University Institute of Engineering

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Bachelor of Engineering

Subject Name: System Programming

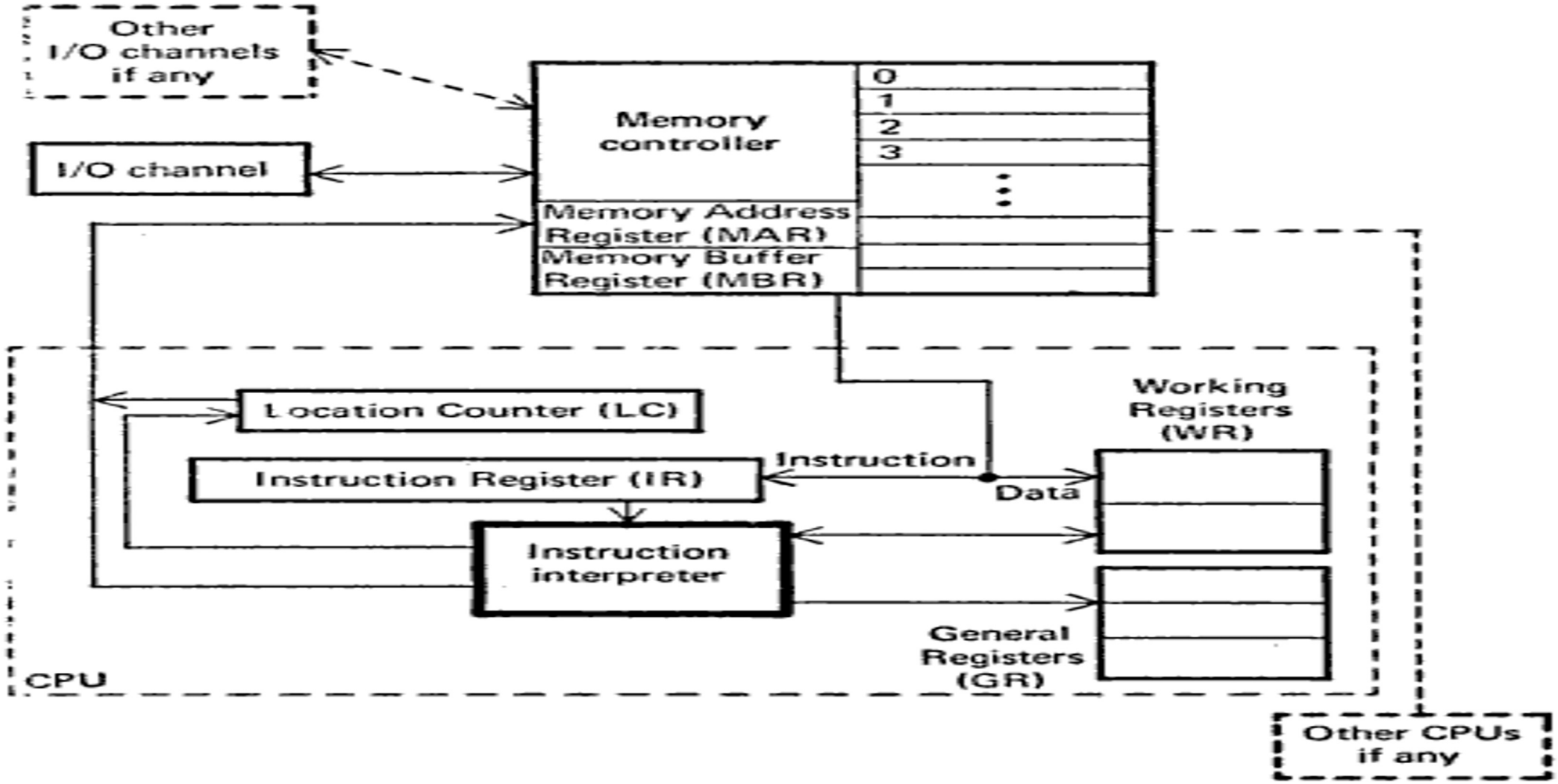
Subject Code: CST-315

Chapter-2.1
Compilers

- Machine Structure
- Interfaces
- Address Space
- Computer Languages

General Machine Structure

- The structure of CPU for a typical Von Neumann Machine is as follows –



General Machine Structure

The components of a general machine are as follows:

- ***Instruction interpreter:*** A group of electronic circuits performs the intent of instruction of fetched from memory.
- ***Location counter:*** LC otherwise called as *program counter PC* or *instruction counter IC,* is a hardware memory device which denotes the location of the current instruction being executed.
- ***Instruction register:*** A copy of the content of the LC is stored in IR.
- ***Working register:*** are the memory devices that serve as “scratch pad” for the instruction interpreter.
- ***General register:*** are used by programmers as storage locations and for special functions.

General Machine Structure

- ***Memory address register (MAR):*** contains the address of the memory location that is to read from or stored into.
- ***Memory buffer register (MBR):*** contain a copy of the content of the memory location whose address is stored in MAR. The primary interface between the memory and the CPU is through *memory buffer register*.
- ***Memory controller:*** is a hardware device whose work is to transfer the content of the MBR to the core memory location whose address is stored in MAR.
- ***I/O channels:*** may be thought of as separate computers which interprets special instructions for inputting and outputting information from the memory.

General Machine Structure

- ***Memory:*** Basic unit, size and addressing scheme.
- ***Registers*** Number of registers, and size, functions, interrelation of each register.
- ***Data:*** Types of data and their storing scheme.
- ***Instruction:*** Classes of instructions, allowable operations and their storing scheme.
- ***Special Features:*** Additional features like interrupt and protections.

General Machine Structure

- **Registers**
- There are a total of **16** ***general purpose registers*** of 32 bits each. In addition there are ***4 floating point register*** of 64 bits each. It also has a 64 **bits** ***program status word (PSW)*** that contains the value of the location counter, protection information and interrupt status.

- The general purpose registers are basically used in arithmetic and logical operations as *base registers* and helps in address formation. The general purpose registers also used as scratch pads for the programmers. Let us take an instruction A 1,901(2,15).
- A(opcode)1(operand in register 1),901(offset) (2(index register),15(base register))

Interface

- **Interface may be defined in two ways-**

1. As per the software ,Interface is a program that allows a user to interact computers in person or over a network. An interface may also refer to controls used in a program that allow the user to interact with the program.

For Eg.

- GUI (Graphical User Interface). This type of interface is what you are using now to navigate your computer and how you got to this page.

Interface

2. As per the hardware interface is a physical device, port, or connection that interacts with the computer or other hardware device

For eg.

IDE and SATA are disk drive interfaces for computer Hard Drives and ATAPI is an early interface for CD-ROM drives.

- **Examples of drive interfaces**
-

- The following list is a list of different internal and external interfaces that connect a drive to a computer.
- ATA ATAPI, eSATA, FireWire , IDE etc

Address Space

- An address space is a range of valid addresses in memory that are available for a program or process. That is, it is the memory that a program or process can access. The memory can be either physical or virtual and is used for executing instructions and storing data.
- On a computer, each process and device is allocated an address space, which holds a certain portion of the processor's address space. The processor's address space is typically restricted to the width of its registers and address bus.

Address Space

Address space is often classified as :

- **flat** : In this the addresses are represented as incrementally increasing integers that start at zero
- **Segmented** : In this the addresses are portrayed as independent segments augmented by offsets.
- The size of an address space can be made larger than that of physical memory by using a memory management technique called **virtual memory.**
- So an address space consists of both physical memory and virtual memory.

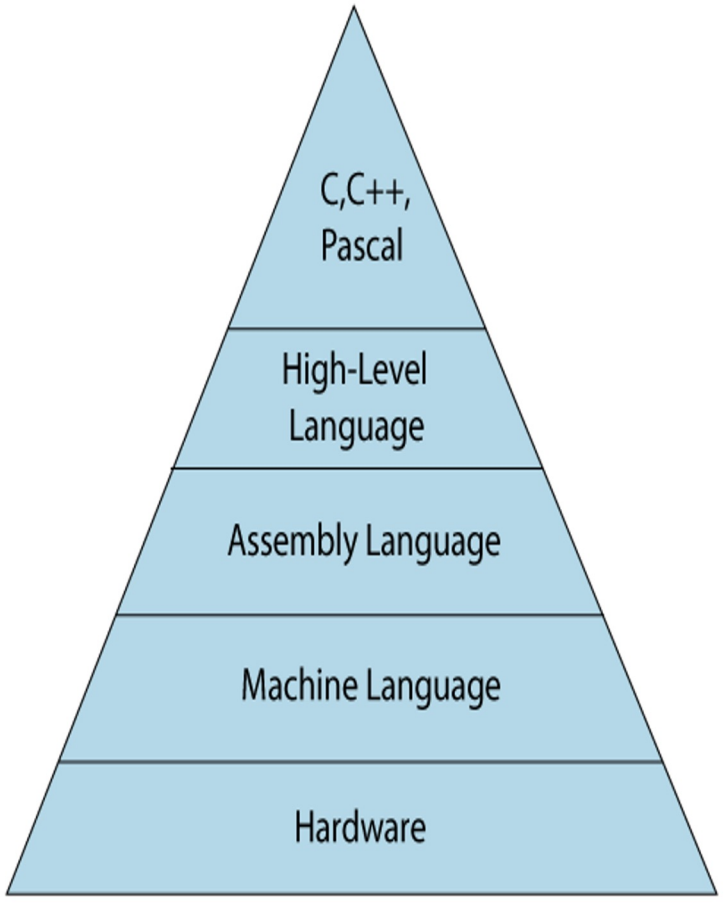
Computer Languages

- The computer language is defined as code or syntax which is used to write programs or any specific applications. It is used to communicate with computers.

- Computer languages can be broadly classified into

  3 major categories :

  1. assembly language
  2. machine language
  3. High-level language.



Types of Computer Languages

- **1. Machine Language**
- The machine language is sometimes referred to as machine code or object code which is a set of binary digits 0 and 1. These binary digits are understood and read by a computer system and interpreted easily. It is considered a native language as it can be directly understood by a central processing unit (CPU).
- Example of machine language for the text “Hello World”:-
- 01001000 0110101 01101100 01101100 01101111 00100000 01010111 01101111 01110010 01101100 01100100

Assembly Language

- The assembly language is considered a low-level language for microprocessors and many other programmable devices. The assembly language is also considered a second-generation language. The assembly language is mostly famous for writing an operating system and also in writing different desktop applications.
- This language has certain drawbacks as

  - It does not contain any variables or functions in programs and also the program is not portable on different processors.

  For Eg. **ADD A,B** -To add the contents of register A & B.

High-Level Language

- The high-level language is easy to understand and the code can be written easily as the programs written are user-friendly in a high-level language. The high-level of language uses the concept of abstraction and also focuses on programming language rather than focusing on computer hardware components .
- Examples of high-level languages are C++, C, JAVA, FORTRAN, Pascal, Perl, Ruby, and Visual Basic.

Advantages:

- The syntax used and the programming style can be easily understood by humans if it is compared to low-level language.
- The only requirement in a high-level language is the need for a compiler. As the program written in a high-level language is not directly understood by the computer system.

High Level Language

Computer Languages In-Demand

- The list of computer languages is never-ending. From time to time the demand for computer languages also fluctuates. Some languages were in demand years ago but as technology changes the demand also changes. Here is the list of computer languages in demand all around the world:
- Ruby
- JavaScript
- Python
- PHP
- Java
- C#
- Objective-C & Swift

References

- [PDF] Systems Programming and Operating Systems by Dhamdhere - Free Download PDF (dlscrib.com)
- [PDF] Principles of Compiler Design By Alfred V. Aho & J.D.Ullman Free Download – Learnengineering.in

THANK YOU