1 Fill in the blanks

1. The hardware realization of multiplier in a microcontroller/microprocessor is an organizational issue.
2. Data bus is always bidirectional and address bus is unidirectional.
3. Major difference between the microcontroller and a microprocessor is

Key difference in both of them is presence of external peripheral, where microcontrollers have RAM, ROM, EEPROM embedded in it while we have to use external circuits in case of microprocessors.

4. In the Von Neumann architecture discussed in the class the MBR handles a word of length ..... (20 or 40). IR ..... IBR ..... MAR ..... and PC .....

5. Output of IR is a part of address bus.
6. Issues or performance measures of whether a hardware multiplier implementation or software realization of multiplier algorithm are ....., ..., ..., and .....
7. Which of the following is (are) volatile? (a) SRAM, (b) EEPROM (c) DRAM (d) NV-RAM

SRAM and DRAM are Volatile memories

2 Answer the following

1.Browse in the internet for a commercial processor which has also FPGA.

https://www.networkworld.com/article/3230929/intel-unveils-hybrid-cpu-fpga-plans.html

2. What is the difference between computer organization and architecture?

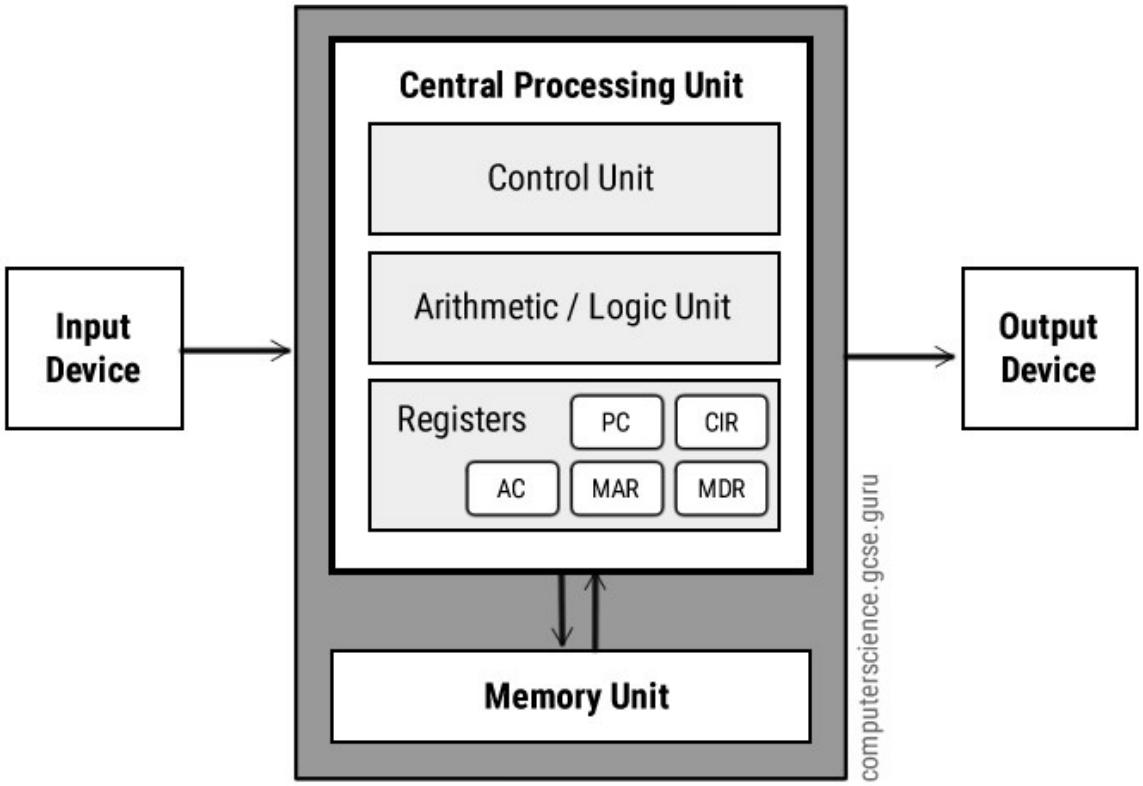
Computer Architecture	Computer Organization
Computer Architecture is concerned with the way	Computer Organization is concerned with the structure and b computer system as seen by the user.

hardware components are connected together to form a computer system.	
It acts as the interface between hardware and software.	It deals with the components of a connection in a system.
Computer Architecture helps us to understand the functionalities of a system.	Computer Organization tells us how exactly all the units in the and interconnected.
A programmer can view architecture in terms of instructions, addressing modes and registers.	Whereas Organization expresses the realization of architectur
While designing a computer system architecture is considered first.	An organization is done on the basis of architecture.
Computer Architecture deals with high-level design issues.	Computer Organization deals with low-level design issues.
Architecture involves Logic (Instruction sets, Addressing modes, Data types, Cache optimization)	Organization involves Physical Components (Circuit design, A Peripherals)

3. Describe the Von Neumann architecture with a block diagram and explain its operation.

Von Neumann architecture was first published by John von Neumann in 1945.His computer architecture design consists of a Control Unit, Arithmetic and Logic Unit (ALU), Memory Unit, Registers and Inputs/Outputs .Von Neumann architecture is based on the stored-program computer concept, where

instruction data and program data are stored in the same memory.



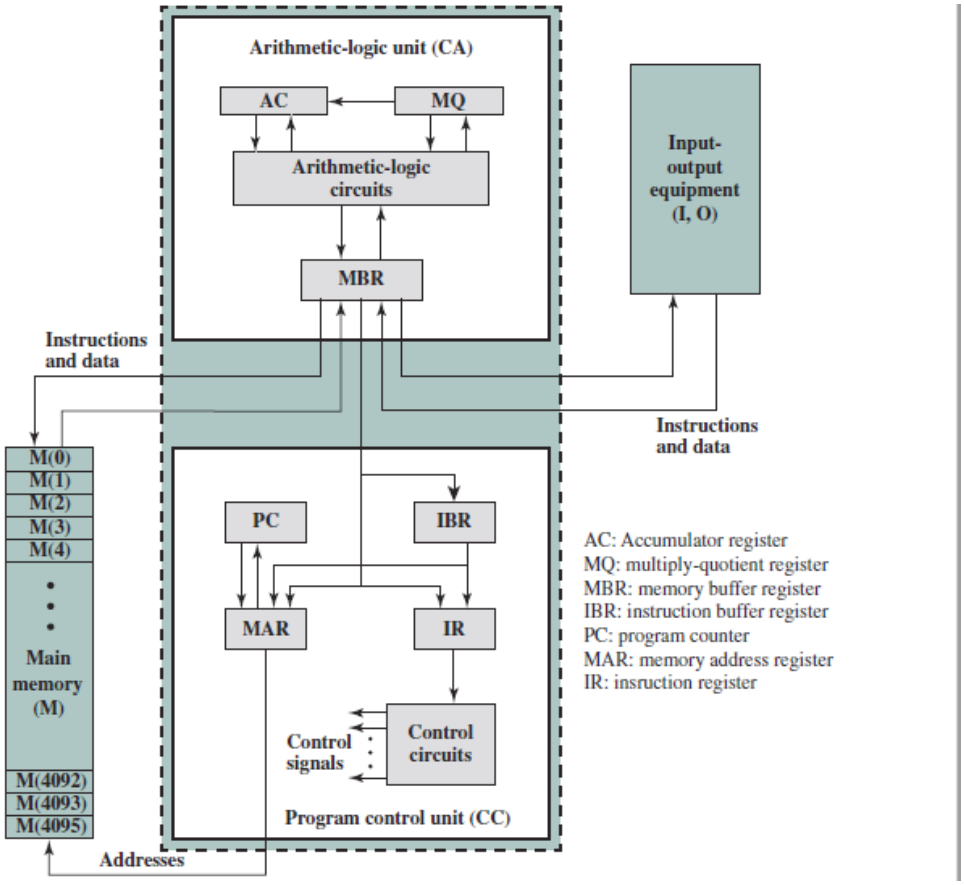
4. What is the major difference between Von Neumann architecture and Harvard architecture?

Von-Neumann architecture
In a Von-Neumann architecture, the same memory and bus are used to store both data and instructions that run the program. Since you cannot access program memory and data memory simultaneously, the Von Neumann architecture is susceptible to bottlenecks and system performance is affected.

Harvard Architecture
The Harvard architecture stores machine instructions and data in separate memory units that are connected by different busses. In this case, there are at least two memory address spaces to work with, so there is a memory register for machine instructions and another memory register for data. Computers

designed with the Harvard architecture are able to run a program and access data independently, and therefore simultaneously. Harvard architecture has a strict separation between data and code. Thus, Harvard architecture is more complicated but separate pipelines remove the bottleneck that Von Neumann creates.

5.



1.Memory buffer register (MBR): Contains a word to be stored in memory or sent
to the I/O unit, or is used to receive a word from memory or from the I/O unit.
2.Memory address register (MAR): Specifies the address in memory of the word
to be written from or read into the MBR.
3. Instruction register (IR): Contains the 8-bit opcode instruction being executed

The IAS operates by repetitively performing an instruction cycle, as shown in
Each instruction cycle consists of two subcycles. During the fetch cycle,
the opcode of the next instruction is loaded into the IR and the address portion is loaded into the MAR. This instruction may be taken from the IBR, or it can be
obtained from memory by loading a word into the MBR, and then down to the IBR,IR, and MAR.

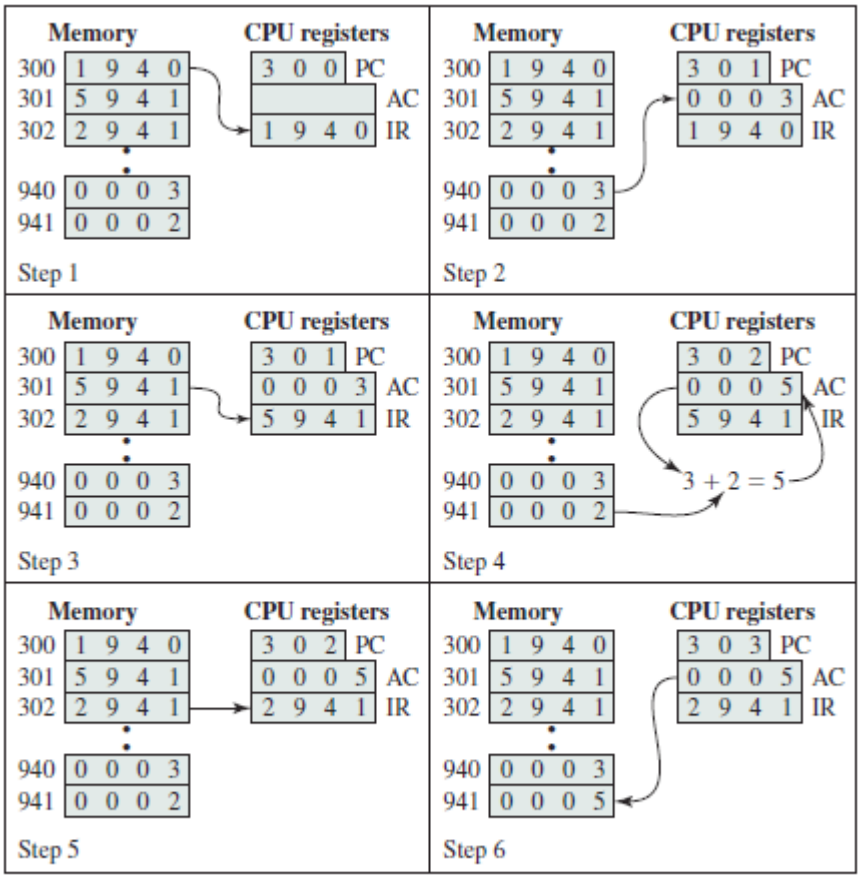


Figure 3.5 Example of Program Execution (contents of memory and registers in hexadecimal)

IR=MBR
Step1 PC=MAR
Step 2-6 MAR=last 3 nibbles of IR

A partial program execution, showing the relevant portions of memory and processor registers.1 The program fragment shown adds the
contents of the memory word at address 940 to the contents of the memory word at address 941 and stores the result in the latter location.

Three instructions, which can be described as three fetch and three execute cycles, are required:

1. The PC contains 300, the address of the first instruction. This instruction (the value 1940 in hexadecimal) is loaded into the instruction register IR, and the PC is incremented. Note that this process involves the use of a memory address register and a memory buffer register. For simplicity, these intermediate registers are ignored.
2. The first 4 bits (first hexadecimal digit) in the IR indicate that the AC is to be loaded. The remaining 12 bits (three hexadecimal digits) specify the address (940) from which data are to be loaded.
3. The next instruction (5941) is fetched from location 301, and the PC is incremented.
4. The old contents of the AC and the contents of location 941 are added, and the result is stored in the AC.
5. The next instruction (2941) is fetched from location 302, and the PC is incremented.
6. The contents of the AC are stored in location 941.

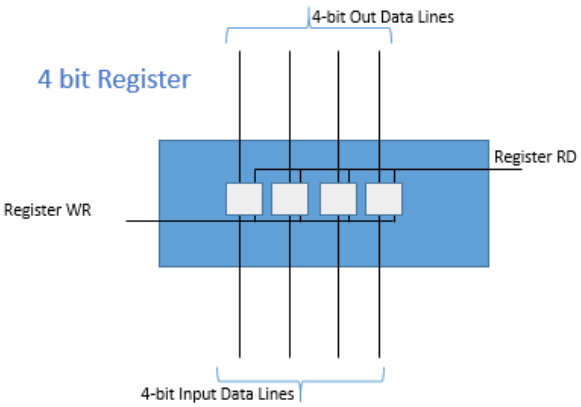
In this example, three instruction cycles, each consisting of a fetch cycle and an execute cycle, are needed to add the contents of location 940 to the contents of 941.

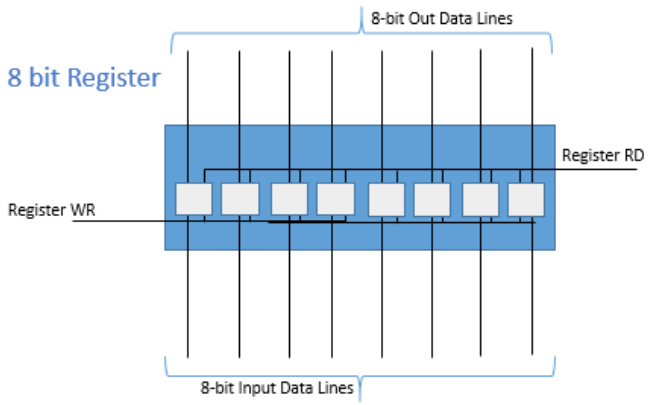
6.PC=PC++

7.

Flip Flops to Registers

- The group of flip flops collectively form a multi bit data (also called data-word) holders: 8/16/32 bit memory are called Registers
- To perform read and write operations from all the flip flops contained in the register, we have a common read enable and a write enable





Memory and Data Bus

- Set of registers together form a Memory
- Each of the register in the memory is identified by an index i
  - Registers are named as R0, R1, R2, R3 ..... R(n-1)
    - For every register: a WR line and a RD line
  - To enable data transfer each register connected to Data Bus and control the RD and WR lines of these registers to execute the desired transfer
  - Width of the data bus: maximum number of bits a Register can hold or needs to transfer