Datapath / Machine Organization

CSE 410

Lecture 08

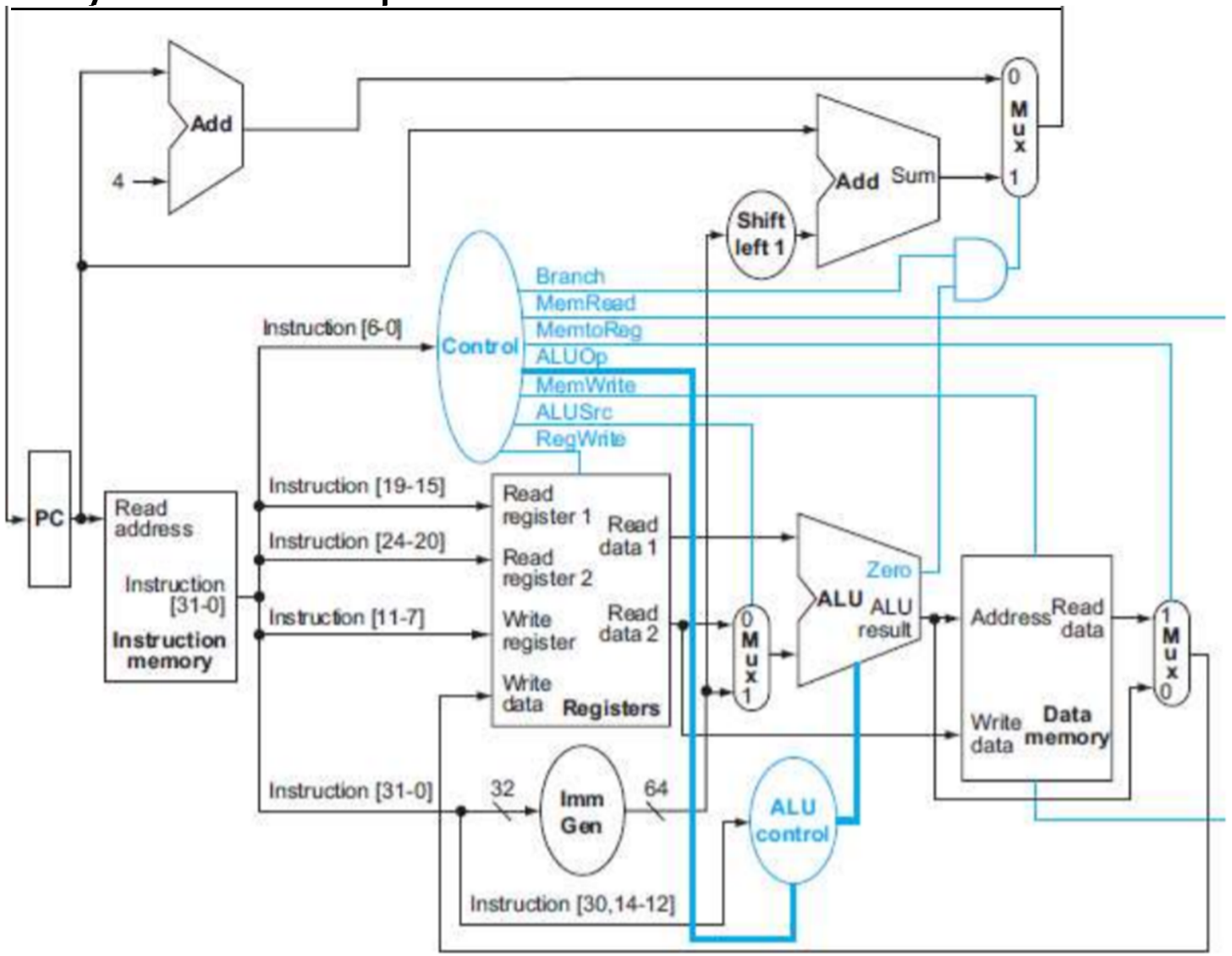
Lecture Outline

- **The Datapath**
  - **Basic Components**
  - **Basic Datapath Layout**
  - **Basic Datapath Operation**

Datapath

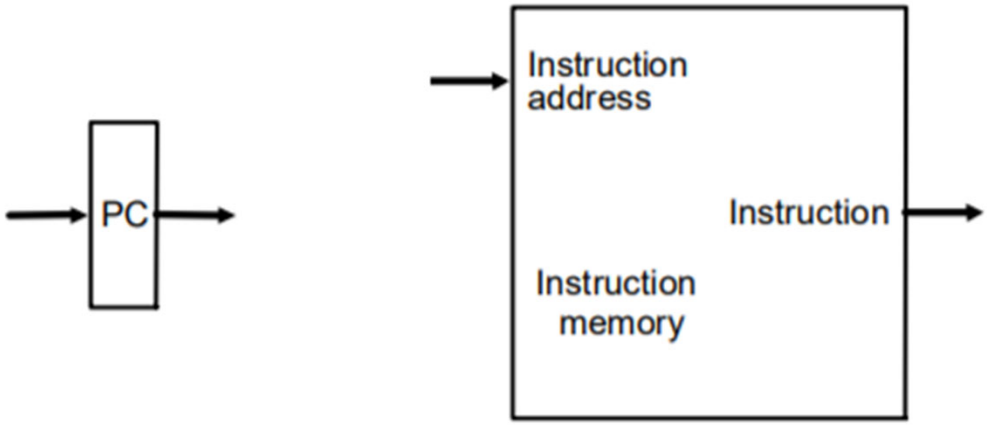
- We've looked at how to create circuits composed of logic (gates) that perform some computing functions
- We now move up a level of abstraction – the basic components are not gates, but rather small circuits that perform more powerful functions, for example
  - adder
  - multiplexor
- This level of design is called "machine organization"

Single-Cycle Datapath



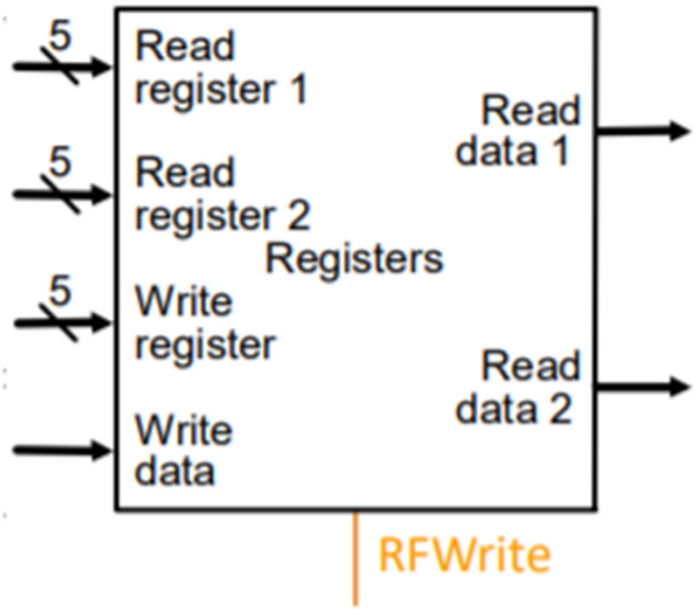
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

Components: Instruction Fetch



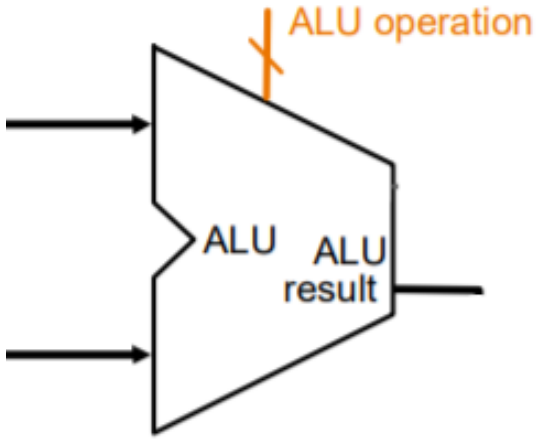
- A hardware register holds the PC
- An instruction fetch component can retrieve words of memory
  - Takes as an input the address to be fetched
    - We'll use the PC
  - Gives as an output the 32-bits at that memory address

Components: Register File



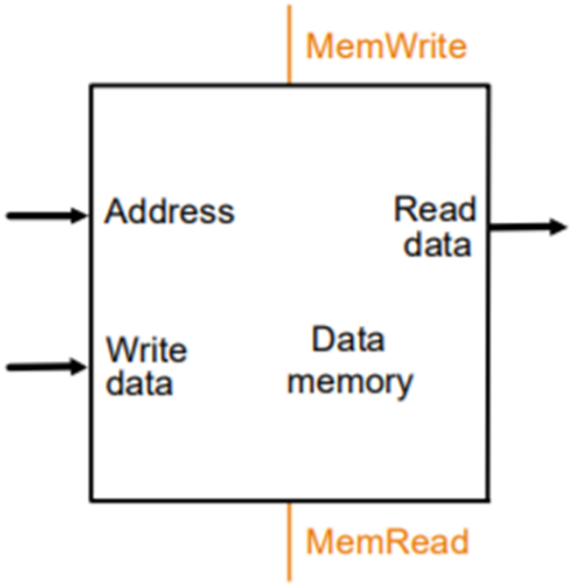
- Contains 32 32-bit registers
- Inputs:
  - Two register numbers, to be read
  - A one-bit “control” input indicating whether or not to write a register
  - A write register number
  - Write register data
- Outputs:
  - Contents of read registers whose numbers were given as inputs

Components: ALU



- Performs add, subtract, mul, div, and, or, etc.
- Inputs:
  - 2 32-bit data items
  - Control line indicating what operation to perform
- Output:
  - Result of the operation on the input values

Components: Memory Interface



- Inputs:
  - An address to be (possibly) read or written
  - Data to be written
  - Control lines indicating whether to do a read or a write (or neither)
- Output
  - Word read from memory, if a read is done

The Control Component

- Control is in charge of making sure the correct items are written (to memory ,register, or PC), depending on what the instruction is
- There is no incentive to not compute something that will not be written
  - E.g., don't care what data is fed to the register file as the write data if the register file won't be written

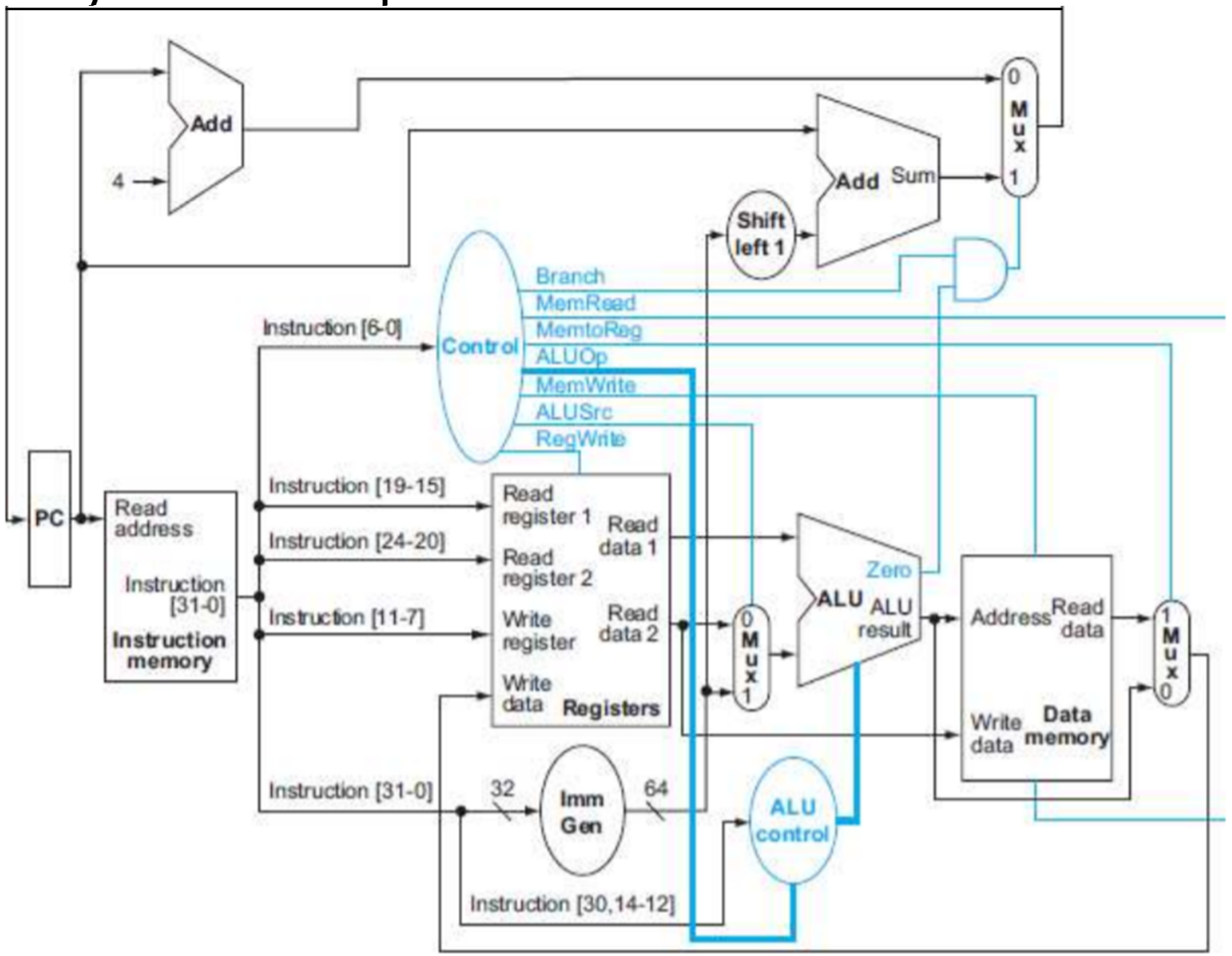
Reminder: Instruction Formats

CORE INSTRUCTION FORMATS

	31 27	26 25	24 20	19 15	14 12	11 7	6 0
R	funct7		rs2	rs1	funct3	rd	Opcode
I	imm[11:0]			rs1	funct3	rd	Opcode
S	imm[11:5]		rs2	rs1	funct3	imm[4:0]	opcode
SB	imm[12|10:5]		rs2	rs1	funct3	imm[4:1|11]	opcode
U	imm[31:12]					rd	opcode
UJ	imm[20|10:1|11|19:12]					rd	opcode

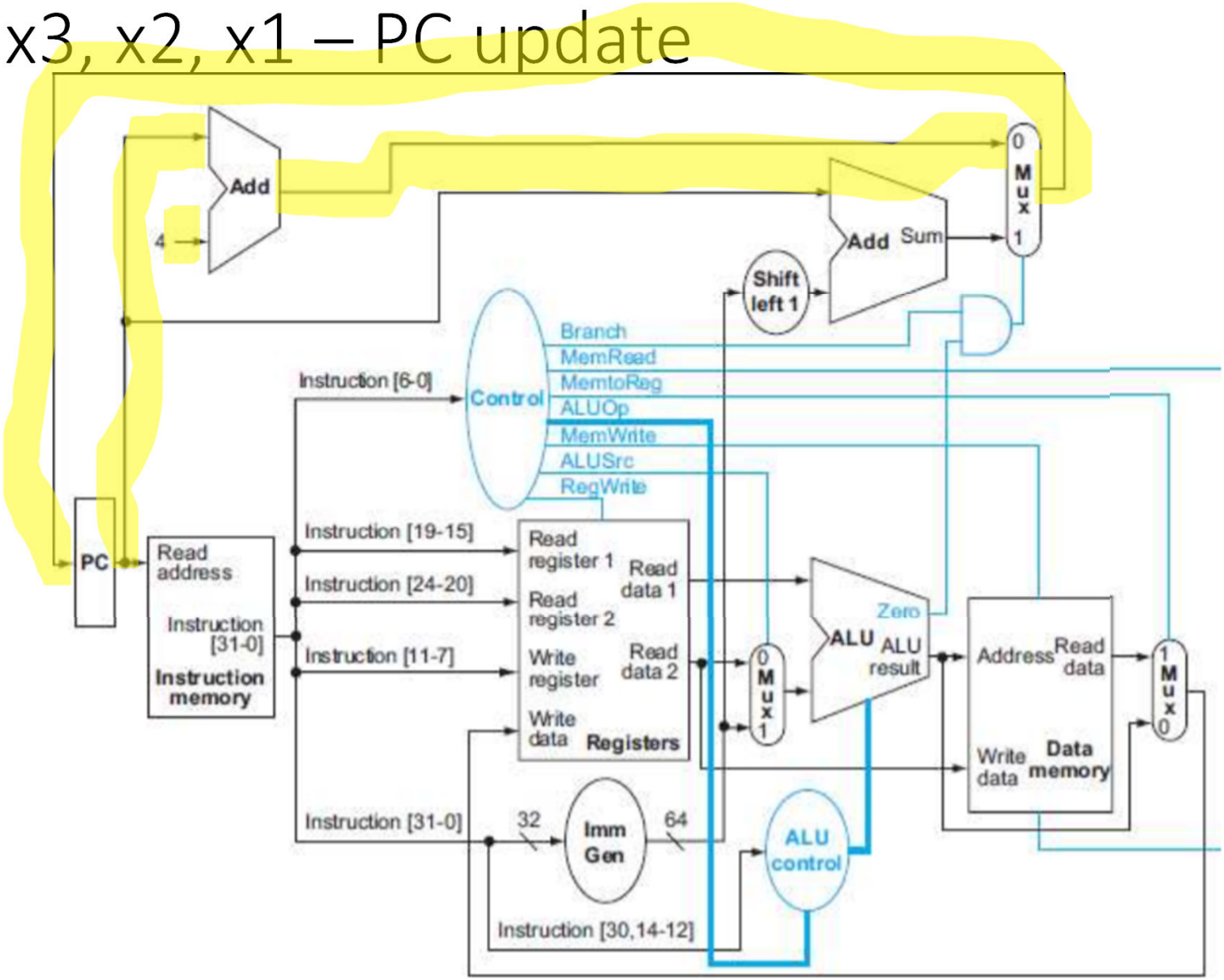
- If a register will be written, reg number in bits 11:7
- If one register read, number in bits 19:15
- If two registers read, second number in bits 24:20
- If there's an immediate:
  - sign bit in bit 31
  - bits 10:5 of immediate in bits 30:25 of instruction

Single-Cycle Datapath



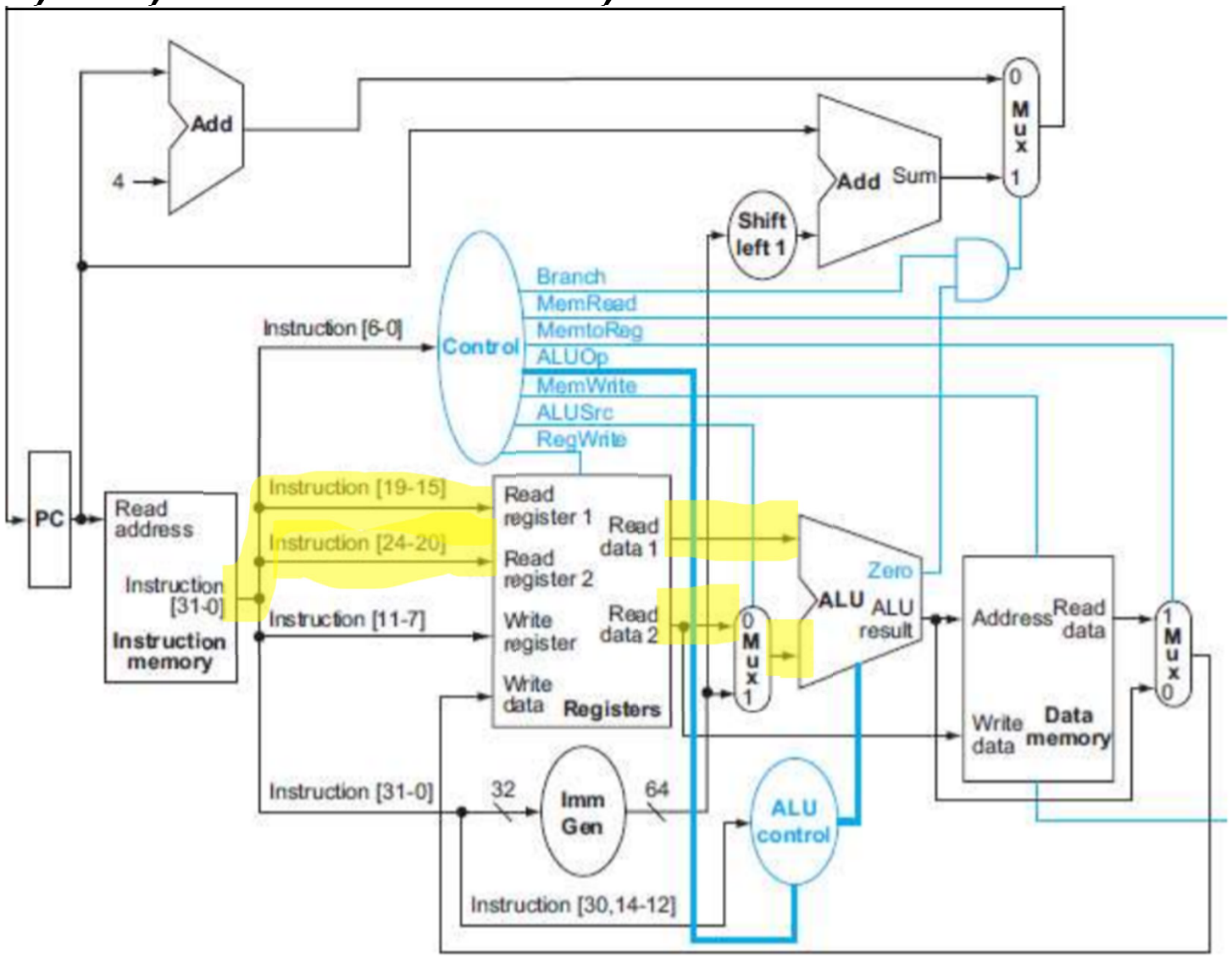
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

add x3, x2, x1 – PC update



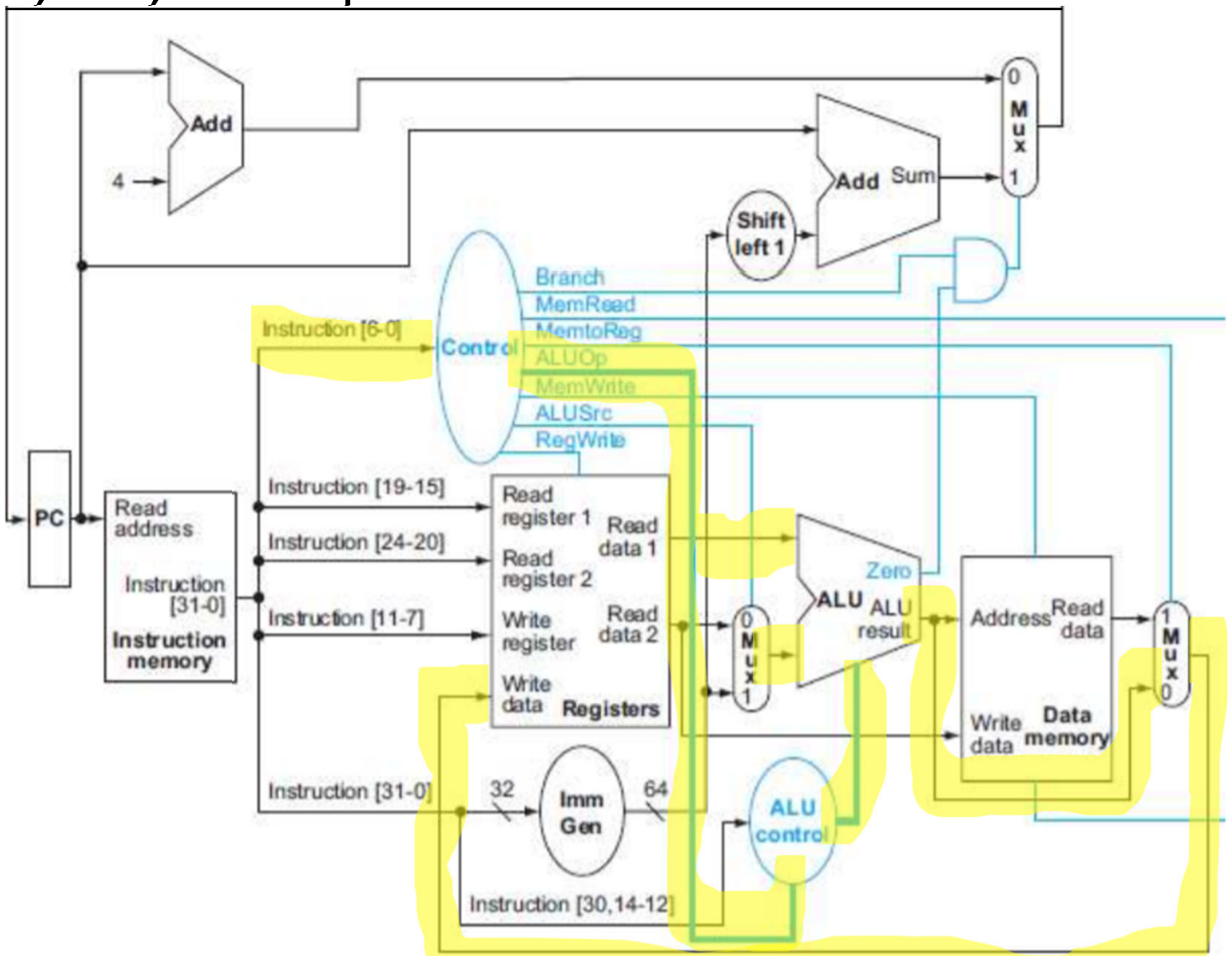
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

add x3, x2, x1 – fetch x2, x1



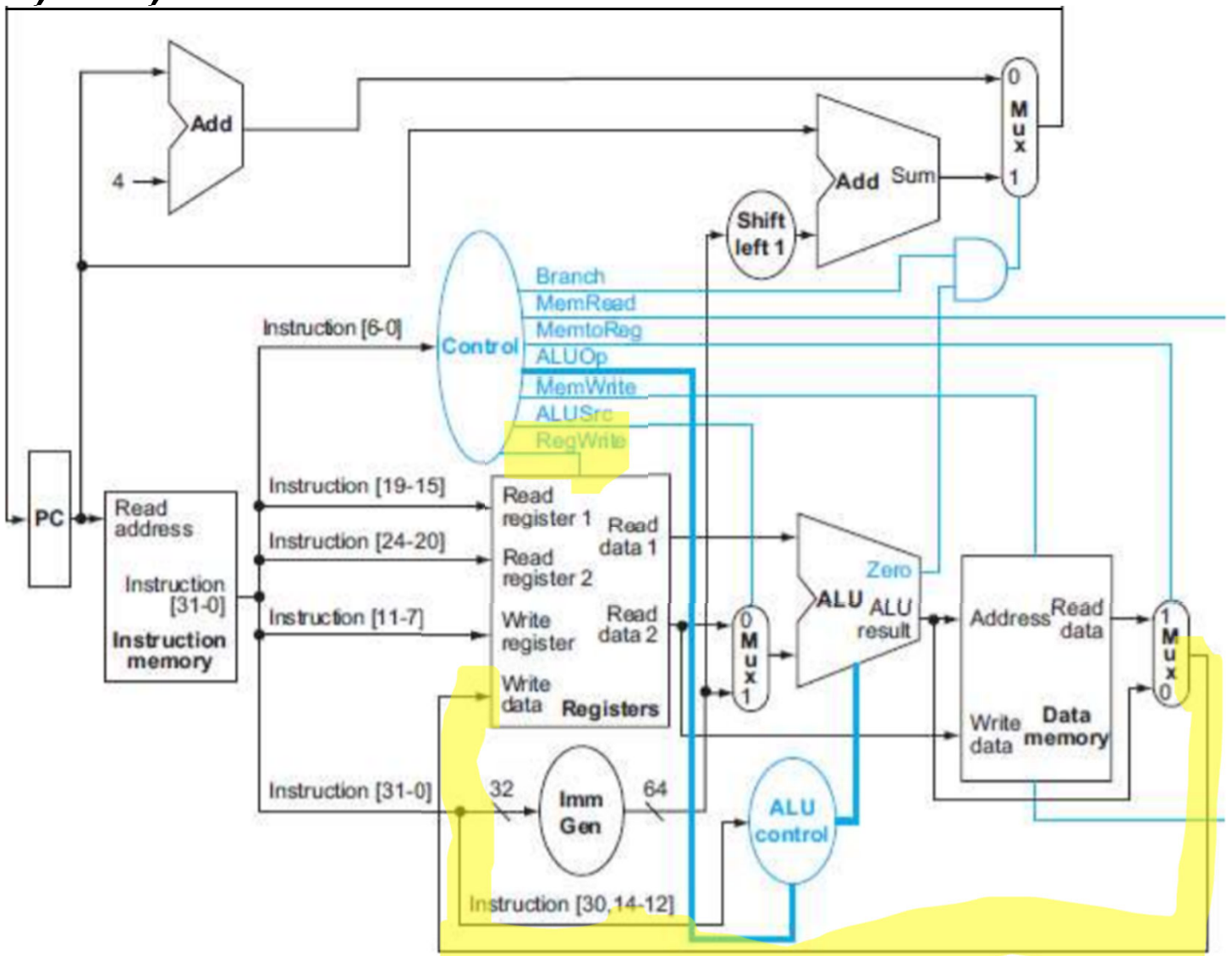
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

add x3, x2, x1 – perform addition



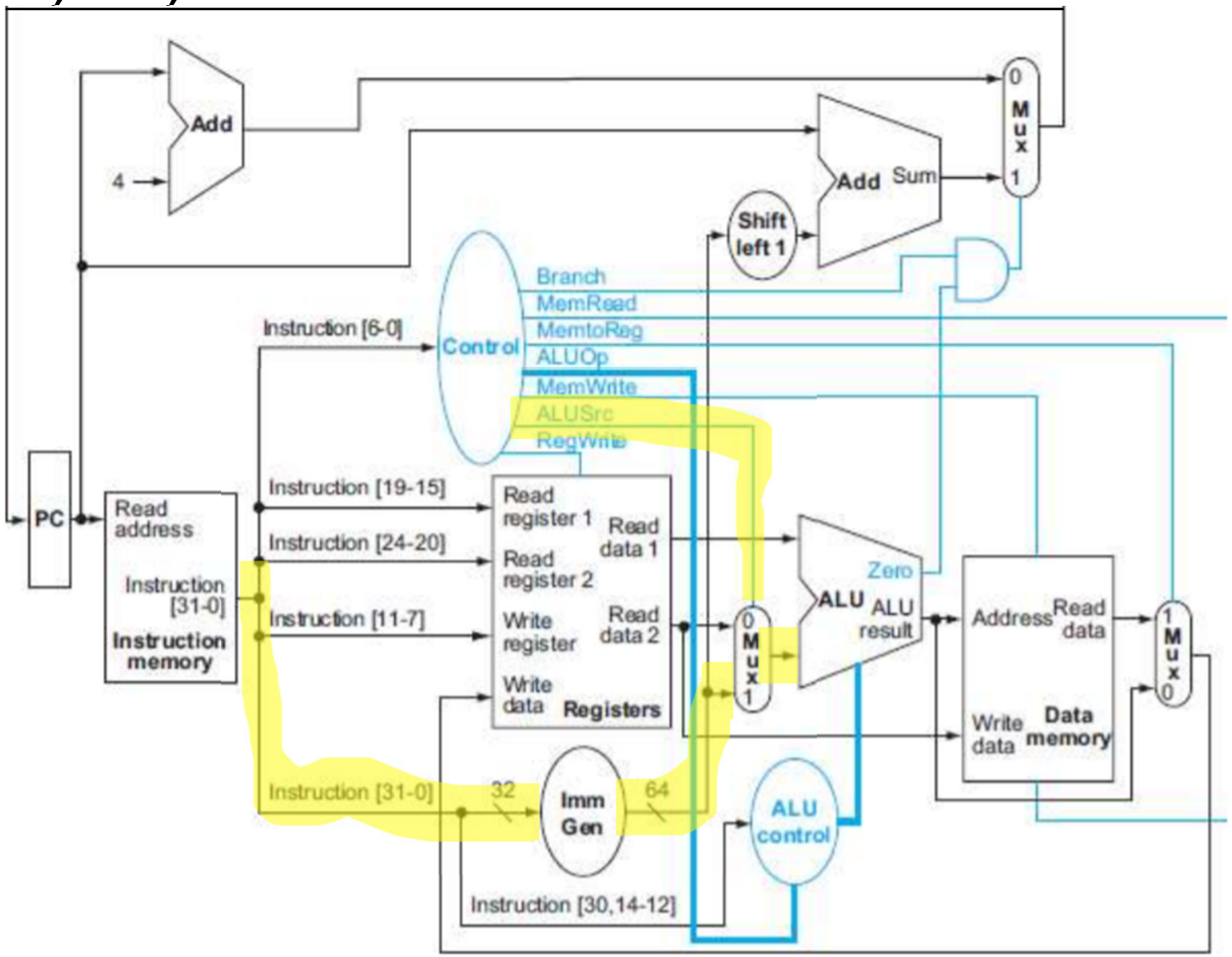
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

add x3, x2, x1 – write result into x3



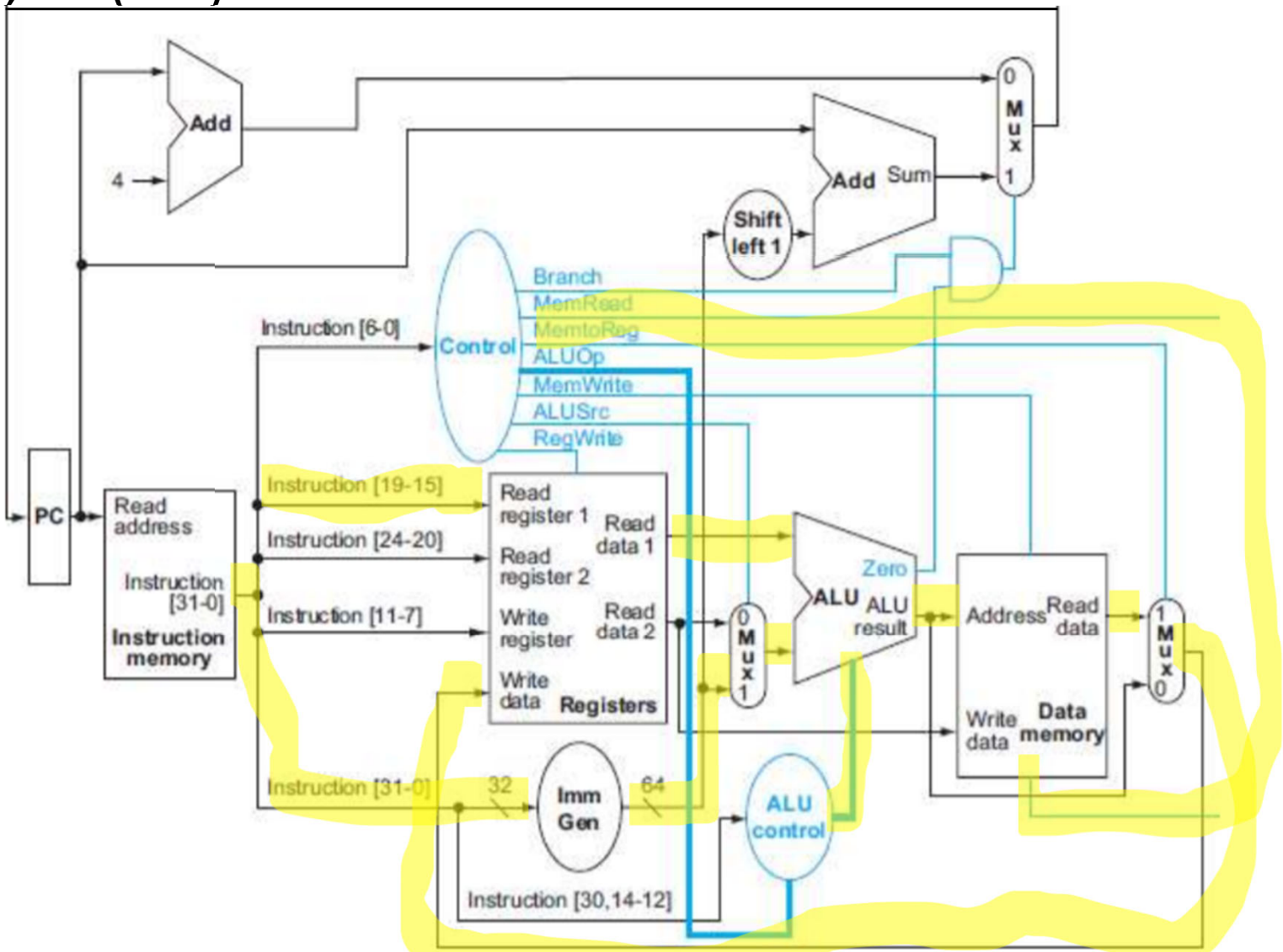
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

addi x3, x2, 1 – differences from add



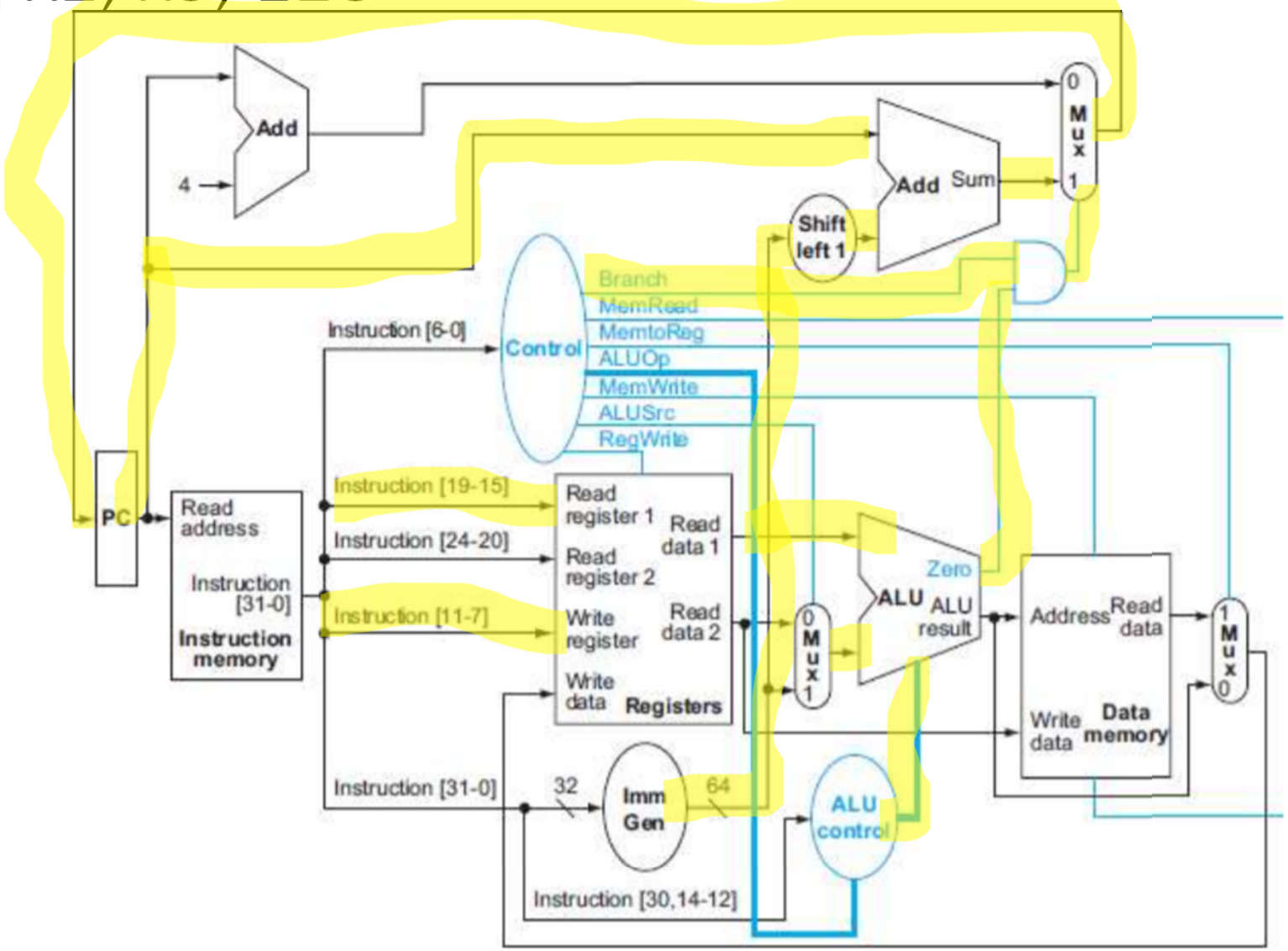
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

lw x3,12(x2)



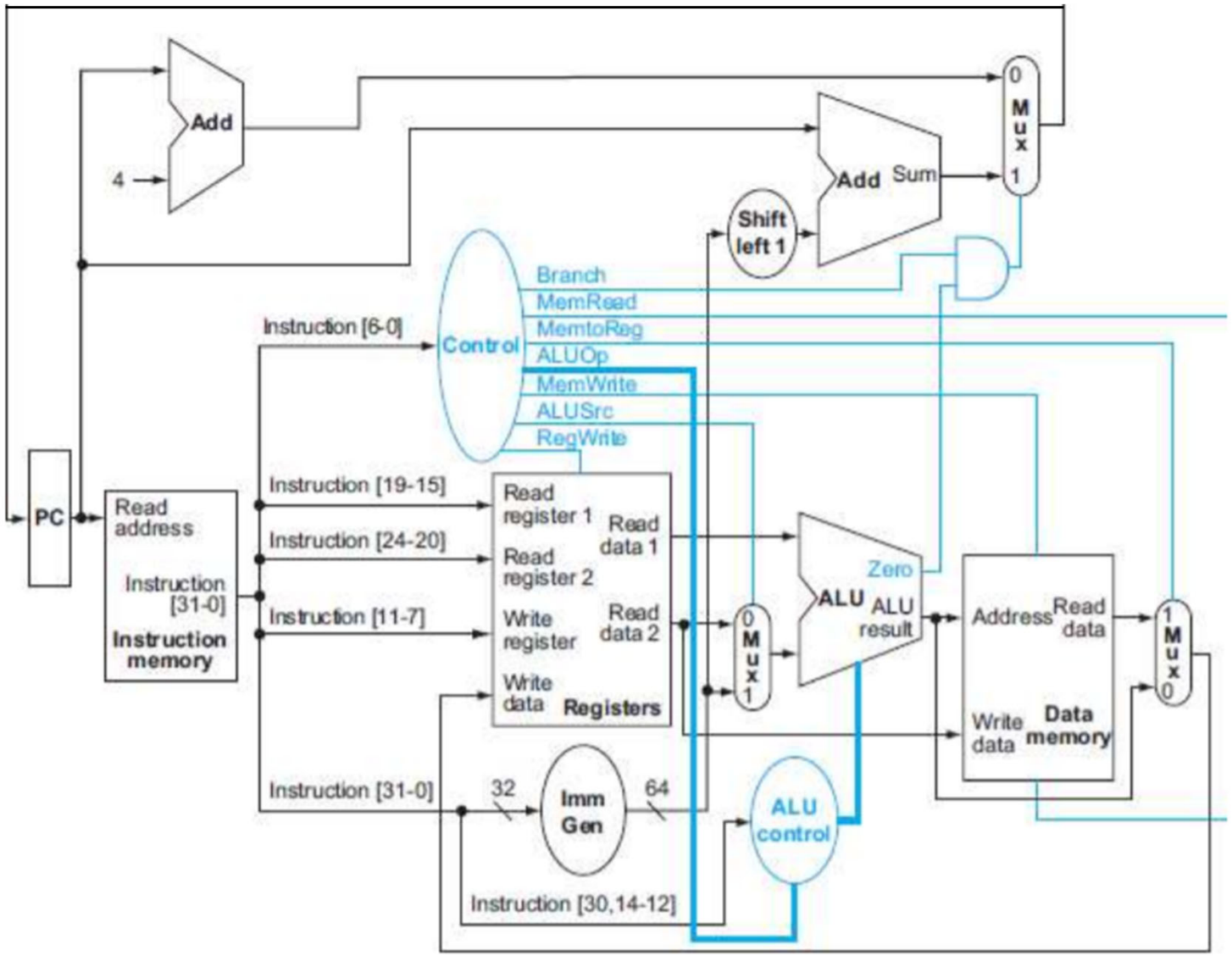
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

beq x2, x3, 128



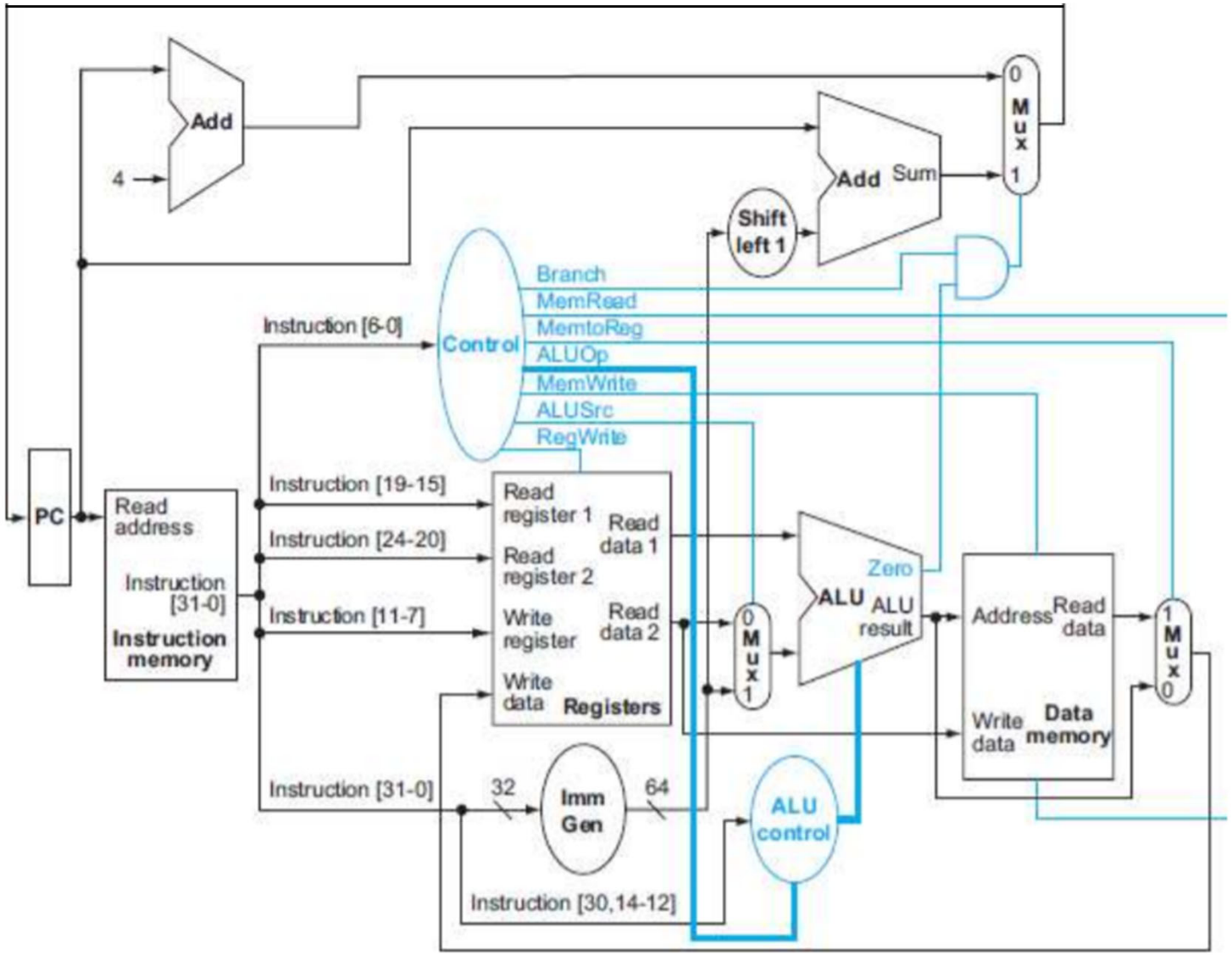
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

sll x4, x3, 29 – what are control values?



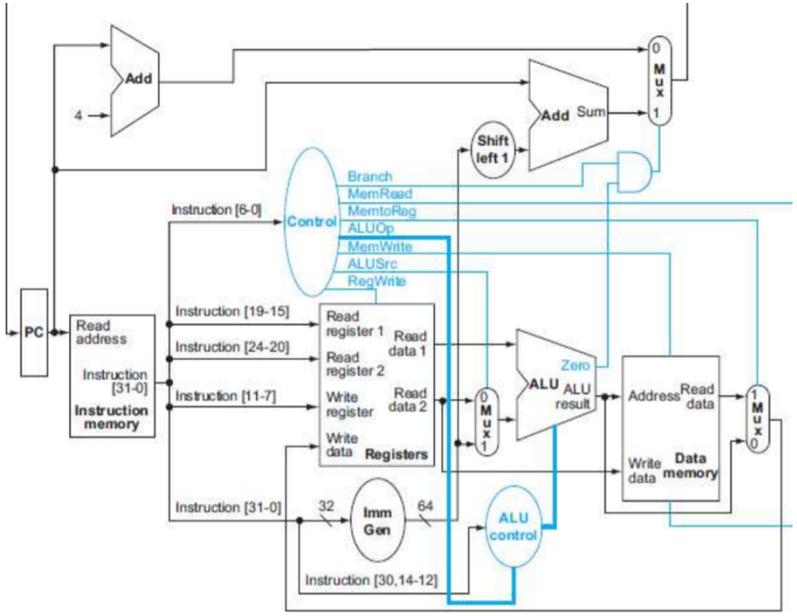
[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

jalr x2, 64(x6) – what's missing in datapath?



[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

Datapath Summary



[Figure 4.17 from book, Copyright © 2018 Elsevier Inc. All rights reserved.]

- Major Components:
  - Control
  - Instruction Fetch &PC Control
  - Register File
  - ALU operation
  - Memory Read/Write
  - Register Write Data