1. Building a computer from parts
    a. You know enough at this point to build your own computer
        i. Add two numbers

        ii. Implement other operations like subtract, AND, OR, XOR, so on

        iii. Calculate a running sum of numbers

        iv. Add based on values stored somewhere

2. Von Neumann architecture
    a. Almost all current machine designs based on concepts developed by John von Neumann

    b. Architecture based on following three key concepts (according to Stallings)

3. Tasks of a computer, as defined by Stallings (from before)
    a. Move data

    b. Process data

    c. Store data

    d. Control

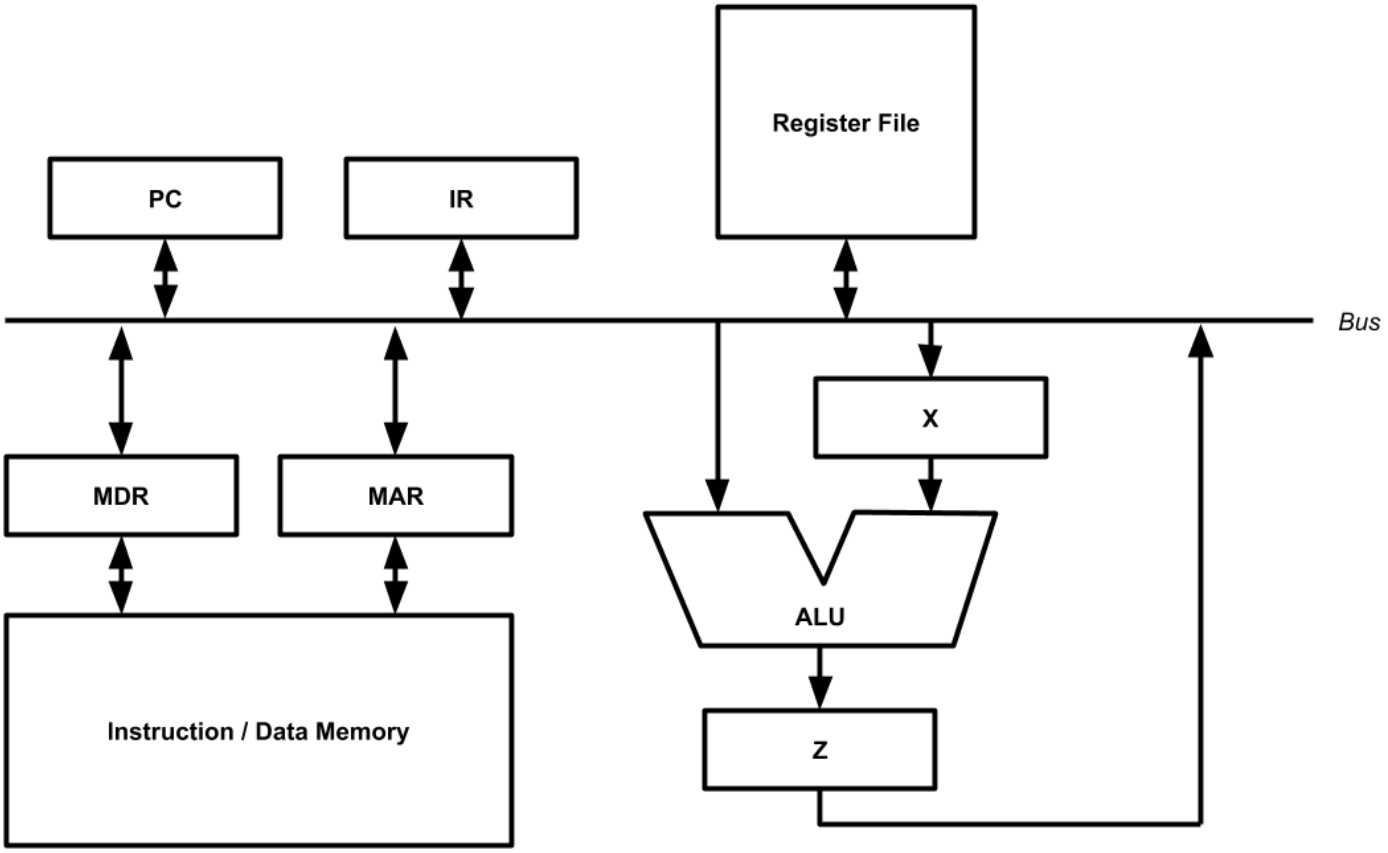
4. Putting together a basic CPU

        i. Alternative

    b. Let's have registers

    c. Keeping track of state

5. Single bus and executing instructions
    a. Simplistic single bus CPU below

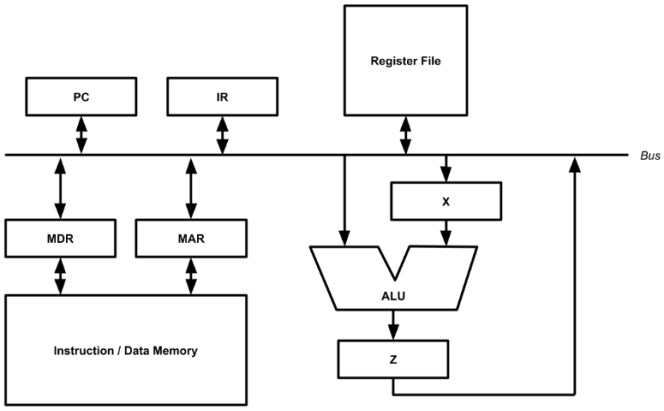


    b. Sequence of actions
        i. Fetch

        ii. Decode

        iii. Execute

c. Example – let's add two memory locations and place result in register file



i. Get value at first memory location
   1. Address of first operand placed on bus from IR
   2. MAR takes in value, memory returns desired value to MDR
   3. MDR places its value on bus, X takes in value for temporary storage

ii. Get value at second memory location
   1. Address of second operand placed on bus from IR
   2. MAR takes in value, memory returns desired value to MDR
   3. MDR places its value on bus

iii. Add things together and place in register file
   1. ALU takes in current value on bus and X, places it output in Z
   2. Z places its value on bus, register file takes in value
   3. IR places register address on bus, register file takes in address

d. Can see that one bus is a huge limiting factor
    i. Lots of contention for the bus, many things want to use it

    ii. Every time we use the bus for something else, need another clock cycle

    iii. Idea

    iv. Solution

e. Couple of reasons why we can't reduce to 1 cycle
    i. Must increment PC to reach next instruction

    ii. Complex addressing modes require multiple trips to memory