Overview of Microcomputer Structure and Operation

Course Teacher:

Md. Fahmidur Rahman Sakib
Lecturer, Department of Computer Science & Engineering
Metropolitan University

Course ID: CSE 237
Course Title: Microprocessor and Interfacing

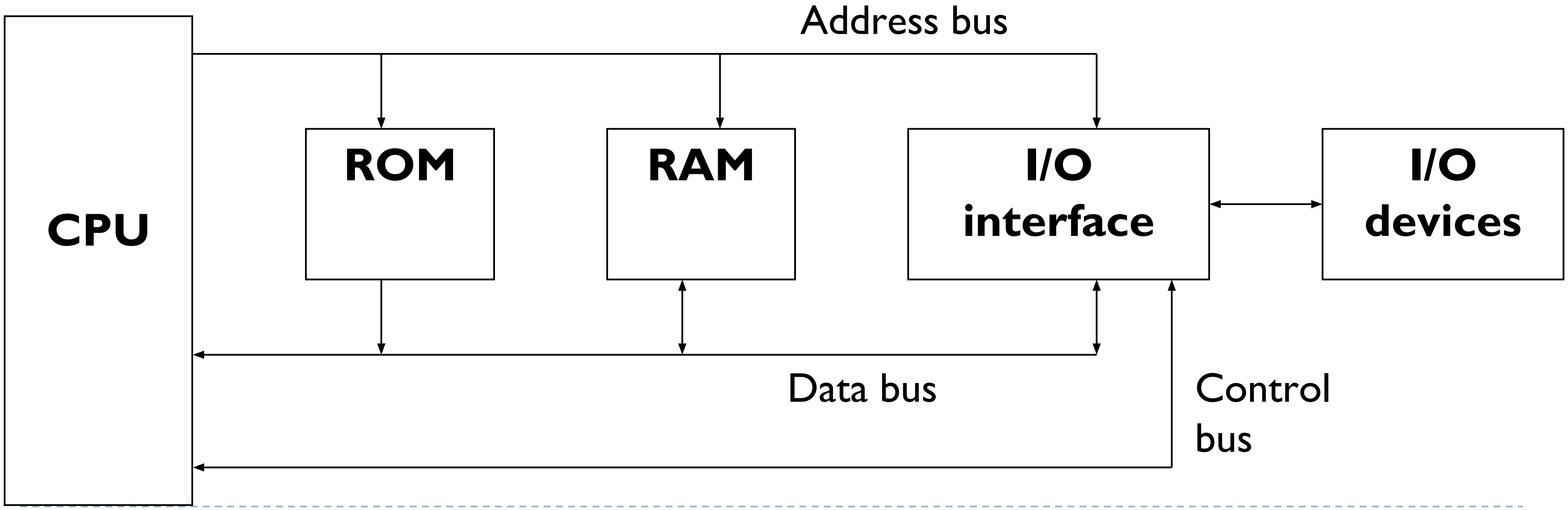
Lecture References:

- **Book:**
  - *Microprocessors and Interfacing: Programming and Hardware,* **Author:** Douglas V. Hall

Block Diagram of a Simple Microcomputer

Components of Microcomputer:

- CPU (Microprocessor)
- Memory (RAM, ROM etc)
- I/O
- System Buses:
  - Address bus
  - Data bus
  - Control bus



CPU - Central Processing Unit

- **FETCH :** Take in binary-coded instructions from memory
- **DECODE :** Analyze or make sense of the instructions
- **EXECUTE :** Carry out the instructions
- Controls overall operation of the computer

- **Important components:** Registers , ALU , Control Unit

Pentium D dual core processors

Memory

- This is where all the binary coded instructions and data are stored. Example: ROM, RAM etc.

RAM (Random Access Memory) :

- Can be read and written to anytime by the CPU.
- It is volatile memory. That means contents of RAM are erased when the power to the computer is turned off.

ROM (Read Only Memory) :

- Can only be read by the CPU.
- It is pre-loaded with data and software that never changes like computer's initial start-up instructions.
- It is non volatile memory. That means contents of ROM are **NOT** erased when the power to the computer is turned off.

I/O Unit

- **Input/output (I/O) units serve as a medium of communication between the user and the computer.**
- **Inputs** are the signals or data received by the system, and **outputs** are the signals or data sent from it.
- Devices that provide input or output to the computer are called ***peripherals***.
- For example:

  keyboard, **mouse** (input)

  display, **printer** (output)

  hard disk (both input & output)

System Bus

- System bus is made up of three types of bus :
  - Address Bus
  - Data Bus
  - Control Bus

- **WRITE operation :** When data is written onto memory location or an I/O port by the processor
- **READ operation :** When data is read from a selected memory location or an I/O port by the processor

Address Bus

- Carries memory address of the instructions which are to be executed
- Information transfer takes place from the MP to the memory or I/O elements. That is why address bus is ***Unidirectional***.
- On these lines the CPU sends out the address of the memory location or I/O port that is to be **written** to or **read** from
- The number of locations that the CPU can address is determined by the number of address lines

For example : microprocessor with 32 bit address bus can address 2^{32} memory locations

Data Bus

- It is a bidirectional bus
- Data can flow in both directions, that is, to or from the microprocessor.
- The size of the data bus varies from one microprocessor to another.
- Usually matches the *word length* of the microprocessor
- Usually a multiple of 8
- We talk of 4-bit, 8-bit, 16-bit , 32-bit and 64-bit processors which refers to the normal word length of the microprocessor

Control Bus

- It carries timing and control signals generated by the CPU that are used to synchronize operation of the individual microcomputer elements.

- It can carry many different signals. For e.g.
  - I/O Read
  - I/O Write
  - Interrupt
  - Memory read
  - Memory write

Fetching & Execution Cycles

The Fetch & Execute Cycle of the CPU is composed of three basic operations :

- **Fetch**
- **Decode**
- **Execute**

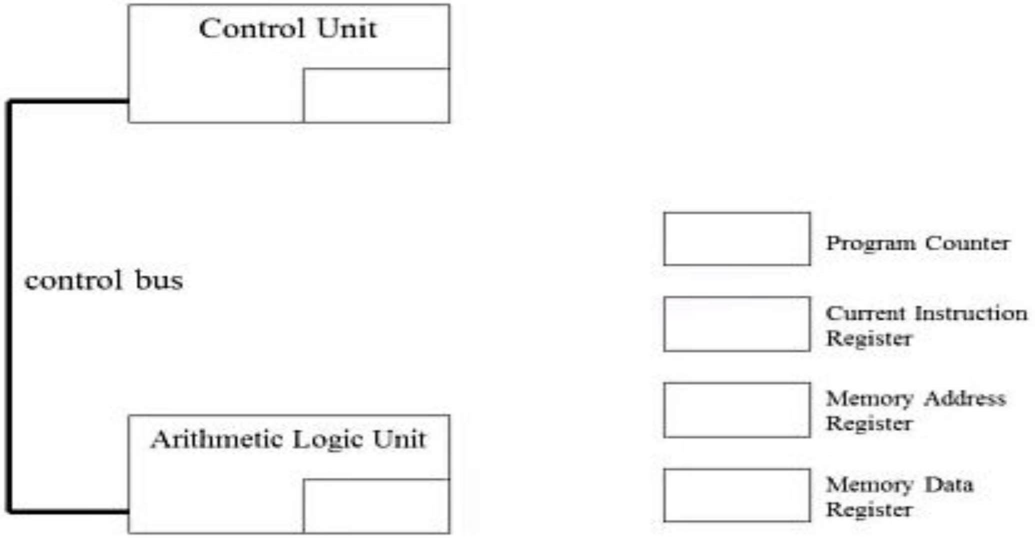
- **Fetch :**
  - The instruction required from memory is stored or copied in the instruction register.
  - Increments the program counter so that it points to the next instruction.

Fetching & Execution Cycles

- **Execute cycle**
  - The actual actions which occur during the execute cycle of an instruction.
  - Depend on both the instruction itself and the addressing mode specified to be used to access the data that may be required.

The CPU's Special Purpose Registers

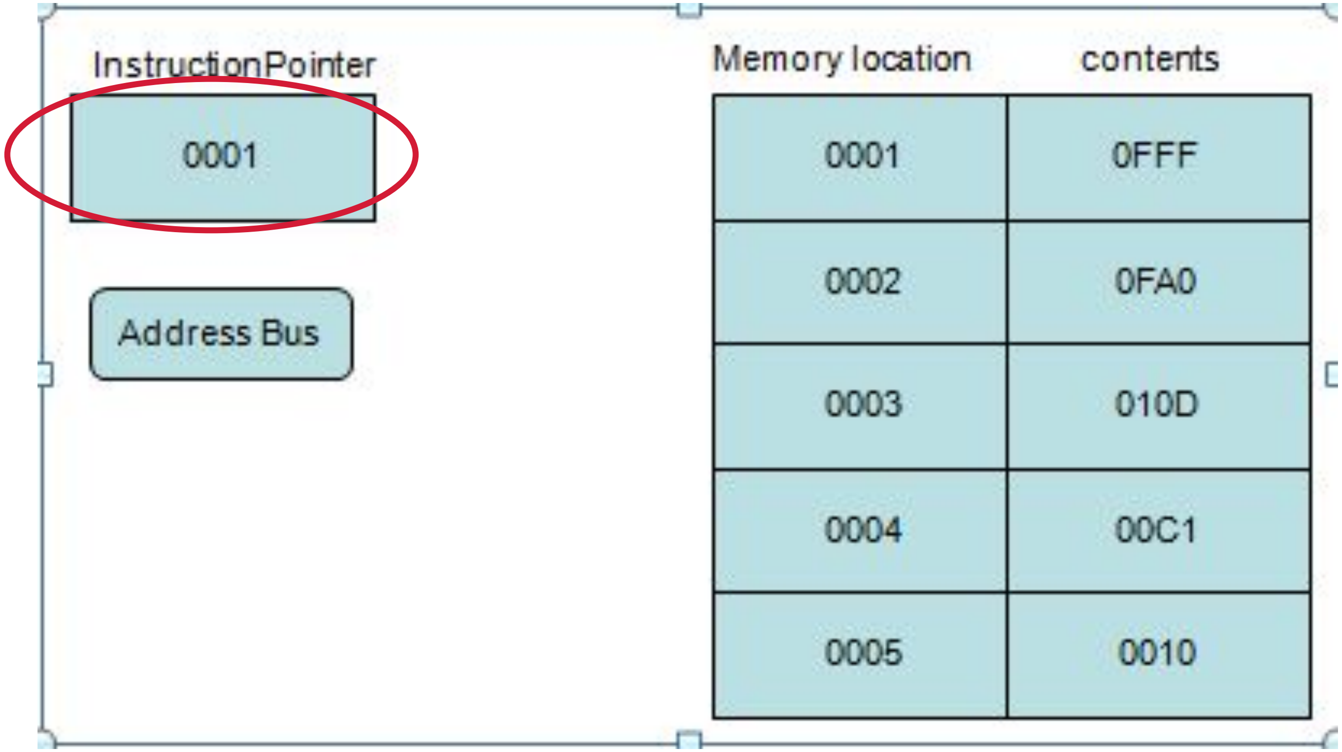
- **Program Counter** : Holds address of next instruction
- **Instruction Register** : Holds the instruction currently being executed or decoded
- **Memory Address Register :** Holds memory address from where data will be fetched
- **Memory Data Register** : Holds the data being transferred to the memory or from the memory by the CPU



Fetching an Instruction

- **Step 1**

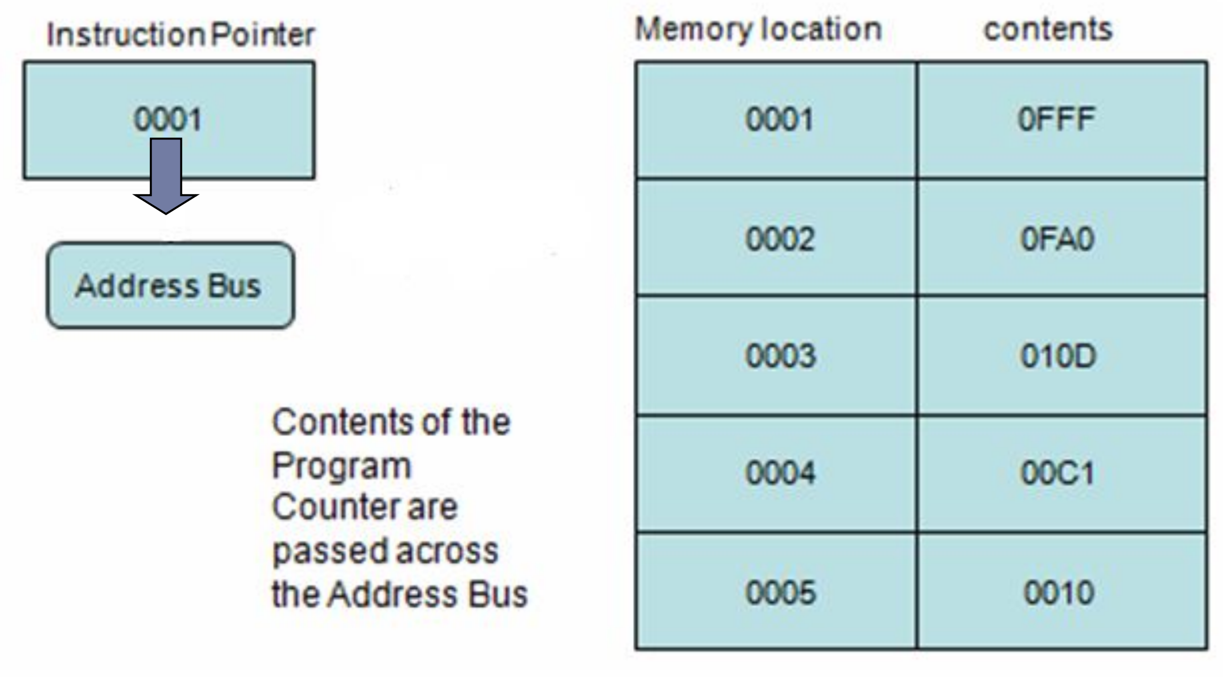
Instruction Pointer (IP) or a program counter is register, that holds the address of the next instruction to be fetched.



Memory location	contents
0001	0FFF
0002	0FA0
0003	010D
0004	00C1
0005	0010

Fetching an Instruction

- **Step 2**

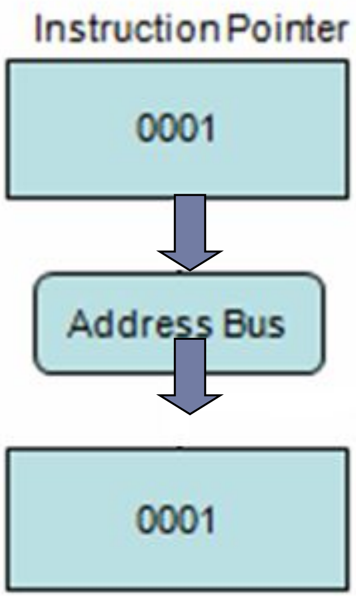


Contents of the Program Counter are passed across the Address Bus

Memory location	contents
0001	0FFF
0002	0FA0
0003	010D
0004	00C1
0005	0010

Fetching an Instruction

- **Step 3**



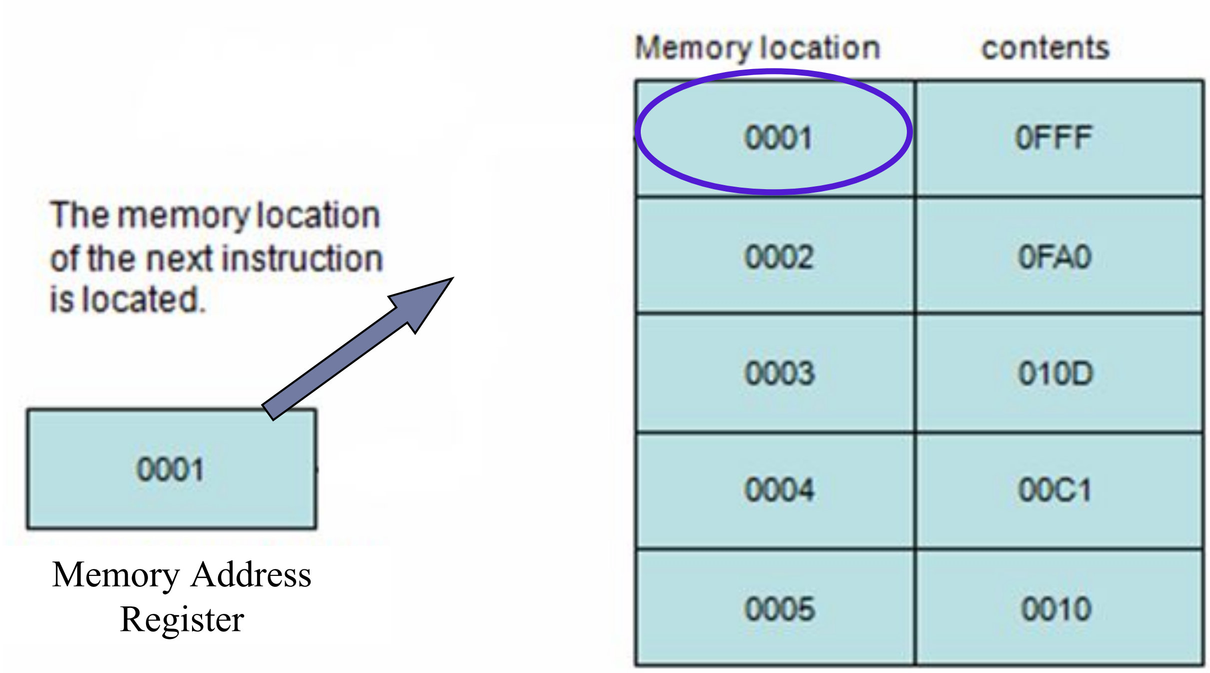
Memory Address Register

The address moves over the address bus to the Memory Address Register

Memory location	contents
0001	0FFF
0002	0FA0
0003	010D
0004	00C1
0005	0010

Fetching an Instruction

- **Step 4**



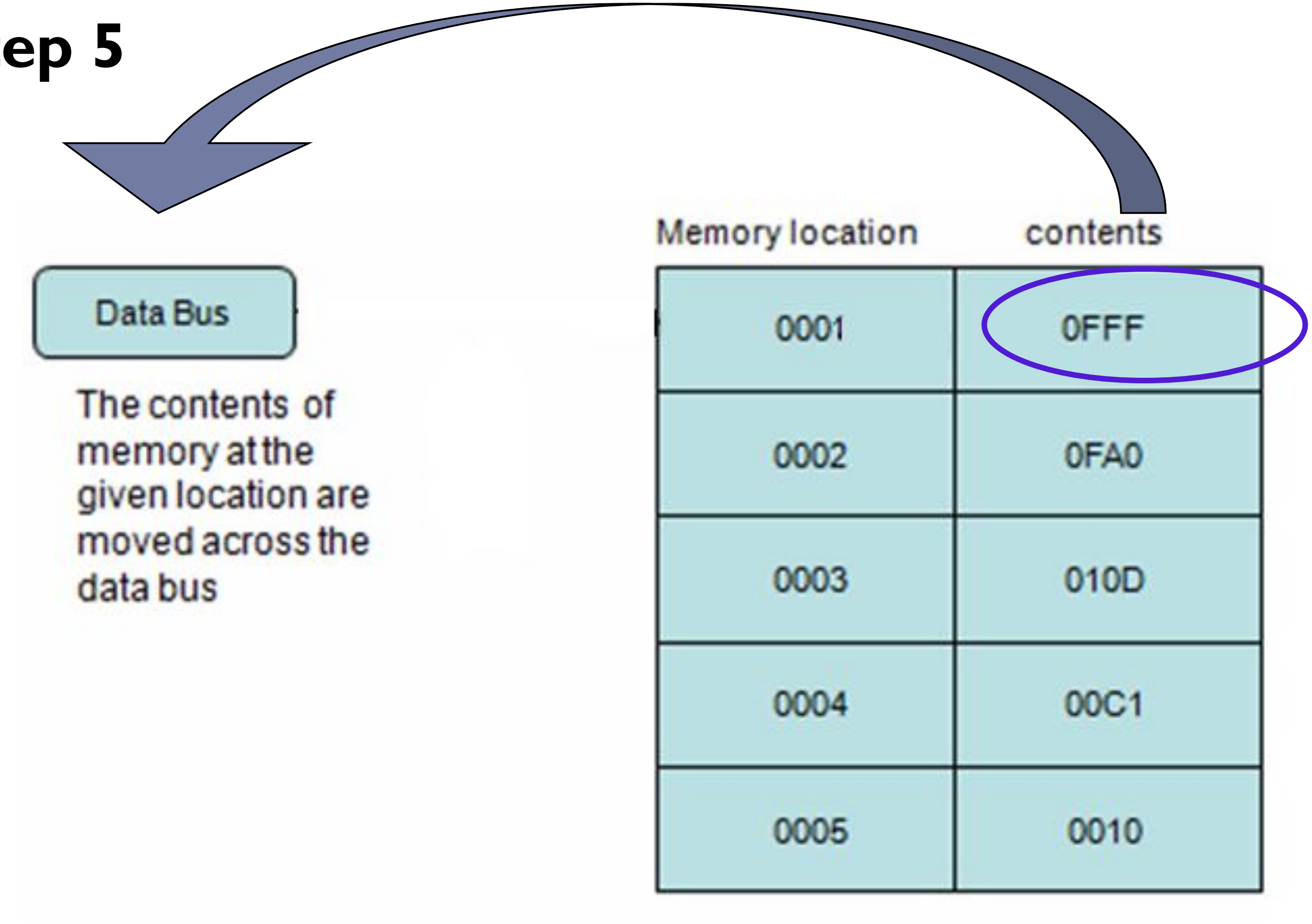
The memory location of the next instruction is located.

Memory Address Register

Memory location	contents
0001	0FFF
0002	0FA0
0003	010D
0004	00C1
0005	0010

Fetching an Instruction

- **Step 5**

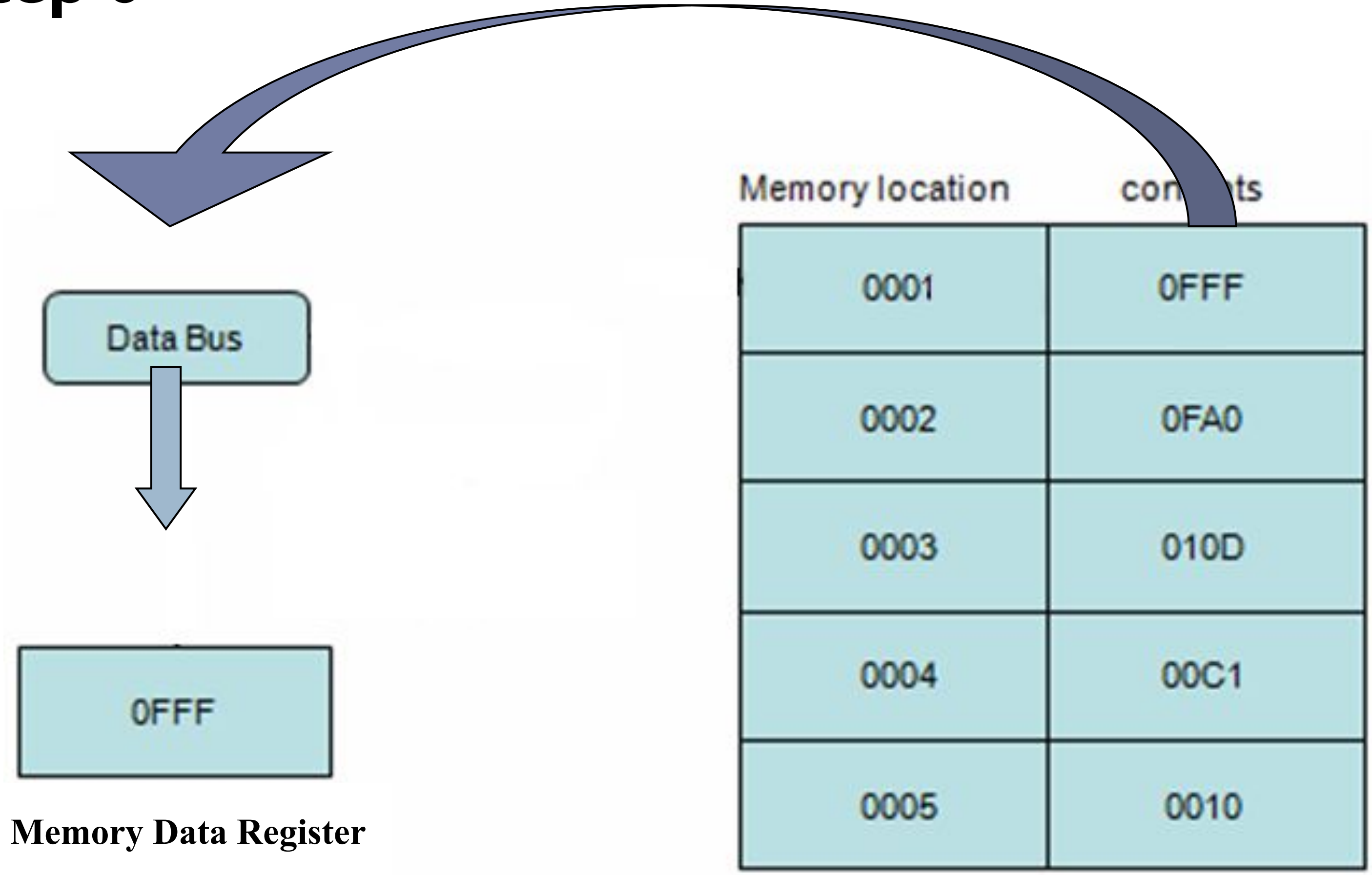


The contents of memory at the given location are moved across the data bus

Memory location	contents
0001	0FFF
0002	0FA0
0003	010D
0004	00C1
0005	0010

Fetching an Instruction

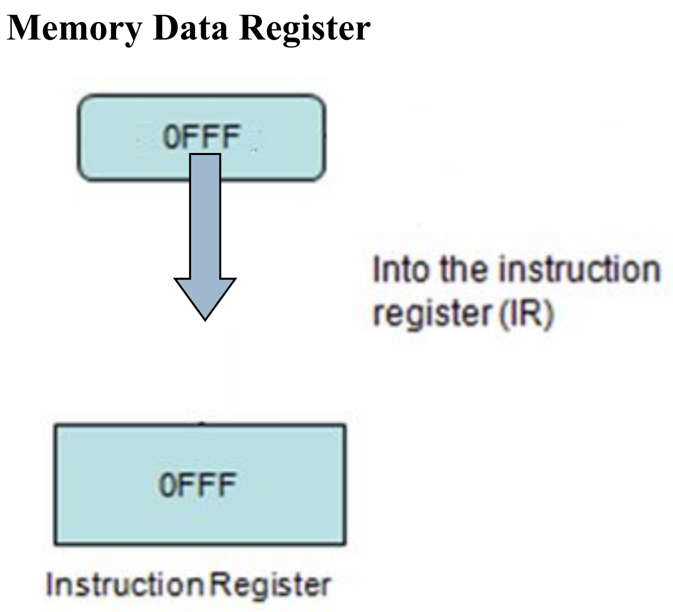
- **Step 6**



Memory location	contents
0001	0FFF
0002	0FA0
0003	010D
0004	00C1
0005	0010

Fetching an Instruction

- **Step 7**



Memory location	contents
0001	0FFF
0002	0FA0
0003	010D
0004	00C1
0005	0010

Food for thought

- What do you understand by a 32 bit Data Bus ?
- BIOS is a special program that orchestrates loading the computer's operating system. Should it be stored in ROM or RAM ?

Thank You!!