



# Fundamentals of Parallel Processing

By Harry F. Jordan, Gita Alaghband

[Download now](#)

[Read Online](#) 

**Fundamentals of Parallel Processing** By Harry F. Jordan, Gita Alaghband

Rapid changes in the field of parallel processing make this book especially important for professionals who are faced daily with new products—and provides them with the level of understanding they need to evaluate and select the products. It gives readers a fundamental understanding of parallel processing application and system development. Chapter topics include parallel machines and computations, potential for parallel computations, vector algorithms and architectures, MIMD computers and multiprocessors, distributed memory processors, interconnection networks, data dependence and parallelism, implementing synchronization and data sharing, parallel processor performance, temporal behavior of parallel programs, and parallel I/O. For computational scientists, software engineers, computer architects, and computer engineers.

 [Download Fundamentals of Parallel Processing ...pdf](#)

 [Read Online Fundamentals of Parallel Processing ...pdf](#)

# Fundamentals of Parallel Processing

By Harry F. Jordan, Gita Alaghband

## Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband

Rapid changes in the field of parallel processing make this book especially important for professionals who are faced daily with new products—and provides them with the level of understanding they need to evaluate and select the products. It gives readers a fundamental understanding of parallel processing application and system development. Chapter topics include parallel machines and computations, potential for parallel computations, vector algorithms and architectures, MIMD computers and multiprocessors, distributed memory processors, interconnection networks, data dependence and parallelism, implementing synchronization and data sharing, parallel processor performance, temporal behavior of parallel programs, and parallel I/O. For computational scientists, software engineers, computer architects, and computer engineers.

## Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband Bibliography

- Rank: #273799 in Books
- Published on: 2002-08-26
- Ingredients: Example Ingredients
- Original language: English
- Number of items: 1
- Dimensions: 9.20" h x 1.20" w x 7.00" l, 2.00 pounds
- Binding: Paperback
- 536 pages

 [Download Fundamentals of Parallel Processing ...pdf](#)

 [Read Online Fundamentals of Parallel Processing ...pdf](#)

## Download and Read Free Online Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband

---

### Editorial Review

#### From the Back Cover

This unique book provides comprehensive coverage of the crucial fundamental concepts of parallel processing. The authors also introduce the integration of parallel architecture, algorithms, and language to provide insight into designing and implementing parallel applications. This book is suitable for advanced courses on the principles of parallel processing and is also a superb professional reference.

#### Beneficial Features:

- Presents subjects by integrating algorithms, languages, and architectures.
- Clearly explains performance analysis and implications resulting from influences of the three components.
- Guides readers with thought-provoking questions at the beginning of each chapter.
- Offers numerous examples to clarify difficult concepts.
- Helps students implement their learning with 141 end-of-chapter problems.
- Conveys performance implications of each new topic with consistent integration of algorithms, languages, and architectures throughout the text.

#### About the Author

**Harry F. Jordan** received the Ph.D. from the University of Illinois. He has been with the University of Colorado at Boulder since 1966 and is now a professor in the Departments of Electrical and Computer Engineering and Computer Science. Professor Jordan's interests in computer systems center on the interface between hardware and software, including supercomputers, multi-processor architecture, and optical computing.

**Gita Alaghband** received the Ph.D. in Electrical Engineering (1986) from the University of Colorado at Boulder. Currently, she is a professor in the Department of Computer Science and Engineering at the University of Colorado at Denver. Dr. Alaghband's research interests in parallel processing include computer architecture, performance evaluation, simulation, application programs, and algorithm designs.

Excerpt. © Reprinted by permission. All rights reserved.

### To the Student

Computing is usually taught from a step-by-step or serial point of view. Algorithms are organized as a sequence of computational steps, programs are written one command after another, and machines are designed to execute a chain of machine instructions by performing a string of microsteps, one after another. While sequential formulation of a problem can lead to a solution, a tremendous performance advantage is available from doing many operations in parallel. The two principal approaches to speeding up a computation are a faster clock rate for the underlying hardware and doing more operations in parallel. Introducing parallel operations to speed up an application is a promising approach, because as tasks become larger, more operations can potentially be done in parallel. To realize this potential, three things must work together. Algorithms must involve many independent operations, programming languages must allow the specification of parallel operations or identify them automatically, and the architecture of the computer

running the program must execute multiple operations simultaneously.

Parallel processing is the result of this combination of algorithm design, programming language structure, and computer architecture all directed toward faster completion of an application. The fundamentals of parallel processing emerge from an understanding of this combination of computing topics and their collaboration to achieve high performance. To gain this understanding, a basic knowledge of computer design and architecture, of programming languages and how they produce machine code, and of the elements of algorithm structure is required. Although some subsections focus exclusively on one of the three aspects of architecture, language, or algorithm, there are no such major divisions in the book. Treatments of all three are combined to expose the fundamental concepts that make up the discipline of parallel processing. We expect the reader to have a basic knowledge of algorithms and programming. To address the real goal of parallel processing—better performance—one must know how the program is executed by a computer at the machine language level. This requires an understanding of the specific organization of hardware elements constituting a machine architecture. Introductory experience in these areas constitutes the prerequisite material for reading this text.

## To the Instructor

The goal of this textbook is to provide a comprehensive coverage of the principles of parallel processing. Integration of parallel architectures, algorithms, and languages is the key in gaining both the breadth and the depth of knowledge and expertise needed in designing and developing successful parallel applications. The book is organized and presented so that it continuously relates these subjects within the topic being studied. Discussions of algorithm designs are followed by the performance implications of each design on parallel architectures.

The rapid changes in technology and the continuous arrival of new architectures, languages, and systems demand a fundamental understanding of the field of parallel processing. The uniqueness of this book is that it treats fundamental concepts rather than a collection of the latest trends. The flow of information is carefully designed so that each section is a natural next step from the previous one. Detailed examples are used to clarify difficult concepts. The issues to be studied are posed early enough to motivate the reader to continue and to give a clear picture of what is to come next and why. The alternative approach of covering "recent" architectures, languages, and systems as a vehicle to teach the fundamental concepts is difficult and quickly dated. It is very hard to get to the heart of a subject without the readers feeling lost and confused about what is really being conveyed. Peeling off some layers of additional information and features is necessary before getting to the fundamentals in every case. For example, is it necessary for a language to provide numerous constructs? Or are some of them considered essential and some additionally provided for ease of use? Are they implemented with efficiency in mind for certain architectures or are they provided for portability? Are the constructs implementation dependent? Will their performance vary by much on different computer architectures? It is never possible to completely understand the trade-offs and the underlying concepts by going over example machines and languages alone. Once the fundamental concepts are understood, they can be applied to any architecture, system, or language.

Parallel processing is a relatively young academic discipline. The authors believe that it has developed to a point where fundamentals can be identified and discussed apart from individual systems. We have focused on presenting the fundamentals by architectural features, system properties, language constructs, algorithm design and implementation implications in a way that is as independent as possible of specific architectures, systems, and languages. In some cases, the original machine, language, or system introducing the concept being presented is covered. However, in a majority of cases we have intentionally refrained from expanding each topic to cover many specific machines or languages for the purpose of concentrating on fundamentals.

Although this is not intended as a parallel programming text, a real programming language presented for each type of major parallelism concept introduced throughout the text. We selected Fortran as the base language whenever possible for several reasons. Much of the earth literature in parallel processing is Fortran-based, and there are numerous parallel Fortran scientific programs and programmers. In addition, Fortran is a simple high-level language Close to the machine level. It is easier to observe and explain the effects of executing Fortran statements on various machine architectures compared to high-level languages with many complex, user-friendly features. The Fortran program designer has much control over programming style, design, implementation, and execution. Fortran is a static language, so in comparison to dynamic languages or languages providing dynamic features, the programmer must be cautioned less regarding the use of high-level features and their parallel performance implications. The simplicity of the language helps keep the focus on parallel concepts and constructs. That multidimensional arrays are supported in Fortran is especially significant for vector processing. Maintaining the same base language throughout the book keeps the presentation consistent, and readers, not needing to switch between languages, will concentrate on parallel issues.

This textbook is designed and organized after many years of teaching and research experience in the field of parallel processing. It is intended for computer science or computer engineering seniors and graduate students. Students studying the book will be able to confidently design and implement new parallel applications, evaluate parallel program and architecture performance, and, most important be able to develop their skills by learning new parallel environments on their own. The major task of an educator is to nurture his or her students so that they can continue to grow and develop in their field of interest independently. This textbook is designed with this important goal in mind and will provide instructors with a comprehensive set of material to educate their students to be productive and successful.

## **Tailoring the Text to a Syllabus**

The first seven chapters constitute an excellent first-semester course in parallel processing. They give an in-depth coverage of parallel algorithm design, vector, multiprocessor, and dataflow architectures, parallel languages for each machine type, synchronization and communication mechanisms, interconnection networks, data dependence, and compiler optimization techniques. The remaining four chapters are intended for advanced treatment of issues studied in the first part of the book. The focus is on various synchronization and communication implementations, influence of implementations on performance, interpretation of machine architecture and program performance, effects of program behavior on performance, and parallel I/O. This part of the book provides an excellent second semester for graduate students. They will gain insight into how to analyze machine architectures, parallel programs and systems, and understand how these components interact and influence overall performance. Many advanced research project ideas can be deduced from topics covered in Chapters 8 through 11.

## **Chapter Contents**

Chapter 1 briefly reviews the evolution of parallelism in computer architectures. It introduces the basic ideas of vector processing, multiprocessing, and parallel operations in algorithms. It establishes a framework for topics in the remaining chapters.

Chapter 2 introduces the key ideas of data dependence. The prefix computation is used to illustrate algorithm characteristics that make different ways of doing the same computation more or less parallel.

Chapter 3 examines the application of the same operation to multiple data items in parallel. It motivates the discussion with some simple algorithms from linear algebra and presents an architecture at the machine language level that incorporates vector operations. Fortran 90 is discussed as a language with high-level

support for the unique features of machine level vector processing. Pipelined vector processing is discussed.

Chapter 4 briefly surveys multiprocessor architectural organizations and establishes the difference between shared and distributed memory multiprocessors. It proceeds to focus on shared memory by describing the extensions to sequential programming that are needed to coordinate multiple processes to perform a task. The OpenMP Fortran extension is used as an illustration of high-level constructs used to support shared memory multiprocessing. The chapter also establishes the basics of pipelined MIMD, or multithreaded, architectures.

Chapter 5 describes distributed memory multiprocessors using the message passing viewpoint to direct attention toward the dominant role of data communication in such architectures. Explicit send and receive programming is introduced, and the message passing interface (MPI) is used as an illustration of high-level language support for such programs. The basics of cache coherence and memory consistency are described in relation to shared memory and distributed memory multiprocessors.

Chapter 6 discusses interconnection networks in depth, including those for vector computers, and shared and distributed memory multiprocessors. Static and dynamic networks are compared and contrasted along with various topologies and their properties. Use of the network to combine messages, as in the NYU Ultracomputer, is discussed.

Chapter 7 is important in relating the ideas of data dependence that underlie the structure of parallel algorithms to the structure of a program. It covers code optimization techniques and topics of concern to a compiler writer having the task of generating code for a parallel computer. This chapter also introduces the ideas of dataflow languages and architectures that allow the elimination of nonessential dependences from programming languages and machines.

Chapter 8 expands the ideas of synchronization introduced in the shared memory discussion of Chapter 4 and integrates them with the data transmission point of view emphasized in Chapter 5. An in-depth understanding of key issues in synchronization is provided by a set of key topics, ranging from synchronization in cooperative communication, managing shared tasks, waiting mechanisms, to how to prove that a synchronization mechanism is implemented correctly.

Chapter 9 focuses specifically on the performance issues that have been continually referred to in previous chapters. It treats various performance models and illustrates their use through case studies of measurements on real systems. The impact of different scheduling and implementations of parallel constructs is discussed.

Chapter 10 relates performance of a parallel program execution to its temporal behavior. Experiments on real systems are used to illustrate performance characterization models. It examines temporal characterization from several viewpoints ranging from behavior in single cache systems, multiprocessor systems with distributed caches, to message passing systems.

Chapter 11 treats various aspects of parallelism in I/O operations. Parallel access disk arrays (RAID) are described as parallel I/O hardware. I/O dependence operations are introduced. Parallel input and output methods on files are discussed. Finally, parallelism in multiprocessors collective I/O operations is covered using MPI-10.

## Users Review

### From reader reviews:

**Katherine Sorenson:**

In this 21st centuries, people become competitive in every single way. By being competitive at this point, people have to do something to make them survive, being in the middle of the crowded place and notice through surrounding. One thing that occasionally many people have underestimated the idea for a while is reading. Yep, by reading a reserve your ability to survive raise then having chance to stay than other is high. For you personally who want to start reading a book, we give you this kind of Fundamentals of Parallel Processing book as beginner and daily reading guide. Why, because this book is more than just a book.

**Adrian White:**

Reading a book tends to be new life style in this particular era globalization. With looking at you can get a lot of information that will give you benefit in your life. Together with book everyone in this world could share their idea. Books can also inspire a lot of people. Lots of author can inspire their reader with their story or even their experience. Not only the storyline that share in the books. But also they write about the data about something that you need illustration. How to get the good score toefl, or how to teach your young ones, there are many kinds of book that you can get now. The authors nowadays always try to improve their proficiency in writing, they also doing some study before they write to the book. One of them is this Fundamentals of Parallel Processing.

**Carrie Mathis:**

Playing with family in a park, coming to see the water world or hanging out with pals is thing that usually you will have done when you have spare time, and then why you don't try matter that really opposite from that. 1 activity that make you not experience tired but still relaxing, trilling like on roller coaster you are ride on and with addition associated with. Even you love Fundamentals of Parallel Processing, you could enjoy both. It is excellent combination right, you still desire to miss it? What kind of hangout type is it? Oh occur its mind hangout folks. What? Still don't understand it, oh come on its called reading friends.

**Silvia Smedley:**

Reading a book being new life style in this yr; every people loves to examine a book. When you read a book you can get a lots of benefit. When you read publications, you can improve your knowledge, because book has a lot of information in it. The information that you will get depend on what sorts of book that you have read. In order to get information about your review, you can read education books, but if you act like you want to entertain yourself you can read a fiction books, this kind of us novel, comics, in addition to soon. The Fundamentals of Parallel Processing offer you a new experience in studying a book.

**Download and Read Online Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband #XLMB4KH3PSA**

# **Read Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband for online ebook**

Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband books to read online.

## **Online Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband ebook PDF download**

**Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband Doc**

**Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband Mobipocket**

**Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband EPub**

**XLMB4KH3PSA: Fundamentals of Parallel Processing By Harry F. Jordan, Gita Alaghband**