

Lab 1**Brief Introduction of COAL and Software Installation**

Computer Organization refers to the way in which the hardware components of a computer system are arranged and interconnected. Computer Organization and Assembly Language (COAL) Programming deals with lower-level computer programming—machine or assembly language, and how these are used in the typical computer system. Assembly language programming develops a very basic and low-level understanding of the computer. In higher level languages there is a distance between the computer and the programmer. This is because higher level languages are designed to be closer and friendlier to the programmer, thereby creating distance with the machine. This distance is covered by translators which are called compilers and interpreters. The aim of programming in assembly language is to bypass these intermediates and talk directly with the computer.

Different architectures of the computer system are adopted to understand the rules, methods, and procedures that describe the execution and functionality of the entire computer system. John von Neumann coined and developed an architecture named **von Neumann architecture**. The computer we are using nowadays is based on the von Neumann architecture. It has some concepts. It is also known as Princeton architecture. It renders a unique design for the electronic digital systems having the following components:

- A Central Processing Unit (CPU) with arithmetic and logic unit (ALU) and processors with attached registers.
- A memory that can store data and instructions.
- External mass storage or secondary storage.
- A Control Unit (CU) with the ability to hold instructions in the program counter (PC) or instruction register (IR).
- Input and output mechanisms and peripherals.

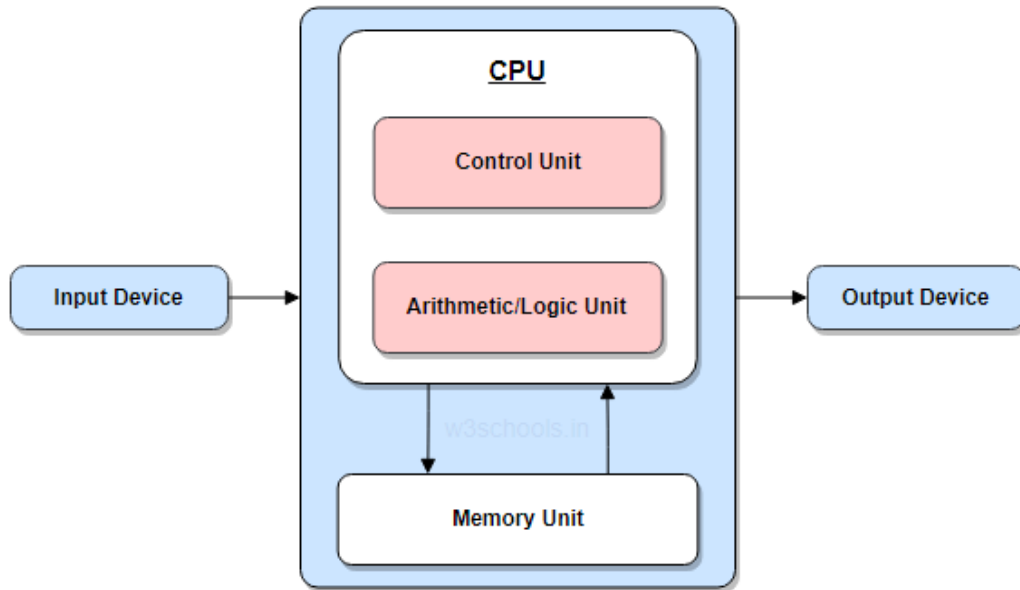


Figure 1: Von Neumann Architecture

The von Neumann design thus constitutes the foundation of modern computing. The Harvard architecture, a similar model, had committed data addresses and buses for reading and writing to memory. It won because von Neumann's architecture was easier to execute in real hardware.

Software Required

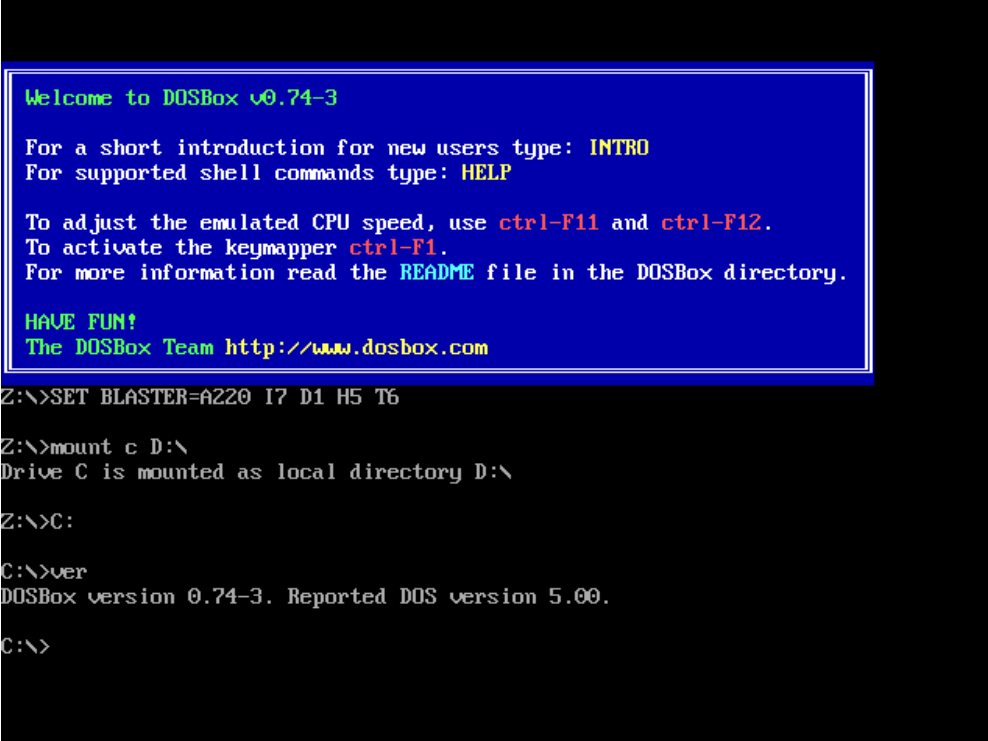
- DOSBox (An Emulator)
- Notepad ++/Visual Studio Code for Assembly Coding

Objective of Using DOSBox

DOSBox is an emulator and the purpose of using it here is to study a specific 16-bit architecture of Intel IBM PC using assembly language on a 32-bit or 64-bit machine.



Figure 2: DOSBox Icon



```
Welcome to DOSBox v0.74-3

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount c D:\
Drive C is mounted as local directory D:\

Z:\>C:

C:\>ver
DOSBox version 0.74-3. Reported DOS version 5.00.

C:\>
```

Figure 3: Interface of the DOSBox

Steps of Installation for Windows

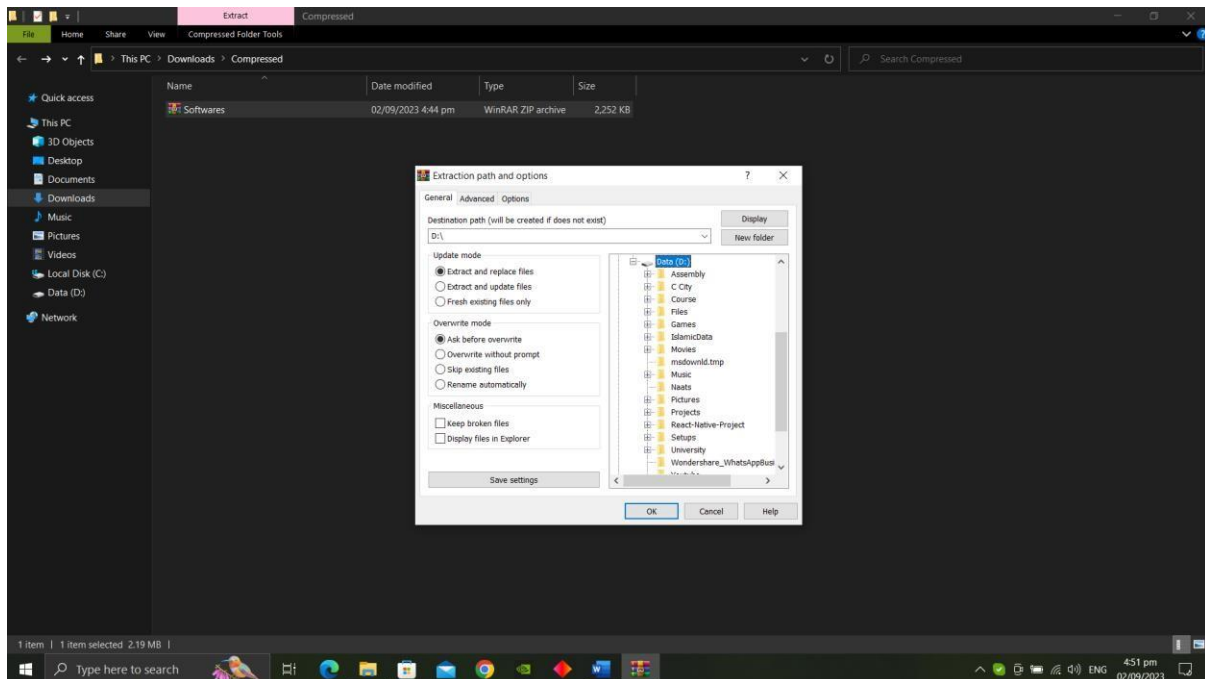
Follow these steps if you have a Windows PC to download and install DOSBox:

- Open a browser and go to the [DOSBox download page](#).
- Locate the latest Windows installer download and select it.
- After the download is completed, run the installer.
- Follow the on-screen directions. It's fairly simple, and the default options should suffice in most situations.

Lab Work

- Installation of the Software
- Getting familiar with the interface of DOSBox and making **.asm** files in NotePad++

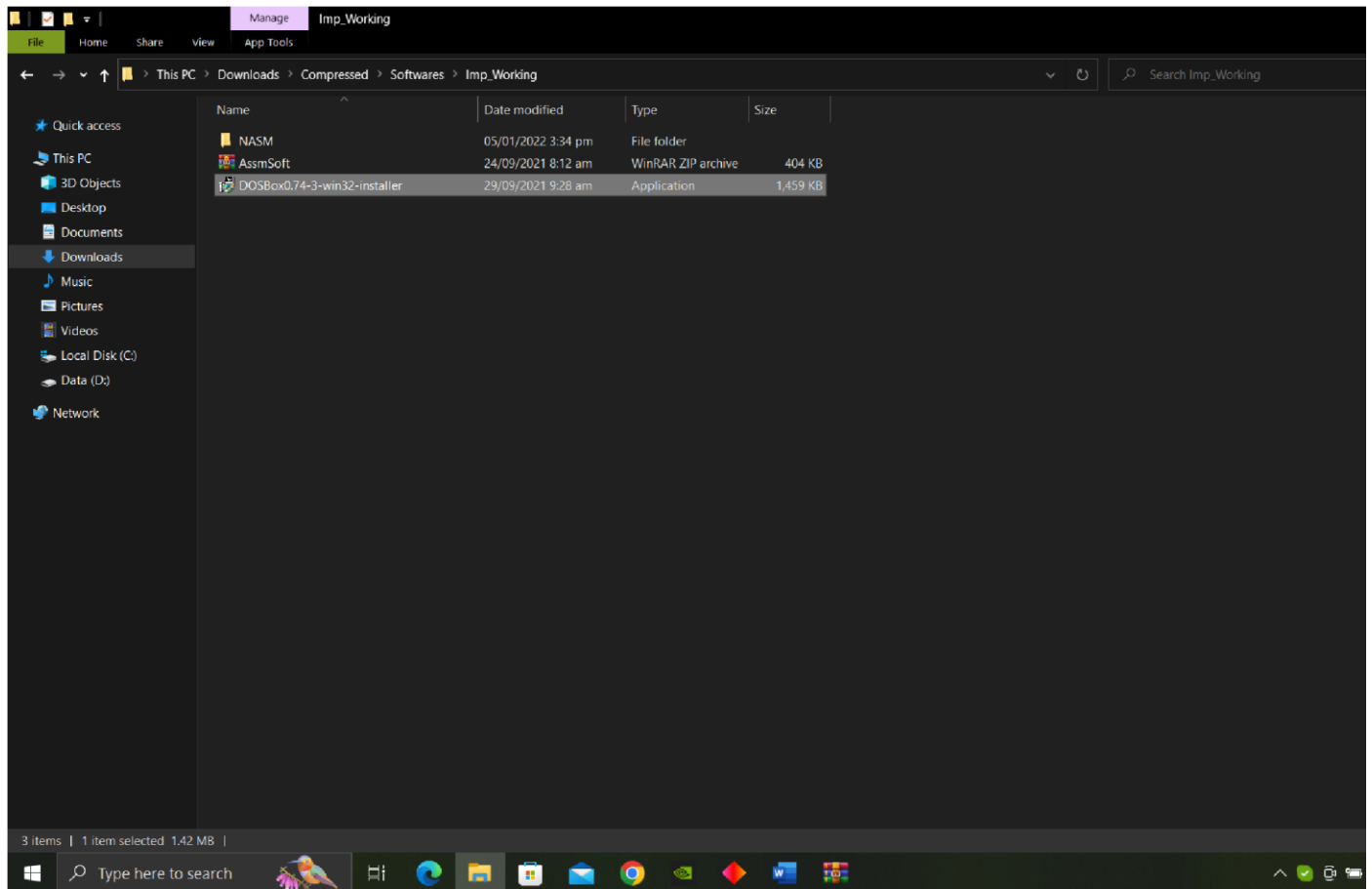
Extract the folder and make a folder in any drive other than name it "Assembly"



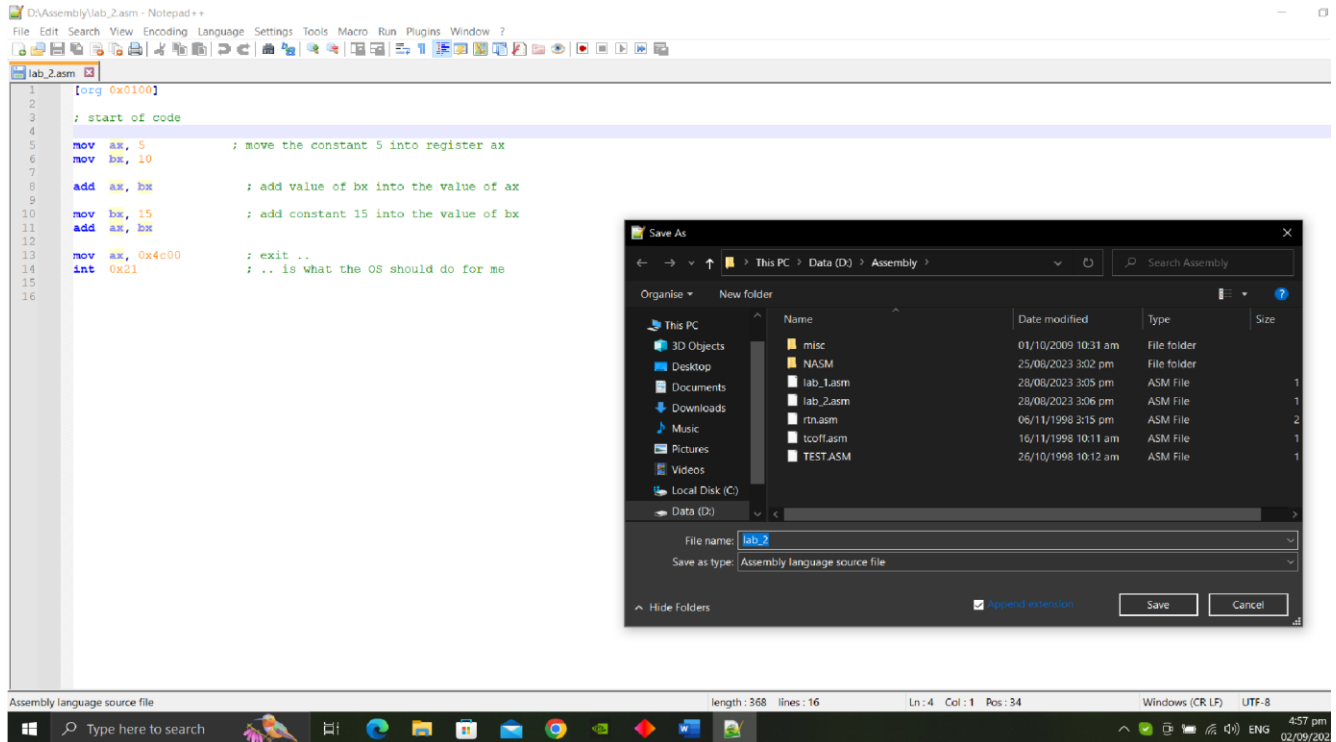
Open DOSBOX installer and go ahead and install it.

Copy the NASM folder to Assembly and also extract the files from Assmsoft to the same directory.

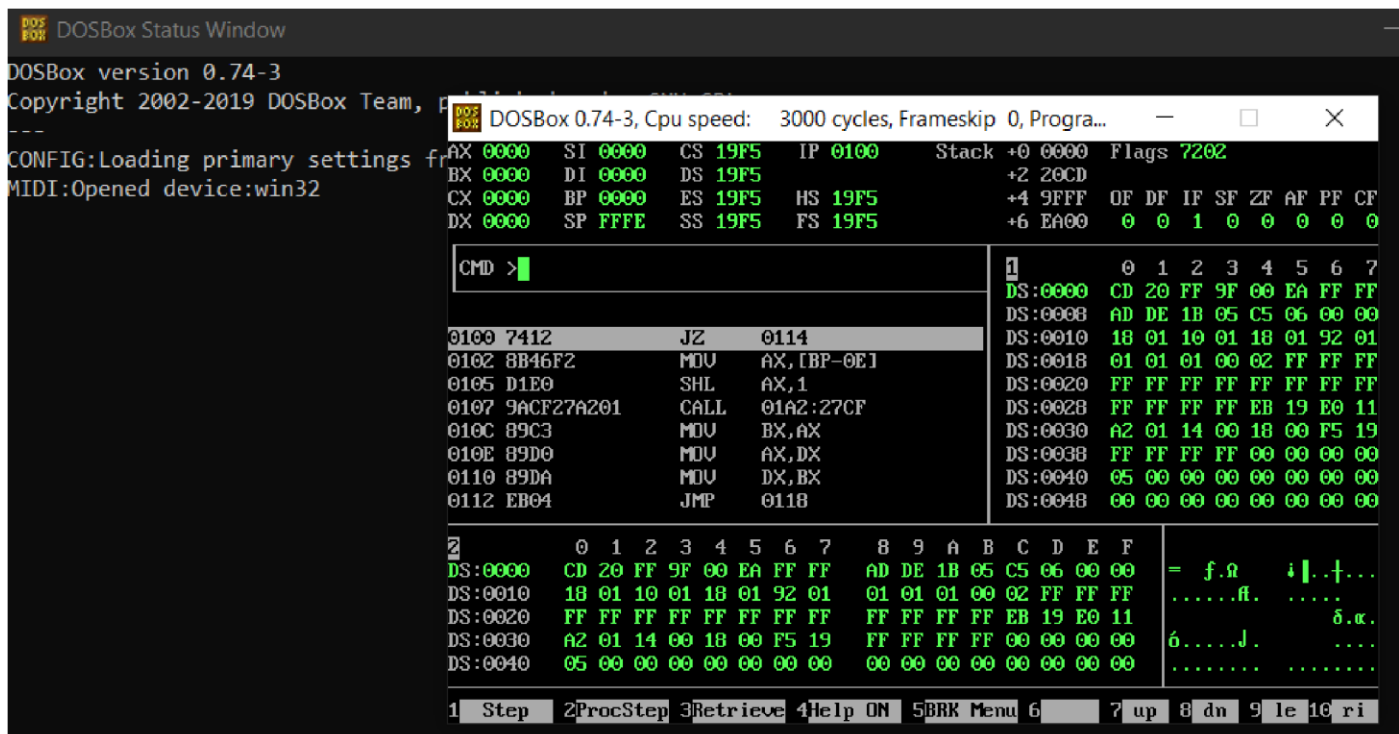
Installation Process is complete

**Title:** Running Our First Code.

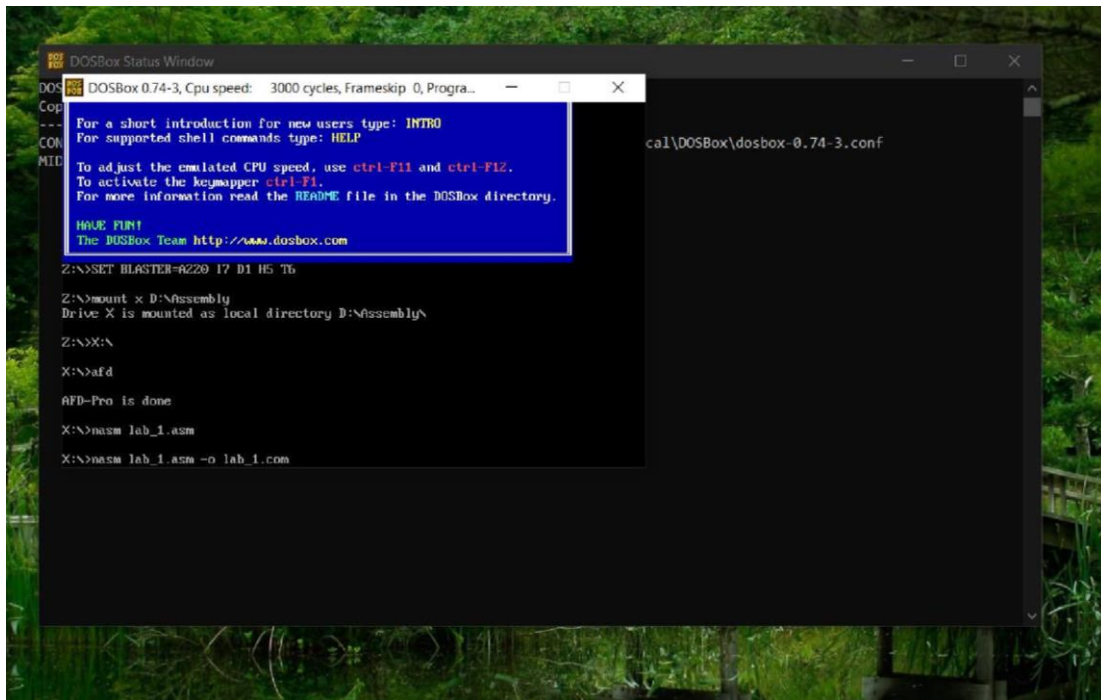
Open Notepad++ and write down the following lines of code and save the file, make sure you save the file as .asm in the same directory as installation.



After you have saved it, Open Dos-box by clicking the icon on your desktop and run in the following commands. Once you have done it go ahead and enter afd command as per in the picture below and view your virtual machine, get familiar to the environment and by pressing f2 each next line of code is executed and you will see how our computer hardware interacts with it and view each value stored in our registers.



Now by pressing f2 each next instruction will be executed



The screenshot shows a DOSBox window with a command prompt. A blue help window is open, displaying instructions for new users. The command prompt shows the following commands and output:

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip: 0, Progra...
DOS:
Cop:
---
COM:
MID:

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount x D:\Assembly
Drive X is mounted as local directory D:\Assembly\

Z:\>X:\
X:\>afid
AFD-Pro is done
X:\>nasm lab_1.asm
X:\>nasm lab_1.asm -o lab_1.com
```

These are the commands we will use to open the asm.com file for viewing execution on our virtual machine.