University of Illinois at Urbana-Champaign
Dept. of Electrical and Computer Engineering

ECE 120: Introduction to Computing

LC-3 as a von Neumann Machine

Build an LC-3 Processor as a von Neumann Machine

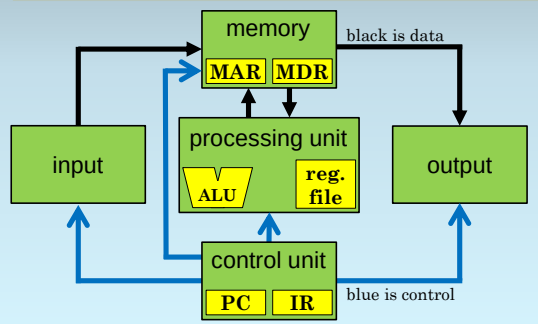
Let's talk about a specific von Neumann machine.

The **Little Computer-3 (LC-3) ISA**
- was developed by Patt & Patel
- as **an educational tool**.

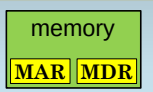
As Yale (Patt) says, it took them **three tries to get it right**, hence LC-3.

In our class, we will build up towards the LC-3 microarchitecture in Appendix C of P&P.

Recall the Five Parts of the von Neumann Model



The LC-3 Memory is 2^{16}×16-Bit



Let's start again with the memory.

In the LC-3 ISA,
- memory has 2^{16} **addresses**, and
- **16-bit addressability**.

Call this number X.

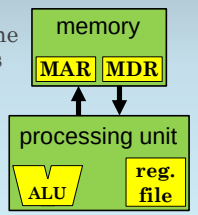
And call this Y.

How many bits in the MAR, X or Y? X (16)

How many bits in the MDR? Y (also 16)

The LC-3 ALU Supports ADD, AND, and NOT Operations

The ALU in the LC-3 supports three operations: **ADD, AND, and NOT**.



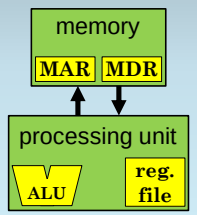
And remember DeMorgan's Law: A+B = [A'B']'

What if someone wants another function?

Apply logical completeness.

The LC-3 Register File Has Eight Registers

The register file contains **eight registers**.



The "R" is just for humans, of course.

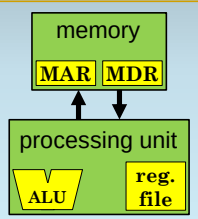
Guess what their names are…

~~Donner, Blitzen~~… **No. R0 through R7.**

The LC-3's Word Size is 16 Bits

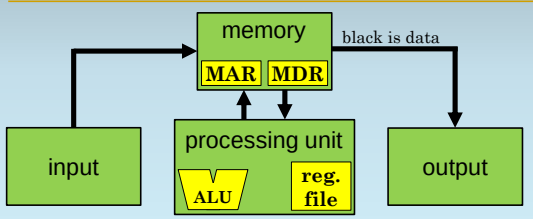
The LC-3's **word size is 16 bits**.

The **ALU operates on 16-bit operands**.



Each register in the register file **stores 16 bits**.

The LC-3 Includes a Keyboard and a Display



The LC-3 has one **input** device: **a keyboard**.

And one **output** device: **a monitor/display**.

Details of the LC-3 Input Device

keyboard input uses two registers
- **Keyboard Status Register (KBSR)**
  used to handshake when a key arrives
- **Keyboard Data Register (KBDR)**
  used to delivers keystrokes, coded as **ASCII**

Using these registers is a topic for ECE220
(see Ch. 8-9 of Patt & Patel).

Details of the LC-3 Output Device

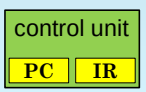
display output also uses two registers
- **Display Status Register (DSR)**
  used to handshake (processor must wait for the display!)
- **Display Data Register (DDR)**
  used to send characters to print, coded as **ASCII**

Again, using these is a topic for ECE220
(see Ch. 8-9 of Patt & Patel).

The LC-3 Also Has a Control Unit

Recall that
- the **program counter (PC)** stores the address of the next instruction, and
- LC-3 memory is **$2^X \times Y$-bit**, where **X** and **Y** are both 16.

How many bits in the PC, X or Y? X (16)

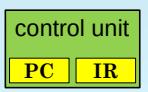


What About the IR?

Recall that the **instruction register (IR)** stores the encoded bits of the instruction being executed.

How many bits in the IR?

Did you read ahead? If not, you can't know the answer yet.



What About the IR?

How do we encode instructions?

The ISA defines a representation.

Instructions may require a variable number of bits (as in x86).

However, **in the LC-3 ISA, every instruction requires 16 bits**.

This design choice is **deliberately equal to the addressability of the memory** so that each memory location holds one instruction.

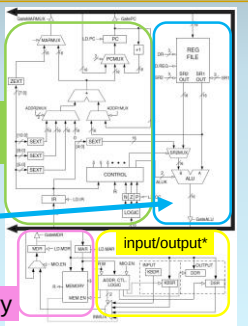
So, yes, the **IR requires 16 bits**.

A Datapath for an LC-3 Processor

Here's a diagram of a datapath for an LC-3 processor (Patt and Patel Figure C.3).

The heavy black line is a 16-bit bus.

*Again, we leave the details of I/O for ECE220.



Control Unit Elements for an LC-3 Processor

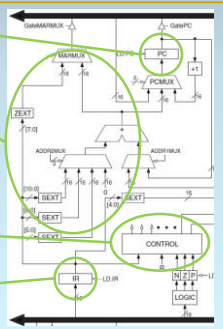
Program Counter (PC)

Miscellaneous instruction execution logic

Let's take a closer look at the control unit.

FSM generating control signals

Instruction Register (IR)

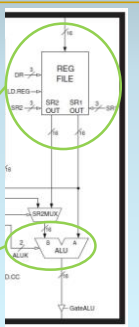


Processing Unit Elements for an LC-3 Processor

And a closer look at the processing unit.

Register file with eight 16-bit registers (two reads and one write per cycle)

Arithmetic Logic Unit (ALU) supports ADD, AND, NOT (all 16-bit, 2's comp. ADD)

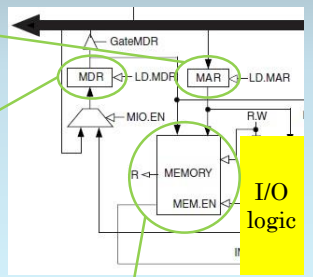


Memory Elements for an LC-3 Processor

Memory Address Register

Memory Data Register

And a closer look at the memory.



I/O logic

2^{16}×16-bit memory