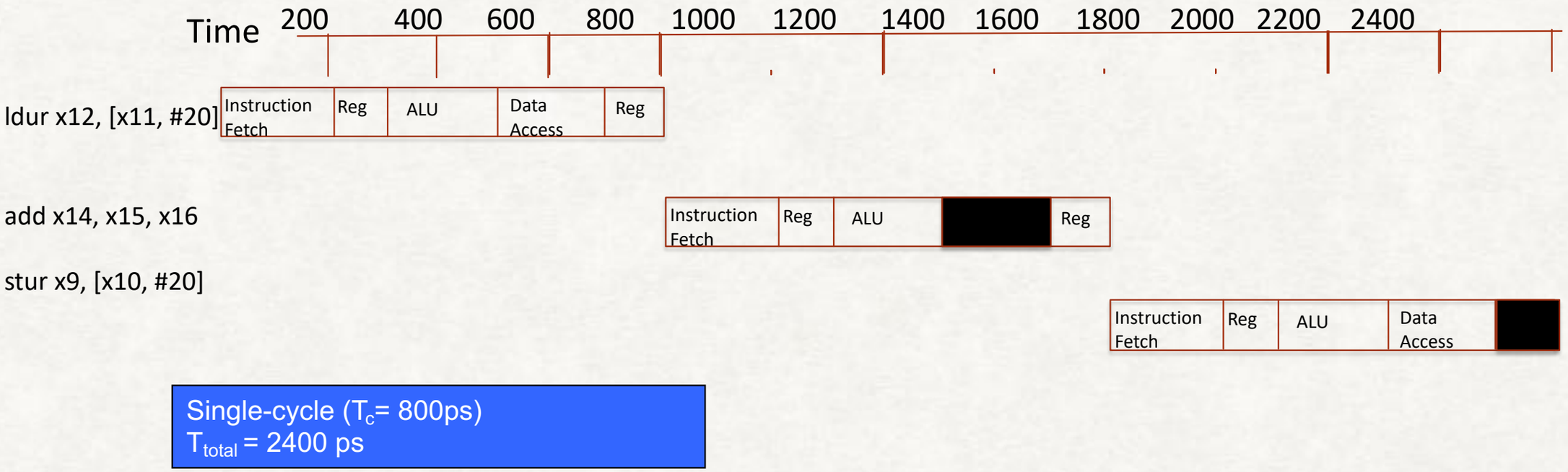
Instruction	Reg2Loc	ALUSrc	MemtoReg	RegWrite	MemRead	MemWrite	Branch	ALUOp1	ALUOp0
R-format	0	0	0	1	0	0	0	1	0
LDUR	X	1	1	1	1	0	0	0	0
STUR	1	1	X	0	0	1	0	0	0
CBZ	1	0	X	0	0	0	1	0	1

Signal name	Effect when deasserted	Effect when asserted
Reg2Loc	The register number for Read register 2 comes from the Rm field (bits 20:16).	The register number for Read register 2 comes from the Rt field (bits 4:0).
RegWrite	None.	The register on the Write register input is written with the value on the Write data input.
ALUSrc	The second ALU operand comes from the second register file output (Read data 2).	The second ALU operand is the sign-extended, lower 32 bits of the instruction.
PCSrc	The PC is replaced by the output of the adder that computes the value of PC + 4.	The PC is replaced by the output of the adder that computes the branch target.
MemRead	None.	Data memory contents designated by the address input are put on the Read data output.
MemWrite	None.	Data memory contents designated by the address input are replaced by the value on the Write data input.
MemtoReg	The value fed to the register Write data input comes from the ALU.	The value fed to the register Write data input comes from the data memory.

DATAPATH IN STAGES

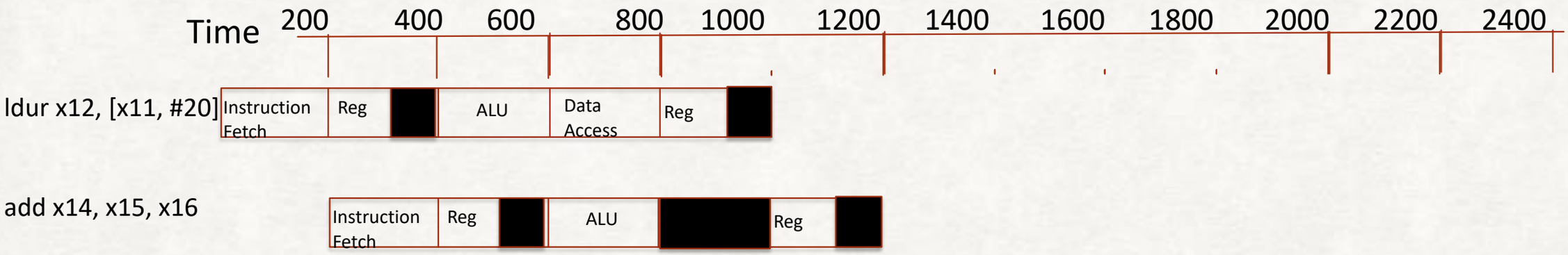


SINGLE CYCLE TIME TO EXECUTE



Single-cycle (T_c= 800ps)
T_{total} = 2400 ps

PIPELINED TIME TO EXECUTE



Pipelined-cycle (T_c= 200ps)
T_{total} = 1400 ps

The cycle time is the length of the longest stage

Cycle time = 200 ps

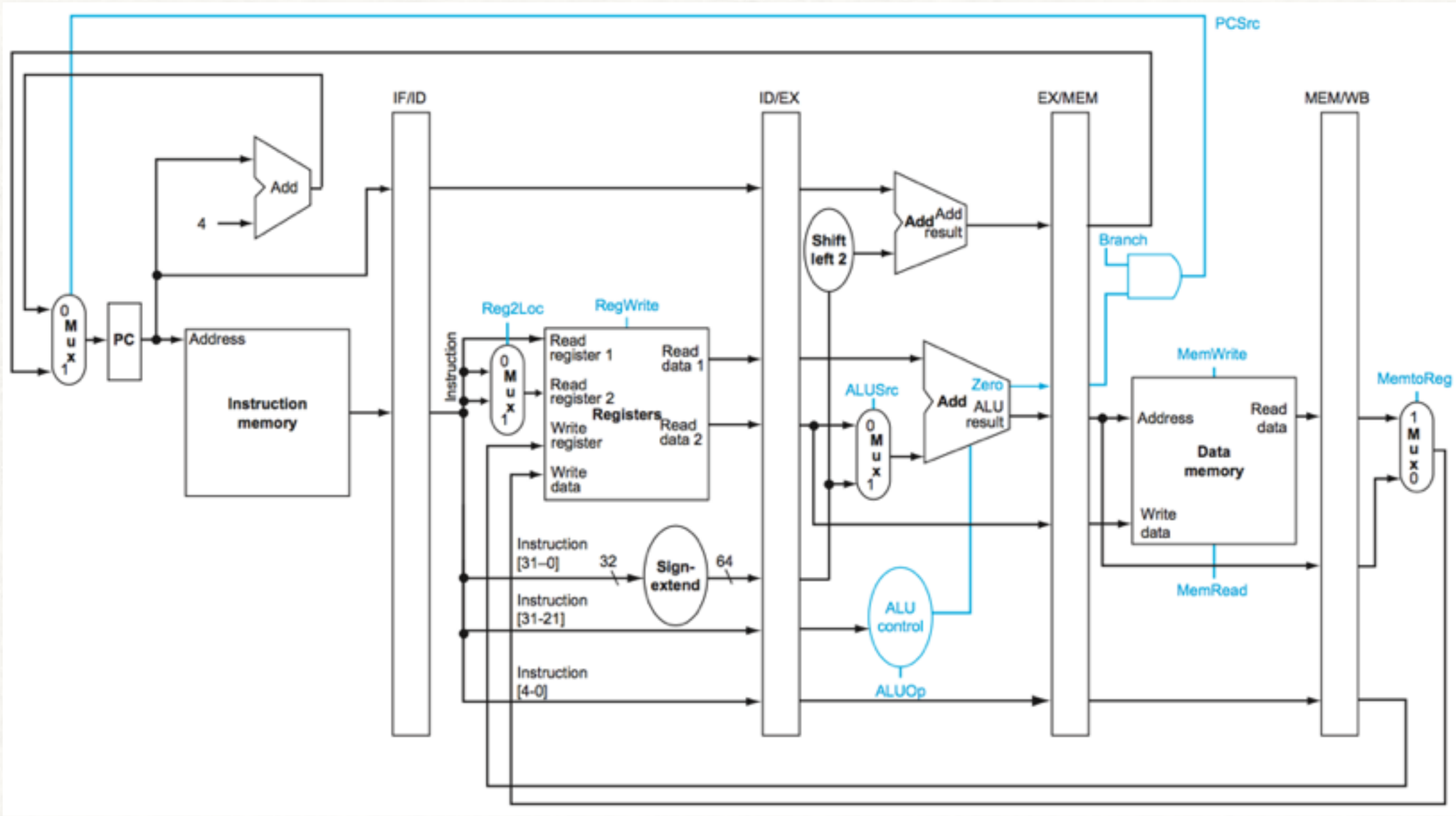
3 instructions took 7 cycles

EXAMPLE PROBLEM

A single cycle processor with a 1500 ps clock cycle can be split into 5 stages with execution times of 300 ps, 200 ps, 250 ps, 400 ps, and 350 ps.

What is the pipelined cycle time?
How long does it take for one instruction to get through the pipeline?
How long does it take for 6 instructions to get through the pipeline, assuming no hazards?
What is the speedup for processing 6 instructions?

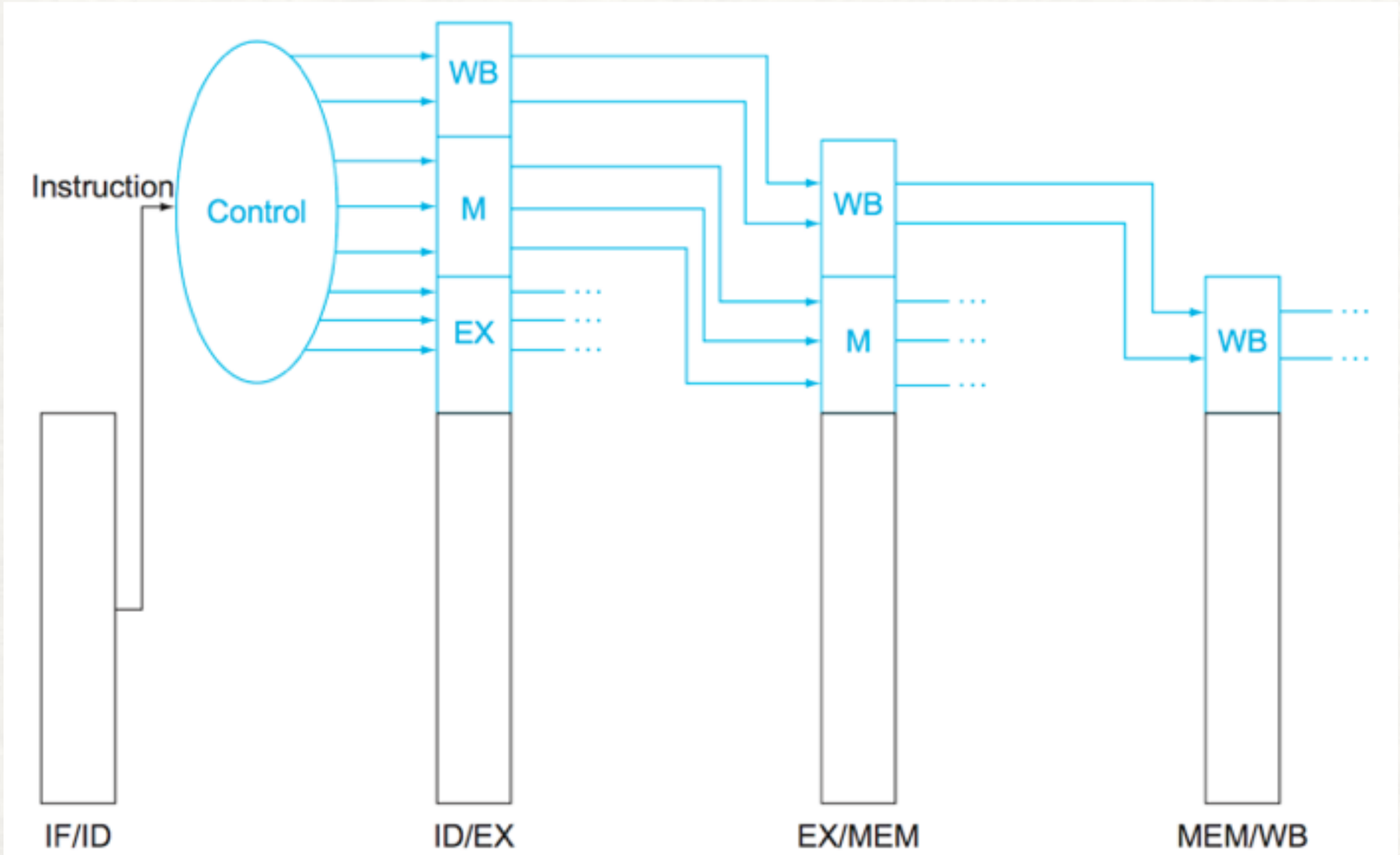
PIPELINE CONTROL SIGNALS



PIPELINE CONTROL SIGNALS

Instruction	Instruction decode stage control lines	Execution/address calculation stage control lines			Memory access stage control lines			Write-back stage control lines	
Instruction	Reg2Loc	ALUOp1	ALUOp0	ALUSrc	Branch	MemRead	MemWrite	RegWrite	MemtoReg
R-format	0	1	0	0	0	0	0	1	0
LDUR	X	0	0	1	0	1	0	1	1
STUR	1	0	0	1	0	0	1	0	X
CBZ	1	0	1	0	1	0	0	0	X

PIPELINE CONTROL SIGNALS



HOW IT WORKS: The Control Signals are generated in the ID stage Control Signals are stored in inter-stage buffers (e.g., ID/EX) Each stage (e.g., EX or MEM) uses the control signals it needs