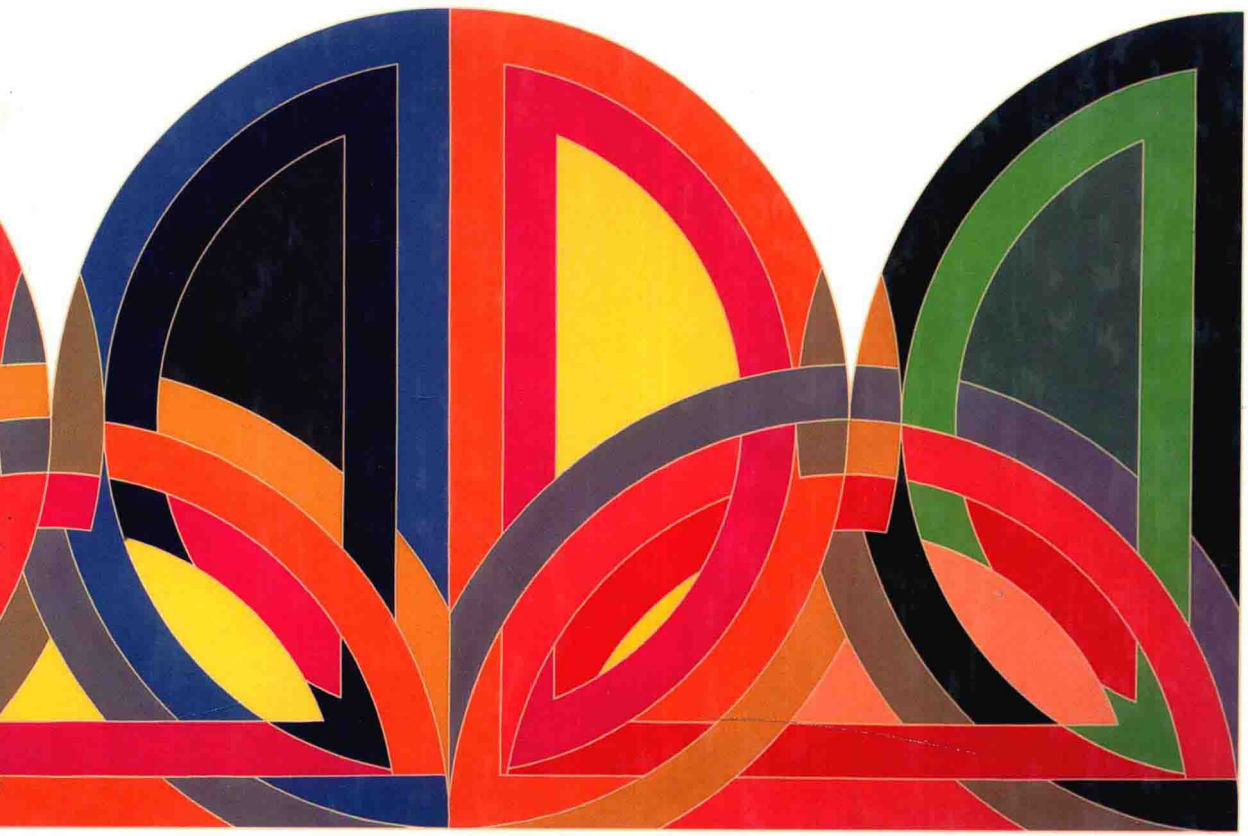
ELSEVIER

PRINCIPLES AND PRACTICES OF INTERCONNECTION NETWORKS

WILLIAM JAMES DALLY & BRIAN TOWLES



MORGAN KAUFMANN

Principles and Practices of Interconnection Networks

William James Dally

Brian Towles

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann is an imprint of Elsevier

ELSEVIER

MORGAN KAUFMANN PUBLISHERS

Publishing Director:	*Diane D. Cerra*
Senior Editor:	*Denise E. M. Penrose*
Publishing Services Manager:	*Simon Crump*
Project Manager:	*Marcy Barnes-Henrie*
Editorial Coordinator:	*Alyson Day*
Editorial Assistant:	*Summer Block*
Cover Design:	*Hannus Design Associates*
Cover Image:	*Frank Stella*, Takht-i-Sulayan-I (1967)
Text Design:	*Rebecca Evans & Associates*
Composition:	*Integra Software Services Pvt., Ltd.*
Copyeditor:	*Catherine Albano*
Proofreader:	*Deborah Prato*
Indexer:	*Sharon Hilgenberg*
Interior printer	*The Maple-Vail Book Manufacturing Group*
Cover printer	*Phoenix Color Corp.*

Morgan Kaufmann Publishers is an imprint of Elsevier
500 Sansome Street, Suite 400, San Francisco, CA 94111

This book is printed on acid-free paper.

Figure 3.13 courtesy of the Association for Computing Machinery (ACM), from James Laudon and Daniel Lenoski, "The SGI Origin: a ccNUMA highly scalable server," Proceedings of the International Symposium on Computer Architecture (ISCA), pp. 241-251, 1997. (ISBN: 0897919017) Figure 10.

Figure 10.7 from Thinking Machines Corp.

Figure 11.5 courtesy of Ray Mains, Ray Mains Photography,
http://www.mauigateway.com/~raymains/.

Library of Congress Cataloging-in-Publication Data

Dally, William J.
Principles and practices of interconnection networks / William Dally, Brian Towles.
p. cm.
Includes bibliographical references and index.
ISBN 0-12-200751-4 (alk. paper)
1. Computer networks-Design and construction.
2. Multiprocessors. I. Towles, Brian. II. Title.

TK5105.5.D3272003
004.6'5–dc22

ISBN: 0-12-200751-4 2003058915

For information on all Morgan Kaufmann publications,
visit our Web Site at *www.mkp.com*

Printed in the United States of America
04 05 06 07 08 5 4 3 2 1

E FOR
PRINCIPLES AND PRACTICES OF INTERCONNECTION NETWORKS

The scholarship of this book is unparalleled in its area. This text is for interconnection networks what Hennessy and Patterson's text is for computer architecture — an authoritative, one-stop source that clearly and methodically explains the more significant concepts. Treatment of the material both in breadth and in depth is very well done . . . a must read and a slam dunk! — Timothy Mark Pinkston, University of Southern California

[This book is] the most comprehensive and coherent work on modern interconnection networks. As leaders in the field, Dally and Towles capitalize on their vast experience as researchers and engineers to present both the theory behind such networks and the practice of building them. This book is a necessity for anyone studying, analyzing, or designing interconnection networks. — Stephen W. Keckler, The University of Texas at Austin

This book will serve as excellent teaching material, an invaluable research reference, and a very handy supplement for system designers. In addition to documenting and clearly presenting the key research findings, the book's incisive practical treatment is unique. By presenting how actual design constraints impact each facet of interconnection network design, the book deftly ties theoretical findings of the past decades to real systems design. This perspective is critically needed in engineering education. — Li-Shiuan Peh, Princeton University

Principles and Practices of Interconnection Networks is a triple threat: comprehensive, well written and authoritative. The need for this book has grown with the increasing impact of interconnects on computer system performance and cost. It will be a great tool for students and teachers alike, and will clearly help practicing engineers build better networks. — Steve Scott, Cray, Inc.

Dally and Towles use their combined three decades of experience to create a book that elucidates the theory and practice of computer interconnection networks. On one hand, they derive fundamentals and enumerate design alternatives. On the other, they present numerous case studies and are not afraid to give their experienced opinions on current choices and future trends. This book is a "must buy" for those interested in or designing interconnection networks. — Mark Hill, University of Wisconsin, Madison

This book will instantly become a canonical reference in the field of interconnection networks. Professor Dally's pioneering research dramatically and permanently changed this field by introducing rigorous evaluation techniques and creative solutions to the challenge of high-performance computer system communication. This well-organized textbook will benefit both students and experienced practitioners. The presentation and exercises are a result of years of classroom experience in creating this material. All in all, this is a must-have source of information. — Craig Stunkel, IBM

Principles and Practices of Interconnection Networks

Acknowledgments

We are deeply indebted to a large number of people who have contributed to the creation of this book. Timothy Pinkston at USC and Li-Shiuan Peh at Princeton were the first brave souls (other than the authors) to teach courses using drafts of this text. Their comments have greatly improved the quality of the finished book. Mitchell Gusat, Mark Hill, Li-Shiuan Peh, Timothy Pinkston, and Craig Stunkel carefully reviewed drafts of this manuscript and provided invaluable comments that led to numerous improvements.

Many people (mostly designers of the original networks) contributed information to the case studies and verfied their accuracy. Randy Rettberg provided information on the BBN Butterfly and Monarch. Charles Leiserson and Bradley Kuszmaul filled in the details of the Thinking Machines CM-5 network. Craig Stunkel and Bulent Abali provided information on the IBM SP1 and SP2. Information on the Alpha 21364 was provided by Shubu Mukherjee. Steve Scott provided information on the Cray T3E. Greg Thorson provided the pictures of the T3E.

Much of the development of this material has been influenced by the students and staff that have worked with us on interconnection network research projects at Stanford and MIT, including Andrew Chien, Scott Wills, Peter Nuth, Larry Dennison, Mike Noakes, Andrew Chang, Hiromichi Aoki, Rich Lethin, Whay Lee, Li-Shiuan Peh, Jin Namkoong, Arjun Singh, and Amit Gupta.

This material has been developed over the years teaching courses on interconnection networks: 6.845 at MIT and EE482B at Stanford. The students in these classes helped us hone our understanding and presentation of the material. Past TAs for EE482B Li-Shiuan Peh and Kelly Shaw deserve particular thanks.

We have learned much from discussions with colleagues over the years, including Jose Duato (Valencia), Timothy Pinkston (USC), Sudha Yalamanchili (Georgia Tech), Anant Agarwal (MIT), Tom Knight (MIT), Gill Pratt (MIT), Steve Ward (MIT), Chuck Seitz (Myricom), and Shubu Mukherjee (Intel). Our practical understanding of interconnection networks has benefited from industrial collaborations with Justin Rattner (Intel), Dave Dunning (Intel), Steve Oberlin (Cray), Greg Thorson (Cray), Steve Scott (Cray), Burton Smith (Cray), Phil Carvey (BBN and Avici), Larry Dennison (Avici), Allen King (Avici), Derek Chiou (Avici), Gopalkrishna Ramamurthy (Velio), and Ephrem Wu (Velio).

Denise Penrose, Summer Block, and Alyson Day have helped us throughout the project.

We also thank both Catherine Albano and Deborah Prato for careful editing, and our production manager, Marcy Barnes-Henrie, who shepherded the book through the sometimes difficult passage from manuscript through finished product.

Finally, our families: Sharon, Jenny, Katie, and Liza Dally and Herman and Dana Towles offered tremendous support and made significant sacrifices so we could have time to devote to writing.

Preface

Digital electronic systems of all types are rapidly becoming *commmunication limited*. Movement of data, not arithmetic or control logic, is the factor limiting cost, performance, size, and power in these systems. At the same time, buses, long the mainstay of system interconnect, are unable to keep up with increasing performance requirements.

Interconnection networks offer an attractive solution to this communication crisis and are becoming pervasive in digital systems. A well-designed interconnection network makes efficient use of scarce communication resources — providing high-bandwidth, low-latency communication between clients with a minimum of cost and energy.

Historically used only in high-end supercomputers and telecom switches, interconnection networks are now found in digital systems of all sizes and all types. They are used in systems ranging from large supercomputers to small embedded systems-on-a-chip (SoC) and in applications including inter-processor communication, processor-memory interconnect, input/output and storage switches, router fabrics, and to replace dedicated wiring.

Indeed, as system complexity and integration continues to increase, many designers are finding it more efficient to route packets, not wires. Using an interconnection network rather than dedicated wiring allows scarce bandwidth to be shared so it can be used efficiently with a high duty factor. In contrast, dedicated wiring is idle much of the time. Using a network also enforces regular, structured use of communication resources, making systems easier to design, debug, and optimize.

The basic principles of interconnection networks are relatively simple and it is easy to design an interconnection network that efficiently meets all of the requirements of a given application. Unfortunately, if the basic principles are not understood it is also easy to design an interconnection network that works poorly if at all. Experienced engineers have designed networks that have deadlocked, that have performance bottlenecks due to a poor topology choice or routing algorithm, and that realize only a tiny fraction of their peak performance because of poor flow control. These mistakes would have been easy to avoid if the designers had understood a few simple principles.

This book draws on the experience of the authors in designing interconnection networks over a period of more than twenty years. We have designed tens of networks that today form the backbone of high-performance computers (both message-passing

and shared-memory), Internet routers, telecom circuit switches, and I/O interconnect. These systems have been designed around a variety of topologies including crossbars, tori, Clos networks, and butterflies. We developed wormhole routing and virtual-channel flow control. In designing these systems and developing these methods we learned many lessons about what works and what doesn't. In this book, we share with you, the reader, the benefit of this experience in the form of a set of simple principles for interconnection network design based on topology, routing, flow control, and router architecture.

Organization

The book starts with two introductory chapters and is then divided into five parts that deal with topology, routing, flow control, router architecture, and performance. A graphical outline of the book showing dependences between sections and chapters is shown in Figure 1. We start in Chapter 1 by describing what interconnection networks are, how they are used, the performance requirements of their different applications, and how design choices of topology, routing, and flow control are made to satisfy these requirements. To make these concepts concrete and to motivate the remainder of the book, Chapter 2 describes a simple interconnection network in detail: from the topology down to the Verilog for each router. The detail of this example demystifies the abstract topics of routing and flow control, and the performance issues with this simple network motivate the more sophisticated methods and design approaches described in the remainder of the book.

The first step in designing an interconnection network is to select a topology that meets the throughput, latency, and cost requirements of the application given a set of packaging constraints. Chapters 3 through 7 explore the topology design space. We start in Chapter 3 by developing topology metrics. A topology's bisection bandwidth and diameter bound its achievable throughput and latency, respectively, and its path diversity determines both performance under adversarial traffic and fault tolerance. Topology is constrained by the available packaging technology and cost requirements with both module pin limitations and system wire bisection governing achievable channel width. In Chapters 4 through 6, we address the performance metrics and packaging constraints of several common topologies: butterflies, tori, and non-blocking networks. Our discussion of topology ends at Chapter 7 with coverage of concentration and toplogy *slicing*, methods used to handle bursty traffic and to map topologies to packaging modules.

Once a topology is selected, a routing algorithm determines how much of the bisection bandwidth can be converted to system throughput and how closely latency approaches the diameter limit. Chapters 4 through 11 describe the routing problem and a range of solutions. A good routing algorithm load-balances traffic across the channels of a topology to handle adversarial traffic patterns while simultaneously exploiting the locality of benign traffic patterns. We introduce the problem in Chapter 8 by considering routing on a ring network and show that the naive *greedy* algorithm gives poor performance on adversarial traffic. We go on to describe oblivious

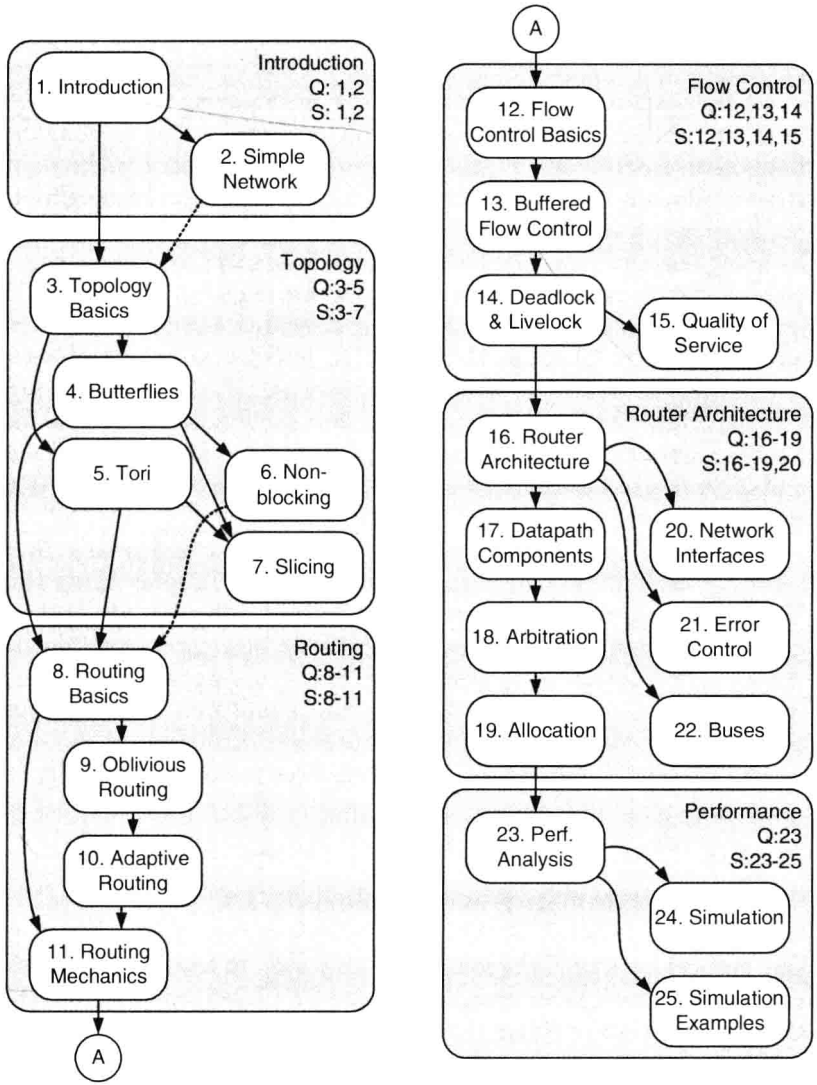


Figure 1 Outline of this book showing dependencies between chapters. Major sections are denoted as shaded areas. Chapters that should be covered in any course on the subject are placed along the left side of the shaded areas. Optional chapters are placed to the right. Dependences are indicated by arrows. A solid arrow implies that the chapter at the tail of the arrow must be understood to understand the chapter at the head of the arrow. A dotted arrow indicates that it is helpful, but not required, to understand the chapter at the tail of the arrow before the chapter at the head. The notation in each shaded area recommends which chapters to cover in a quarter course (Q) and a semester course (S).

routing algorithms in Chapter 9 and adaptive routing algorithms in Chapter 10. The routing portion of the book then concludes with a discussion of routing mechanics in Chapter 11.

A flow-control mechanism sequences packets along the path from source to destination by allocating channel bandwidth and buffer capacity along the way. A good flow-control mechanism avoids idling resources or blocking packets on resource constraints, allowing it to realize a large fraction of the potential throughput and minimizing latency respectively. A bad flow-control mechanism may squander throughput by idling resources, increase latency by unnecessarily blocking packets, and may even result in deadlock or livelock. These topics are explored in Chapters 12 through 15.

The policies embedded in a routing algorithm and flow-control method are realized in a router. Chapters 16 through 22 describe the microarchitecture of routers and network interfaces. In these chapters, we introduce the building blocks of routers and show how they are composed. We then show how a router can be pipelined to handle a flit or packet each cycle. Special attention is given to problems of *arbitration* and *allocation* in Chapters 18 and 19 because these functions are critical to router performance.

To bring all of these topics together, the book closes with a discussion of network performance in Chapters 23 through 25. In Chapter 23 we start by defining the basic performance measures and point out a number of common pitfalls that can result in misleading measurements. We go on to introduce the use of queueing theory and probablistic analysis in predicting the performance of interconnection networks. In Chapter 24 we describe how simulation is used to predict network performance covering workloads, measurement methodology, and simulator design. Finally, Chapter 25 gives a number of example performance results.

Teaching Interconnection Networks

The authors have used the material in this book to teach graduate courses on interconnection networks for over 10 years at MIT (6.845) and Stanford (EE482b). Over the years the class notes for these courses have evolved and been refined. The result is this book.

A one quarter or one semester course on interconnection networks can follow the outline of this book, as indicated in Figure 1. An individual instructor can add or delete the optional chapters (shown to the right side of the shaded area) to tailor the course to their own needs.

One schedule for a one-quarter course using this book is shown in Table 1 . Each lecture corresponds roughly to one chapter of the book. A semester course can start with this same basic outline and add additional material from the optional chapters.

In teaching a graduate interconnections network course using this book, we typically assign a research or design project (in addition to assigning selected exercises from each chapter). A typical project involves designing an interconnection network (or a component of a network) given a set of constraints, and comparing the performance of alternative designs. The design project brings the course material together

Table 1 One schedule for a ten-week quarter course on interconnection networks. Each chapter covered corresponds roughly to one lecture. In week 3, Chapter 6 through Section 6.3.1 is covered.

Week	Topic	Chapters
1	Introduction	1, 2
2	Topology	3, 4
3	Topology	5, (6)
4	Routing	8, 9
5	Routing	10, 11
6	Flow Control	12, 13, 14
7	Router Architecture	16, 17
8	Arbitration & Allocation	18, 19
9	Performance	23
10	Review	

for students. They see the interplay of the different aspects of interconnection network design and get to apply the principles they have learned first hand.

Teaching materials for a one quarter course using this book (Stanford EE482b) are available on-line at `http://cva.stanford.edu/ee482b`. This page also includes example projects and student papers from the last several offerings of this course.

About the Authors

Bill Dally received his B.S. in electrical engineering from Virginia Polytechnic Institute, an M.S. in electrical engineering from Stanford University, and a Ph.D. in computer science from Caltech. Bill and his group have developed system architecture, network architecture, signaling, routing, and synchronization technology that can be found in most large parallel computers today. While at Bell Telephone Laboratories, Bill contributed to the design of the BELLMAC32 microprocessor and designed the MARS hardware accelerator. At Caltech he designed the MOSSIM Simulation Engine and the Torus Routing Chip, which pioneered wormhole routing and virtual-channel flow control. While a Professor of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology, his group built the J-Machine and the M-Machine, experimental parallel computer systems that pioneered the separation of mechanisms from programming models and demonstrated very low overhead synchronization and communication mechanisms. Bill is currently a professor of electrical engineering and computer science at Stanford University. His group at Stanford has developed the Imagine processor, which introduced the concepts of stream processing and partitioned register organizations. Bill has worked with Cray Research and Intel to incorporate many of these innovations in commercial parallel computers. He has also worked with Avici Systems to incorporate this technology into Internet routers, and co-founded Velio Communications to commercialize high-speed signaling technology. He is a fellow of the IEEE, a fellow of the ACM, and has received numerous honors including the ACM Maurice Wilkes award. He currently leads projects on high-speed signaling, computer architecture, and network architecture. He has published more than 150 papers in these areas and is an author of the textbook *Digital Systems Engineering* (Cambridge University Press, 1998).

Brian Towles received a B.CmpE in computer engineering from the Georgia Institute of Technology in 1999 and an M.S. in electrical engineering from Stanford University in 2002. He is currently working toward a Ph.D. in electrical engineering at Stanford University. His research interests include interconnection networks, network algorithms, and parallel computer architecture.

Contents