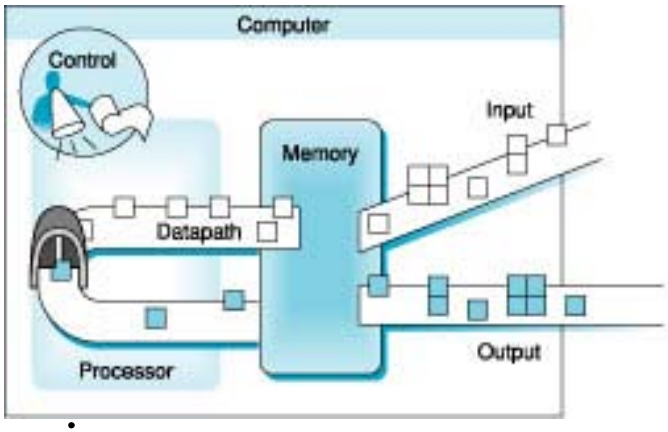
Lecture 8: Processor design – single cycle

- Motivation:
  - Learn how to design a simple processor
- Two main parts:
  - Datapath: performs the data operations as commanded by the program instructions
  - Control: controls the datapath, memory and I/O according to the program instructions
- Using:
  - Combinational and sequential circuits described in previous lecture



Design steps / Lecture outline

- Step 1: Determine the components required
- Step 2: Build the datapath
- Step 3: Build the control

- Show the execution of 1-2 instructions on the designed machine

Determine the components

Processor task

- Instruction fetch from memory
- Read registers
- Execution
  - Data processing instructions
  - Data transfer instructions
  - Branch instructions

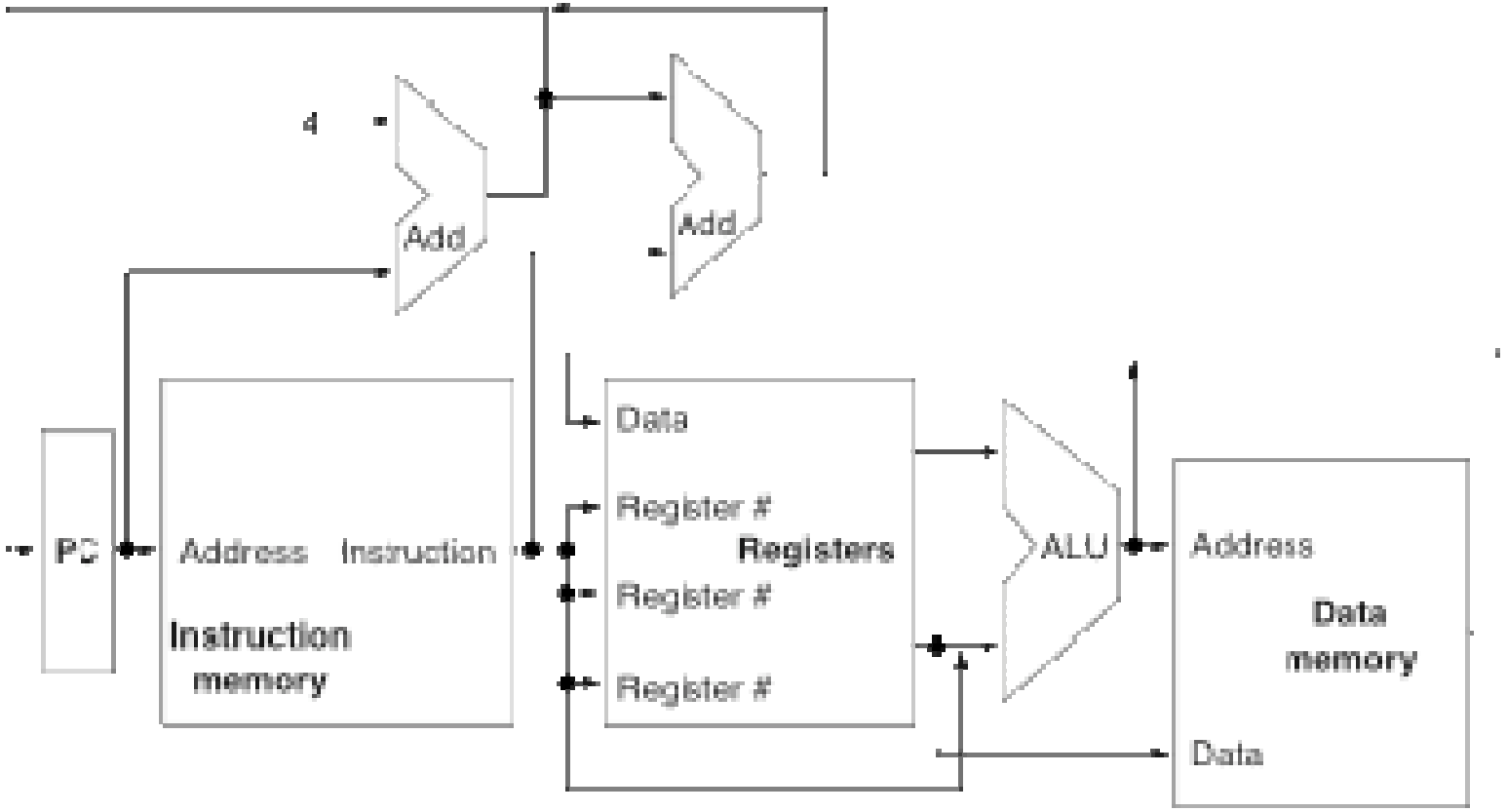
Component list

- PC register
- Memory (instructions)
- Adder: PC+4
- Register file
  - 2 read, 1 write
- ALU
- Memory (data)
- Adder: branch target

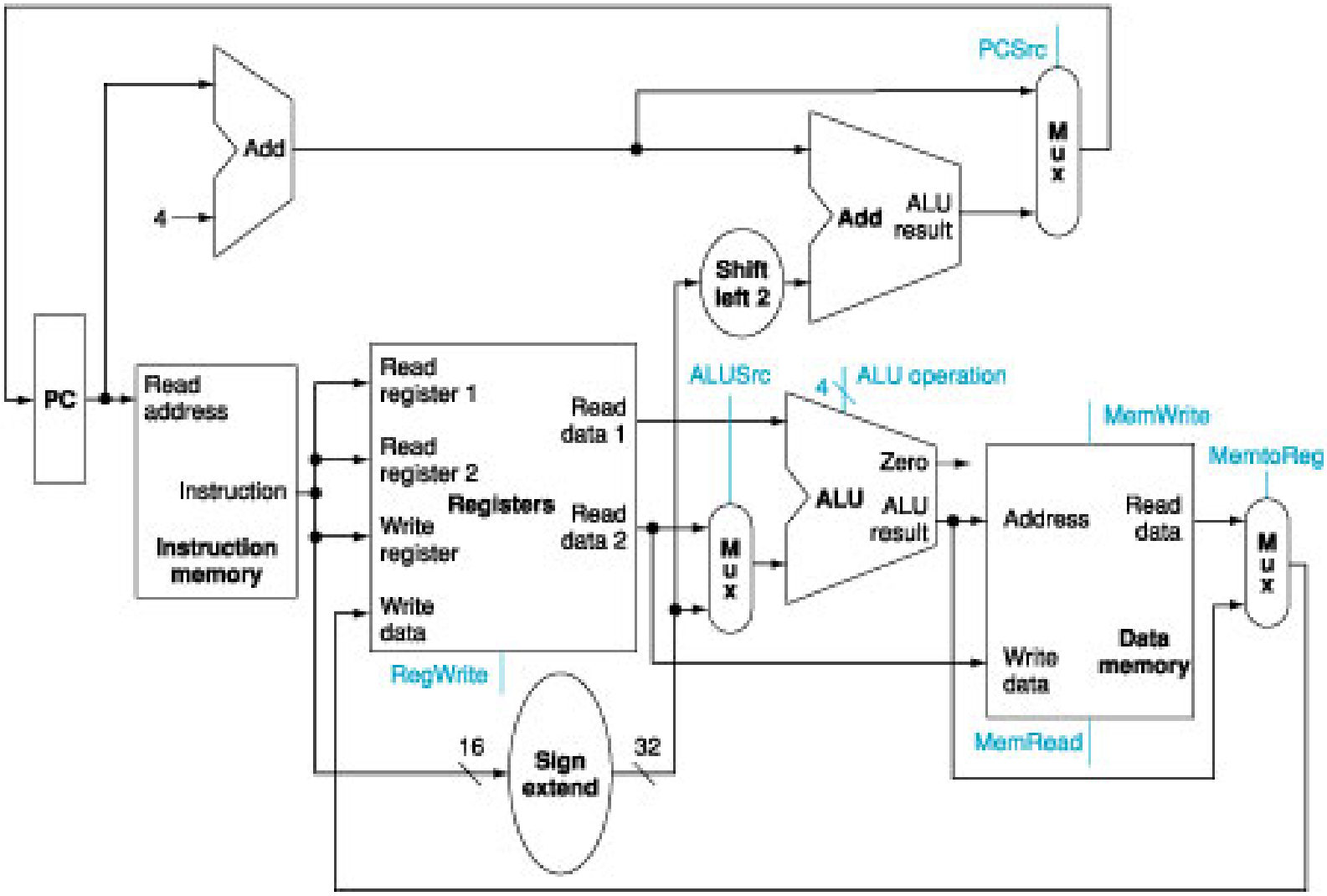
Design guidelines

- Simple is fast; small is also fast
- Exploit instruction similarities as much as possible to simplify circuits
  - This is why RISC processors have simple, regular instruction sets
- Be optimistic! You can always ignore a result (e.g. the 2nd register read) if it is not needed
  - But before it is stored in a register/memory

Datapath – draft #1



Datapath: single cycle implementation



How to design the control part

- For all control signals determine which value selects what operation, input, etc.
- Make truth table of control signal values for each instruction, or instruction group
- Convert table to combinational circuit

- The above is OK for 1-level decoding, but sometimes 2 levels are useful, e.g. for the ALU operation

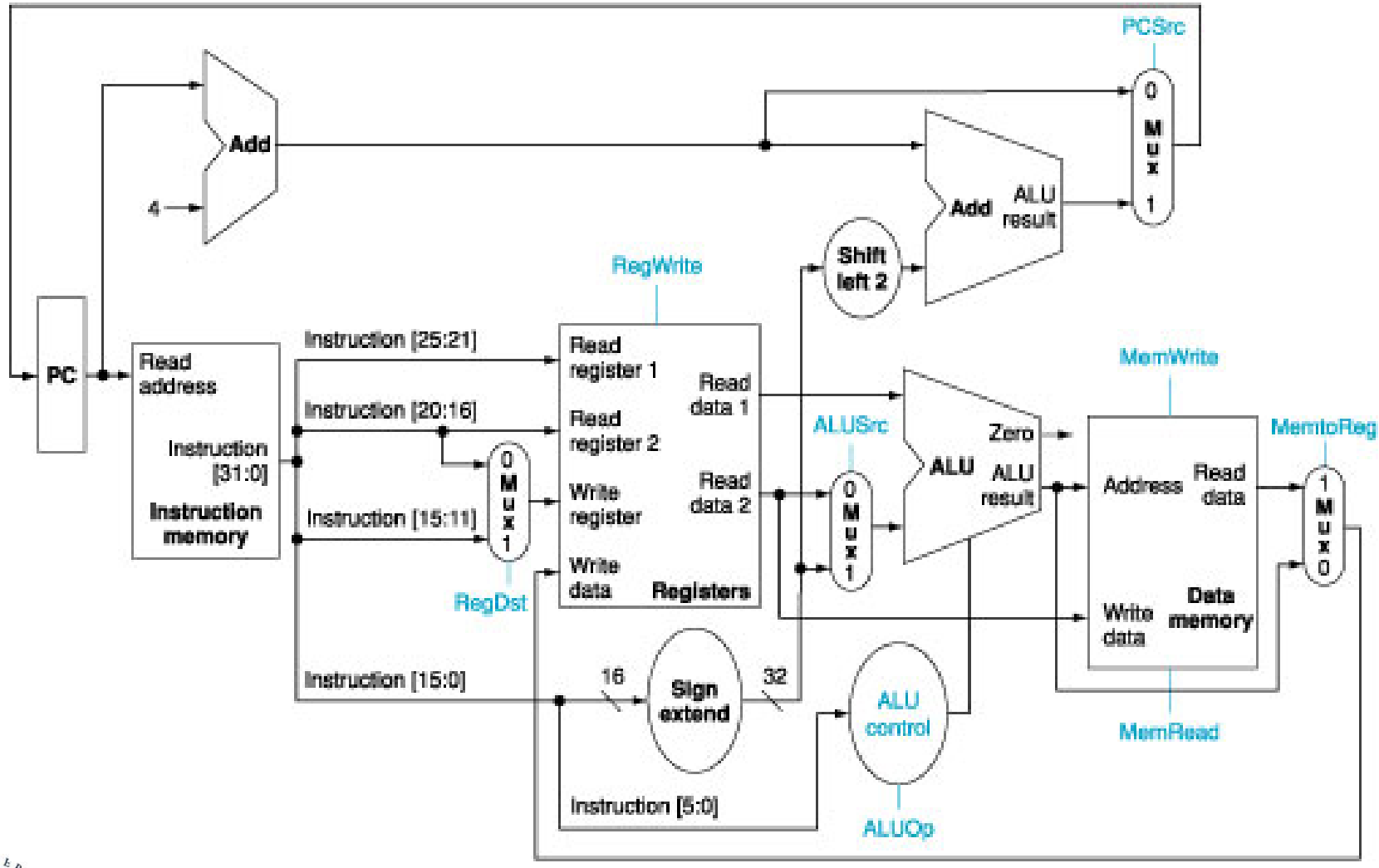
ALU control

- ALU operation:
  - Data transfers – add
  - Branches – sub
  - All other – determined by funct field, I[5:0]

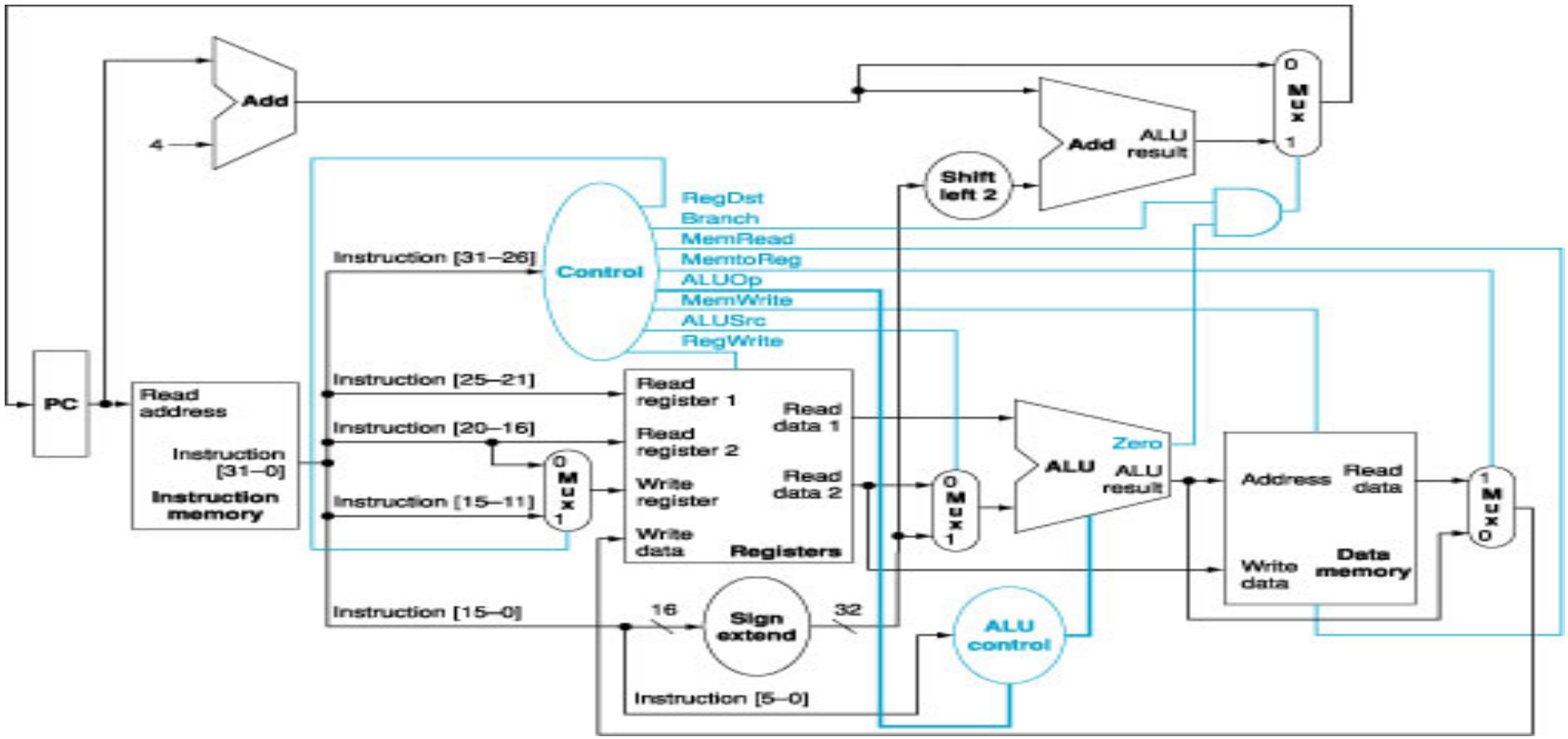
ALU operation			
0000	AND	0110	subtract
0001	OR	0111	set-on-less
0010	add		than

- Main control specifies which of the 3 op types
- Second level provides final ALUoperation control signal

Datapath: control signals

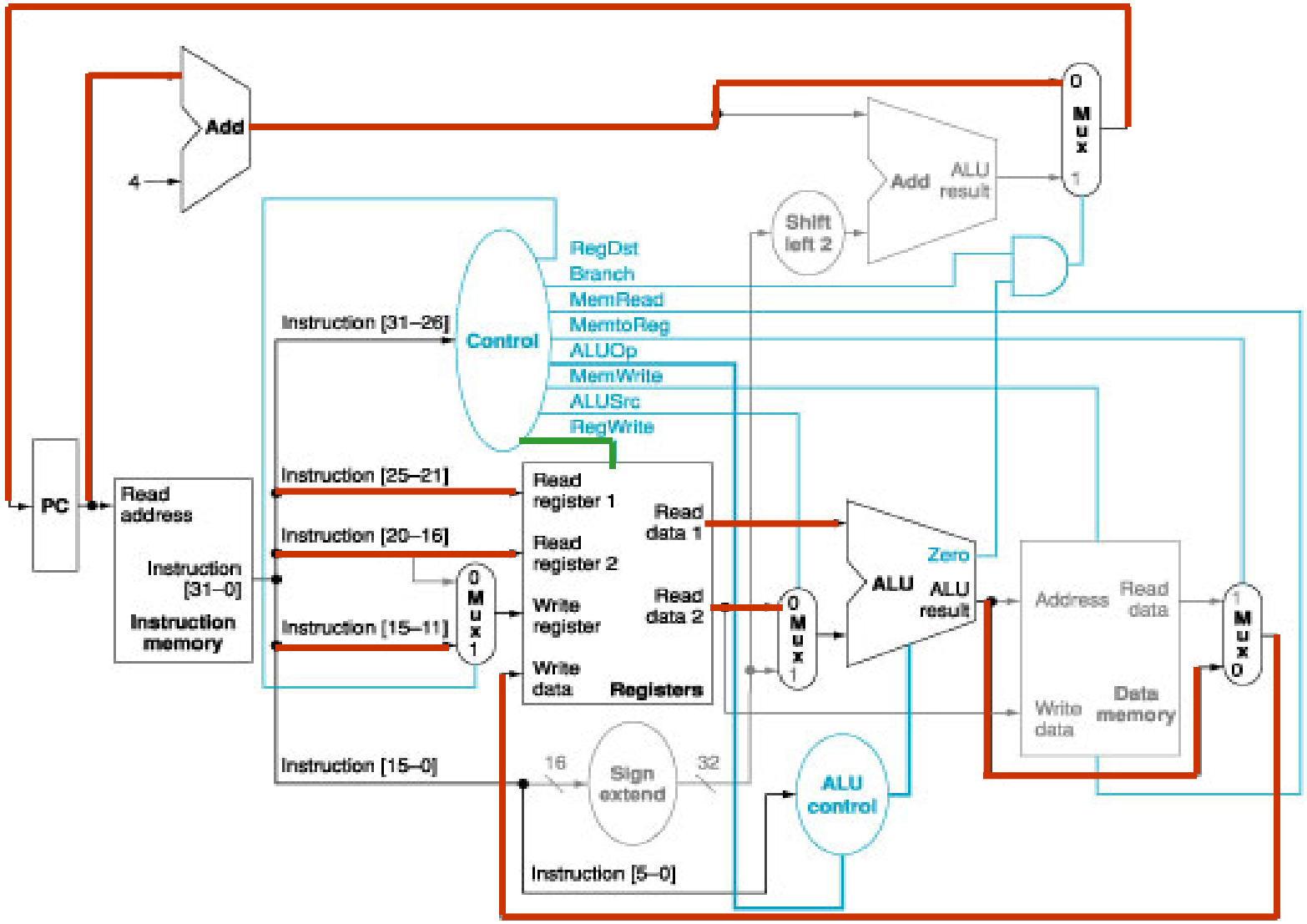


Final datapath and control truth table

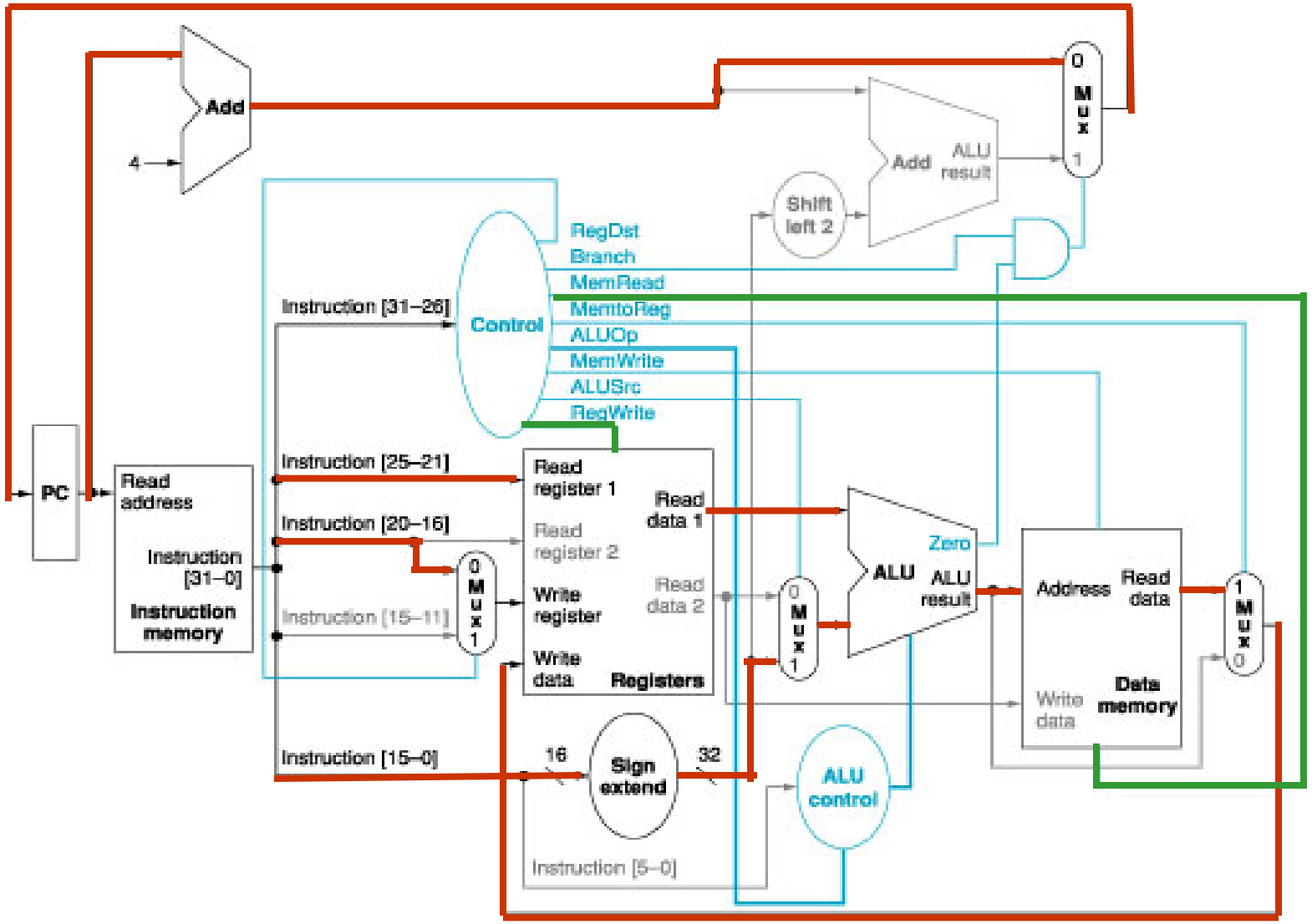


Instruction	RegDst	ALUSrc	Memto-Reg	Reg Write	Mem Read	Mem Write	Branch	ALUOp1	ALUp0
R-format	1	0	0	1	0	0	0	1	0
`lw`	0	1	1	1	1	0	0	0	0
`sw`	X	1	X	0	0	1	0	0	0
`beq`	X	0	X	0	0	0	1	0	1

R-type instruction execution



lw execution



beq (taken) execution