# 1. Main Memory

**a)** **Memory** is a component of the computer that saves instructions and data.

i. Types of memory: **ROM**, **RAM**, **Associate Memory** and **Cache Memory** etc. (see point f)

ii. Basic Unit: **Bit** which uses 0 and 1. (Combination of 8 bits is a byte)

iii. Memory has a many **cells** (**words**) and **locations,** each can save one data.

**b)** **Memory Address:** The address number of each cell (word) of memory, which can be used to locate each data or instruction in the memory.

i. If memory has n cells (words), it has an address from 0 until n-1.

ii. If m bit data is used to store address, then the number of cells (words) are $2^m$

**c)** **Byte Arrangement:** The arrangement of bytes in a cell (word). There are two arrangements:

i. Big Endian: Numbering starts from left

ii. Little Endian: Numbering starts from the right

iii. Problems may arise when sending records between machines of different byte arrangements, but software can be used to change records to the machines own numbering schemes.

| | | | |
|---|---|---|---|
| 0 | 1 | 2 | 3 |
| 4 | 5 | 6 | 7 |
| 8 | 9 | 10 | 11 |
| 12 | 13 | 14 | 15 |

Big Endian

| | | | |
|---|---|---|---|
| 3 | 2 | 1 | 0 |
| 7 | 6 | 5 | 4 |
| 11 | 10 | 9 | 8 |
| 15 | 14 | 13 | 12 |

Little Endian

**d)** **Error Correction Code:** Computer memory faces failures and error that may change the saved data. To control these errors, most memory overcome this by:

i. **Error Detection Code:** Through this method, a few additional bits are added into a memory cell (word). When data is read from memory, these bits are checked for errors. If it detects an error:

ii. **Hammin Alkhwarizmi**: is used to correct these errors.

iii. **Hamming Distance:** No. of different bits between two binary strings (using XOR)

iv. The codeword which has the minimum Hamming distance with the valid codeword is taken to be correct.

**e)** **Types of memory**

i. **RAM (Random Access Memory):** Type of memory that we can read and write to. Types of RAM include SRAM, DRAM, EDO-DRAM, CDRAM, SDRAM, RDRAM etc

1. **Static RAM:** Consists of small **transistors** which can retain its contents as long as power remains.
2. **Dynamic RAM:** Consists of small **capacitors** which can be charged and discharged. As capacitors tend to leak charge, the state is maintained by refreshing each bit every few million nanoseconds to avoid data from disappearing. As DRAM is made with tiny capacitors, it can be built with a big capacity.

ii. **ROM (Read Only memory):** Type of memory that can only be read. Data is stored at the manufacturing stage. In most computers ROM is used to store Bootstrap loader (which functions to start the OS when computer is turned on)

1. **PROM:** Programmable ROM, similar to ROM chip but allows data to be written (field programming.)
2. **EPROM:** Erasable Programmable ROM is a ROM that can be reprogrammed and erased.
3. **EEPROM:** Electrically Erasable Programmable ROM, data can be erased or reprogrammed by applying higher than normal electrical voltage.

iii. **Associative Memory:** Type of memory chip where memory is accessed by searching through the content of the data rather than its address. This is very fast. More expensive than RAM because it has the capability to save data and has logic circuits to erase the contents.

iv. **Cache Memory:** A memory system where the memory is located in a smaller and high speed storage, which makes the process of accessing memory significantly faster. It operates in the following manner:
    1. Performing **Search** for the entry word
    2. If entry is found, it is **taken**
    3. If not, the block of memory that has the particular entry cell (word) will be taken and **saved** into cache memory.

**f)** **Memory Chip:** The memory chip semiconductor is in the form of integrated circuits.

**g)** **Memory Organisation:** Memory can be organised in two ways; memory with capacity for cells (words) or capacity for cell size.

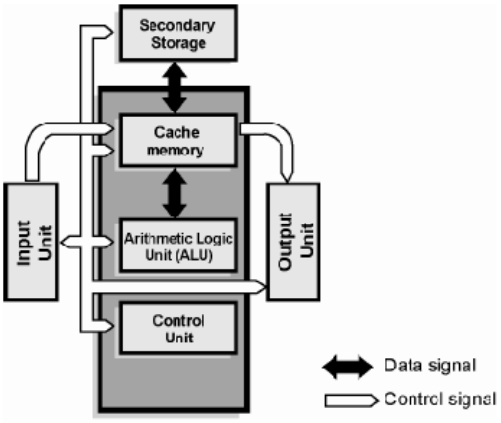

Figure 2.1: Location of CPU in a computer

# 2. Central Processing Unit

**a)** **CPU:** An electric component, consisting of digital circuits and each circuit has a different processing function. Its main duty of processing information is done by executing a program with a set of instructions in the main memory, that program operates by fetching instructions, checking, and then executing instructions. CPU is composed of 3 components:

i. **Control unit:** Controls all CPU operations, by giving proper control instructions.
ii. **Arithmetic and Logic Unit (ALU):** Deals with all arithmetic (+-/x) and logical (AND OR NOT) operations.
iii. **Register:** High speed memory units to store specific controlled information and data temporarily. This memory has several registers, each for specific functions. Main registers are:
    1. **Programme Counter (PC):** Refers to the next instruction to be carried out
    2. **Instruction Register (IR):** Hold the instructions currently being executed.

**b)** **Instruction Execution:** Are referred to as the machine language for that CPU. Fetch-Decode-Execute cycle:

i. **Fetch** the instruction from memory and transfer to IR
ii. **Decode** the PC value
iii. **Identify** the type of instruction
iv. Is the data **available**?
v. If **yes** to (iv), identify the location of data, and transfer to the data registry
vi. If **no** to (iv), Execute instruction
vii. **Store** the result in suitable place
viii. **REPEAT**

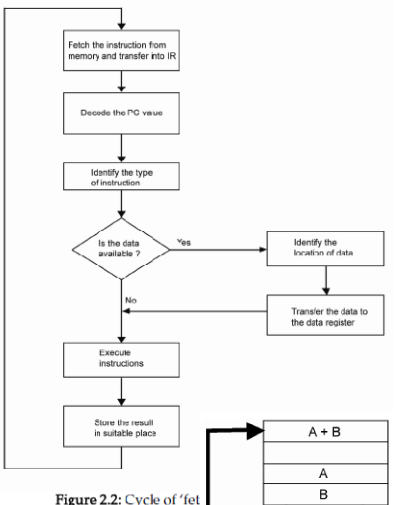

Figure 2.2: Cycle of 'fet

**c)** **CPU Organisation (Datapaths):** Has a few registers (1-16) and an ALU. Data is entered into two ALU input registers labelled A and B. ALU performs mathematical operations and result is stored in the output register.

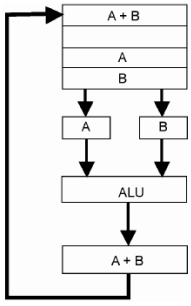

Figure 2.3: Datapaths for CPU

**d)** **Interrupts:** To ensure accurate and fast output, few instructions such as interrupts were created. It is an external signal sent to and executed by CPU, which require immediate response. CPU will halt all current executions and save all required data. Then CPU will serve the interrupts.

There are 3 types

i. **External Interrupts (EI):** Mostly by IO devices as they are slower than CPU processing.
ii. **Internal Interrupts (II):** Occur due to invalid data or instructions (Eg attempt to divide by zero)
iii. **Software Interrupts:** Produced by software used by OS

**e) Microprocessor:** Early processors were huge, with much effort by engineers to produce smaller CPU's, they invented CPU in the size of a chip, which is called the Microprocessor.

i. **Intel 4004:** First microprocessor, 640b memory
ii. **Intel 8008:** An advancement on 4004, uses the 8-bit tech and addresses up to 16kb of memory
iii. **Intel 8080:** Four times more address memory and 10 times faster than 8008
iv. **Intel 8085:** Not much different from 8080, Able to operate software at higher speed (Most successful)
v. **Intel 8086/8088:** Used in earlier IBM units. 40 pins. 16 bit datapath. But 8088 could read/write with inly 8-bit memory while 8086 used 16-bits. Both chips address upto 1 Mb of memory.

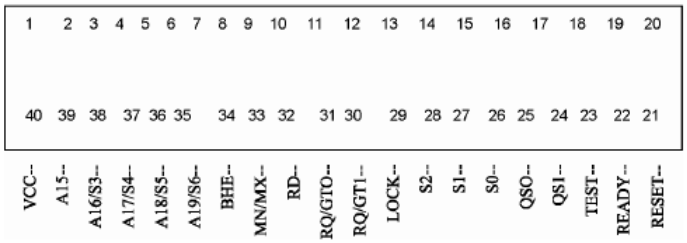

Figure 2.6: 8088 microprocessor with pins in maximum mode

-**A0-A19** – Contains Memory addresses of IO
-**35th – 38th** – pins are Multiplexed in order to have addresses in the start of bus cycle and info stat at end
-**33rd** Pin – **MN/MX** – Chip is in min or max mode
-**INTR** (IO devices) and **NMI** (Internal)is to give interrupts to CPU
-**RESET** - to restart

vi. **Intel 8087:** AKA Arithmetic Co-Processor Chip; Speed up arithmetic and mathematical operations
vii. **Intel 80286:** Substitute for 8088, has 3 advantages to ensure its 5 times more efficient. It has Kernal and User modes (Allows multitasking applications), Data reaches upto 16-bits, Performance is faster than 8088. It has 4 Units:
    1. **Bus Unit:** Conducts all operations of the CPU.
    2. **Instruction Unit:** Encodes the bytes transmitted by the Bus Unit for execution.
    3. **Execution Unit:** Executes instructions given by Instruction unit
    4. **Address Unit:** Deals with physical and virtual address (technique to extend memory to the allocated address) functions
viii. **Intel 80386:** First 32-bit CPU. Has 132 Pins. Came in SX (32-bit address, 16-bit data) and DX (32-bit) versions. Besides high performances, advantages over 80286 are:
    1. All registers and instructions of 8/16/32 bits can be executed at same rate
    2. Allows virtual memory of 246bits
    3. Handles physical memory upto 4Gb (16MB in 80286)
ix. **Intel 80486:** Similar to 80386, 168 pins. Has 1.2 million transistors. Speeds upto 100 Mhz. Comes in SX (with arithmetic co-processor) and DX (not with arithmetic co-processor) versions.
x. **Intel Pentium:** 237 pins arranges similar to 8088 chip (pins only on two sides).

# 3. Input / Output Device

Used to enter and retrieve data from the computer

**a) Terminal:** A computer terminal consists of a keyboard, monitor and other electronic tools that control both devices. There are 3 types of terminals:

i. **Character-Map Terminal:** Use Video RAM to display characters arranged as 25x80 on the monitor. To display, CPU will copy particular character into the Video RAM. Each character, attached with an attribute byte that explains how the character should be displayed. Video Board then produces the character from Video RAM repeatedly.

ii. **Bit-Map Terminal:** Same as Character-Map terminal, but its Video RAM consists of larger bit arrangement of image elements called pixels. Screen can consist of at least 300x320 pixels (and more). Disadvantages are:
    1. It needs a large Video RAM
    2. In terms of efficiency, backups from the Video RAM to the screen are required to display an image on the screen.

iii. **RS-232C Terminal:** A standard terminal computer interface, introduced to enable use of terminals for all types of computers. It has 25 connector pins. Standard of the RS-232C is in the Mechanical size, Connector form, Voltage level and the meaning of each signal in a pin.

**b) Keyboard:** Keys in the keyboard acts as switches to allow electricity connection to occur

**c) Monitor (CRT):** A box which generally consists of a CRT and its power supplier. CRT has a gun which shoots electron beams towards the tube screen. A colour monitor has three types of guns, red, blue and green.

**d) Mouse:** A Mouse is a small plastic component that is usually placed on the table, next to the terminal. When the mouse is moved, the cursor on the screen also moves, enabling the user to point at a particular item. A Mouse has buttons with different functions, enabling the user to select an item from the menu. Mouses generally send a serial of 3 bytes to computer in every 100 ms; first byte indicates x value changes, second y value changes, third byte informs if button is pressed or not. 3 types of mouses are:

i. **Mechanical Mouse:** Has 2 rubber balls arranged vertically underneath. When mouse moves parallel to the main axis, one ball will roll, when it goes vertical, the other ball will roll.
ii. **Optical Mouse:** Got no balls, it has a LED and a light detector.
iii. **Optomechanical Mouse:** Similar to the mechanical mouse, 2 balls each rotating 90 degrees. But each ball has a LED in the middle. When mouse is moved, the balls rotate and light shines through the openings. Amount of light signals detected indicates the amount of movements.

**e) Printer:** A device used to produce a hard copy of documents. Some types:

i. **Impact Printer:** Operates like a type writer. A piece of metal (or plastic) is attached to the printer; the piece of metal strikes a ribbon of ink and transfers the ink onto paper.
ii. **Dot-Matrix Printer:** Printer has 7 to 24 pins that are activated electromagnetically, which will scan each line printed. The result is seven horizontal lines with each line having 5 × 80 = 400 dots. Whether each dot can be printed or not depends on the letters that will be printed. The printing quality can be improved using two techniques i.e by adding the number of needles and by using overlapping printing dots.
iii. **Laser Printer:** Operates like a Photocopy machine. A high voltage is passed to the Drum and covered with light sensitive material. Later laser light will scan the whole drum. This light is modified to produce clear and dark dots. The first line of dots that have been scanned is transferred to the toner. The toner will be attract to the dots that have electrostatic charge and produce an image for that line. Then the drum that is coated with the toner is rolled or pressed over the paper and transfers the black powder to the paper. The paper is passed through the hot roller and the toner is transferred onto the paper. During rotation, the drum is discharged and cleaned from any excess toner. It is then prepared to be charged and coated again for next page.

# 4. Secondary Storage and Communication Devices

The Secondary storage is an important tool to provide backup in event of data loss and data damage occurs. Communication devices enable data exchange and data to be transmitted beyond geographical boundaries.

Information can be shared to help people in many fields.

a) **Magnetic Tape:** Similar to a home video tape, a thin plastic tape that has been covered with magnetisable materials. Information is read/stored in a sequence from the square matrix bits. Electric current passed through the tape enables the information to be written on the tape in small dots (bits).

b) **Magnetic Disk:** Piece of metal with 5-10 inch in diameter that is coated with a magnetic layer at both sides.

c) **Floppy Disk:** The floppy disc or diskette is a rounded plastic piece (Mylar) that is flat and can spin inside a jacket. Data is stored in the form of electromagnetic charge (presence or absence) over the metal oxide layer that covers the Mylar plastic.

d) **Optical Disk:** Has a higher storage capacity. A high powered laser beam is used to form microscopic pits on the disc. A low power laser beam is directed on the surface. Reflection from the beam is detected when there is no pit (1), if pit is there, the reflection beam disperses and cannot be detected (0).

e) **Modulation:** The process of varying one waveform in relation to another waveform.
f) **Demodulation** is the act of extracting the original information-bearing signal from a modulated wave.

g) **Modem:** The device that receives characters from a computer in two levels of signal form, one bit at one time and sends bits in a serial way in a modulation amplitude form, frequency or phase is known as modem.

h) **Asynchronous Transmission:** The receiver and transmitter do not have to be ready at the same time. To enable the receiver to know the beginning character; the bit is sent before each character. To increase the reliability, 1 or 2 bit is stopped from being sent after each character.
i) **Synchronous Transmission:** A start bit and stop bit are required because both the transmitter and receiver are in a ready state simultaneously. Synchronous communication usually happens at a very higher speed bit. In this method, once the modem is moderated, it can send characters continuously. To maintain the moderator, signals are still transmitted although there is no actual data that required to be sent.

j) **Simplex Channel:** Transmission method which is only capable of sending data in one direction only as one end has the transmitter and the other end has the receiver. Eg Radio and television.
k) **Half-Duplex Channel:** This channel is able to transmit and receive data in two directions but not simultaneously. During a transmission, a modem acts as sender and another modem as receiver. Eg walkie-talkie.
l) **Full-Duplex Channel:** This channel can transmit and receive data in a two directions simultaneously. It does not waste time during exchange of direction. Eg telephone

# 5. Earlier Computer System and the Von Neumann Machine

a) **Babbage Analytical Machine:** Charles Babbage invented the Difference Engine which was only able to add and subtract. Babbage furthered his work to invent an analytical engine. This engine, which was the first computer, has four components:
   i. **Store (memory):** consists of 1000 words from 50 decimal digits that are used to hold variables and results.
   ii. **The Mill:** is where the numbers are operated on. The Mill receives numbers from the store, and executes addition, subtraction, multiplication, division and returns the results to the store.
   iii. **Input section:** is the card reader (punch card)

iv. **Output section:** produces the output of card reader.

**b)** **ENIAC:** This computer consists of 18000 vacuum tubes and 1500 relays. It weighs around 30 tonnes and uses up to 140 kilowatt. It has 20 registers with each register handling 10 decimal digits. It was programmed with 6000 multi-positioned switches and linked to a variety of sockets and cables.

**c)** **Von Neumann Machine:** The Von Neumann machine was used in EDSAC, the first computer that used the stored program concept. The Von Neumann machine has five sections; Memory, Arithmetic and Logic Unit (ALU), Control Unit, Input and Output device. The Memory has 4096 words and each word has 40 bits (0 or 1). Each word holds 20 instruction bits or 39 marked integer bits. Out of the 20 bits used for instructions, 8 bits are used to indicate types of instruction and the other 12 bits to indicate one of the 4096 words in the memory. ALU consists of registers with 40 bits known as accumulator to store the result of arithmetic operations.

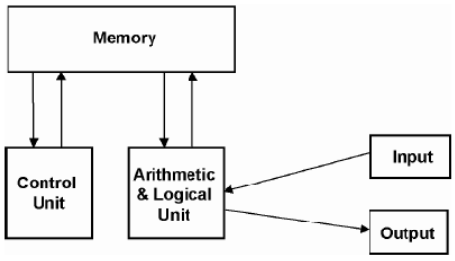

Figure 5.3: The Initial Von Neumann machine

# 6. Bus System

**a)** **Bus System:** A system of connecting computer components using the road concept, where roads doesn't connect houses directly to each other but provide a path to where each house is connected to that path.

**b)** **Computer Bus:** Consisting of 50-100 parallel wires coated with copper, which connect the components of a computer system in a board. Few types of computer bus are:

i. **Internal Bus:** Connects components in a chip.
ii. **System Bus:** Connects all the chips that form a computer system.
iii. **Local Bus:** Specifically connects two components directly (Eg processor and co-processor)

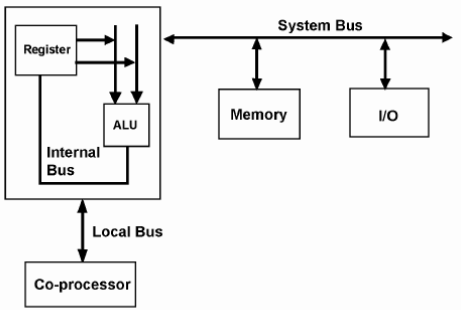

Figure 6.2: Varieties of bus in a computer system

**c)** **Bus Operation:** Master devices initiates data transfer while slave devices wait for instructions.

i. **Bus Driver:** Chips connected to the master buses which makes sending signal strong
ii. **Receiver Bus:** Chips connected to the slave buses which makes receiving signals strong
iii. **Transceiver:** Chips that makes sending and receiving signals stronger.

**d)** **Main Buses:**

i. **Data Bus:** Transfer data from one place to another.
ii. **Address Bus:** States whether the address in a memory address or a I/O device address.
iii. **Controller Bus:** Carries control signals such as read or write signals.

**e)** **Categories of Buses:**

i. **Synchronous Bus:** (Most computers use this) Driven by a crystal swinger which acts as clock and produce signals consisting of rectangle waves with repeats between 5MHz and 50MHz. All activities involve one or few bus cycles.
ii. **Asynchronous Bus:** Not clock controlled.

**f)** **Bus Arbitration:** To determine which device is the master bus at any one time. CPU has the lowest priority in getting bus, only if no other devices are using the bus as CPU can wait whereas IO devices cant or loose data.

i. **VAX SBI:** Uses 16 connectors for demands by priority bus, one of each device. When a device needs to use bus, it makes a demand through its demand channel. If it is of a higher priority than to other devices, it will be given the control bus, else it has to wait for next cycle.

**g) Microcomputer:**

i. **DMA (Direct memory Access):** Completely controlled by the CPU(Unlike IOP), ensures IO operations run without interrupting CPU executions.

# 7. PC Organisation

**a) Changes in the Bus System:**

i. **EISA (Extended Industry Standard Architecture):** bus system used in PC
ii. **VESA (Video Electronics Standard Association)**: bus to connect disk and monitor directly to the processor.
iii. **SCSI (Skuzzy):** Accepted standard by ANSI, connects IO devices such as hard disk unit and printer to PC.
iv. **PCI (Peripheral Control Interphase):** Based on Pentium, enables fast data transfer, supports one or multi processors.

# 8. RISC Machine

**a) RISC Architecture:** Based on the few principles below:

i. **Single Instructions for each data cycle:** So that a single cycle doesn't take a long time to complete.
ii. **Load/Save Architecture:** Enabling writing of data from register into memory, to enable instructions to access memory (as memory involves more than one cycle, the first point is against this, so RISC machine instructions cannot access memory.
iii. **No Microcodes:** By voiding need for translation at this stage (microcodes translate), it is faster by executing directly from hardware.
iv. **Fixed format instructions:** As its instructions are implemented directly in hardware, each bit in an instruction is used as an input to control CPU components, so number of bits in one instructions is fixed (Unfiex numbers will cause difficulties in CPU architecture.
v. **Small number of instructions:** No. Of bits in RISC architecture is smaller and lesser.

**b) Put complexity in compiler:** Compiler technology is one of the factors that enabled development of RISC machines, as it takes the task to translate complex high-level languages

**5 important Steps in Designing RISC Machines**

i. As RISC involves reduces instructions, any application that uses this machine need to be analysed to determine the main operation that needs to be provided.
ii. Design an optimal datapath, for the main operation
iii. Design instructions that form the main operation by using that particular datapath
iv. Increase new instructions only if it does not cause any delay in the machine
v. Repeat this process for the other resources such as cache memory, main memory and co-processor arithmetic.

**c) Pipelining:** A technique that enables the processor to process more than one instruction at a time. A machine is partitioned to several functional units. Phases of instruction execution in RISC machines are **Fetching**, **Decoding**, **Executing**, and **writing** to memory.

| | CYCLE | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| Fetch | A | B | C | | | |
| Decode | | A | B | C | | |
| Execute | | | A | B | C | |
| Write | | | | A | B | C |

Figure 8.2: 4 phases pipelining

**d) Register Usage:** To enable execution of instructions in a single datapath, RISC compilers use many registers, to reduce transfer of data from the memory.

**e) Registers in RISC1:** Divided into 4 groups, each with 8 registers

i. Group holding **Global registers and indicators:** Not specified to any instruction, used by most if the instructions throughout program
ii. Group holding **Entry Parameters** for an instruction
iii. Group holding **local variables,** if it exceeds 8, it will be stored in a stack in main memory.
iv. Group holding **Exit Parameters**

**f) Advantages of RISC:**

i. **Speed:** As instructions in the RISC are processed by pipelining, it is 2 to 4 times faster than CISC.
ii. **Simple:** As RISC executes short instructions, it used only a small portion of the chip, enabling other parts such as Memory management functions and floating-point units to be placed on the same chip.
iii. **Short Cycle:** Since RISC processors are simpler compared to CISC, it can be designed faster, taking advantage of rapid advancement of technology.

**g) Disadvantages of RISC:**

i. **Quality of Instruction code:** RISC processors capability depends on the code to be executed, if compiler fails to do its job well, the processor will take more time to execute instructions.
ii. **Increasing size of code:** RISC needs many instructions to complete one task, so the size of code will be a problem (Size of a program compiled by RISC will be bigger than that same program compiled by CISC)
iii. **System Design:** RISC needs high speed memory system to receive instructions, usually has many caches.

**h) RISC vs CISC:** Issue not resolved. Among the issues debated are:

i. **Which is more suitable to execute higher level language program?**
If "suitable" refers to speed then:
1. **What kind of language?**
C language or pascal, not for FORTRAN
2. **What type of program should be used?**
Recursive program, like Hanoi Tower function suitable for RISC machine
3. **What about input/output?**
RISC machine build with a simple input output device
4. **Should the floating-point unit be included?**
Without the support from the specific devices, RISC is unable to perform calculations involving floating-point units. (Most RISC machines cannot perform multiplication or division)
5. **What about the impact of the compiler?**
Speed achieved not because of capability but efficiency of computer

ii. **Which has a better microprocessor chip?**
The RISC chip is much simpler than CISC chip. Easier to design and completed in a short time period. Microprograms enable CISC chips compatible with its predecessor to be produced.

iii. **Which compiler program is easier to write?**
As instructions in RISC are simpler, so writing compiler is more difficult. However, as instruction set is simpler, the compilation of the construct of a high level language into the RISC machine language can only be done by one way. In CISC, this compilation can be done in many ways. Without any attention to detail, the machine-language program produced may not be efficient.