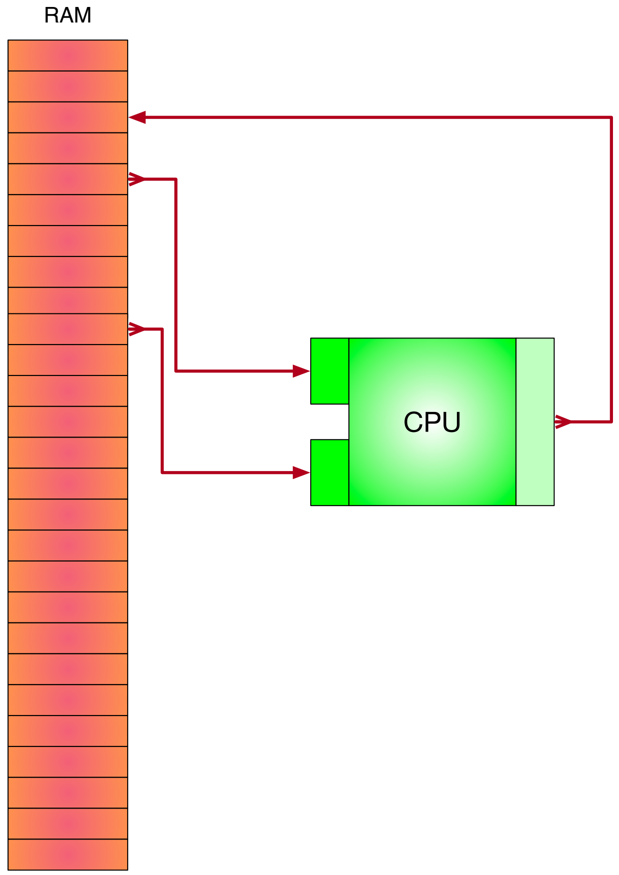
Variables

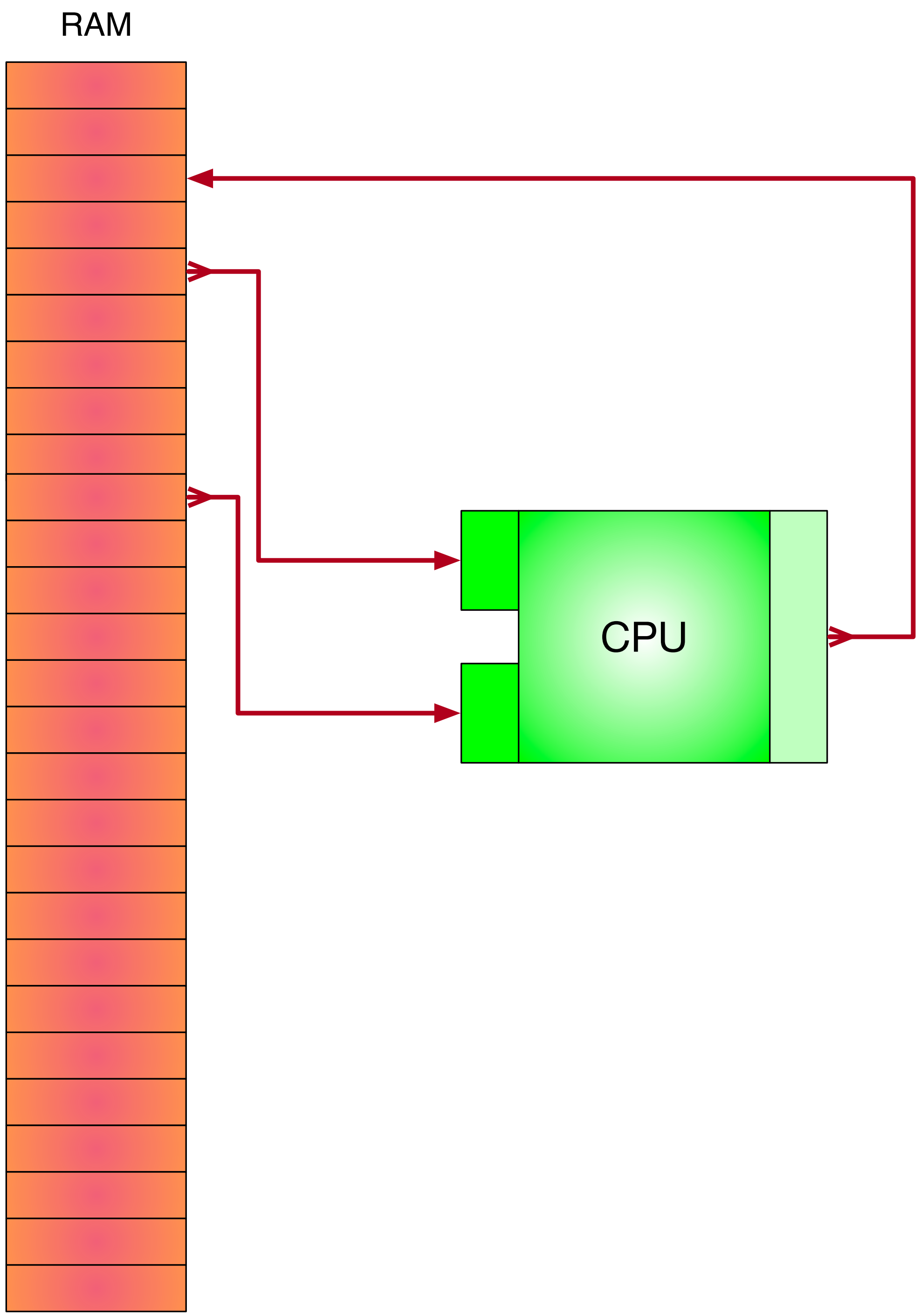
Thomas Schwarz, SJ

Nature of Computing

- Computers give to their users the impression of simplicity
  - Even though that simplicity only existed in the very beginning
  - We can imagine a computer in the simple RAM model



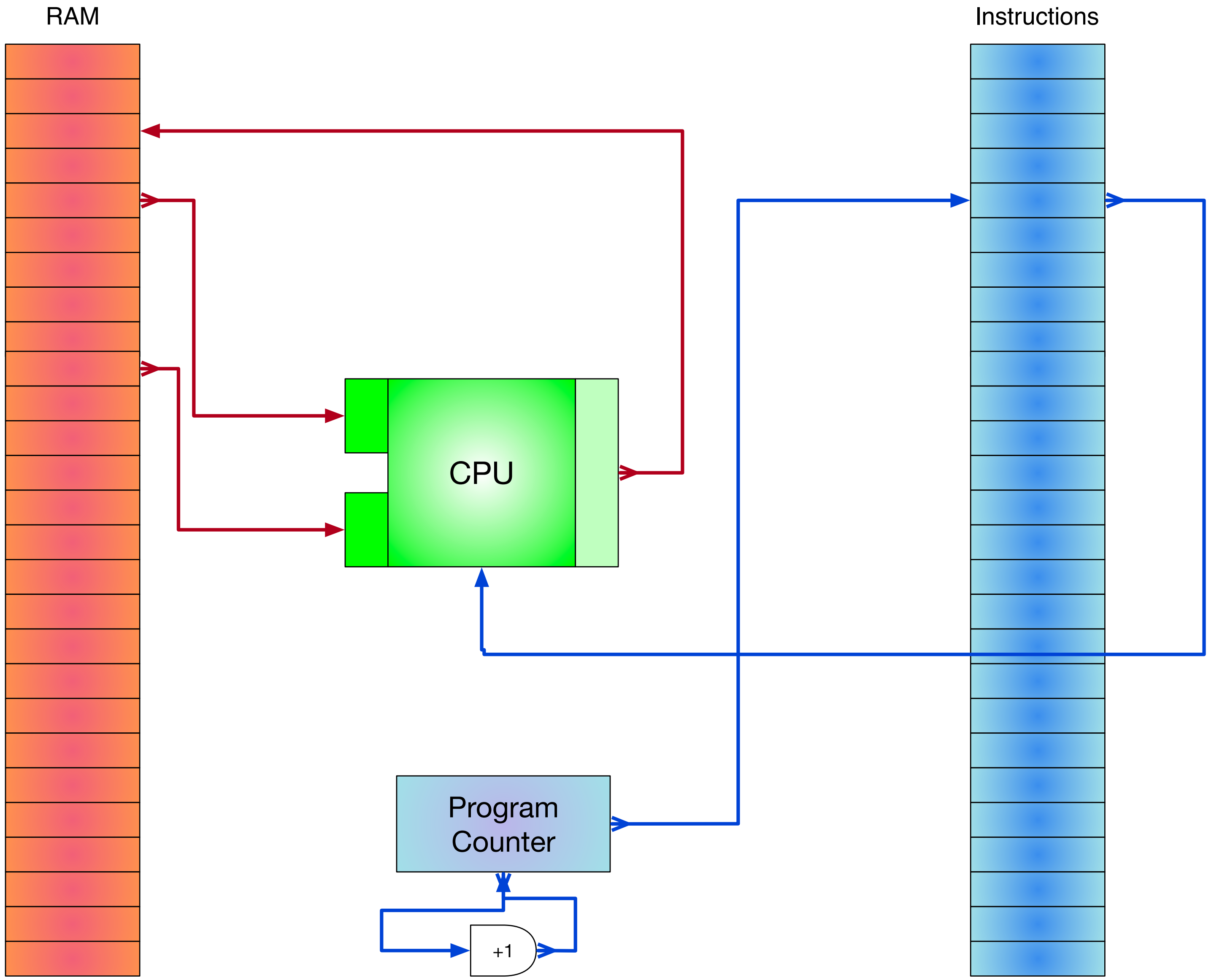
Nature of Computing



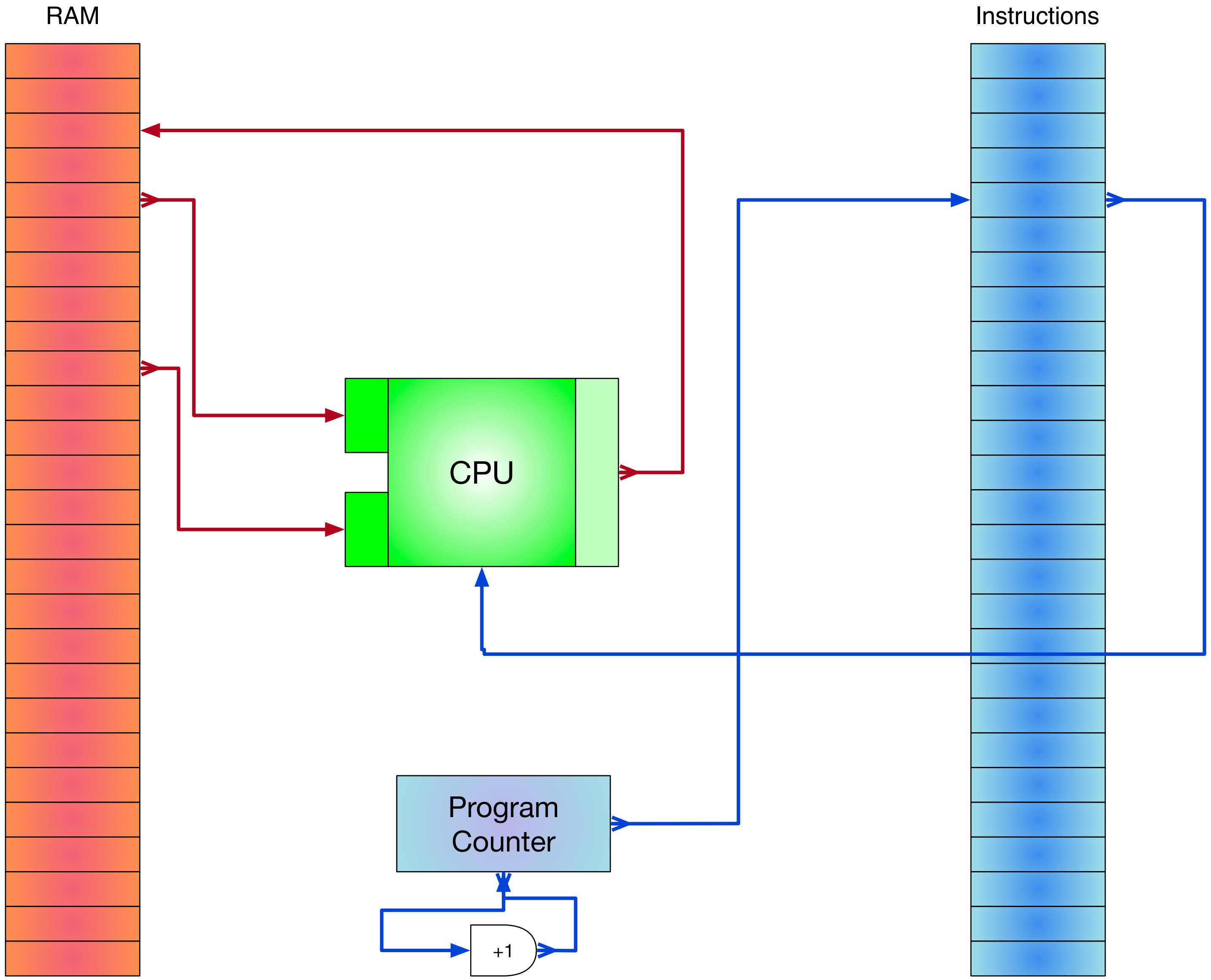
- Processing usually processes data stored in cells in Random Access Memory (RAM)
- Typically two operands taken from different cells yield a new value,
- which is stored in a RAM cell

Nature of Computing

- The instructions are stored elsewhere
    - Let's imagine a separate RAM (Harvard architecture)

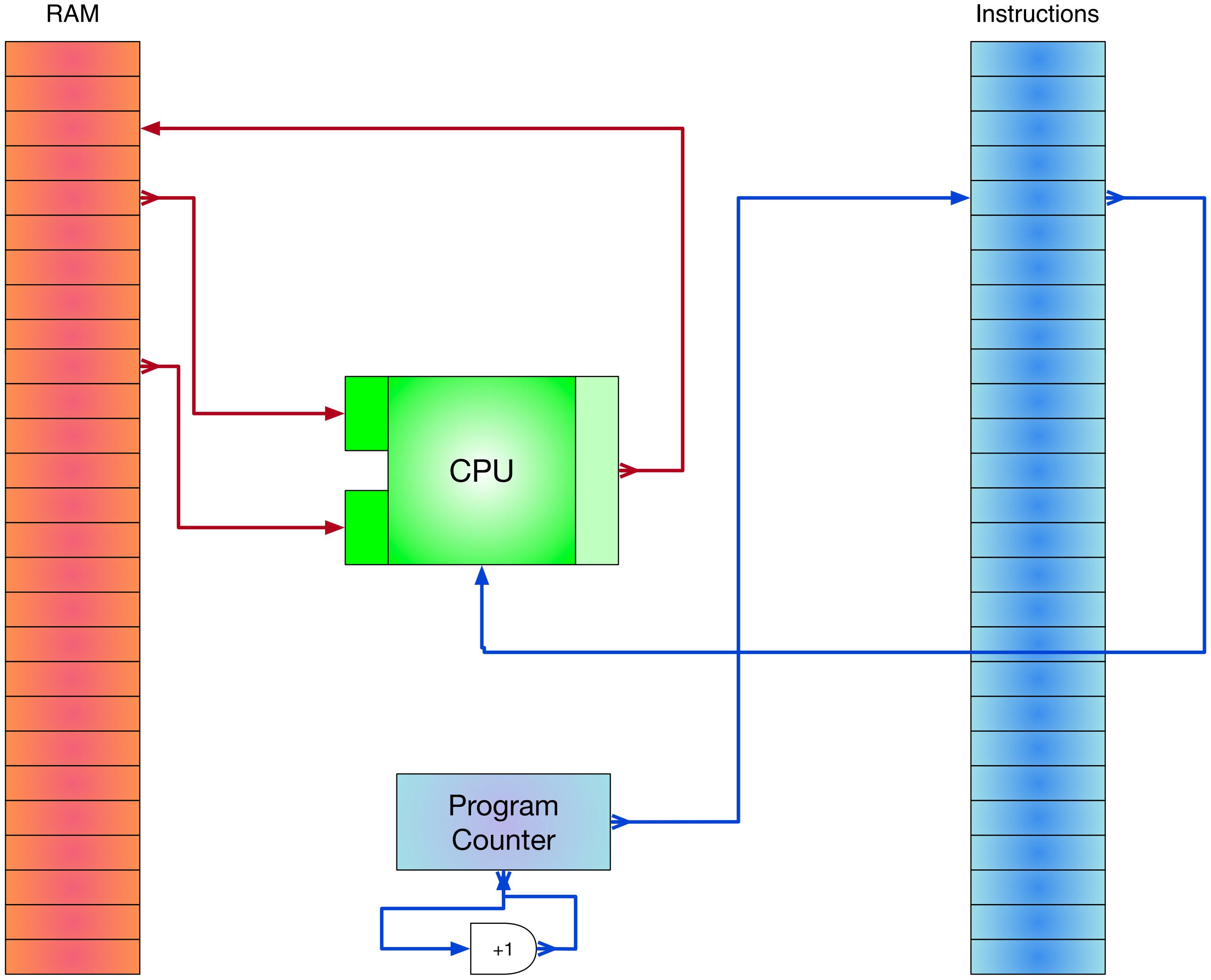


Nature of Computing



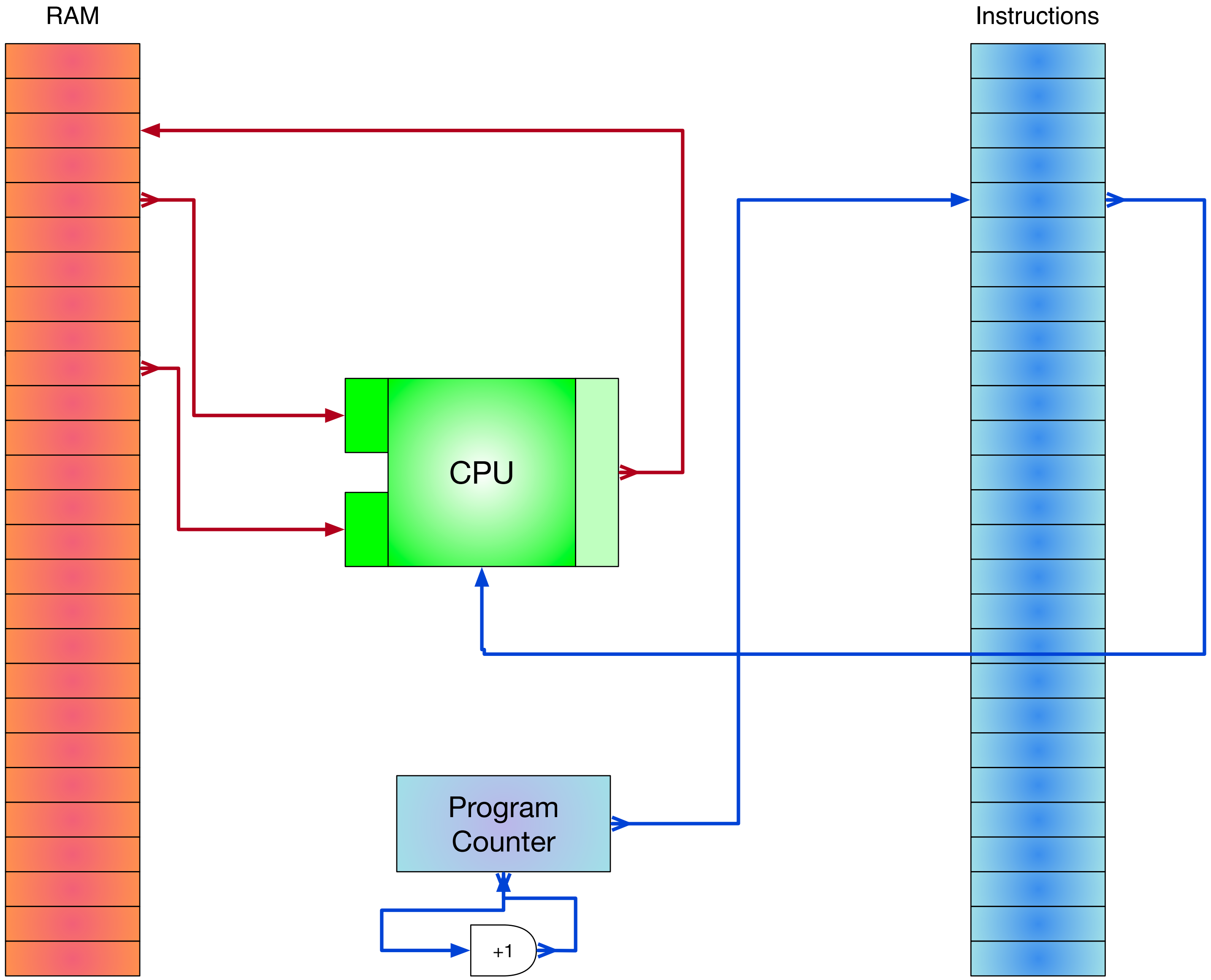
- A program counter selects the instruction
- Which is then guided to the CPU

Nature of Computing



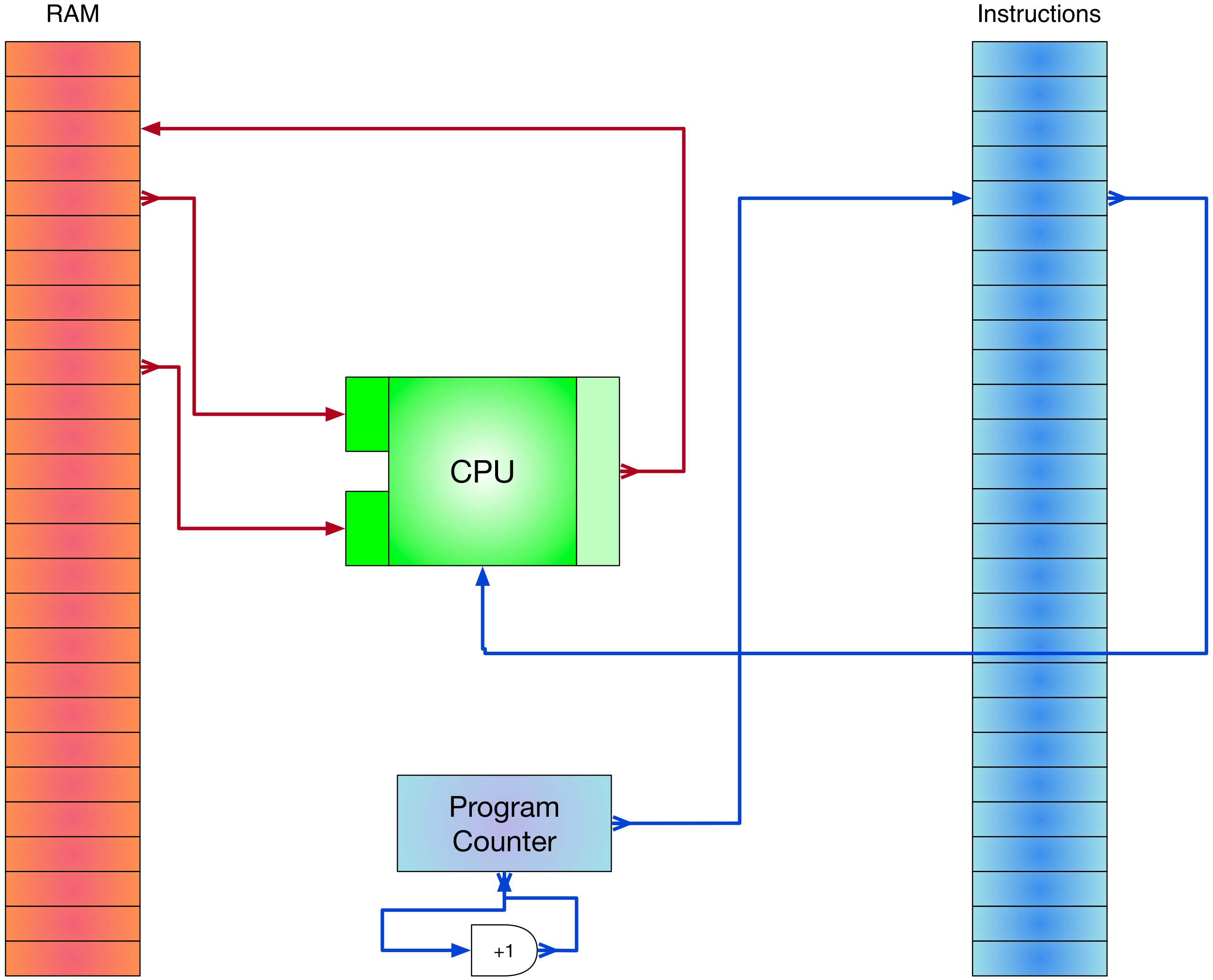
- The CPU decodes the instruction
  - Instructions are binary arrays in machine code

Nature of Computing



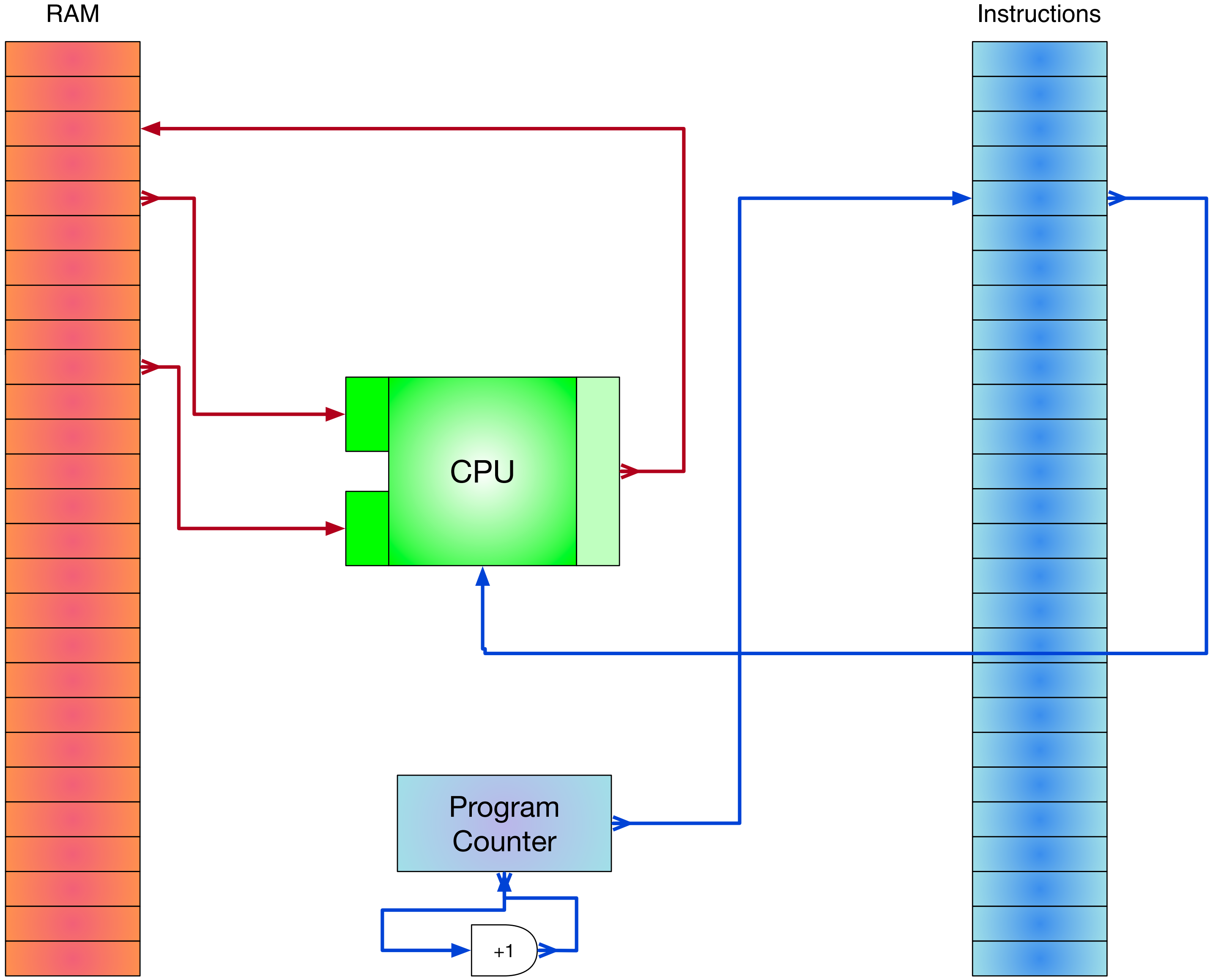
- After the instruction has been decoded, it is executed
  - By fetching the operands from data RAM, execution proper and storing the result elsewhere in RAM

Nature of Computing



- After execution
  - The program counter is set to the next instruction
  - Usually by incrementing the program counter

Nature of Computing



- And then the cycle starts again:
    - Fetch instruction
    - Execute instruction

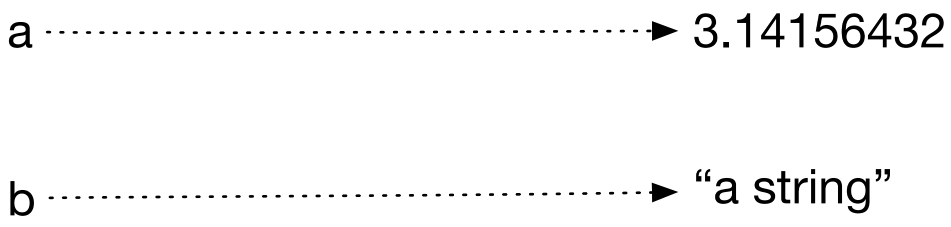
Nature of Computing

- Reality check:
  - RAM
    - RAMs are implemented in slow DRAM (50 nsec cycles) and fast caches (5 nsec cycles)
      - And that is still simplifying a lot
    - Typically, we have an instruction and a data cache
    - But RAM is stored by both types of data
  - Instruction types
  - Flow control

Variables

- Variables in Python are names for storage locations and objects stored
- You create a variable through the assignment operation

```
a = 3.14156432
b = “a string”
```

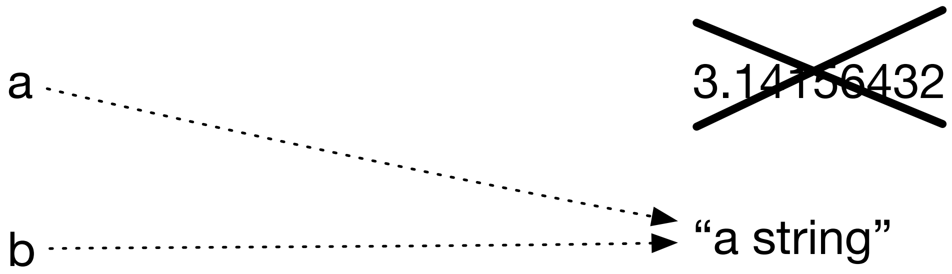


a 3.14156432

b “a string”

- Create two variables and assign values to them
- Variable *a* is of type floating point and variable *b* is of type string

```
a = b
```



a ~~3.14156432~~

b “a string”

- After reassigning, both variable names refer to the same value
- The floating point number is garbage collected

Variables

- A variable has a current type that can change
- A variable has a name
  - Names start with a letter or an underscore
  - Consists of letters, digits, and underscores
  - There are naming conventions that should be adhered to
    - The leading underscore and the double leading underscore mean something, so don't use them now

Variable Naming

- Universal variable naming conventions
  - Variable names should be readable
    - Ideally, you code sounds like English prose
  - They should be self-explanatory, but not verbose
    - This is part of developing good coding style and will take time

Variable Naming

- Two main schools:
    - Camelcase (medial capitals)
        - New words are separated by capitalizing the first letter
            - `annualInterestRate`
                - Small “a” because it is a value
                - Capital “I” and “R” to make the compound more readable
                    - in lieu of spaces that cannot be part of a variable name

Variable Naming

- Snake case
    - Separate components by a medial underscore
        - Example:
            - `annual_interest_rate`
        - Might be easier to read, but involves more key strokes

Variable Naming

- Constants:
  - Sometimes, variables never change their value
    - If this natural behavior, then we have a *constant*
      - Example: number of seconds in a year
  - Constants use all-capital letters separated by underscores (“Screaming snake case”)

```
NUMBER_OF_SECONDS_PER_YEAR = 365.25*24*60*60
```

- Using an expression is preferable to using a “magic number”

Variables

- Garbage collection
  - Python maintains information on use of storage locations
  - If there is no way to access this storage location
    - The storage location is “freed”
    - I.e.: It can be reused
      - Reuse will of course “destroy” the previous data

Variable Lifetime

- A variable is created by an assignment operation
    - Variable content and type can be changed through further assignments
- Variable is no longer accessible
    - Storage content is “freed”