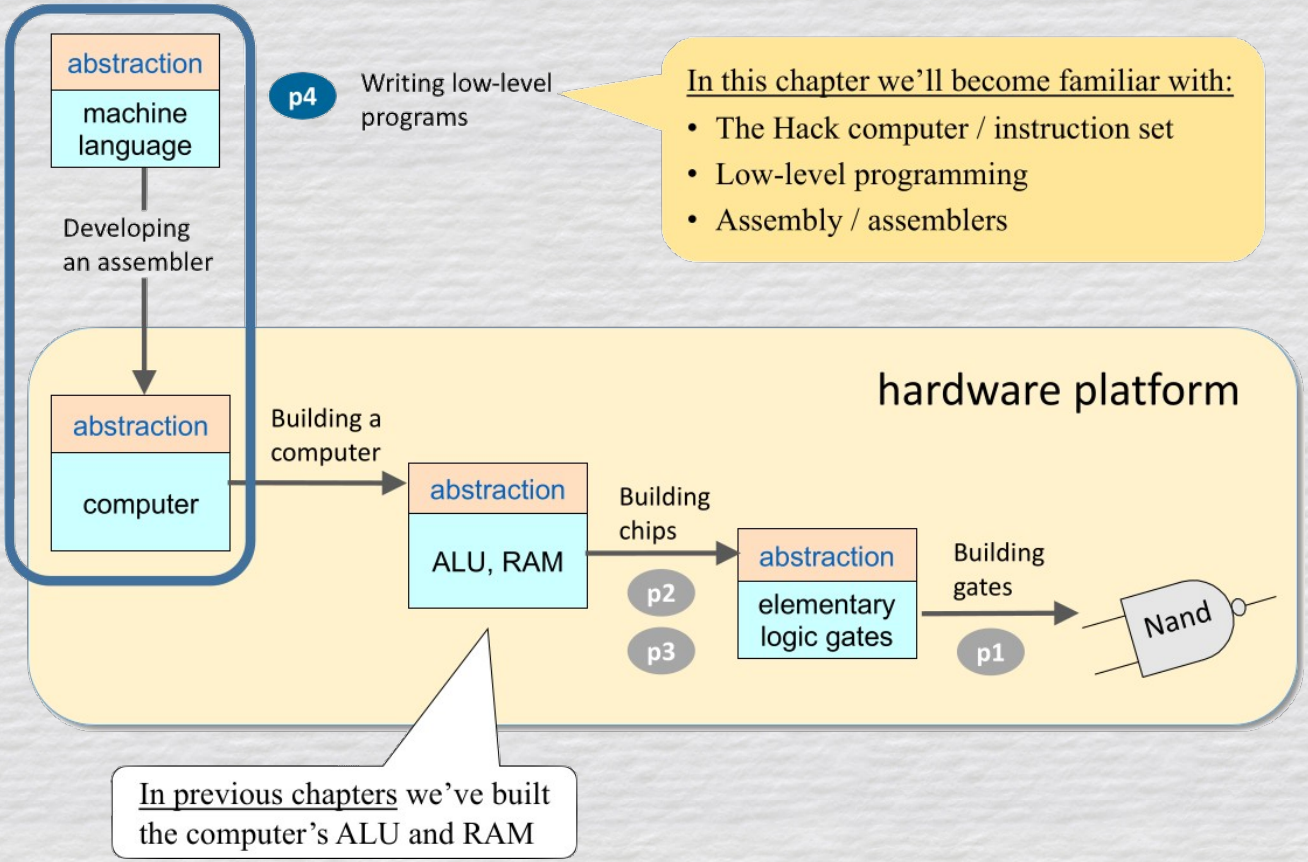
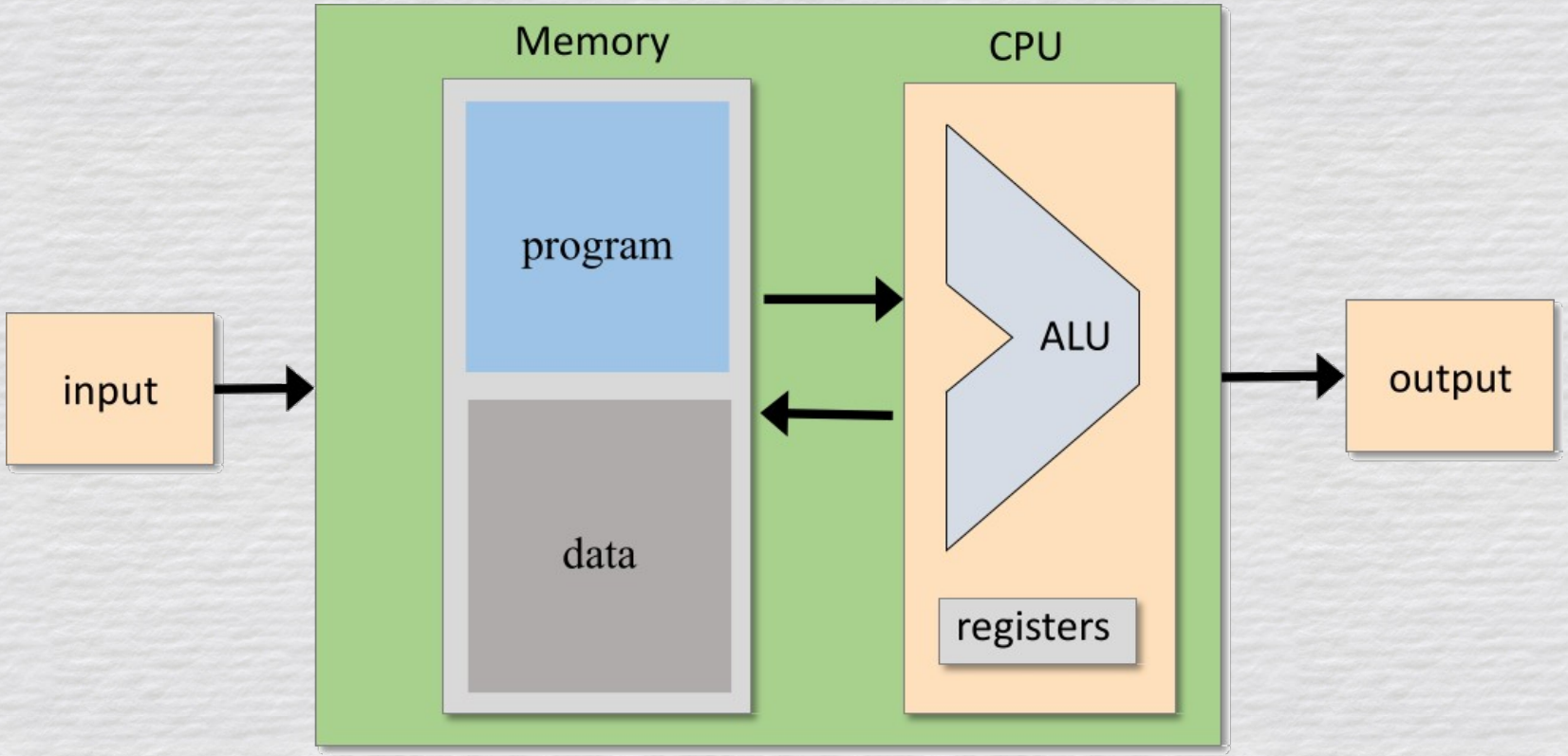
Computer Organization

HACK Memory System

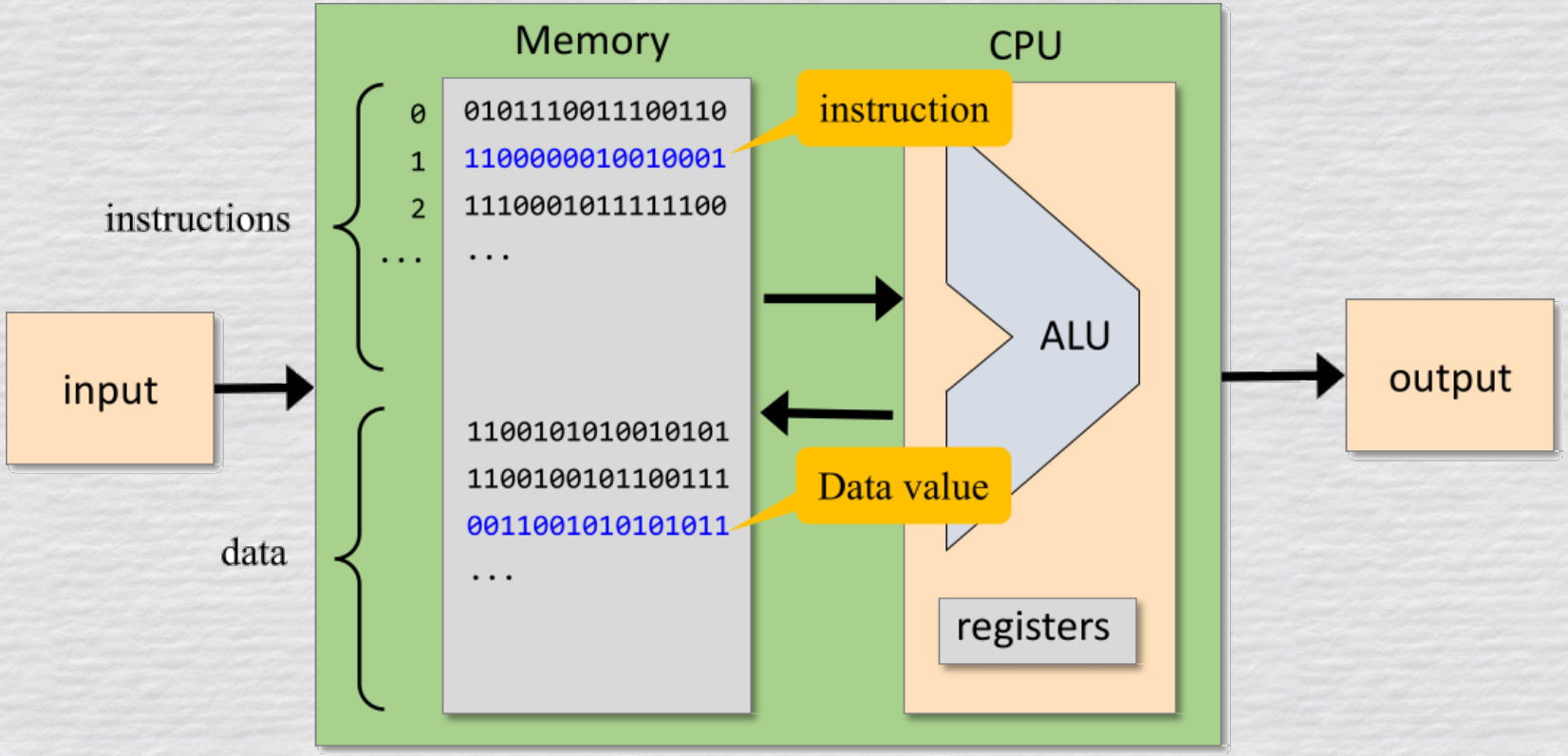
Roadmap


abstraction
machine language
Developing an assembler
abstraction
computer
p4
Writing low-level programs
In this chapter we'll become familiar with:
• The Hack computer / instruction set
• Low-level programming
• Assembly / assemblers
hardware platform
Building a computer
abstraction
ALU, RAM
Building chips
p2
p3
abstraction
elementary logic gates
Building gates
p1
Nand
In previous chapters we've built the computer's ALU and RAM

Computer – Overview



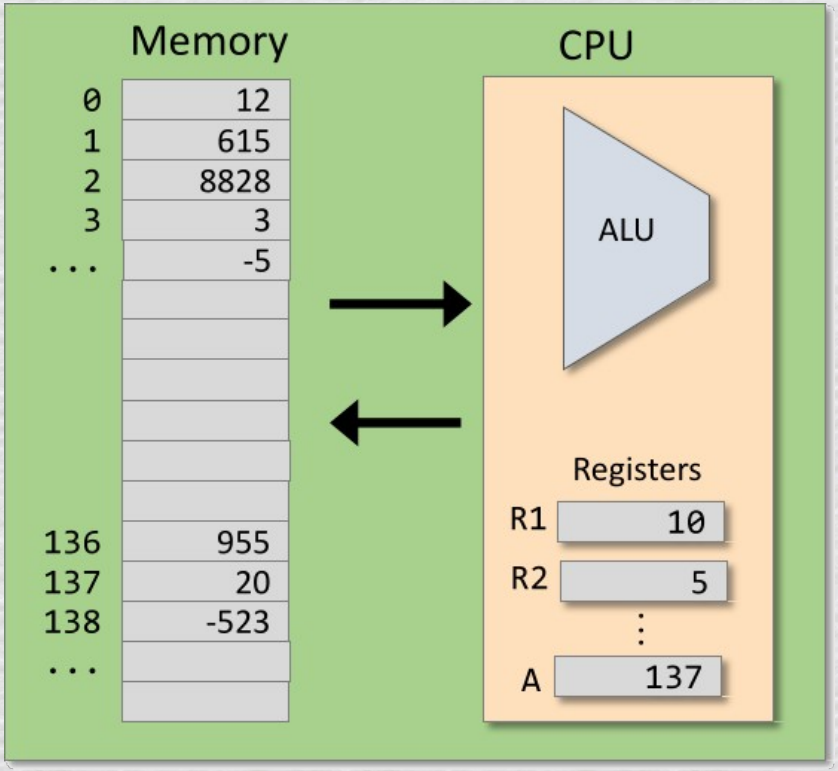
Computer – Overview



Computer – Registers

Memory Registers

- Many of these exist
- Slower
- Accessed via memory addresses

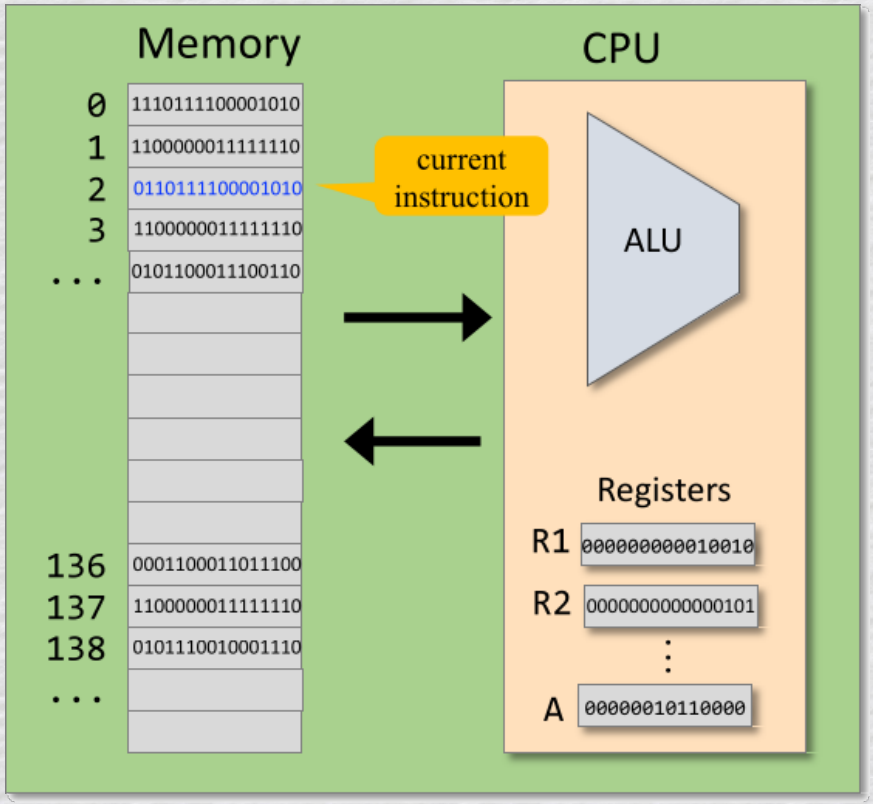


CPU Registers

- Very few of these exist
- Very fast
- CPU accesses these directly

Computer – Instructions

- Computer executes instructions primarily in an iterative fashion
- This only deviates if told to branch to a specific instruction in memory
- This behaviour is handled via the Program Counter



Branching

Unconditional Branching

- Executing instructions rather than the next iteration
- Simply jumps to a new instruction

Basic version

```
...
// Adds 1 to R1, repetitively
13  add R1,1
...
...
27  goto 13
...
...
```

- Line numbers
- Physical addresses

Symbolic version

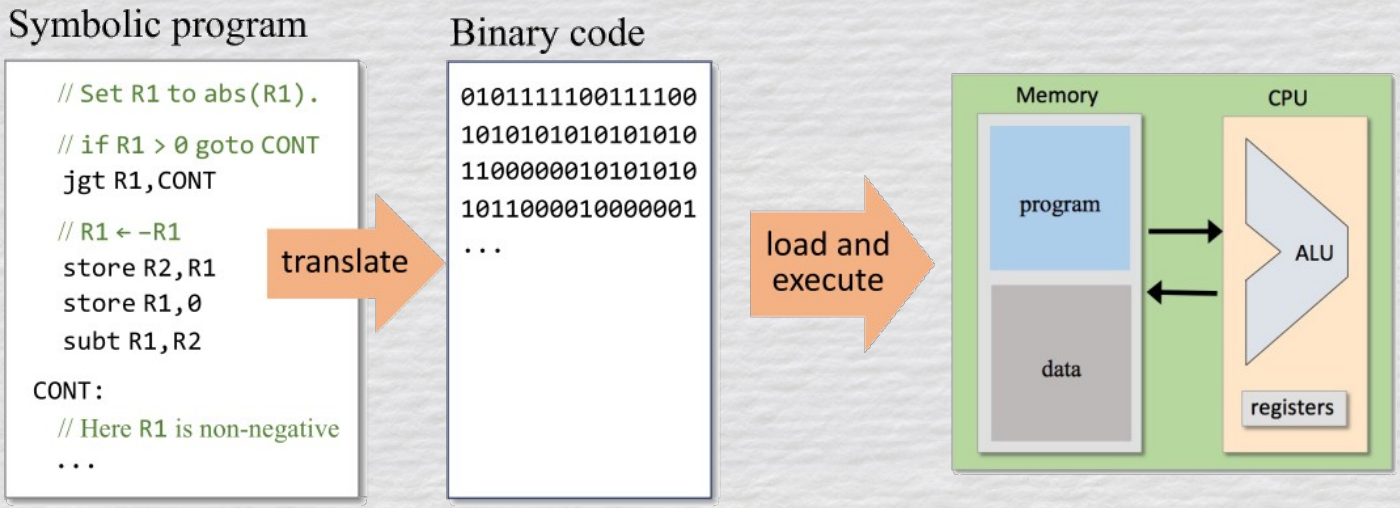
```
...
// Adds 1 to R1, repetitively
(LOOP)
    add R1,1
    ...
    goto LOOP
    ...
```

- No line numbers
- Symbolic addresses

Branching – cont.

Conditional Branching

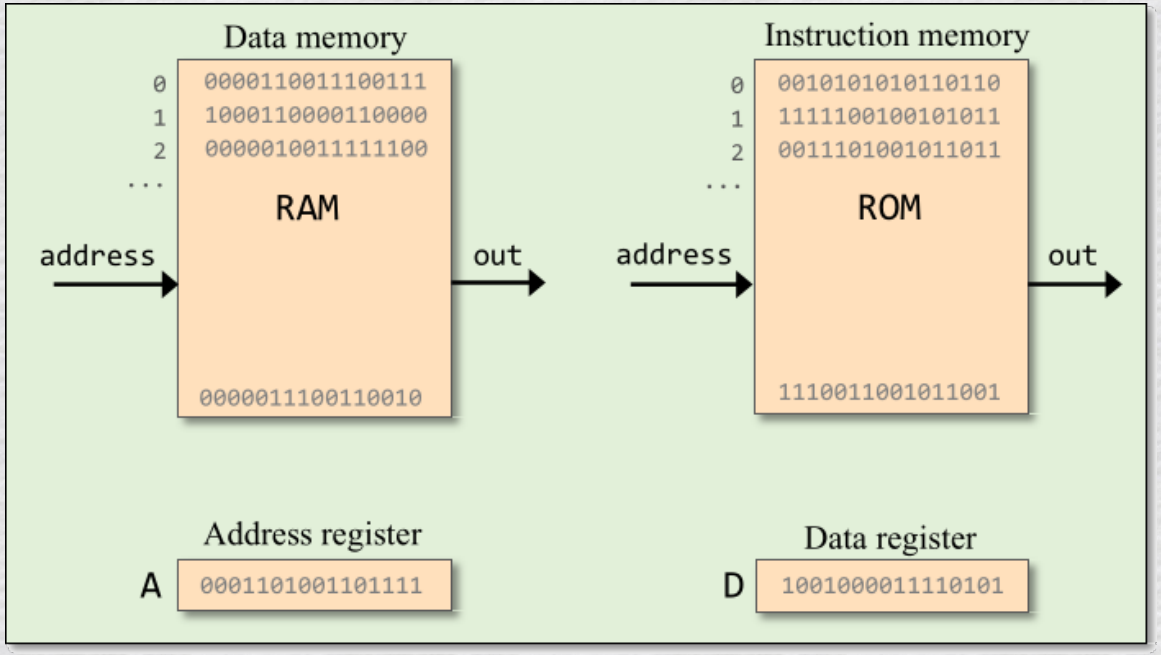
- Executing instructions based on specific conditions
- Able to deviate from a linear nature



Hack Computer

RAM, ROM, & Registers

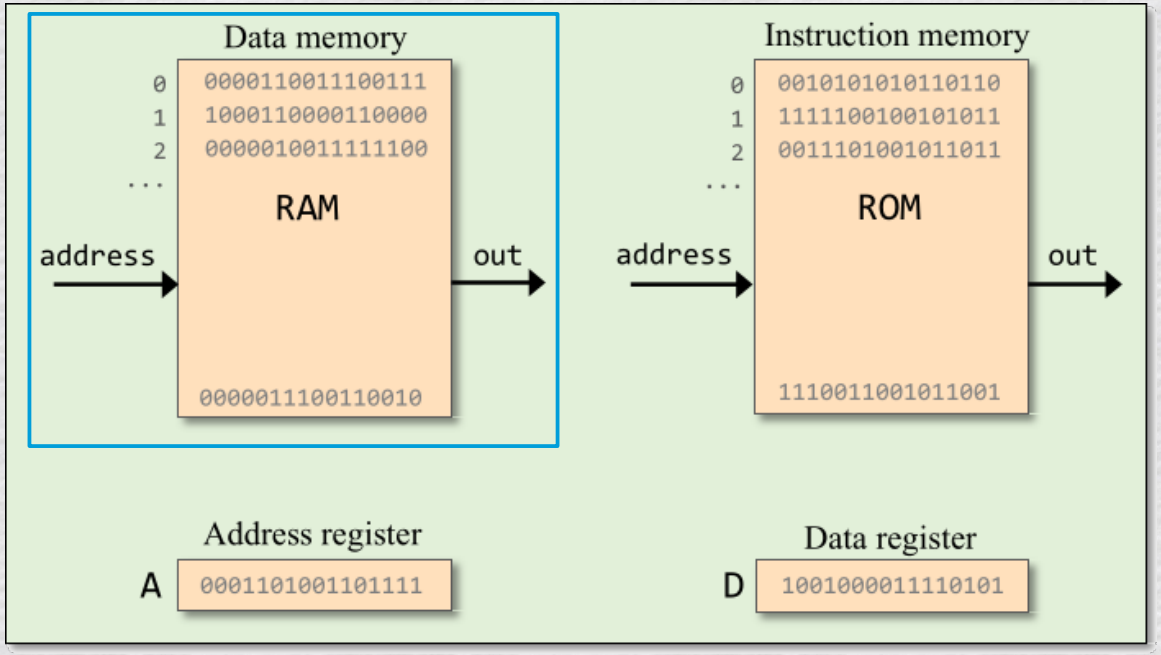
- All of these components comprise the memory of our overall system



Hack Computer – RAM

RAM

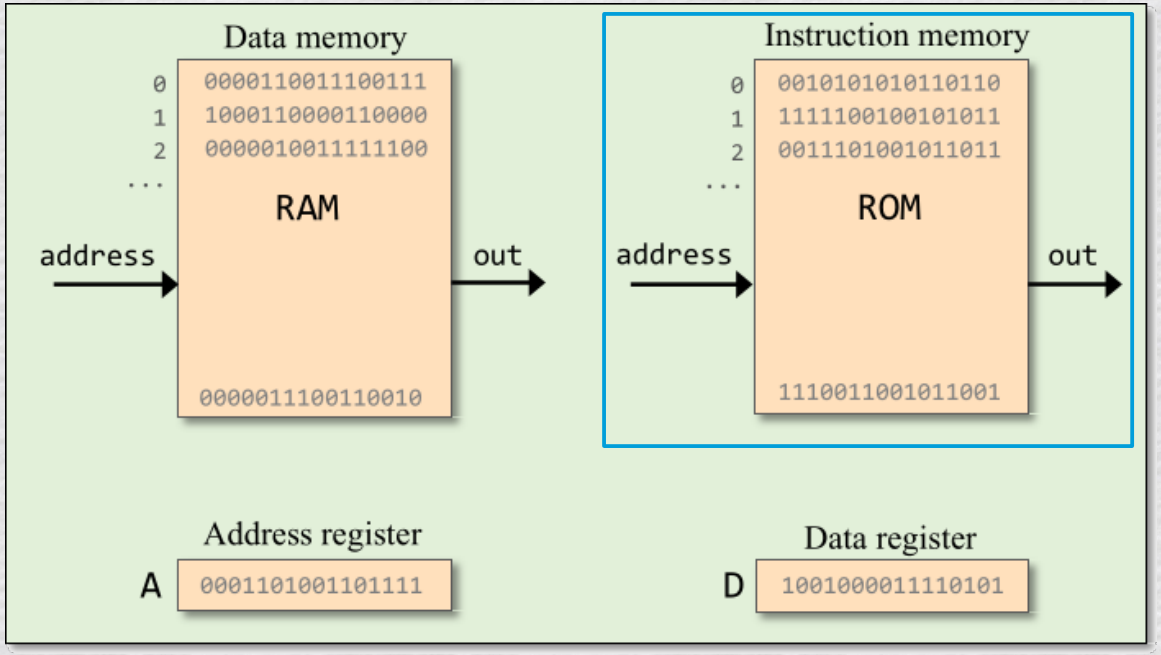
- Read-write data memory
- Addressed by the *A* register
- Selected Register, RAM[*A*], is seen as the *M* register



Hack Computer – ROM

ROM

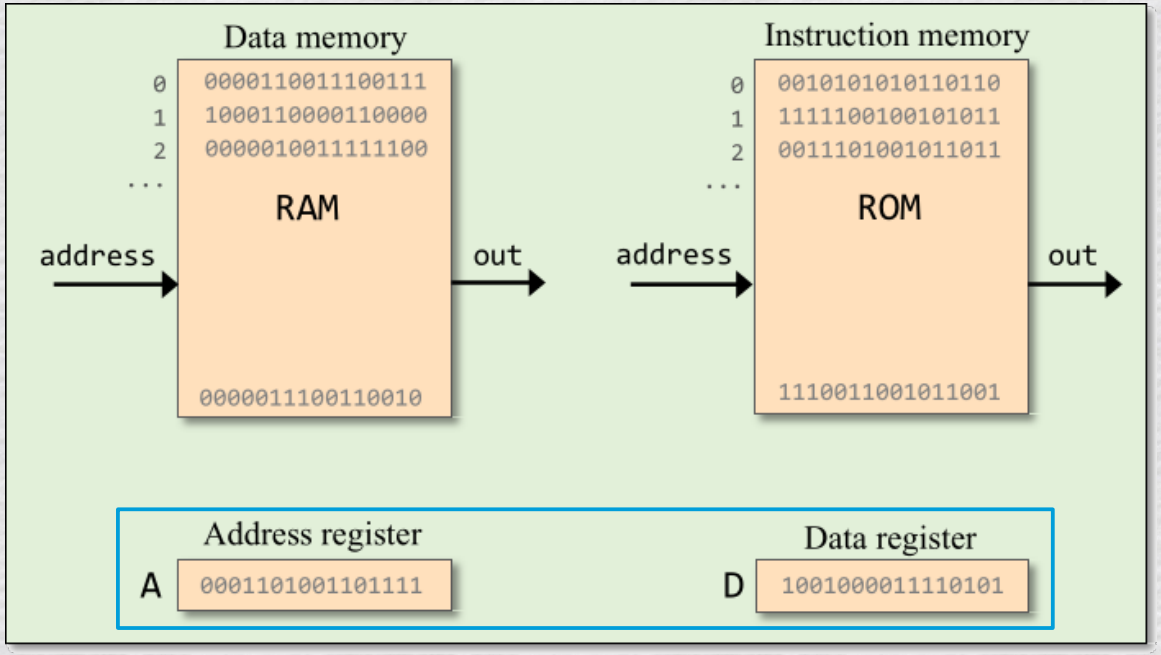
- Read-only instruction memory
- Addressed by the *A* register
- Selected Register, ROM[*A*], contains the current instruction



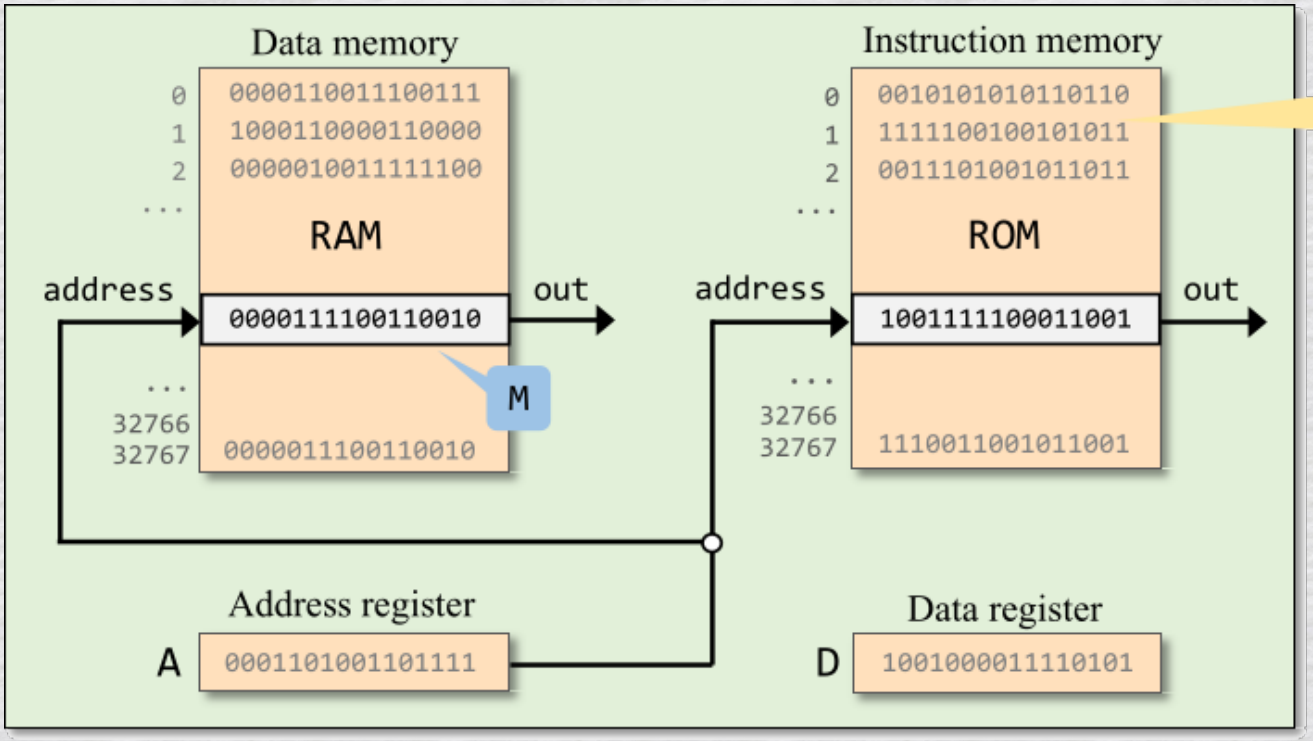
Hack Computer – Registers

Registers

- Data Register - *D*
  - Exists outside of the memory
- Address Register - *A*
  - Used to select ROM & RAM
- Memory Register - *M*
  - Current memory address



Hack Computer – Overview



Loaded with a sequence of 16-bit Hack instructions

(Conceptual, partial view of the Hack computer architecture)