Instruction Cycle

Lecture 04

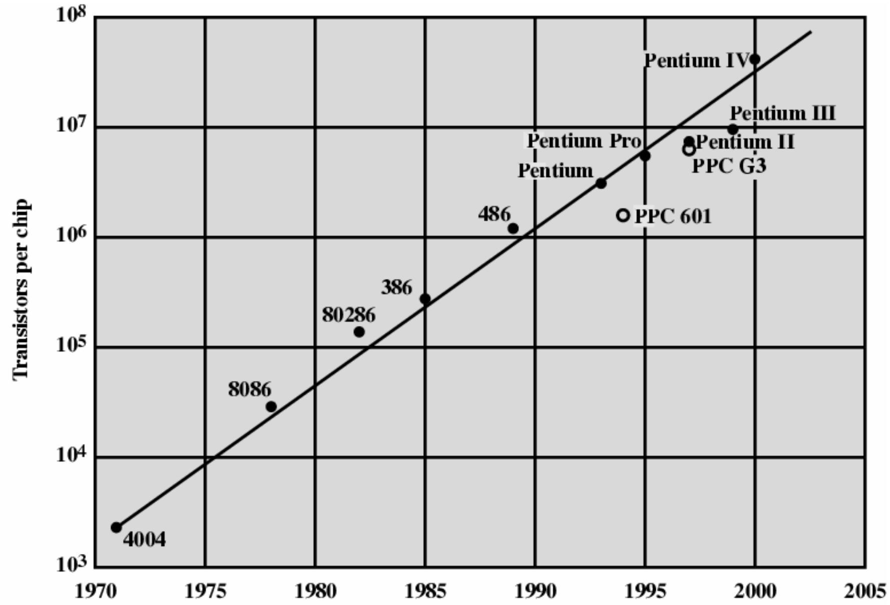
Generations of Computer

- Vacuum tube - 1946-1957
- Transistor - 1958-1964
- Small scale integration - 1965 on
  - ➢ Up to 100 devices on a chip
- Medium scale integration - to 1971
  - ➢ 100-3,000 devices on a chip
- Large scale integration - 1971-1977
  - ➢ 3,000 - 100,000 devices on a chip
- Very large scale integration - 1978 -1991
  - ➢ 100,000 - 100,000,000 devices on a chip
- Ultra large scale integration – 1991 -
  - ➢ Over 100,000,000 devices on a chip

Moore’s Law

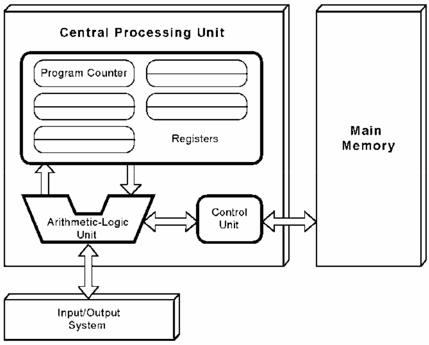
- Increased density of components on chip
- Moore’s Law (1965) co-founder of Intel
  - Gordon Moore, Intel Co-founder
  - “The density of transistors in an integrated circuit will double every two year.”
- Since 1970’s development has slowed a little
  - Number of transistors doubles every 18 months
  - Cost of a chip has remained almost unchanged
- Higher packing density means shorter electrical paths, giving higher performance

Growth in CPU Transistor Count



The von Neumann Model

- The computers employ a fetch-decode-execute cycle to run programs as follows:
  - The control unit fetches the next instruction from memory using the program counter to determine where the instruction is located
  - The instruction is decoded into a language that the ALU can understand.
  - Any data operands required to execute the instruction are fetched from memory and placed into registers within the CPU
  - The ALU executes the instruction and places results in registers or memory



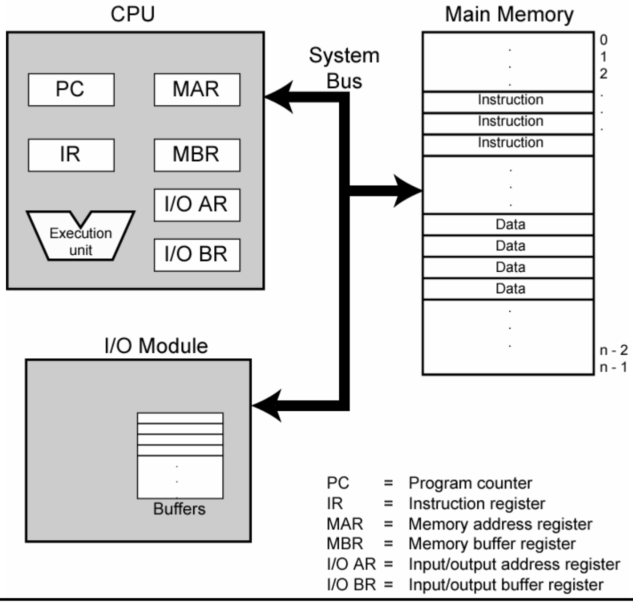
Computer Components

- Almost all computer designs are based on concept developed by John von Neumann at the institute for Advanced Studies (IAS)
- The key concepts of von Neumann Architecture are
  - Data and instructions are stored in a single read-write memory
  - The contents of memory are addressable by location without regard the type of data contained therein
  - Execution occurs in a sequential fashion- unless modified from one instruction to the next
- A small set of basic logic components that can be combined in various ways to store binary data and to perform arithmetic and logic operation on that data

What is a program?

- A general purpose configuration of hardware is to perform various functions on data depending on control signal applied to the hardware
  - ➢ The system accepts data and produces results
  - ➢ Rewiring of hardware is not required, but the set of control signal is applied “ programming?”
- A sequence of steps
  - ➢ For each step, an arithmetic or logical operation is done on some data
  - ➢ For each operation, a different set of control signals is needed

Computer Components: Top Level View



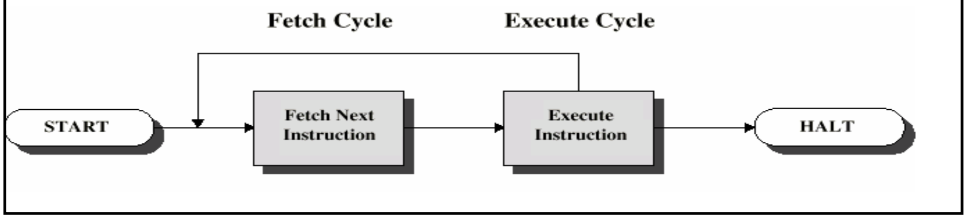
PC	=	Program counter
IR	=	Instruction register
MAR	=	Memory address register
MBR	=	Memory buffer register
I/O AR	=	Input/output address register
I/O BR	=	Input/output buffer register

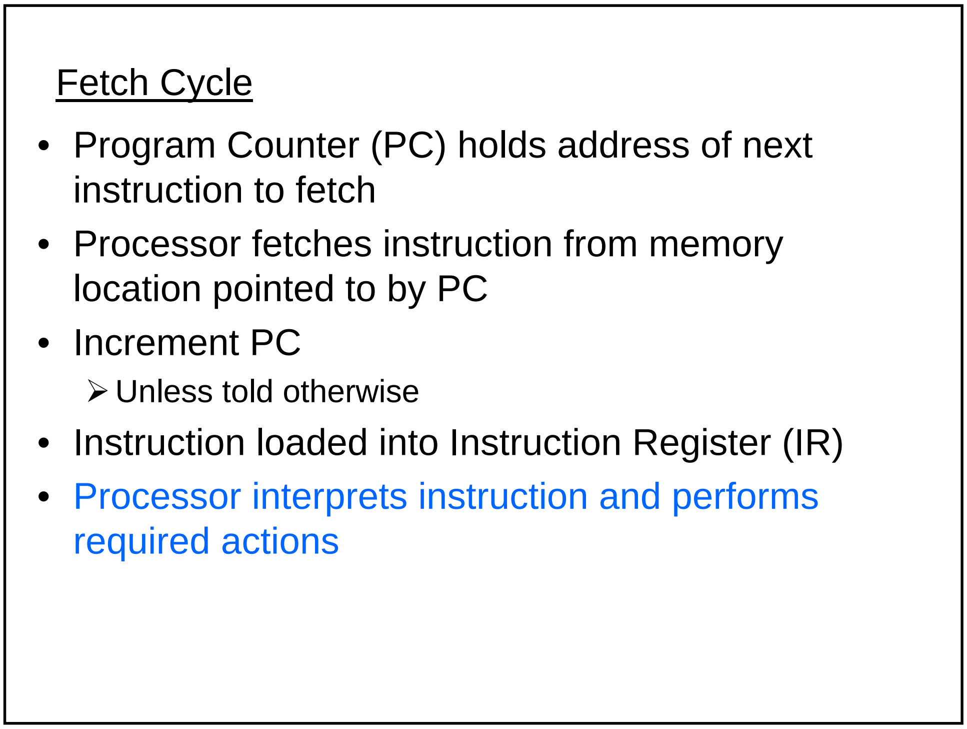
Instruction Cycle

- Basic function performed by a computer is execution of a program
  - ➢ consists of set of instructions stored in the memory
  - ➢ Processor executes the instructions specified in the program
- Instruction processing consists of two steps:
  - ➢ Processor reads – Fetches from memory one at time
  - ➢ Execute each instruction
- Program execution – repeating the steps over and over
  - ➢ Processing required for a single instruction is called " instruction cycle"

Instruction fetch and execute

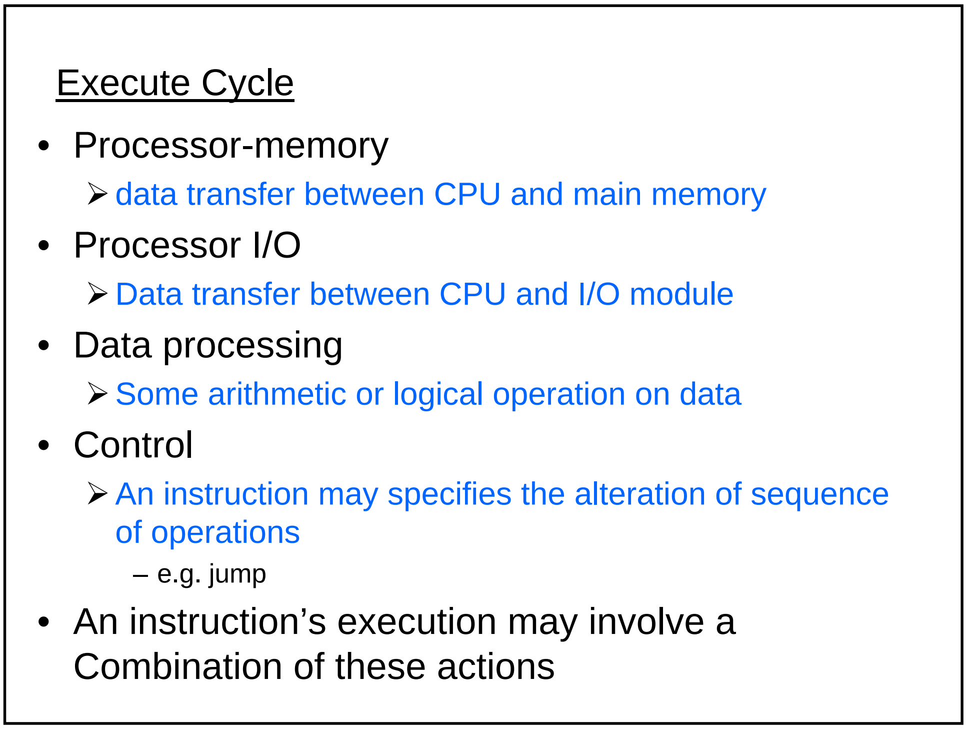
- At the beginning of each instruction cycle
  - ➢ Processor fetches an instruction from the memory
    - – A register "PC"- program counter holds the address of the instruction to be fetched next
    - – Processor always increment the PC after each instruction unless told otherwise
  - ➢ The fetched instruction is loaded into a register located in the processor – " the instruction Register (IR)"
  - ➢ The instruction contains bits – to specify the action the processor is to take





Fetch Cycle

- Program Counter (PC) holds address of next instruction to fetch
- Processor fetches instruction from memory location pointed to by PC
- Increment PC
  - ➢ Unless told otherwise
- Instruction loaded into Instruction Register (IR)
- Processor interprets instruction and performs required actions



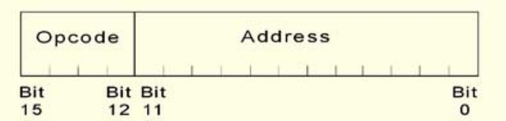
Execute Cycle

- Processor-memory
  - ➢ data transfer between CPU and main memory
- Processor I/O
  - ➢ Data transfer between CPU and I/O module
- Data processing
  - ➢ Some arithmetic or logical operation on data
- Control
  - ➢ An instruction may specifies the alteration of sequence of operations
    - – e.g. jump
- An instruction’s execution may involve a Combination of these actions

Example of Program Execution

Example

Add the contents of the memory at address 940 to the contents of the memory at address 941 and store the result in the next location



Internal CPU registers:

Program counter (PC): Address of next instruction

Instruction Register (IR): Current instruction

Accumulator (AC): Temporary Storage

Partial List of Opcodes:

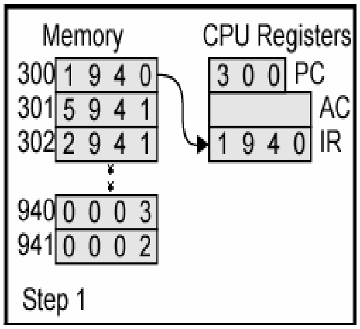
0001 = (1h) = Load AC from Memory

0010 = (2h) = Store AC to memory

0101 = (5h) = Add to AC from Memory

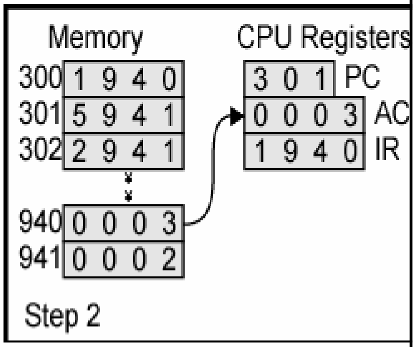
1. Program Execution

- PC contains 300 – the address of the first instruction
  - ➢ This instruction (the value 1940h) is loaded into the IR
  - ➢ PC is incremented (301)
- The process involve the use of
  - ➢ Memory Address Register (MAR)
  - ➢ Memory Buffer Register (MBR)



2. Program Execution

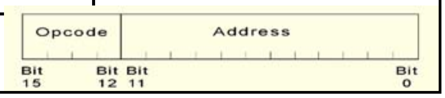
- The first nibble in the IR indicate that the AC is to be loaded
  - ➢Remaining 12 bits specifies the
    - – Address 940 from which data are to be loaded



1940h= 0001 1001 0100 0000

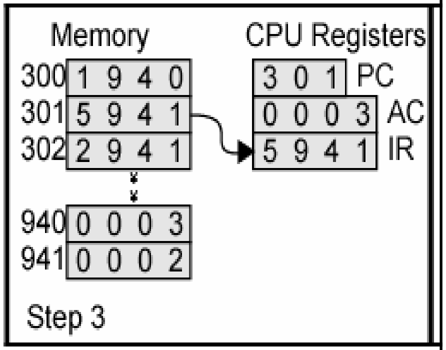
000	1001 0100 0000

Load contents to AC



Example of Program Execution

- The next instruction 5941 is fetched from the location 301
- PC is incremented



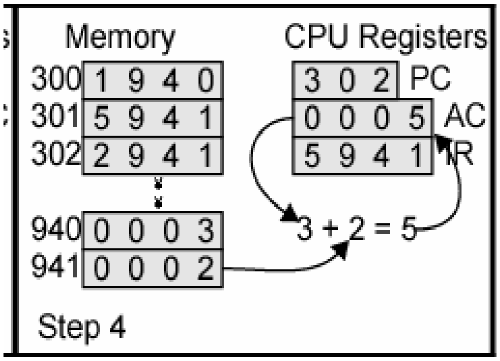
5941h= 0101 1001 0100 0001

010	1001 0100 0001

Add contents to AC

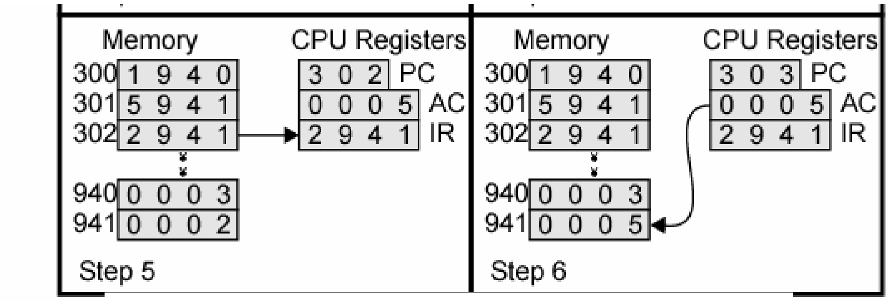
Example of Program Execution

- The old contents of AC and the contents of the location 941 are added
- The result is stored in AC



Example of Program Execution

- The next instruction 2941 is fetched from location 302
  - ➢ PC is incremented
- The contents of AC are stored in location 941



2941h= 0010 1001 0100 0001

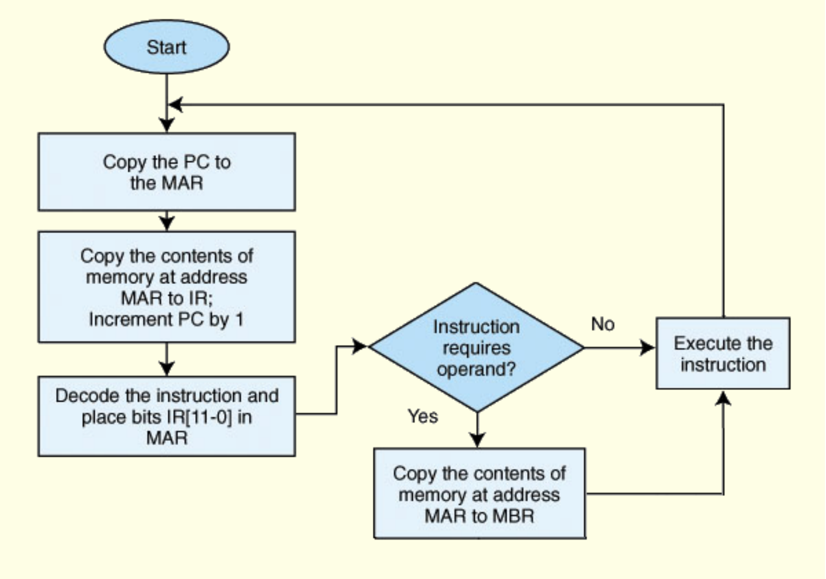
0010	1001 0100 0001

Store the contents of AC at address x

Instruction Processing

- The *fetch-decode-execute cycle* is the series of steps that a computer carries out when it runs a program.
  - ➢We first have to *fetch* an instruction from memory, and place it into the IR.
  - ➢Once in the IR, it is *decoded* to determine what needs to be done next.
  - ➢If a memory value (operand) is involved in the operation, it is retrieved and placed into the MBR.
- With everything in place, the instruction is *executed*.

Instruction Processing



Start
Copy the PC to the MAR
Copy the contents of memory at address MAR to IR; Increment PC by 1
Decode the instruction and place bits IR[11-0] in MAR
Instruction requires operand?
No
Yes
Execute the instruction
Copy the contents of memory at address MAR to MBR