

## Volume Two: An Introduction to Machine Architecture

- Chapter One: System Organization  
A gentle introduction to the components that make up a typical PC.
- Chapter Two: Memory Access and Organization  
A discussion of the 80x86 memory addressing modes and how HLA organizes your data in memory.
- Chapter Three: Introduction to Digital Design  
A low-level description of how computer designers build CPUs and other system components.
- Chapter Four: CPU Architecture  
A look at the internal operation of the CPU.
- Chapter Five: Instruction Set Architecture  
This chapter describes how Intel's engineers designed the 80x86 instruction set. It also explains many of their design decisions, good and bad.
- Chapter Six: Memory Architecture  
How memory is organized for high performance computing systems.
- Chapter Seven: The I/O Subsystem  
Input and output are two of the most important functions on a PC. This chapter describes how input and output occurs on a typical 80x86 system.
- Chapter Eight: Questions, Projects, and Laboratory Exercises  
See what you've learned in this topic!

This topic, as its title suggests, is primarily targeted towards a machine organization course. Those who wish to study assembly language programming should at least read Chapter Two and possibly Chapter One. Chapter Three is a low-level discussion of digital logic. This information is important to those who are interested in design-

**Volume Two:**

**Machine Architecture**

ing CPUs and other system components. Those individuals who are main interested in programming can safely skip this chapter. Chapters Four, Five, and Six provide a more in-depth look at computer systems' architecture. Those wanting to know how things work "under the hood" will want to read these chapters. However, programmers who just want to learn assembly language programming can safely skip these chapters. Chapter Seven discusses I/O on the 80x86. Under modern 32-bit operating systems you will not be able to utilize much of this information unless you are writing device drivers. However, those interested in learning how low-level I/O takes place in assembly language will want to read this chapter.