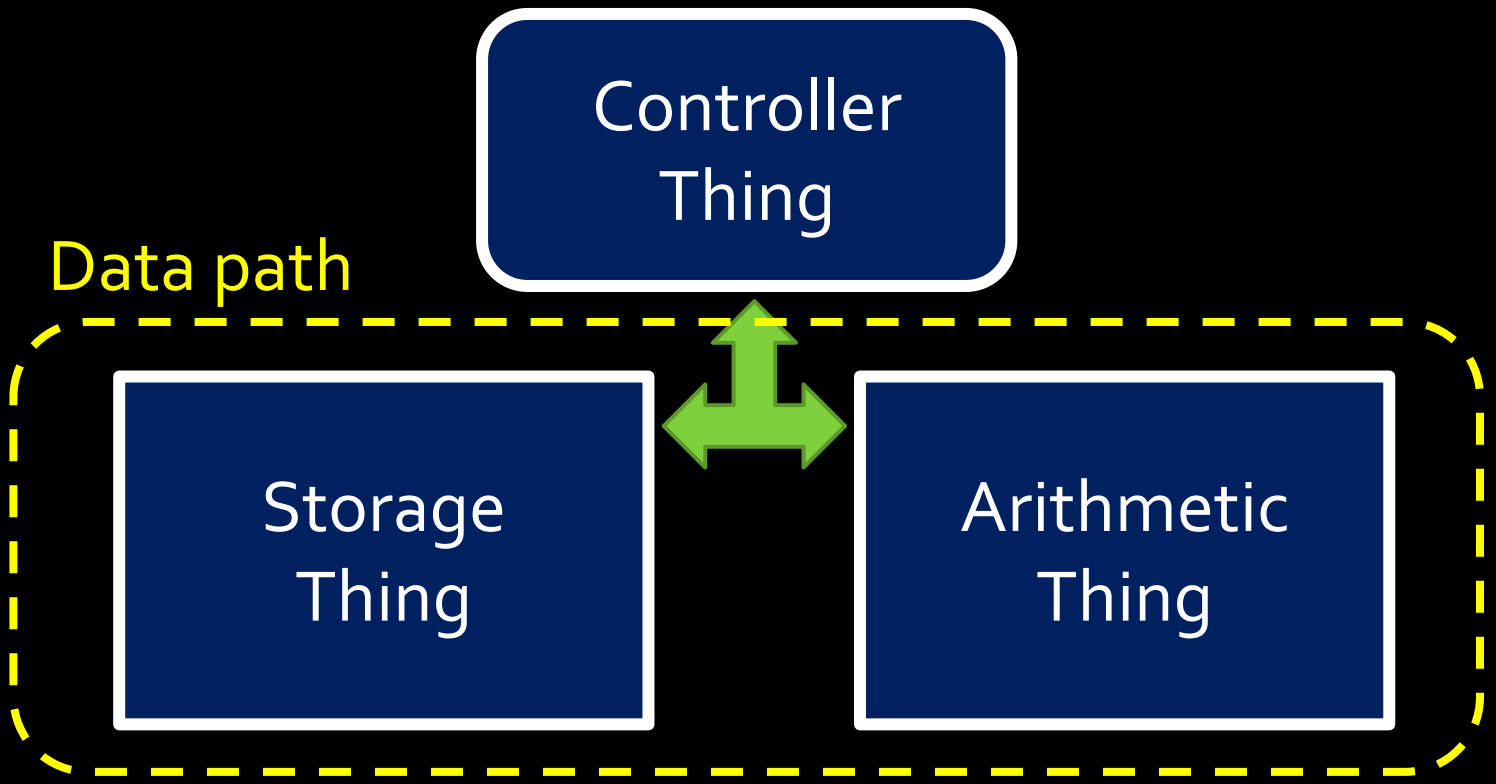
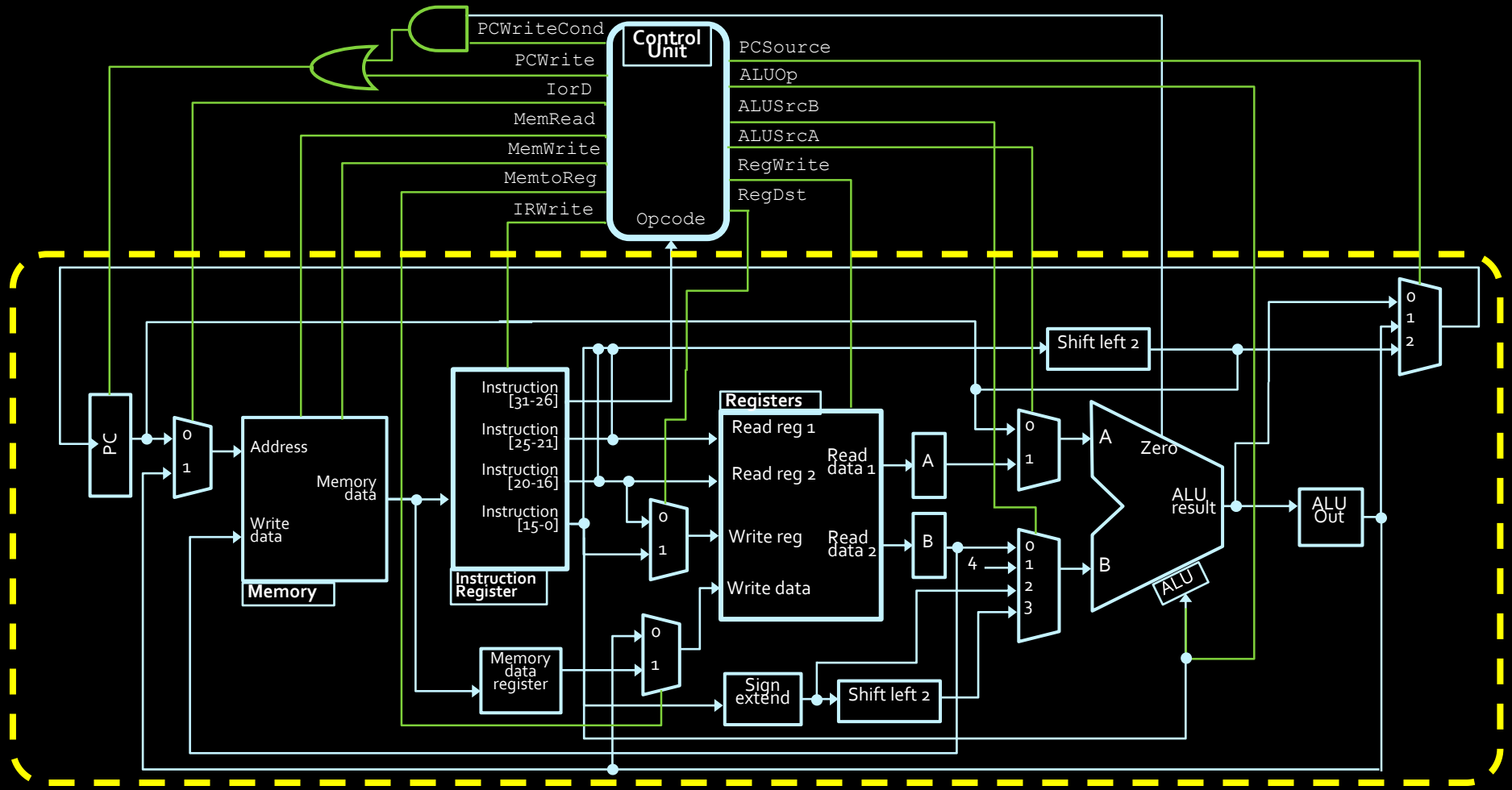
Week 7, part D: Intro to Control

aka: the Controller Thing

- We learned what the Arithmetic and Storage Things do
- And we can build a data path



MIPS Datapath Diagram

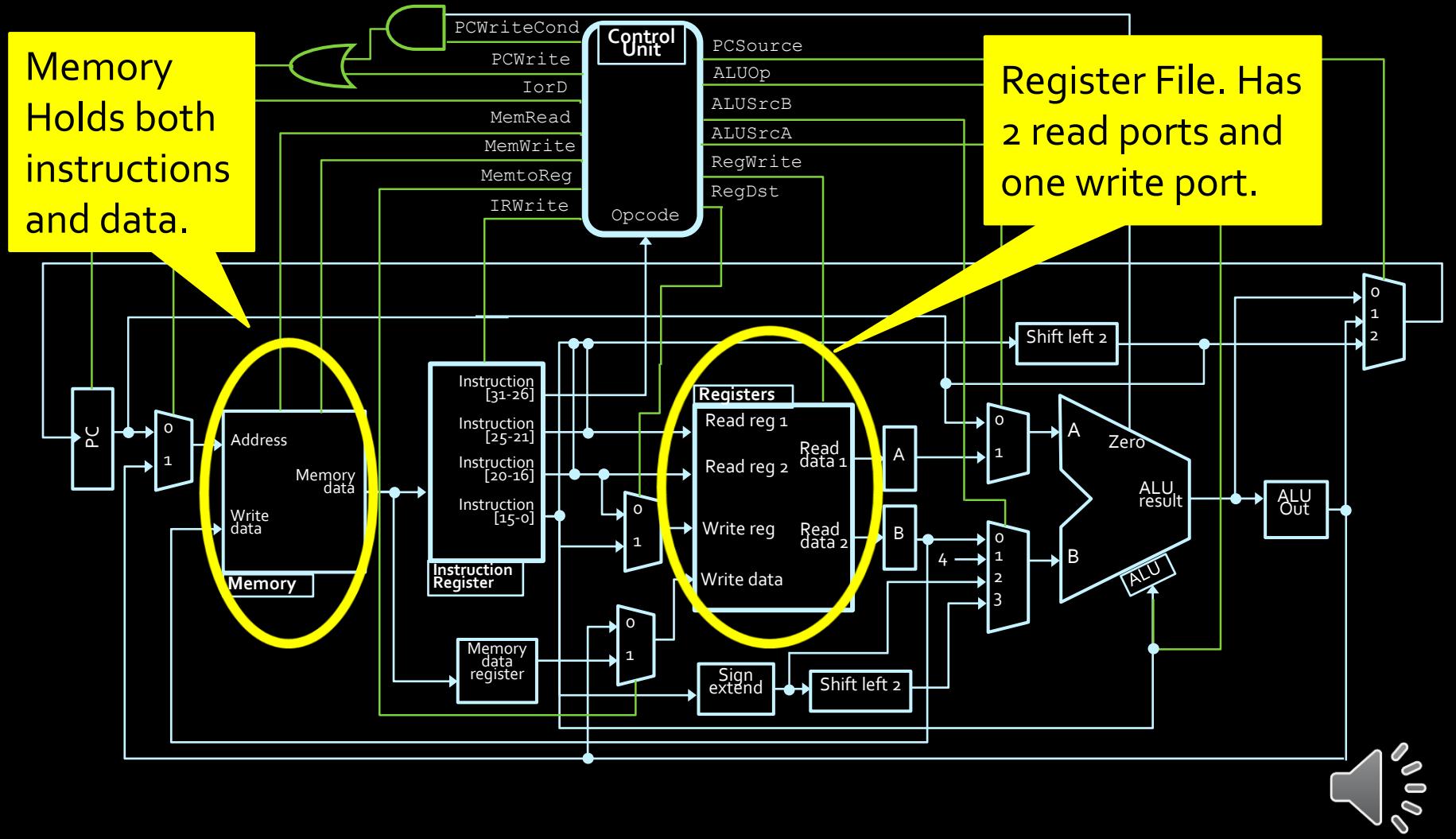


Data path

The Control Unit

- Reminder: the control unit determines for a data path:
  - Where the data is coming from (the source),
  - Where it's going to (the destination), and
  - How the data is being processed (the operation).
- It does so by sending the right control signals to muxes, registers, and other components.

Data Sources and Destinations



The Control Unit

- We've seen control units that always compute the same thing.
- …a microprocessor is programmable.
- How will the control unit know what operation to perform?
  - It gets information from an instruction.
  - This instruction also contains other information about the operation to the rest of the processor.
  - The control unit is responsible for loading the next instruction to run, after completing the current one.

Executing a program

- What actually happens when you run an executable program?
  (e.g., Logisim, calc, Fortnight)
  1. OS loads a series of instructions into memory
  2. Location of first instruction is provided to CPU
  3. CPU executes instructions one at a time

  - (this is a little simplified)

Instructions

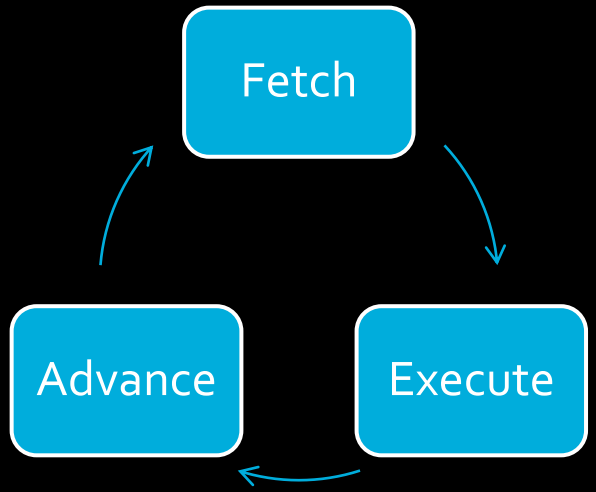
What is an instruction?

- A binary string of bits
  - Always 32 bits in our MIPS processor!
  - Length depends on architecture.
    - Can even be variable! x86: 1-15 bytes
- Tells the processor (the control unit) what to do
- Also known as a control word.

Instructions

- The processor's infinite loop:
  1. Fetch instruction from memory.
  2. Execute it.
  3. Advance to next instruction.

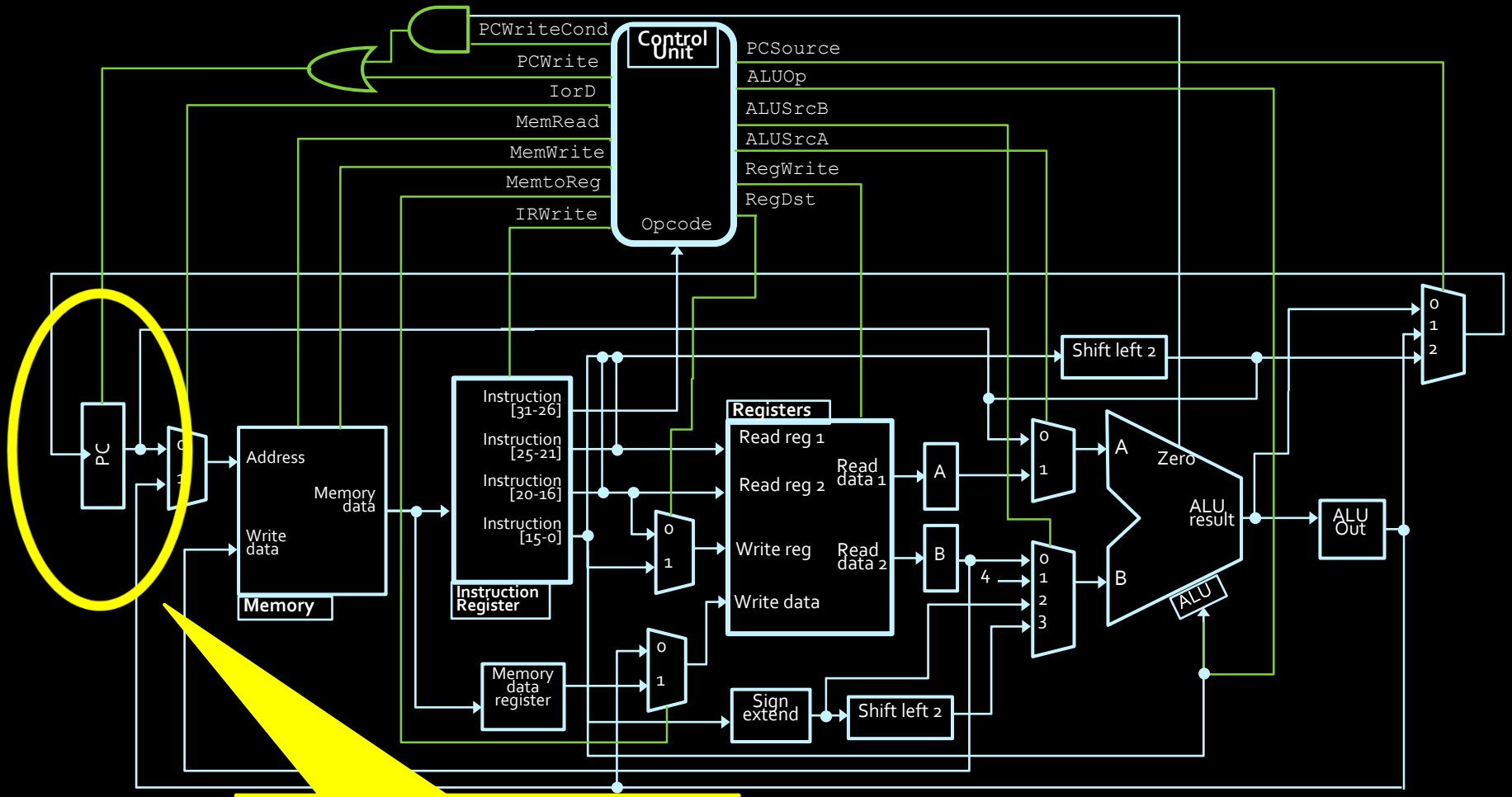
- Really, that's it!



Fetching Instructions

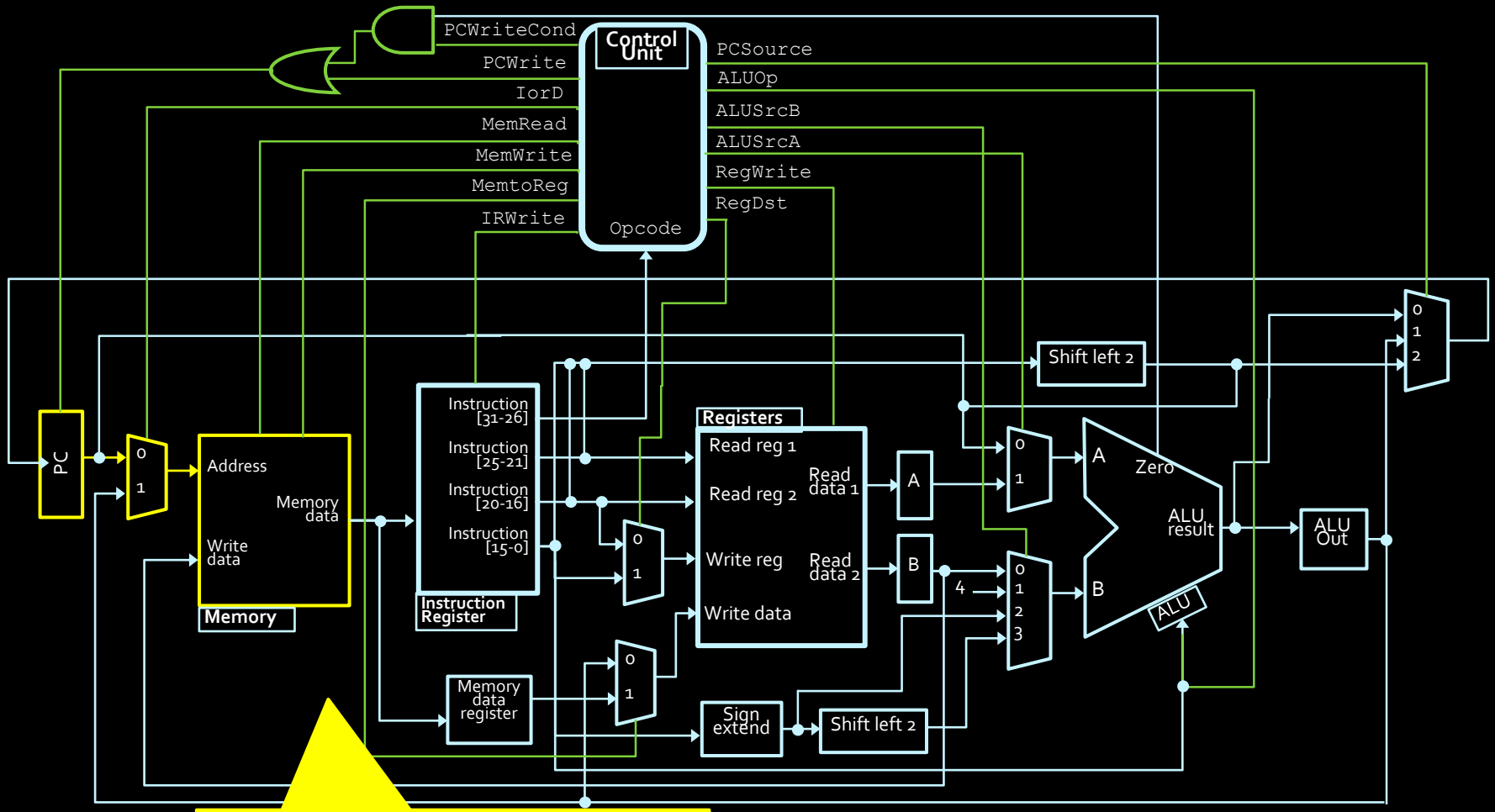
- Instructions are fetched from memory.
- How do we know which instruction to execute?
  - Store address in special register:
    Program Counter (PC)
  - After every instruction fetch, PC is incremented by 4
    - Because each instruction word is 4 bytes.
  - Can also be set by output of ALU.
    - Allow us to 'jump' to another part of the code!
  - Address in PC must be word-aligned (divisible by 4)
    - We'll talk more about that later.

The Program Counter



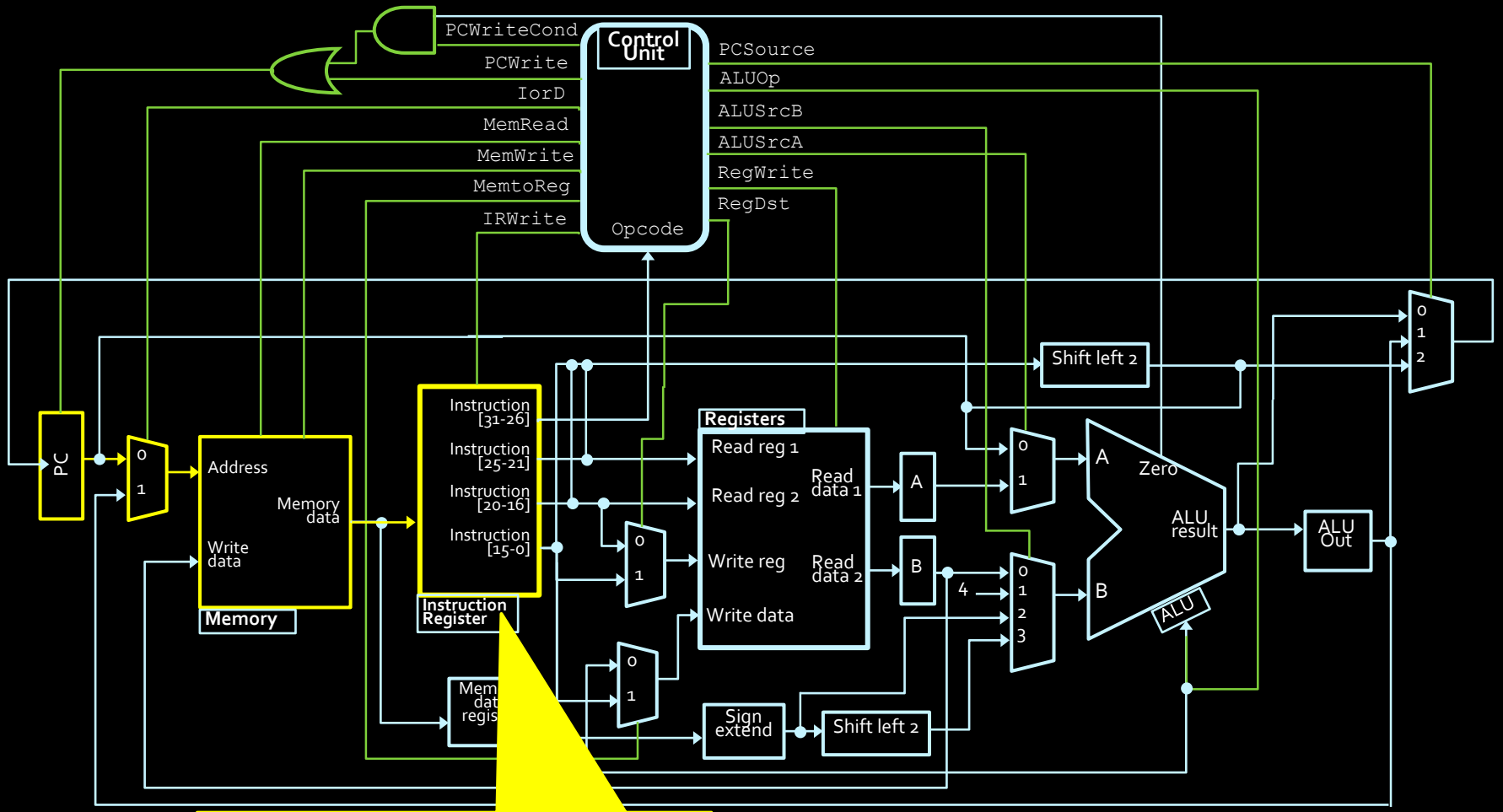
Program counter holds address of instruction

Fetching Instruction



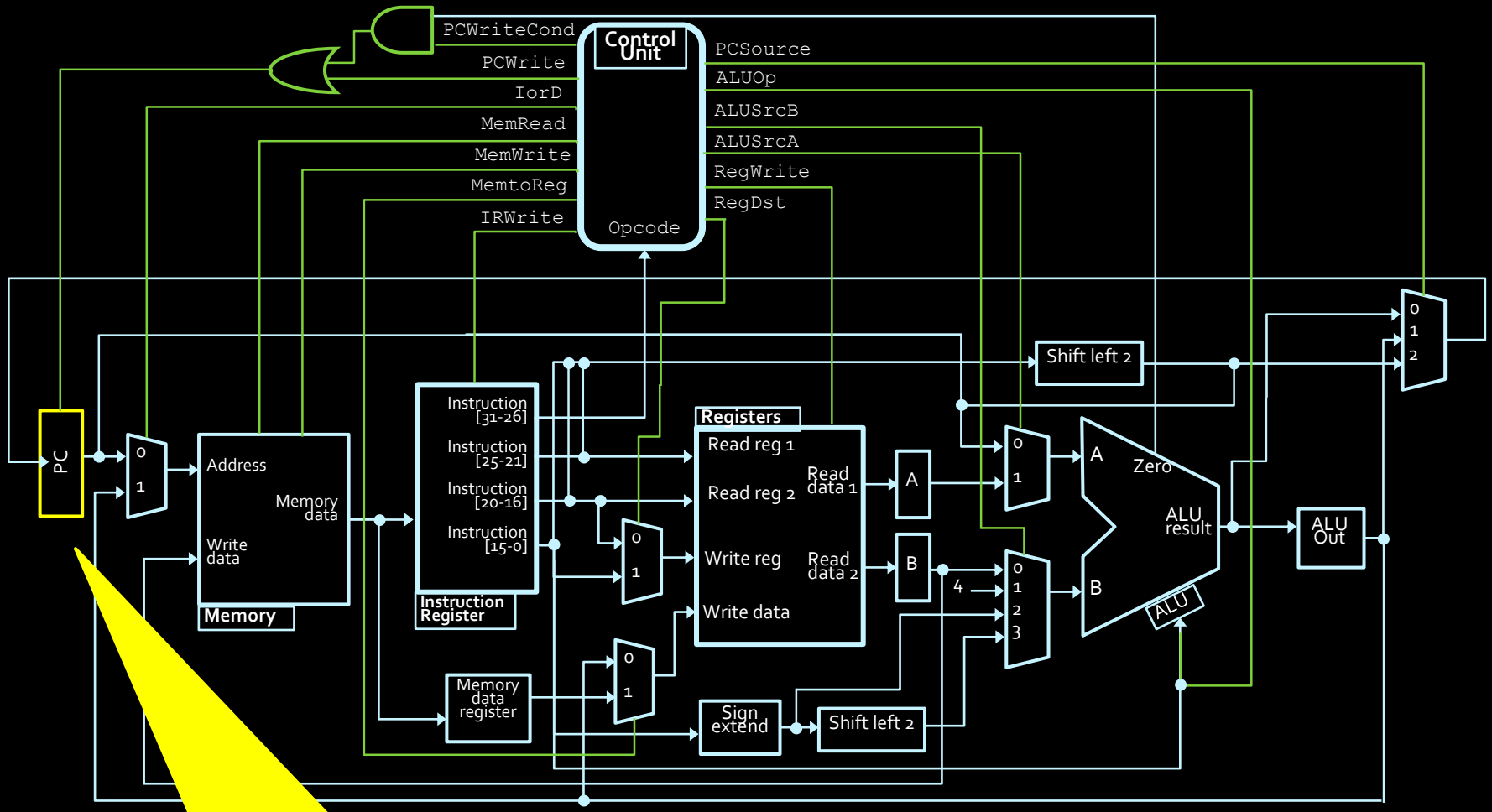
Feed into memory unit to fetch next instruction

Fetching Instruction



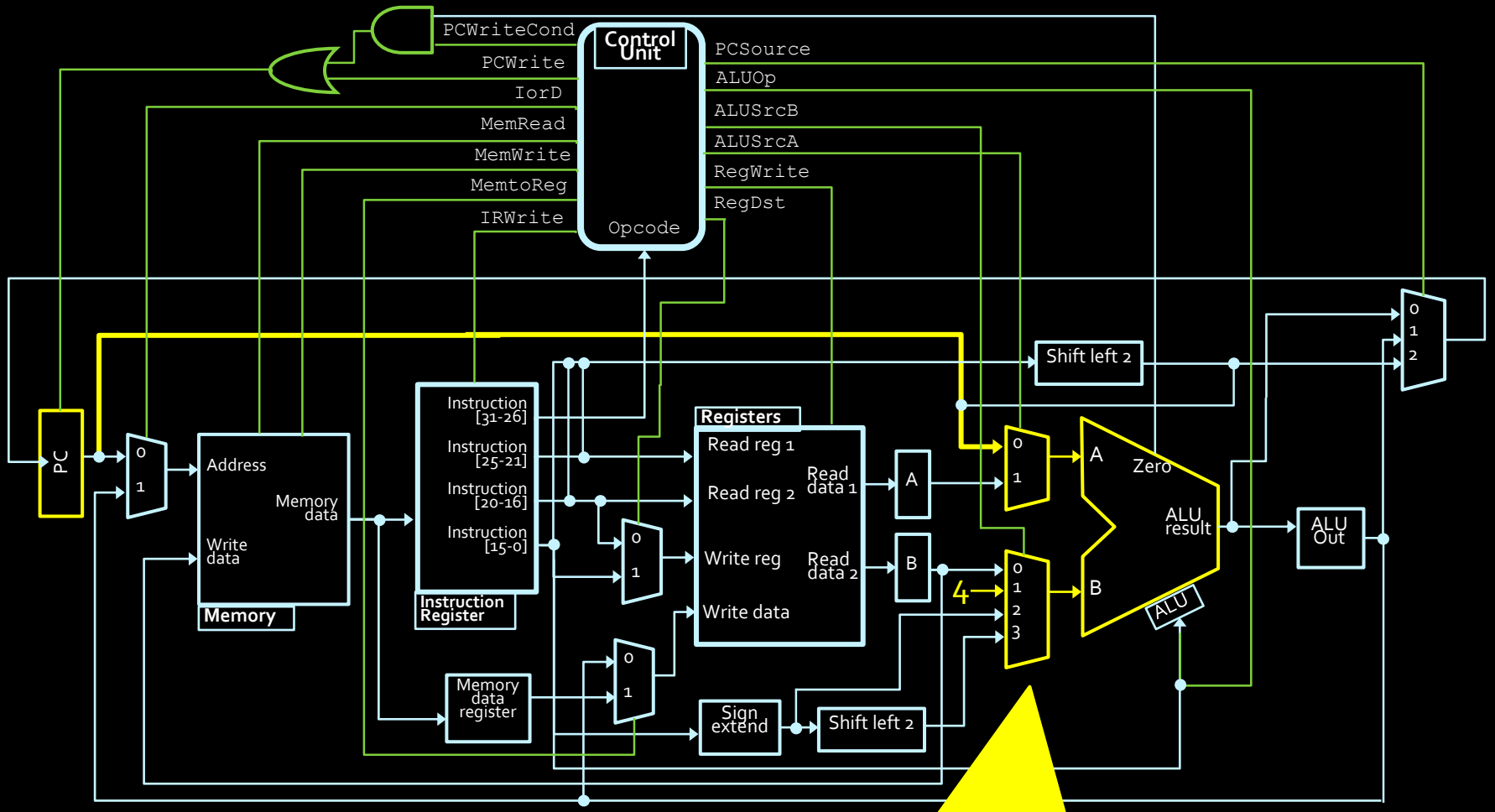
Store fetched instruction in special register

Increment Program Counter



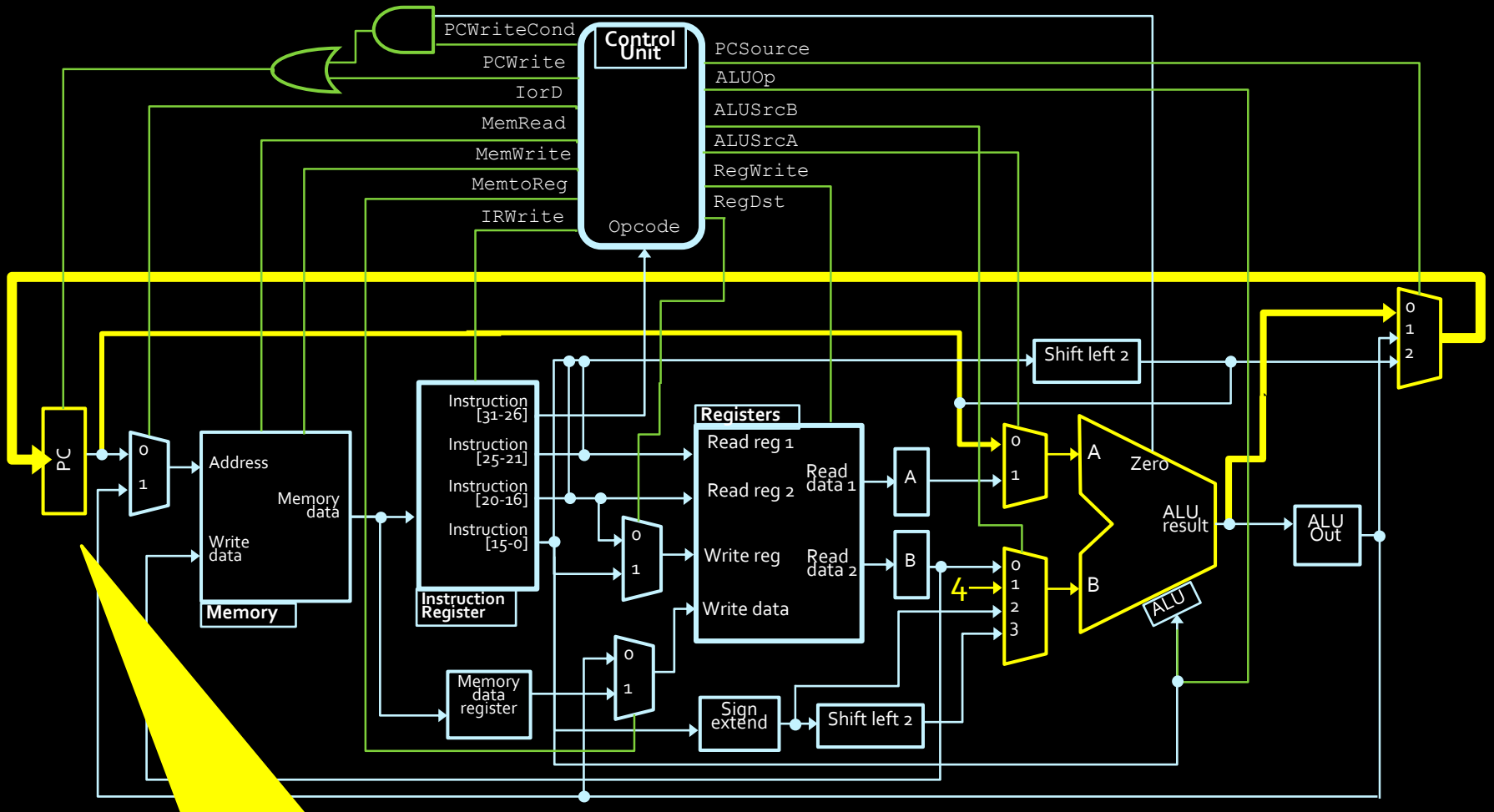
Take current address in PC…

Increment Program Counter



Increment address in PC by 4

Increment Program Counter



Store back into PC register

Understanding instructions

- Okay... Here's your instruction... GO

```
00000000 00000001 00111000 00100011
```

→ Move to next part