

# VLSI

## Design Methodology Development



THOMAS DILLINGER

FREE SAMPLE CHAPTER

SHARE WITH OTHERS



---

# **VLSI DESIGN METHODOLOGY DEVELOPMENT**

---

Thomas Dillinger

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at [corpsales@pearsoned.com](mailto:corpsales@pearsoned.com) or (800) 382-3419.

For government sales inquiries, please contact [governmentsales@pearsoned.com](mailto:governmentsales@pearsoned.com).  
For questions about sales outside the U.S., please contact [intlcs@pearson.com](mailto:intlcs@pearson.com).

Visit us on the Web: [informit.com/aw](http://informit.com/aw)

*Library of Congress Control Number:* 2019937605  
Copyright © 2019 Pearson Education, Inc.

All rights reserved. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, request forms and the appropriate contacts within the Pearson Education Global Rights & Permissions Department, please visit [www.pearsoned.com/permissions/](http://www.pearsoned.com/permissions/).

ISBN-13: 978-0-13-573241-0

ISBN-10: 0-13-573241-7

ScoutAutomatedPrintCode

**Acquisitions Editor**  
Malobika Chakraborty

**Managing Editor**  
Sandra Schroeder

**Senior Project Editor**  
Lori Lyons

**Copy Editor**  
Catherine D. Wilson

**Production Manager**  
Abirami Ulaganathan

**Indexer**  
Erika Millen

**Proofreader**  
Christopher Morris

**Cover Designer**  
Chuti Prasertsith

**Compositor**  
codeMantra

*To Pat, for his inspiration  
and  
To Martha, who loved to write*

*This page intentionally left blank*

---

# CONTENTS AT A GLANCE

Preface.....	xiv
<b>Topic I: Overview of VLSI Design Methodology</b>	
Chapter 1 Introduction.....	13
Chapter 2 VLSI Design Methodology .....	131
Chapter 3 Hierarchical Design Decomposition .....	193
<b>Topic II: Modeling</b>	
Chapter 4 Cell and IP Modeling .....	223
<b>Topic III: Design Validation</b>	
Chapter 5 Characteristics of Functional Validation .....	259
Chapter 6 Characteristics of Formal Equivalency Verification .....	301
<b>Topic IV: Design Implementation</b>	
Chapter 7 Logic Synthesis .....	319
Chapter 8 Placement .....	363
Chapter 9 Routing .....	373

## Topic V: Electrical Analysis

Chapter 10	Layout Parasitic Extraction and Electrical Modeling.....	405
Chapter 11	Timing Analysis .....	443
Chapter 12	Noise Analysis.....	475
Chapter 13	Power Analysis.....	495
Chapter 14	Power Rail Voltage Drop Analysis .....	509
Chapter 15	Electromigration (EM) Reliability Analysis .....	529
Chapter 16	Miscellaneous Electrical Analysis Requirements .....	559

## Topic VI: Preparation for Manufacturing Release and Bring-Up

Chapter 17	ECOs.....	595
Chapter 18	Physical Design Verification .....	607
Chapter 19	Design for Testability Analysis .....	625
Chapter 20	Preparation for Tapeout .....	677
Chapter 21	Post-Silicon Debug and Characterization (“Bring-up”) and Product Qualification .....	693
Epilogue	.....	705
Index	.....	711

Register your copy of *VLSI Design Methodology Development* at [informit.com](http://informit.com) for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to [informit.com/register](http://informit.com/register) and log in or create an account. Enter the product ISBN 9780135732410 and click Submit. Once the process is complete, you will find any available bonus content under “Registered Products.”

---

# C O N T E N T S

<b>Preface .....</b>	<b>xiv</b>
<b>Topic I: Overview of VLSI Design Methodology 1</b>	
I.1 Methodology Guidelines for Logical and Physical Design Hierarchy Correspondence 6	
I.2 Managing Inter-Block Glue Logic 8	
<b>Chapter 1 Introduction.....</b>	<b>13</b>
1.1 Definitions 13	
1.2 Intellectual Property (IP) Models 21	
1.3 Tapeout and NRE Fabrication Cost 42	
1.4 Fabrication Technology 44	
1.5 Power and Clock Domains On-chip 105	
1.6 Physical Design Planning 113	
1.7 Summary 126	
References 127	
Further Research 129	
<b>Chapter 2 VLSI Design Methodology .....</b>	<b>131</b>
2.1 IP Design Methodology 131	
2.2 SoC Physical Design Methodology 141	
2.3 EDA Tool and Release Flow Management 165	
2.4 Design Methodology “Trailblazing” and Reference Flows 168	

2.5	Design Data Management (DDM)	171
2.6	Power and Clock Domain Management	175
2.7	Design for Testability (DFT)	177
2.8	Design-for-Manufacturability (DFM) and Design-for-Yield (DFY) Requirements	184
2.9	Design Optimization	185
2.10	Methodology Checks	186
	References	190
	Further Research	190

### **Chapter 3 Hierarchical Design Decomposition ..... 193**

3.1	Logical-to-Physical Correspondence	193
3.2	Division of SRAM Array Versus Non-Array Functionality	197
3.3	Division of Dataflow and Control Flow Functionality	198
3.4	Design Block Size for Logic Synthesis and Physical Design	202
3.5	Power and Clock Domain Considerations	206
3.6	Opportunities for Reuse of Hierarchical Units	207
3.7	Automated Test Pattern Generation (ATPG) Limitations	208
3.8	Intangibles	211
3.9	The Impact of Changes to the SoC Model Hierarchy During Design	212
3.10	Generating Hierarchical Electrical Abstracts Versus Top-Level Flat Analysis	214
3.11	Methodologies for Top-Level Logical and Physical Hierarchies	216
3.12	Summary	218
	References	219
	Further Research	219

## **Topic II: Modeling 221**

### **Chapter 4 Cell and IP Modeling ..... 223**

4.1	Functional Modeling for Cells and IP	223
4.2	Physical Models for Library Cells	240
4.3	Library Cell Models for Analysis Flows	241
4.4	Design for End-of-Life (EOL) Circuit Parameter Drift	251
4.5	Summary	253
	References	253
	Further Research	254

**Topic III: Design Validation 257****Chapter 5 Characteristics of Functional Validation .....259**

- 5.1 Software Simulation 259
- 5.2 Testbench Stimulus Development 262
- 5.3 Hardware-Accelerated Simulation: Emulation and Prototyping 268
- 5.4 Behavioral Co-simulation 275
- 5.5 Switch-Level and Symbolic Simulation 275
- 5.6 Simulation Throughput and Resource Planning 281
- 5.7 Validation of Production Test Patterns 284
- 5.8 Event Trace Logging 288
- 5.9 Model Coverage Analysis 289
- 5.10 Switching Activity Factor Estimates for Power Dissipation Analysis 295
- 5.11 Summary 296
  - References 297
  - Further Research 298

**Chapter 6 Characteristics of Formal Equivalency Verification .....301**

- 6.1 RTL Combinational Model Equivalency 301
- 6.2 State Mapping for Equivalency 302
- 6.3 Combinational Logic Cone Analysis 305
- 6.4 Use of Model Input Assertions for Equivalency 306
- 6.5 Sequential Model Equivalency 307
- 6.6 Functional and Test-Mode Equivalence Verification 309
- 6.7 Array Equivalence Verification 310
- 6.8 Summary 313
  - References 314
  - Further Research 314

**Topic IV: Design Implementation 317****Chapter 7 Logic Synthesis .....319**

- 7.1 Level of Hardware Description Language Modeling 319
- 7.2 Generation and Verification of Timing Constraints 320
- 7.3 Technology Mapping to the Cell Library 328
- 7.4 Signal Repowering and “High-Fan-out” Net Synthesis (HFNS) 335

7.5	Post-Synthesis Netlist Characteristics	339
7.6	Synthesis with a Power Format File	340
7.7	Post-Technology Mapping Optimizations for Timing and Power	343
7.8	Hold Timing Optimization	348
7.9	Clock Tree Synthesis (CTS)	350
7.10	Integration of Hard IP Macros in Synthesis	353
7.11	Low-Effort Synthesis (LES) Methodology	354
7.12	Summary	359
	References	360
	Further Research	360
<b>Chapter 8</b>	<b>Placement</b> .....	<b>363</b>
8.1	Global Floorplanning of Hierarchical Units	363
8.2	Parasitic Interconnect Estimation	366
8.3	Cell Placement	367
8.4	Clock Tree Local Buffer Placement	369
8.5	Summary	370
	References	370
	Further Research	370
<b>Chapter 9</b>	<b>Routing</b> .....	<b>373</b>
9.1	Routing Introduction	373
9.2	Global and Detailed Routing Phases	378
9.3	Back End Of Line Interconnect “Stacks”	383
9.4	Routing Optimizations	387
9.5	Summary	399
	References	400
	Further Research	400
<b>Topic V: Electrical Analysis</b>		<b>403</b>
<b>Chapter 10</b>	<b>Layout Parasitic Extraction and Electrical Modeling</b> .....	<b>405</b>
10.1	Introduction	405
10.2	Cell- and Transistor-Level Parasitic Modeling for Cell Characterization	411
10.3	Decoupling Capacitance Calculation for Power Grid Analysis	431
10.4	Interconnect Extraction	433
10.5	“Selected Net” Extraction Options	438

---

10.6	RLC Modeling	439
10.7	Summary	439
	References	440
	Further Research	442
<b>Chapter 11</b>	<b>Timing Analysis</b>	<b>443</b>
11.1	Cell Delay Calculation	443
11.2	Interconnect Delay Calculation	446
11.3	Electrical Design Checks	452
11.4	Static Timing Analysis	453
11.5	Summary	469
	References	470
	Further Research	472
<b>Chapter 12</b>	<b>Noise Analysis</b>	<b>475</b>
12.1	Introduction to Noise Analysis	475
12.2	Static Noise Analysis, Part I	476
12.3	Noise Impact on Delay	481
12.4	Electrical Models for Static Noise Analysis	485
12.5	Static Noise Analysis, Part II	488
12.6	Summary	491
	References	492
	Further Research	493
<b>Chapter 13</b>	<b>Power Analysis</b>	<b>495</b>
13.1	Introduction to Power Analysis	495
13.2	Models for Switching Activity Power Dissipation	497
13.3	IP Power Models	501
13.4	Device Self-Heat Models	502
13.5	Design-for-Power Feedback from Power Analysis	504
13.6	Summary	505
	References	506
	Further Research	506
<b>Chapter 14</b>	<b>Power Rail Voltage Drop Analysis</b>	<b>509</b>
14.1	Introduction to Power Rail Voltage Drop Analysis	509
14.2	Static I*R Rail Analysis	512

14.3	Dynamic P/G Voltage Drop Analysis	513
14.4	Summary	526
	References	526
	Further Research	527
<b>Chapter 15</b>	<b>Electromigration (EM) Reliability Analysis</b>	<b>529</b>
15.1	Introduction to EM Reliability Analysis	529
15.2	Fundamentals of Electromigration	535
15.3	Power Rail Electromigration Analysis: powerEM	545
15.4	Signal Interconnect Electromigration Analysis: sigEM	548
15.5	Summary	555
	References	555
	Further Research	556
<b>Chapter 16</b>	<b>Miscellaneous Electrical Analysis Requirements</b>	<b>559</b>
16.1	SleepFET Power Rail Analysis	559
16.2	Substrate Noise Injection and Latchup Analysis	562
16.3	Electrostatic Discharge (ESD) Checking	568
16.4	Soft Error Rate (SER) Analysis	576
16.5	Summary	590
	References	590
	Further Research	591
<b>Topic VI: Preparation for Manufacturing Release and Bring-Up</b>		<b>593</b>
<b>Chapter 17</b>	<b>ECOs</b>	<b>595</b>
17.1	Application of an Engineering Change	595
17.2	ECOs and Equivalency Verification	599
17.3	Use of Post-Silicon Cells for ECOs	600
17.4	ECOs and Design Version Management	602
17.5	Summary	605
	References	606
	Further Research	606
<b>Chapter 18</b>	<b>Physical Design Verification</b>	<b>607</b>
18.1	Design Rule Checking (DRC)	607
18.2	Layout-Versus-Schematic (LVS) Verification	610

---

18.3	Electrical Rule Checking (ERC)	616
18.4	Lithography Process Checking (LPC)	618
18.5	DRC Waivers	620
18.6	Summary	622
	Further Research	622
<b>Chapter 19</b>	<b>Design for Testability Analysis</b>	<b>625</b>
19.1	Stuck-at Fault Models and Automated Test Pattern Generation (ATPG)	625
19.2	DFT Design Rule Checking	636
19.3	Memory Built-in Self-Test (MBIST)	638
19.4	Logic Built-in Self-Test (LBIST)	645
19.5	Delay Faults	659
19.6	Bridging Faults	664
19.7	Pattern Diagnostics	665
19.8	Summary	672
	References	673
	Further Research	674
<b>Chapter 20</b>	<b>Preparation for Tapeout</b>	<b>677</b>
20.1	Introduction to Tapeout Preparation	677
20.2	Foundry Interface Release Tapeout Options	678
20.3	Tapeout Checklist Review	684
20.4	Project Tapeout Planning	689
	Further Research	692
<b>Chapter 21</b>	<b>Post-Silicon Debug and Characterization (“Bring-up”) and Product Qualification</b>	<b>693</b>
21.1	Systematic Test Fails	693
21.2	“Shmoo” of Performance Dropout Versus Frequency	695
21.3	Product Qualification	698
21.4	Summary	702
	Reference	702
	Further Research	703
<b>Epilogue</b>		<b>705</b>
<b>Index</b>		<b>711</b>

---

## P R E F A C E

This book describes the steps associated with the design and verification of Very Large Scale Integration (VLSI) integrated circuits, collectively denoted as the *design methodology*. The focus of the text is to describe the key features and requirements of each step in the VLSI methodology. The execution of each step utilizes electronic design automation (EDA) software tools, which are invoked by a script that manages the design configuration, assembles the input data, allocates the IT job resources, and interprets the output results. The script is commonly referred to as the *flow* for the specific methodology step. This book covers both the underlying EDA tool algorithms applied and the characteristics of the related flow. Specific attention is given to the criteria used to assess the status of a design project as it progresses toward the release to fabrication.

The audience for the text is senior-level undergraduates and first-year graduate students studying microelectronics. Professional engineers will also likely find topics of interest to expand the breadth of their expertise. In many cases, the discussion of a specific step extends beyond the design engineering considerations to include the perspective of a project manager, a design automation engineer, a fabrication technology support engineer, and, to be sure, a member of the project methodology team.

It has been my experience that graduating engineers pursuing microelectronic hardware design would benefit from broad exposure to all facets of a VLSI design project and an understanding of the interdependencies between the various engineering teams. The goal of this book is to provide a comprehensive discussion of a VLSI design methodology at a level of technical detail appropriate for a two-semester, project-oriented course of study.

The book is targeted toward a discussion style of presentation rather than formal lectures. The text often highlights the trade-offs that are evaluated when selecting a specific approach for a design methodology step. An interactive discussion among the students provides an opportunity comparable to the engineering environment as part of a design team.

There are no chapter problems provided in this book. However, many universities participate in EDA vendor programs that provide access to individual software tools. This text would work extremely well in combination with such a program. After reviewing a step as a constituent of the overall design methodology, students would be able to exercise the corresponding EDA tool. Projects of larger scope could be incorporated to align with individual student interests—including flow scripting, pursuing power/performance/area evaluations, designing (cell-based) circuits, and developing methodology policies and the software utilities to verify those design standards. Projects would typically culminate in a final presentation to the class.

The text is divided into six major topics. Topic I, “Overview of VLSI Design Methodology,” is rather lengthy, intended to provide background on microelectronic hardware design. Students with prior exposure to these topics could quickly review this material. The subsequent topics include Topic II, “Modeling,” Topic III, “Design Validation,” Topic IV, “Design Implementation,” Topic V, “Electrical Analysis,” and Topic VI, “Preparation for Manufacturing Release and Bring-Up.” The chapters in each topic describe individual flow steps. There is admittedly some overlap in the chapter discussions. For example, the task of embedding an engineering change order (ECO) in a design database nearing release to fabrication is mentioned in multiple chapters and described in detail in Chapter 17, “ECOs.” This repetition reflects the importance of the ECO methodology for a design project. Another example is the pervasive impact of lithographic multipatterning in advanced fabrication process nodes. The decomposition of the design data for a mask layer into (individually resolvable) subsets needs methodology support throughout design implementation, analysis, and physical verification flows. The influence of multipatterning is therefore described in multiple chapters.

The references provided with each chapter are rather sparse and in no way reflect the exceptional research that has enabled the complexity of current VLSI designs. The references listed are often among the landmark papers in their specific disciplines. A search for the technical papers that have recently cited these references will enable the reader to develop a more comprehensive

bibliography. The website <http://www.vlsidesignmethodology.com> provides links to errata and additional technical publications of interest.

Several technical areas deserve greater depth than the length of this book allows. Readers are therefore encouraged to pursue the “Further Research” sections provided at the end of each chapter.

Many colleagues have provided great insights to assist with the development of this text. The collaboration over the years with Tom Lin, Mark Firstenberg, Tim Horel, and Bob Deuchars has been pivotal. The technical review recommendations from Professor Azadeh Davoodi at the University of Wisconsin–Madison have been extremely beneficial. The support from Bob Masleid, Tammy Silver, William Ruby, Charles Dancak, and Dan Nenni is greatly appreciated. Bob Lashley deserves special mention, as his expertise and inspiration have been invaluable. Finally, thanks to my family for their encouragement, especially my wife, Suzi.

**Tom Dillinger**  
Livermore, California

---

## ABOUT THE AUTHOR

**Thomas Dillinger** has more than 30 years of experience in the microelectronics industry, including semiconductor circuit design, fabrication process research, and EDA tool development. He has been responsible for the design methodology development for ASIC, SoC, and complex microprocessor chips for IBM, Sun Microsystems/Oracle, and AMD. He is the author of the book *VLSI Engineering* and has written for SemiWiki.

*This page intentionally left blank*

# Placement

## 8.1 Global Floorplanning of Hierarchical Units

Prior to detailed placement of cells in block netlists (and the global cells at the top of the SoC hierarchy), a physical floorplan of the chip design is required. As briefly described in Section 3.1, the floorplan typically represents the first level of the SoC model hierarchy; it is uncommon to further develop a “floorplan within a floorplan” for the physical design of subsequent levels of the SoC hierarchy. The glue logic functionality at the top hierarchical level is commonly allocated to *channels* between block floorplan boundaries. An alternative methodology would be to define abutting block floorplan regions and insert global glue logic within various blocks. The advantage of the reduced channel area is offset by the additional dependency of global cell and route data on block-level physical verification and electrical analysis.

The physical floorplan data include the global power and ground grids and global clock distribution, typically originating from a PLL hard IP macro that serves as the clock reference source. The power and ground grids in the channels require specific design consideration, as the glue logic circuits include

high-drive-strength cells with high switching activity (e.g., signal repowering buffers, state-repeating register banks).

The floorplan may include allocated routing track segments for major signal busses between blocks in the overall SoC architecture, including global repeaters. These preroutes assist with the definition of the block-level floorplan pins, area, and aspect ratio.

The development of floorplan pins is a critical facet of SoC design planning. The pin definition for each block's primary inputs and primary outputs includes the following:

- The pin width, corresponding to the interconnect wire width to use with the global signal
- The pin metal layer for the interface between global and block routing
- The pin multipatterning decomposition assignment

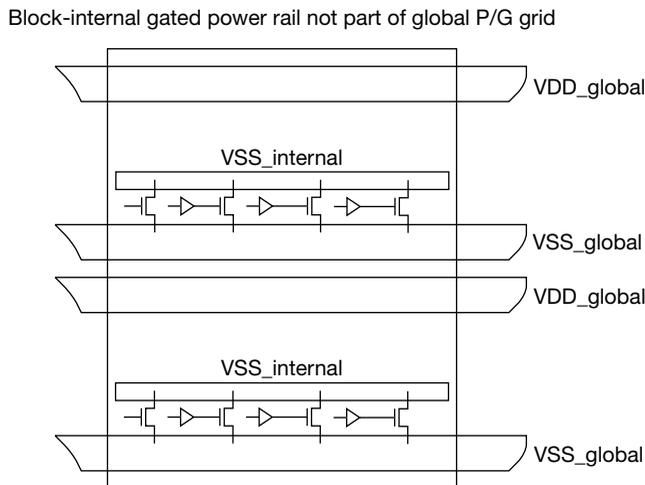
For advanced process nodes, depending on the metal layer, the pin definition may also need to include a multipatterning assignment that is consistent with the “color” associated with the pin's routing track. Alternative methods for pin location assignment include the following:

- **Internal pin locations**—The pin may not be assigned to the block perimeter; rather, it might be given internal coordinates. The goal of using internal pin locations would be to improve timing. As mentioned in Section 7.2, the accuracy of block-level timing closure is improved if the cells connected to block PIs/POs are placed in close proximity to the pin. An internal pin location may allow optimal placement of block netlist cells with connectivity to both global and internal signals. A high density of internal pins may have an adverse impact on global routing, however, to accommodate both over-the-block global routes and pin accessibility.
- **A flexible range of locations**—A pin may be allocated to a range of locations (e.g., a segment of a specific floorplan edge) but not assigned fixed coordinates. In this case, the block placement flow includes pin location assignment as part of cell assignment; rather than using fixed pin locations to influence cell placement, the algorithm is able to include pin placement as an optimization objective. The methodology decision to use flexible pin locations as input to the block cell placement flow introduces an interdependency between global route planning and block physical implementation.

Floorplan areas allocated to hierarchical design blocks are typically rectangular, although EDA vendor tools for physical design may support rectangular definitions. The aspect ratio of each floorplan block is a key factor in subsequent routing and path timing closure. A high aspect ratio block has a skewed ratio of available horizontal to vertical wiring tracks, and thus it may have difficulty subsequently closing on routing.

The SoC floorplan includes blocks associated with the chip input/output pad circuits, usually located on the die perimeter. Mixed-signal IP cores are also typically associated with unique floorplan blocks, such as PLLs, data conversion functions (ADCs, DACs), and high-speed interface SerDes IP. These blocks also require unique power/ground distribution design. The I/O circuits are likely to use additional voltages different from internal cells (e.g., VDDIO, VDD\_1\_2, VDD\_1\_5). Mixed-signal cores require separate low-noise supply rails (e.g., VDDA, GNDA) that are electrically distinct from the rails for digital switching networks.

Power-gating design is reflected within each block, as represented by the power format file description (described in Section 7.6). The internal power and ground distribution to enable deep sleep behavior is not extended globally, as depicted in Figure 8.1.



**Figure 8.1** The block internal power (or ground) distribution to support power gating is not extended globally.

## 8.2 Parasitic Interconnect Estimation

The placement flow utilizes a number of measurement criteria when selecting a candidate location for each cell or for candidate pairs of cells to swap their current coordinates. The process involves a combination of geometric and timing-driven calculations, including the following:

- Total estimated network wire length to realize all connections, using one of various net topology estimates (e.g., bounding box, star, Steiner tree) (Timing estimates from physical synthesis provide [negative slack] nets that may be given additional weighting in the total geometric wire length summation calculation.)
- Interconnect segments crossing a coarse grid overlay of the block floorplan to assess wiring track demand versus availability (to avoid congestion)
- Cell interconnect delay calculation for timing-driven placement optimization

The representation of interconnect delays during placement involves estimates of the  $R \cdot C$  parasitics and a simplified computationally fast delay calculation algorithm (e.g., an *Elmore delay* model for the estimated net topology; see Section 11.1). The SoC methodology team needs to collaborate with the EDA vendor and the foundry to determine how to best estimate the interconnect parasitic delay during cell placement. This estimate needs to reflect the different (per unit length)  $R$  and  $C$  measures of the multiple horizontal and vertical metal routing layers available within the block. During cell placement, an average  $R \cdot C$  delay measure across the available metal routing layers is used. An estimate for parasitic via resistances could also be included in the interconnect delay model.

In addition, the methodology team may use the physical synthesis timing data to derive “non-default” constraints for subsequent cell placement and routing:

- Preferred metal layers for routing critical nets
- Wider width segments (e.g., 1.5X or 2X width rather than 1X)

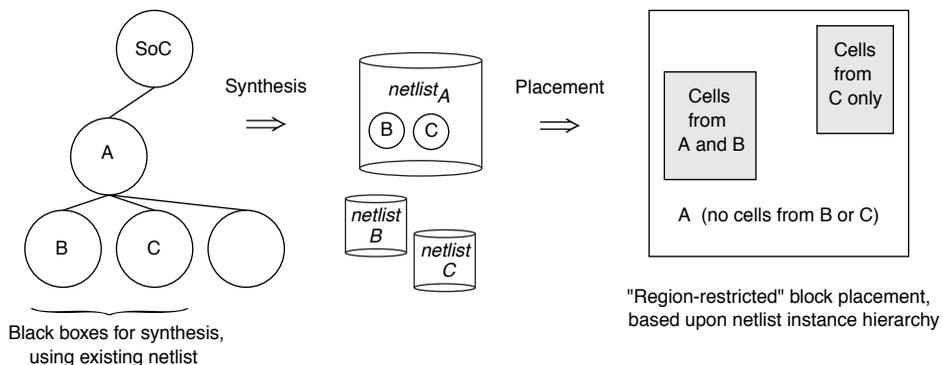
The EDA placement tool applies a different set of parasitic interconnect estimates for nets with non-default rules. Again, collaboration with the

EDA vendor and foundry is required to define how multiple wire load models for different classes of nets should be calculated for timing-driven placement optimizations.

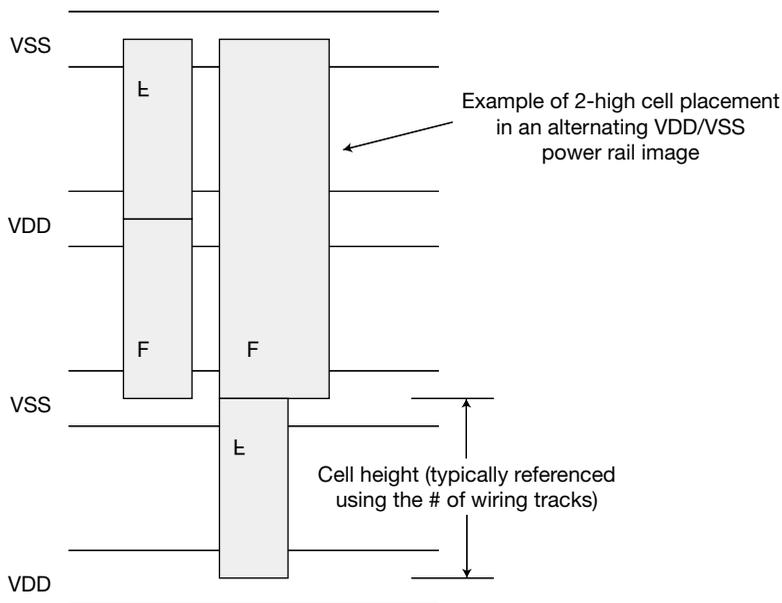
### 8.3 Cell Placement

The SoC block designer relies on the (timing-driven) cell placement flow to provide a routable solution with minimal timing issues for a netlist with (tens of) millions of instances. Placement algorithms have evolved to provide greater netlist capacity with reasonable runtime. To help physical designers achieve improved predictability and confidence in timing closure, the EDA vendor placement tools have incorporated additional features that apply input constraints:

- Preplaced cells and hard IP macros
- Relative placement groups of cells (a set of cells with relative alignment coordinates that are placed/moved as a unit)
- Restrictive area allocation within the floorplan block for subsets of the cell netlist (see Figure 8.2)
- Guidelines for maximum local cell utilization percentage (to allow for the addition of a suitable density of decoupling capacitance cells, substrate and well contact cells, and dummy logic cells for ECOs)
- Ability to place cells with cell height that spans two rows of the placement image (see Figure 8.3)



**Figure 8.2** The block placement flow may be provided with restricted areas for placement of subsets of the block netlist cells. This subset would typically be identified by a specific string in the (flattened netlist hierarchy) instance name.



**Figure 8.3** The cell library may contain physical cells spanning two rows in height.

For current fabrication process nodes, additional cell adjacency restrictions must be observed during placement. Lithographic uniformity of (critical dimension) device gates may require the insertion of dummy gates between cells and at the ends of cell rows. The transition between cells of different  $V_t$  types may also require dummy gate cells to reduce the device variation from  $V_t$  mask overlay and implant dosage. Depending on the design of the cell image, the placement algorithm may also need to insert device well continuity *filler cells* in vacant locations. The methodology team needs to review the cell library techfile data and fabrication process design rules to ensure that any specific placement restrictions and/or dummy cell insertion guidelines are coded for the EDA placement tool.

Throughout the evolution of EDA placement tools, the goal has consistently been to provide a result that is ultimately routable and achieves timing targets, with runtimes that scale with the increasing block netlist instance size. Prior to the introduction of physical synthesis, placement tools consisted of *constructive cell/macro location assignment* followed by *iterative optimization* (or “successive refinement”) steps. The physical synthesis methodology has

resulted in a shift in EDA placement tool development emphasis to improving the iterative solutions. Numerous algorithms have been developed to select candidate cells to reposition and evaluate new proposed locations and/or to successively resolve placement overlaps from an existing assignment, with optimization objectives that address routing congestion and estimated path timing improvements.<sup>[1,2]</sup>

## 8.4 Clock Tree Local Buffer Placement

A key aspect of the placement flow is the special consideration to be given to the clock buffers in the netlist, typically added by the CTS step in the synthesis flow (see Section 7.9). The CTS algorithm attempts to balance the (estimated) loading on the branches of the clock tree in the network, whether originating from a single clock pin or connecting to a global clock grid. During cell placement, the common algorithmic approach is to select clusters of flops in close proximity and place a clock buffer in the final branch of the tree within the area spanned by the flop cluster. Once all clock tree endpoints are placed, a similar approach selects clusters of clock buffers and places a buffer from the preceding level of the tree appropriately; this process iterates recursively to the root level of the clock tree. The clock buffer placement algorithm results in output netlist updates, as the (logically equivalent) sinks at each level of the tree may be swapped during the clustering phase of the placement algorithm. The introduction of clock gating to the CTS tree implies that the cells at each level of the tree are not necessarily logically equivalent; clustering of placed sinks needs to observe gated `clk_enable` functionality.

For block placement with preplaced hard IP macros, the related clock buffers may also be preplaced accordingly. For relative placement groups, clock buffers may be included in the group definition. An increasing design trend is to offer multi-bit registers as an atomic cell library offering to minimize the clock routing and loading among bits. These registers are also likely to be part of relative placement groups with clock buffers (and decoupling capacitance cells).

During block routing, the attention to clock signals focuses on balancing the arrival latency at endpoints, primarily through  $R \cdot C$  interconnect segment allocation. Performance optimization features in the routing flow may result in changes to the drive strength of logic path cells and flops; clock buffer tree

cells may likewise need to receive drive strength updates in routing. For drive strength increases, any resulting cell area overlaps to the placement output locations need to be (incrementally) resolved during routing.

## 8.5 Summary

The incorporation of constructive placement algorithms in logic synthesis flows has resulted in a shift in focus for EDA vendors providing placement tools. Iterative optimization and legalization of the initial physical location cell assignment from synthesis requires judicious selection of candidate cells for re-positioning, with fast and accurate evaluation of interconnect parasitic estimates. This focus on estimation efficiency is required to support an increasing number of cell instances in a design block. In addition, tools are applying a richer set of designer input constraints to direct the resulting cell placement to a solution optimized for routability, path timing closure, and power dissipation reduction. Increasingly, physical implementation design resources for an SoC project are being re-directed from executing cell placement to addressing the complexities of interconnect routing optimizations for electrical and reliability analysis flows, such as timing, power, noise, and electromigration. Nevertheless, the quality of results for the cell placement flow is crucial to achieving subsequent design closure in routing.

## References

- [1] Breuer, M., et al., “Fundamental CAD Algorithms,” *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, Volume 19, Issue 12, December 2000, pp. 1449–1475.
- [2] Pan, D., Halpin, B., and Ren, H., “Timing-Driven Placement,” Chapter 21 in *Handbook of Algorithms for Physical Design Automation*, edited by C. J. Alpert, D. Mehta, and S. Sapatnekar, Auerbach Publications, 2008.

## Further Research

### Estimated Wire Length

Placement algorithms are dependent on wire length estimation calculations. Constructive placement methods often use total estimated wire length for all nets as a measurement criterion. Subsequent iterative optimization algorithms may add “weighting factors” to timing-critical and high-switching-activity nets as part of the wire length minimization objective.

Describe the various net topology alternatives commonly used for wire length estimation (e.g., Steiner tree, star, bounding box). Describe the advantages and disadvantages of the different topologies in terms of computation time and accuracy trade-offs.

### **Constructive Placement and Physical Synthesis**

The physical synthesis flow provides an initial placed netlist, and serial/parallel repowering cells are added during synthesis. Placement tools incorporate both constructive and iterative optimization steps and signal repowering features. As a result, a flow option could be provided to disregard the placement assignments from physical synthesis altogether and apply the constructive placement step on the complete block netlist.

Describe the trade-offs in exercising a constructive placement step on the full block netlist. Describe the sample experiments and quality-of-results criteria that could assist with this trade-off decision.

*This page intentionally left blank*

## A

- abgen, 136
- absolute failure, 542
- abstract generation (abgen), 136
- ABV (assertion-based validation), 264–265
- activation energy, 537
- activity factor, 246, 296–297
- admittance, 444, 516–519
- AEC (Automotive Electronics Council), 699
- aggressive P/G grid design, 510
- aggressor net, 477. *See also* noise analysis
- air gap, 129
- aliases, 641
- alpha particles, 577–578
- analog IP tests, 180, 563–564
- analog layout design, 130
- analysis flows, library cells models for
  - cell delay models, 241–245, 255
  - cell input pin noise sensitivity, 247–249
  - cell models for synthesis and testability analysis, 241
  - clock buffers and sequential cells, 249–251
  - power analysis characterization, 245–247
- analysis mixed-signal coverage, 294–295
- annealing of cell placements, 145
- antennas, 148–149
  - diodes, 396–399
  - rules, 191
- application programming interfaces (APIs), 2, 415
- application-specific integrated circuits (ASICs)
  - ASIC direct release, 510–512
  - overview of, 15–17
- area pins, 138–139
- array generation, 31
- array repair (fuses), 181, 314–315
- array weak bit, 243–244
- arrays
  - array compilers, 197
  - array equivalence verification, 310–313, 314
  - array generation, 31
  - array repair (fuses), 181, 314–315
  - methodology flows, 197–198
  - process variation-aware design, 91–92
  - register files, 198
- arrival time (AT), 147, 455
- ASIC. *See* application-specific integrated circuits (ASICs)
- assertion statements, 132, 238–239
- assertion-based validation (ABV), 239, 264–265
  - assert statement, 238
- ATE (automated test equipment), 629
- ATPG (automated test pattern generation), 163, 208–211, 356, 627, 631–636
- audits, tapeout, 189–190
- automated test equipment (ATE), 629
- automated test pattern generation (ATPG), 163, 208–211, 356, 627, 631–636
- Automotive Electronics Council (AEC), 699
- autonomous linear feedback shift register (LFSR), 648
- avalanche breakdown (ESD protect device), 62–63
- avalanche current, 573

**B**

- back-end-of-line (BEOL), 46–50, 679–680
- backfill cells
  - abstracts, 38–39
  - gate array cells, 35–36
  - model constraints, 39–40
  - model properties, 39–40
  - overview of, 33
  - process design kit (PDK), 40–42
  - standard cell logic, 36–37
  - views, 38–39
- backward inference, 626, 674
- bad machine error injection testcases, 267
- bad machine path coverage, 294
- balanced repowering trees, 335–336
- “balanced” routing, 388
- bandgap voltage reference generators, 102
- bathtub curve, 532
- BC (best-case) process, 87, 89–90, 243
- BDDs (binary decision diagrams), 315
- behavioral co-simulation, 275
- behavioral modeling, 319
  - data structures, 228
  - definition of, 224
  - inferred state, 229–230
  - modules/entities, 230–231
  - regs, 227
  - representation of simulation time, 225–226
  - resolution function, 227
  - scope, 231–232
  - signals, 227
  - SystemC, 254
  - types, 228–229
  - variables, 227
  - wires, 227
- BEM (boundary element method), 406, 442
- BEOL (back-end-of-line), 46–50, 679–680
- best-case (BC) process, 87, 89–90, 243
- biased outputs, 648
- BiCMOS, 59
- bidirectional DC current density ( $j_{AC}$ ), 539–540
- binary decision diagrams (BDDs), 315
- binning, 607
- BISD. *See* built-in self diagnosis (BISD)
- BIST. *See* built-in self-test (BIST)
- black box IP, 152
- Black’s equation, 537–538
- Blech length, 538, 557
- blind build part, 630
- block netlists, 371
  - block size, 202
  - logic synthesis, 203–204
  - low-effort synthesis (LES), 204–205
  - physical design, 205–206
- placement
  - cell placement, 367–369
  - clock tree local buffer placement, 369–370
  - estimated wire length, 370–371
  - global floorplanning of hierarchical units, 363–365
  - parasitic interconnect estimation, 366–367
- blockage map, 22
- blockages, 137
- block-based design, 202
  - block-level noise analysis, 493–494
  - block-level timing constraint budgeting, 473–474
  - electrical abstracts, 214–216
  - logic synthesis, 203–204
  - low-effort synthesis (LES), 204–205
  - physical design, 205–206
- blocking, 227
- body (model builds), 231
- body bias, 63
- body effect, 63–66, 79–80, 329–330
- borophosphosilicate glass (BPSG), 579
- boundaries, 137
- boundary element method (BEM), 406, 442
- boundary scan, 180, 209, 635
- bounding boxes, 325
- Box, George, 221
- BPSG (borophosphosilicate glass), 579
- branch configuration (DDM), 173–174
- breakpoints, 231, 261
- bridging faults, 164, 388, 664–665
- budgeting, timing constraint, 473–474
- buffering, 331, 336–339
- bug trackers, 187, 192
- built-in self diagnosis (BISD), 641–645. *See also*
  - built-in self-test (BIST)
  - built-in self repair (BISR), 642–645
- built-in self-test (BIST), 112, 134–136
  - logic built-in self-test (LBIST)
    - compaction, 649–650
    - hierarchical testing, 658–659
    - overview of, 645
    - pattern generation, 645–648
    - pseudo-random pattern generation (PRPG), 650–652
    - test compression, 652–658
    - X-values in fault simulation, 650
  - memory built-in self-test (MBIST), 163–164, 638–645
- bulk CMOS devices. *See also* soft error rate (SER)
  - body effect, 63–66
  - contacted gate pitch, 68

- latchup, 69–70
  - low-resistivity gate material, 66–67
  - overview of, 59–63
  - raised source/drain nodes, 67
  - spacer dielectric, 67–71
  - burn-in testing, 104, 208–211, 625–626, 703
- C**
- C effective model reduction for delay calculation, 374–375
  - capacitance
    - coupling, 435–436
    - decoupling, 431–433
    - effective capacitance ( $C_{\text{eff}}$ ), 116, 326–328, 374–375, 443–446
    - sidewall coupling capacitance, 374–375
    - total ( $C_{\text{total}}$ ), 326–328
  - CCRs (channel-connected regions), 278
  - CDF (cumulative density function), 483, 531
  - CDM (charged-device model), 158, 568
  - $C_{\text{eff}}$  algorithm, 374–375
  - cell and IP modeling
    - end-of-life circuit parameter drift, 251–253
    - functional modeling
      - behavioral modeling, 224–232, 254
      - initialization, 235
      - netlists, 234–235
      - overview of, 223
      - register-transfer level (RTL) modeling, 232–234
      - test model views, 235–236
      - X- and don't-care modeling, 236–239
    - library cell models for analysis flows
      - cell delay models, 241–245, 255
      - cell input pin noise sensitivity, 247–249
      - cell models for synthesis and testability analysis, 241
      - clock buffers and sequential cells, 249–251
      - power analysis characterization, 245–247
      - physical models for library cells, 240
    - cell delay calculation, 443–446
    - cell input pin noise sensitivity, 247–249
    - cell instance names, 339–340
    - cell library models, 8, 17
      - cell abstract, 5, 38–39
      - models for analysis flows
        - cell delay models, 241–245, 255
        - cell input pin noise sensitivity, 247–249
        - cell models for synthesis and testability analysis, 241
        - clock buffers and sequential cells, 249–251
        - power analysis characterization, 245–247
        - physical models, 240
        - technology mapping, 203, 328–335
  - cell placement, 27, 94, 200–201, 240, 325, 367–369
  - cell power characterization, 430–431
  - cell swaps, 203
  - cell-level electromigration analysis, 549–551
  - cell-level parasitic modeling, 411
    - cell characterization, 419–422
    - cell extraction, 411–413
    - cell power characterization, 430–431
    - extraction corners, 417–419
    - input pin noise characterization, 426–430
    - layout-dependent effects (LDEs), 414–417
    - logically symmetric inputs, 424
    - multiple-input switching (MIS), 423–424
    - out-of-range delay calculation, 422
    - sequential circuit characterization, 424–425
    - voltage values, 422–423
  - CFG (control-flow graph), 293
  - channel-connected regions (CCRs), 278
  - channel-less design, 9
  - channels, 9, 363
    - definition of, 124
    - FET, 60–61
    - scribe, 124
  - characterization, 38–39, 221, 245–247, 419–422
    - cell power, 430–431
    - input pin noise, 426–430
    - IP characterization data, 494
    - n-Sigma, 241–245, 255
    - power analysis, 245–247
    - sequential circuit, 424–425
  - charged-device model (CDM), 158, 568
  - CHECKEDOUT tag, 173
  - checkerboard patterns, 675
  - checkin (DDM), 173
  - checklist, tapeout, 677
    - overview of, 684–685
    - post-tapeout engineering change orders (ECOs), 688–689
    - tapeout waivers, 685–688
  - checkout (DDM), 173
  - checkpoint/restart feature, 151
  - chemical-mechanical polishing (CMP), 51, 87, 129, 148, 155–156, 383
  - chicken bits, 694–695
  - chip bump packaging, 95
  - chip finishing, 148–149
  - chip package co-design, 97–101
  - chip thermal management, 101–104
  - chip-package co-design, 4
    - chip-package model analysis, 96–97

- early chip-package floorplanning, 95–96
  - overview of, 94
- chip-package-system (CPS) simulation, 97–98
- circuit under test (CUT), 645–646
- circuit-limited yield (CLY), 90, 666
- circuit-switching, power dissipation model for, 497–500, 507
- cladding, 435
- clamp circuit, 573
- clk\_enable-to-clk timing test, 454
- clock arrival analysis, 438
- clock buffers, 249–251
- clock distribution, 124–126, 175–177
- clock domain crossing, 5, 110–113, 206–207
- clock gating, 107–109
- clock tree local buffer placement, 369–370
- clock tree synthesis (CTS), 336, 350–353
- clock-to-output delay arc, 424
- cloning, 186
- clustering, 325
- CLY (circuit-limited yield), 90, 666
- CMOS devices
  - bulk CMOS devices. *See also* soft error rate (SER)
    - body effect, 63–66
    - contacted gate pitch, 68
    - cross-over current, 61
    - device punchthrough, 62–63
    - latchup, 69–70
    - low-resistivity gate material, 66–67
    - overview of, 59–63
    - raised source/drain nodes, 67
    - spacer dielectric, 67–71
    - subthreshold leakage currents, 62–63
    - transition time, 61
  - FinFET devices
    - advantages of, 78–80
    - characteristics of, 77–78
    - constraints, 80–83
    - design methodology, 83–85
    - electromigration, 82–83, 392–393
    - foundry interface release tapeout options, 681–682
    - overview of, 74–77
    - self-heating, 82–83, 253, 502–504
    - snapback mode, 574
    - thermal resistance, 82–83
  - fully depleted silicon-on-insulator (FD-SOI), 71–74
- CMOS transfer gate, 60–63
- CMP (chemical-mechanical polishing), 51, 87, 129, 148, 155–156, 383
- coefficient of thermal expansion (CTE), 700
- collection volume, 579–580
- combinational logic, 301–302, 305–306, 584
- common data model, 2
- compactors, 641
- compilers, array, 197
- complementary metal-oxide-semiconductor. *See* CMOS devices
- concurrent sequential processes (CSPs), 223, 319
- conditional coverage, 291–292
- conductance matrix, 516–519
- configuration management, 192
- configuration specification (configspeg), 172–173, 231
- congestion, routing, 33
- connectivity checks, 143
- conservative P/G grid design, 510
- constant failure rate, 533
- constraints
  - model, 39–40
  - timing, 361, 457
    - effective capacitance ( $C_{\text{eff}}$ ), 326–328
    - generation and verification of, 320–326
    - total capacitance ( $C_{\text{total}}$ ), 326–328
- constructive placement, 368–369, 371
- contact resistance, 442
- contacted gate pitch, 68
- context cells, 53, 84–85
- continuing process improvement (CPI), 41, 620–621
- control flow / dataflow
  - dataflow cell synthesis, 200
  - dataflow library cell design, 199–200
  - definition of, 198–199
  - general block modeling, 201–202
  - relative cell placement, 27, 200–201
- control-flow graph (CFG), 293
- core wrap architecture, 30
- correspondence points, 6–8, 193–196
- cosmic rays, 578
- coupling capacitance, 435–436
- coverage. *See* simulation model statement coverage
- covergroups, 290
- coverpoints, 290
- CPI (continuing process improvement), 41, 620–621
- CPS (chip-package-system) simulation, 97–98
- CRC (cyclic redundancy check), 683
- critical area, 92
- critical charge ( $Q_{\text{crit}}$ ) for a soft error upset (SEU), 580–581
- cross-coupled devices, 34

- cross-over current, 61, 497
  - CSPs (concurrent sequential processes), 223, 319
  - CTE (coefficient of thermal expansion), 700
  - CTS (clock tree synthesis), 336, 350–353
  - cumulative density function (CDF), 483, 531
  - current crowding, 442, 542, 575
  - CUT (circuit under test), 645–646
  - cut masks, 54–59, 378, 682–684
  - cutout overlay cells, 217–218
  - cycle simulation, 233
  - cyclic decomposition error, 56
  - cyclic layout, 54
  - cyclic redundancy, 376–377
  - cyclic redundancy check (CRC), 683
- D**
- DAGs (directed acyclic graphs), 302
  - dark silicon, 504
  - data repositories, 171
  - dataflow
    - dataflow cell synthesis, 200
    - dataflow library cell design, 199–200
    - definition of, 198–199
    - general block modeling, 201–202
    - relative cell placement, 27, 200–201
  - “days per mask layer” target metric, 601
  - DDM. *See* design data management (DDM)
  - de facto standards, 19
  - debugging
    - array equivalence verification, 310–313, 314
    - with array repair, 314–315
    - data structures for, 315
    - functional and test-mode equivalence verification, 309–310
    - layout-versus-schematic netlist checking (LVS), 616
    - model input assertions for, 306–307
    - sequential logic equivalency, 307–309
  - decompression logic, 653
  - decoupling capacitance, 431–433
  - deep n-well, 563
  - deep sleep operating mode, 87
  - defect density-limited yield, 92
  - delamination, 694, 700
  - delay
    - C effective model reduction for delay calculation, 374–375
    - cell, 443–446
    - cell delay models, 241–245, 255
    - delay adjust, 458–459
    - delay-based functional simulation, 468, 474
    - delay-based timing verification, 467–469
    - delta, 225
    - derating delay multipliers, 451
    - Elmore delay, 366, 375, 446–451
    - FO4, 329–330
    - interconnect, 446–451
    - noise impact on
      - I\*R voltage drop effect, 483–484
      - modeling, 481
      - static timing analysis, 482–483
    - out-of-range delay calculation, 422
    - state-dependent, 459
  - delay arcs, 424
  - delay faults, 164
    - definition of, 659
    - gross delay fault, 627–628, 661–662
    - hold latches, 660–661
    - internal cell delay faults, 662–664
    - launch-from-shift, 660
    - path delay faults, 662
  - delta delay, 225
  - Dennard, Robert, 44–45
  - Dennard scaling rules, 44–45, 129
  - dependency traceback, 262
  - depth-of-focus (DoF), 155–156
  - derate tables, 464–466
  - derating multipliers, 90, 243, 451, 462–466
  - design data management (DDM)
    - branch configuration, 173–174
    - checkout/checking, 173
    - configuration specification (“configspeg”), 172–173
    - data repositories, 171
    - latest configuration, 173–174
    - main configuration, 173–174
    - metadata, 171
    - overview of, 171–172
    - physical design data, 174–175
    - power and clock domain management, 175–177
    - virtual filesystem interfaces, 171
  - design for testability (DFT), 163. *See also* design rule checking (DRC)
    - automated test pattern generation (ATPG), 627, 631–636
    - backward inference, 626, 674
    - bridging faults, 664–665
    - controllability and observability, 135, 625
    - delay faults. *See* delay faults
    - embedded deterministic test (EDT), 653
    - embedded macro test, 134
    - forward propagation, 626, 674

- logic built-in self-test (LBIST)
  - compaction, 649–650
  - hierarchical testing, 658–659
  - overview of, 645
  - pattern generation, 645–648
  - pseudo-random pattern generation (PRPG), 650–652
  - test compression, 652–658
  - X-values in fault simulation, 650
- low-effort synthesis (LES), 356–357
- memory array fault models, 674–675
- memory built-in self-test (MBIST), 638–645
- overview of, 177–179
- parametric faults and tests, 629–631
- pattern diagnostics, 665–666
  - diagnostic network analysis, 666–667
  - diagnostic pattern-to-fault correlation, 668–670
  - failing dies, 672, 675
  - overview of, 665–666
  - test response compaction, 670–672
- scan chains, 179
- stuck-at faults, 626–629
- summary of, 183–184
- test escape, 625
- test pattern generation, 179–182
- test pattern ordering, 179–182
- voltage and temperature test options, 183
- design freeze milestones, 595–597
- design implementation, 328–335
  - design optimization, 185–186
  - design platforms, 162
  - design robustness, 529
  - logic synthesis. *See* logic synthesis
  - multipatterning, 377
  - placement. *See* placement
  - routing, 373–374. *See* routing
- design methodology new releases
  - reference flows, 168–169
  - tool evaluation, 169–170
  - trailblazing, 170–171
- design rule checking (DRC), 152, 356–357, 607–610, 636–638
  - DRC waivers, 156–157, 620–621
  - routing and, 375–376
  - voltage-dependent design rule checks, 159–160
- design rule manual (DRM), 50–59
  - chemical-mechanical polishing (CMP), 51
  - context cells, 53
  - cut masks, 54–59
  - dummy shapes, 52–53
  - layer data, 54–59
  - line segment orientations, 53–54
  - metal spacing, 52
  - multipatterning, 54–59
  - pitch, 51
- design validation. *See* formal equivalency verification; functional validation; placement
- design version management, 602–604
- design-for-manufacturability (DFM), 153–156, 184–185, 398–399
  - chemical-mechanical polishing (CMP) analysis, 155–156
  - EDA DFM services, 156
  - lithography process checking (LPC), 154–155
- design-for-power feedback, 504–505
- design-for-test (DFT) insertion, 134. *See also* design for testability (DFT)
- design-for-yield (DFY)
  - overview of, 153, 184–185, 398–399, 687
  - routing optimization, 395–399
    - antenna diode insertion, 396–399
    - line extensions past vias, 395–396
    - redundant vias, 396
    - wire spreading, 395
- detailed routing assignment, 378–383
- detectable faults, 631
- deterministic sequences, 653. *See also* embedded deterministic test (EDT)
- device punchthrough, 62–63
- device self-heat models, 502–504
- device under test (DUT), 263, 571
- device width, representation of, 612–613
- device-level placement, routing and, 400–401
- DFM. *See* design-for-manufacturability (DFM)
- DFT. *See* design for testability (DFT)
- DFY. *See* design-for-yield (DFY)
- diagnostics, pattern
  - diagnostic network analysis, 666–667
  - diagnostic pattern-to-fault correlation, 668–670
  - failing dies, 672, 675
  - overview of, 665–666
  - test response compaction, 670–672
- dictionary, fault, 668
- dies
  - die thermal maps, 502, 507
  - failing, 672, 675
- difference checking, 214
- diffusion current, 578
- direct aggressor coupling, 477, 493
- direct release (ASIC), 510–512
- directed acyclic graphs (DAGs), 302
- directed testcases, 263

direction, noise, 485  
 dishing, 148  
 dissipation, power, 245–247, 497–500, 507  
 DoF (depth-of-focus), 155–156  
 dogbone topology, 153  
 domains, 3–4  
   clock, 110–113  
   power domain constraints, 105–107  
 dominant signals, 664  
 don't-care modeling, 236–239  
 DRAM (dynamic RAM), 579  
 DRC. *See* design rule checking (DRC)  
 drift current, 578  
 drive strength, 22–24  
 driving point admittance, 444  
 DRM. *See* design rule manual (DRM)  
 dummy gates, 84  
 dummy shapes, 40, 52–53  
 dummy transistors, 34  
 DUT (device under test), 263, 571  
 DVFS (dynamic voltage, frequency scaling), 341  
 dynamic P/G voltage drop analysis  
   analysis results, 521–522  
   global and partition-level models for, 519–521  
   global power delivery frequency response, 523  
   matrix solution, 516–519  
   P/G rail capacitance, 514–515  
   simultaneous switching output (SSO) analysis,  
     523–526  
 dynamic power grid voltage analysis, 431–433  
 dynamic RAM (DRAM), 579  
 dynamic voltage frequency scaling (DVFS), 101, 341  
   boost/throttle modes, 422–423  
   noise analysis and, 491–492  
   shmoo plots for, 697

## E

early mode, 321–322  
 ECC (error correcting code), 587–588  
 ECOs. *See* engineering change orders (ECOs)  
 EDA. *See* electronic design automation (EDA)  
 EDC (error detection and correction), 70–71  
 EDT (embedded deterministic test), 653  
 effective capacitance ( $C_{\text{eff}}$ ), 116, 326–328, 374–375,  
 443–446  
 effective slew, 430  
 e-fuse array repair, 182  
 electric fields, 535  
 electrical abstracts, 214–216  
 electrical analysis. *See also* layout parasitic extrac-  
 tion (LPE); soft error rate (SER)  
   chip electrical analysis, 97–101

  electromigration (EM) reliability analysis.  
     *See* electromigration (EM)  
   electrostatic discharge (ESD) checking  
     electrical overstress (EOS), 576  
     ESD protection circuits, 572–576  
     overview of, 568–571  
   noise analysis. *See* noise analysis  
   power analysis. *See* power analysis  
   power rail voltage drop analysis. *See* power rail  
     voltage drop analysis  
   routing optimization, 389–393  
     electromigration, 392–393  
     noise optimization, 391  
     power optimization, 391–392  
     timing optimization, 390  
   for signoff flow, 160–163  
 SleepFET power rail analysis  
   I\*R voltage drop effect, 559–560  
   sleep-to-active state transition, 561–562  
 substrate noise injection  
   analog IP tests, 563–564  
   I/O pad circuit design, 564–565  
   latchup analysis, 562–563  
   latchup qualification, 565–567  
   overview of, 562–563  
 timing analysis  
   cell delay calculation, 443–446  
   cell delay models, 241–245  
   electrical design checks, 452–453  
   interconnect delay calculation, 446–451  
   static timing analysis (STA), 453–469  
 electrical analysis flow status, 188–189  
 electrical overstress (EOS), 158, 576  
 electrical rule checking (ERC), 157–160, 616–618  
   electrostatic discharge (ESD) protection,  
     158–159  
   voltage-dependent design rule checks, 159–160  
 electromigration (EM), 82–83, 437  
   bidirectional DC current density ( $j_{\text{AC}}$ ), 539–540  
   Black's equation, 537–538  
   Blech length, 538, 557  
   cell-level, 549–551  
   for clocks, 551  
   design robustness and reliability, 529  
   detailed routing and, 392–393  
   electric fields, 535  
   electron momentum transfer, 535  
   EM failure, 542–544  
   extraction model, 544–545  
   failure in time (FIT) rate, 533–534  
   flowcharts for, 556  
   healing factor, 538

- introduction to, 529
- mean time to failure (MTTF), 529
- median time to failure (MTTF), 530, 532
- metal grains, 536
- metal hillocks, 536
- metallurgy and, 557
- nucleation, 535
- peak current density (jPEAK), 541–542
- power rail electromigration analysis (powerEM), 545–548
- resistive fails, 557
- signal interconnect electromigration analysis (sigEM), 548–555
- sum of failure rates, 534–535
- thermal gradients, 535
- thermomigration, 535
- unidirectional DC current density ( $j_{DC}$ ), 539–540
- voids, 535
- wearout region, 530, 533
- wire temperature increase due to resistive Joule heating ( $J_{RMS}$ ), 540–541
- electron momentum transfer, 535
- Electronic Design Automation (EDA)
  - EDA platform, 2
  - EDA reference flow, 2–5
  - EDA tool evaluation, 3
  - methodology managers, 165–168
  - release flow management, 165–168
- electronic design automation (EDA), 2. *See also* simulation
  - flow scripts, 2
  - platforms, 2
  - vendor placement tools, 367–369
- electrostatic discharge (ESD) checking, 158–159, 568–571, 700
  - electrical overstress (EOS), 576
  - ESD clamp circuit, 572–576
- Elmore delay, 366, 375, 446–451
- EM. *See* electromigration (EM)
- embedded deterministic test (EDT), 653
- embedded macro test, 134, 460
- emerging technologies, 705–710
- emulation (simulation acceleration), 268–272, 274
- end point detect, 662
- end-of-life circuit parameter drift, 251–253
- engineering change orders (ECOs)
  - design version management, 602–604
  - equivalency verification, 599–600
  - overview of, 595–597
  - physical design updates, 597
    - HDL model hierarchy, 598
    - metal fill, 597–598, 606
    - post-silicon cells for, 600–602
    - post-tapeout, 688–689
    - project tapeout planning, 603–604
  - SoC ECO mode
    - chip finishing, 148–149
    - functional model updates, 143–144
    - netlist updates, 145–148
- engineering teams, 211–212
- EOF (end-of-life) circuit parameter drift, 251–253
- EOS (electrical overstress), 158, 576
- equivalency verification
  - array equivalence verification, 310–313, 314
  - with array repair, 314–315
  - combinational logic cone analysis, 305–306
  - data structures for, 315
  - engineering change orders (ECOs) and, 599–600
  - functional and test-mode equivalence verification, 309–310
  - inverse equivalent, 304
  - model input assertions for, 306–307
  - RTL combination model equivalency, 301–302
  - sequential logic equivalency, 307–309
  - state mapping for, 302–304
  - summary of, 313–314
- equivalent faults, 626
- equivalent NAND logic gates, 13–14
- ERC (electrical rule checking), 157–160, 616–618
  - electrostatic discharge (ESD) protection, 158–159
  - voltage-dependent design rule checks, 159–160
- error correcting code (ECC), 587–588
- error detection and correction (EDC), 70–71
- error injection testcases, 267
- errors. *See also* soft error rate (SER)
  - capture/recovery, 179
  - DRC (design rule checking), 156–157
  - layout-versus-schematic netlist checking (LVS), 616
  - multi-bit, 584–589
    - circuit optimizations, 585–587
    - process optimizations, 584–589
    - system design optimizations, 587–589
  - soft, 69–70, 163
- ESD checking. *See* electrostatic discharge (ESD) checking
  - ESD clamp circuit, 572–576
  - ETM (extracted timing model), 215
- Euler method, 516–517
- EUV (extreme ultraviolet), 54
- event monitors, 132
- event queue, 259–261
- event trace logging, 288–289

- event-driven simulation, 259–261
    - breakpoints, 261
    - event queue, 259–261
  - executable specification, 264
  - expression coverage, 291–292
  - extracted timing model (ETM), 215
  - extraction. *See* layout parasitic extraction (LPE)
  - extraction model (electromigration), 544–545
  - extreme ultraviolet (EUV), 54
- F**
- fabrication technology. *See also* FinFET devices
    - back-end-of-line (BEOL) process, 46–50
    - bulk CMOS devices. *See* CMOS devices, bulk CMOS devices
    - chip thermal management, 101–104
    - chip-package co-design
      - chip electrical analysis, 97–101
      - chip-package model analysis, 96–97
      - early chip-package floorplanning, 95–96
      - overview of, 94
    - clock domains, 110–113
    - clock gating, 107–109
    - design rule manual (DRM), 50–59
      - chemical-mechanical polishing (CMP), 51
      - context cells, 53
      - cut masks, 54–59
      - dummy shapes, 52–53
      - layer data, 54–59
      - line segment orientations, 53–54
      - metal spacing, 52
      - multipatterning, 54–59
      - pitch, 51
    - front-end-of-line (FEOL), 46–50, 679–680
    - fully depleted silicon-on-insulator (FD-SOI), 71–74
    - half nodes, 46
    - operating modes, 86–89
    - power domain constraints, 105–107
    - process, voltage, and temperature (PVT) corners, 86–89
    - process migration
      - overview of, 92–93
      - parameterized layout cells (pCells), 93
      - relative cell placement, 94
    - process retargeting
      - overview of, 92–93
      - parameterized layout cells (pCells), 93
      - relative cell placement, 94
    - process variants, 47–48
    - process variation-aware design
      - array design, 91–92
      - best-case (BC) process, 89–90
      - high-sigma Monte Carlo, 92–93
      - worst-case (WC) process, 89–90
    - scaling factor, 44–45
    - second sourcing, 47
    - self-aligned, quad-patterning (SAQP), 59
    - self-aligned double patterning (SADP), 57–58
    - shrink nodes, 46
    - temperature sensing, 102–104
    - VLSI process nodes, 44–45
  - failing dies, 672, 675
  - failure, electromigration, 542–544
  - failure in time (FIT) rate, 533–534
  - failure rate
    - definition of, 531
    - EM failure, 542–544
    - failure in time (FIT) rate, 533–534
    - mean time to failure (MTTF), 529
    - median time to failure (MTTF), 530, 532
    - sum of failure rates, 534–535
  - false paths, 323, 458–459, 472
  - Fast Signal Database (FSDb), 19
  - fatal assertions, 132
  - fault diagnosis, 38
    - bridging faults, 164, 388, 664–665
    - definition of, 634
    - delay adjust, 164
    - delay faults
      - definition of, 659
      - gross delay fault, 627–628, 661–662
      - hold latches, 660–661
      - internal cell delay faults, 662–664
      - launch-from-shift, 660
      - path delay faults, 662
    - detectable faults, 631
    - equivalent faults, 626
    - fault dictionary, 668–670
    - injected faults, 626
    - node transition faults, 662
    - parametric faults and tests, 629–631
    - simulation, 627, 674
    - stuck-at faults, 626–629
    - undetectable faults, 631
  - FD-SOI, 502–504
  - feature creep, 694–695
  - feedback tap, 647
  - FEOL (front-end-of-line), 46–50, 679–680
  - FET channel, 60–61
  - FIB (focused ion beam), 672
  - field-programmable gate arrays (FPGAs), 272–273

- files
  - power intent specification, 104
  - register, 198
  - techfiles, 40
  - trace, 233, 298–299
- filler cells, 368
- FinFET devices
  - advantages of, 78–80
  - characteristics of, 77–78
  - constraints, 80–83
  - design methodology, 83–85
  - electromigration, 82–83, 392–393
  - foundry interface release tapeout options, 681–682
  - overview of, 74–77
  - self-heating, 82–83, 253, 502–504
  - snapback mode, 574
  - thermal resistance, 82–83
- finite state machines (FSMs), 198–199, 294
- firm IP, 32, 208
- first-pass design, 35
- fishbone topology, 350
- FIT (failure in time), 533–534
- fixed P/G grids, 510–511
- flightlines, 95
- flip-flop cells, 424
- flip/mirror placement, 137–138
- floating random walk (FRW), 406, 442
- floorplanning, 141
  - concept of, 363–365
  - early chip-package floorplanning, 95–96
  - floorplan channels, 363
  - optimization decisions, 113–114
  - pin assignment, 114–119
- flop test, 429
- flop/register values, 179
- flow scripts, 2
- FO4 delay, 329–330
- focused ion beam (FIB), 672
- forbidden pitches, 191, 394–395
- formal equivalency verification
  - array equivalency verification, 310–313, 314
    - with array repair, 314–315
  - combinational logic cone analysis, 305
  - data structures for, 315
  - functional and test-mode equivalency verification, 309–310
  - inverse equivalent, 304
  - model input assertions for, 306–307
  - RTL combination model equivalency, 301–302
  - sequential logic equivalency, 307–309
  - state mapping for, 302–304
  - summary of, 313–314
- forward propagation, 626, 674
- FPGAs (field-programmable gate arrays), 272–273
- fracturing, 433–434
- frequency, 296–297
- front-end-of-line (FEOL), 46–50, 679–680
- frozen design database
  - overview of, 42
  - signoff flow. *See* signoff flow
  - text analysis for, 163–165
- FRW (floating random walk), 406, 442
- FSDB (Fast Signal Database), 19
- FSMs (finite state machines), 198–199, 294
- full-chip model size reduction, 160–161
- full-sleep power state, 106
- fully depleted silicon-on-insulator (FD-SOI), 71–74
- functional equivalence verification, 309–310
- functional exclusions, noise analysis, 485–486
- functional modeling, 143–144
  - behavioral modeling. *See* behavioral modeling
  - initialization, 235
  - netlists, 145–148, 234–235
  - overview of, 223
  - register-transfer level (RTL) modeling, 232–234
  - test model views, 235–236
  - X- and don't-care modeling, 236–239
- functional simulation
  - delay-based, 474
  - with low-effort synthesis (LES) model, 358–359
- functional validation
  - behavioral co-simulation, 275
  - event trace logging, 288–289
  - hardware-accelerated simulation, 267–268
    - emulation, 268–272, 274
    - prototyping, 272–274
  - production test patterns, 284–288
  - resource planning, 281–284
  - simulation model statement coverage. *See* simulation model statement coverage
  - simulation throughput, 281–284
  - software simulation
    - event-driven, 259–261
    - HDL source/configuration cross-reference, 262
    - interactive mode, 261
    - waveform display, 261
  - status, 187–188
  - switching activity factor estimates, 296–297
  - switch-level simulation, 276
  - symbolic simulation, 277–281
  - testbench development. *See* testbenches
- functions, loopback, 135

## fuses

- e-fuses, 182
- soft fuses, 182

**G**

- galloping patterns, 675
- gate array cells, 35–36
- gate length bias, 681
- gate line edge roughness (LER). *See* line edge roughness (LER)
- gate pitch, 68
- gates, equivalent NAND logic, 13–14
- GDS (Graphic Database System), 19
- GDS-II (graphic design system representation), 683
- general purpose I/O cells (GPIO), 26, 164
- generators, array, 31
- glitch power, 467
- global clock (grid, H-tree), 124–126, 350, 501–502
- global model for dynamic I\*R analysis, 519–521
- global net design planning, 494
- global n-sigma variation models, 461
- global power, 523, 614–615
- global repeater insertion, 474
- global routing, 122–124, 378–383
- glue logic, 6, 8–11, 216–218
- GPIOs (general-purpose I/O cells), 26, 164
- GPUs (graphics/image processing units), 29
- gradients, thermal, 535
- grains, 536
- graph-based timing analysis, 472
- Graphic Database System (GDS), 19
- graphic design system representation (GDS-II), 683
- graphics/image processing units (GPUs), 29
- gray box extraction, 162
- gray box layout cells, 162
- grids, clock, 125
- gross delay fault, 627–628, 661–662
- guard rings, 69–70, 563–564

**H**

- half-node process shrink, 46
- half-rule cell layout spacing, 23–24
- hard IP, 30–31, 136–140, 208, 353
- hardware attach interface, 273
- hardware description language (HDL) modeling, 32, 223. *See also* cell and IP modeling
  - concurrent sequential process (CSP), 223, 319
  - delta delay, 225
  - engineering change orders (ECOs) and, 598
  - immediate assignment, 225
  - levels of HDL modeling, 319–320
  - module (Verilog), 19, 32
  - preprocessors, 361
  - source/configuration cross-reference, 262
  - statements
    - assertion, 238–239
    - immediate assignment, 225
    - wait, 223
  - truth tables, 360–361
  - VHDL (VHSIC Hardware Description Language), 19, 32
- hardware-accelerated simulation, 268
  - emulation, 268–272, 274
  - prototyping, 272–274
- HAST (highly accelerated temperature/humidity stress test), 700
- hazard rate, 531
- HBM (human-body model), 158, 568
- HDL. *See* hardware description language (HDL) modeling
- healing factor, 538
- HFNS (high-fan-out net synthesis), 335–339
- hierarchical extraction of IP macros, 410–411
- hierarchical testing, 658–659
- hierarchy, logical and physical
  - automated test pattern generation (ATPG), 208–211
  - block-based design, 202
    - logic synthesis, 203–204
    - low-effort synthesis (LES), 204–205
    - physical design, 205–206
  - correspondence points, 6–8, 193–196
  - difference checking, 214
  - electrical abstracts, 214–216
  - floorplanning, 363–365
  - glue logic, 216–218
  - hierarchical units, 207–208
  - impact of changes to, 212–214
  - logic equivalency checking (LEC) flow, 193–195
  - logical-to-physical correspondence, 193–196
  - non-technical issues, 211–212
  - power and clock domain, 206–207
  - Rent's Rule, 203, 219
  - reuse of hierarchical units, 207–208
  - SRAM memory arrays, 197
    - array compilers, 197
    - dataflow and control flow, 198–202
    - methodology flows, 197–198
    - register files, 198
- high-fan-out net synthesis (HFNS), 335–339
- highly accelerated temperature/humidity stress test (HAST), 700

- high-sigma Monte Carlo, 92–93, 255
  - high-temperature operating life (HTOL) testing, 698–699, 703
  - high-up victim noise transients, 487–488, 493
  - hillocks, 536
  - hold latches, 660–661
  - hold time fixes, 401
  - hold time padding, 362
  - hold timing optimization, 348–349, 362
  - hot carrier effect, 252
  - hot lot fabrication scheduling, 43–44
  - hot spots, 206–207, 359, 496
  - HTOL (high-temperature operating life) testing, 698–699, 703
  - H-tree clock distribution, 124–125, 350
  - human-body model (HBM), 158, 568
- I**
- I\*R voltage drop effect, 483–484, 559–560
  - IBIS (I/O Buffer Information Specification) model, 576, 591
  - ICG (integrated clock gate), 107–109, 332
  - icicle topology, 153
  - IEEE (Institute of Electrical and Electronics Engineers), 19
  - IEEE JTAG 1149. *See* boundary scan
  - IEEE P1500. *See* wrap test
  - ignore lists, 615
  - ILM (interface-level model), 161, 215
  - ILT (inverse lithography technology), 154, 191
  - immediate assignment, 225
  - impact ionization, 573
  - importance sampling, 255
  - incremental recalculation, 337
  - inductance
    - inductance extraction, 408
    - partial, 515
  - industry standards, 19
  - inertial signal delay, 468–469
  - infant fail, 165, 208–211, 532
  - inferred state, 229–230
  - initialization, 228, 235
  - injected faults, 626
  - input pin noise characterization, 426–430
  - Institute of Electrical and Electronics Engineers (IEEE), 19
  - instruction set architecture (ISA), 132
  - integrated clock gate (ICG), 107–109, 332
  - integration, physical, 141–143
  - integration snapshots, 142, 174
  - intellectual property (IP). *See also* cell and IP modeling; modeling
    - backfill cells
      - abstracts, 38–39
      - gate array cells, 35–36
      - model constraints, 39–40
      - model properties, 39–40
      - overview of, 33–34
      - process design kit (PDK), 40–42
      - standard cell logic, 36–37
      - views, 38–39
    - black box IP, 152
    - design methodology
      - built-in self-test (BIST), 134–136
      - logic validation, 131–132
      - performance model, 132–133
      - physical model, 136–140
      - test models, 134–136
      - wrap test architecture, 134–136
    - design validation
      - logic synthesis model, 133–134
      - test patterns, 134–136
    - firm IP, 32, 208
    - general-purpose I/Os (GPIOs), 26
    - hard IP, 30–31, 208
    - macro-cells, 26–29
    - macros, hierarchical extraction of, 410–411
    - soft IP, 32, 207
    - standard cells, 21–25
    - suppliers, 20
    - templates, 21
  - intelligent fill, 608
  - interactive mode, 261
  - interconnect delay calculation, 446–451
  - interconnect extraction
    - coupling capacitance, 435–436
    - k-factor multiplier for coupling capacitance, 435–436
    - overview of, 433–434
    - parasitic netlist reduction, 436–437
    - resistivity, 435
  - interface-level model (ILM), 161, 215
  - inter-level metal dielectrics, 129
  - internal cell delay faults, 662–664
  - International Technology Roadmap for Semiconductors (ITRS), 44
  - interoperable PDK format (iPDK), 40–41
  - interposer, 101
  - inter-symbol interference (ISI), 97–98
  - inverse equivalent, 304
  - inverse lithography technology (ILT), 154, 191
  - inversion
    - temperature, 86–87
    - weak vs. strong, 61

- I/O boundary scans, 209
- I/O Buffer Information Specification (IBIS) model, 576, 591
- I/O electrical measurement tests, 180
- I/O pad circuit design, 563–565
- I/OGND, 564
- $I_{on}$  versus  $I_{off}$  curve, 24–25
- IOVDD, 564
- IP. *See* intellectual property (IP)
- iPDK (interoperable PDK format), 40–41
- ISA (instruction set architecture), 132
- ISI (inter-symbol interference), 97–98
- isolation values, 341
- issues tracking, 187
- iterative optimization, 368–369
- ITRS (International Technology Roadmap for Semiconductors), 44
  
- J**
- JEDEC (Joint Electron Device Engineering Council), 565–567, 699
- jogs, 122
- Joint Electron Device Engineering Council (JEDEC), 565–567
  
- K**
- k-factor multiplier for coupling capacitance, 435–436
- Kirchhoff's current law (KCL), 513
  
- L**
- large-scale integration (LSI), 13
- laser cuts, 672
- latchup, 69–70
  - definition of, 562–563
  - I/O pad circuit design, 563–565
  - qualification, 565–567
- late mode, 321–322
- launch-from-shift, 660
- layer data, 54–59
- layout parasitic extraction (LPE)
  - boundary element method (BEM), 406, 442
  - cell- and transistor-level parasitic modeling. *See* transistor-level parasitic modeling
  - decoupling capacitance, 431–433
  - extraction methodology, 409–411
  - floating random walk (FRW), 406, 442
  - inductance extraction, 408
  - interconnect extraction
    - coupling capacitance, 435–436
    - k-factor multiplier for coupling capacitance, 435–436
    - overview of, 433–434
    - parasitic netlist reduction, 436–437
    - resistivity, 435
  - overview of, 405–407
  - RLC modeling, 439
  - “selected net” extraction, 438–439
  - via/contact resistance, 442
- layout-dependent effects (LDEs), 136, 414–417
- layout-versus-schematic netlist checking (LVS), 150–151, 600, 610–612, 665
  - device width, representation of, 612–613
  - error debugging, 616
  - global power connections, 614–615
  - miscellaneous layout cells, 615–616
  - series connection equivalence, 613–614
  - stop lists, 610
- LBIST. *See* logic built-in self-test (LBIST)
- LDD (lightly doped drain), 64
- LDEs (layout-dependent effects), 136, 414–417
- leakage power improvement at constant performance, 25
- LEC (logic equivalency checking) flow, 193–195
- LER (line edge roughness), 78
- LES. *See* low-effort synthesis (LES)
- LET (linear energy transfer), 581–583
- level shifter cells, 341
- levelization, 233
- LFSR (linear feedback shift register), 646–648
- libraries. *See* cell library models
- lightly doped drain (LDD), 64
- line edge roughness (LER), 78
- line extensions past vias, 395–396
- line segment orientations, 53–54
- linear energy transfer (LET), 581–583
- linear feedback shift register (LFSR), 646–648
- linting, 185
- lithography process checking (LPC), 154–155, 618–620
- local clock distribution, 124–126
- local decoupling capacitance, 34
- local n-sigma variation models, 461
- logging, event trace, 288–289
- logic built-in self-test (LBIST), 163–164
  - compaction, 649–650
  - hierarchical testing, 658–659
  - overview of, 645
  - pattern generation, 645–648
  - pseudo-random pattern generation (PRPG), 650–652
  - test compression, 652–658
  - X-values in fault simulation, 650
- logic equivalency, 301–302

- combinational logic cone analysis, 305–306
  - logic cone, 194, 301–302, 305–306
  - logic equivalency checking (LEC) flow, 193–195
  - logic path depth, 147
  - logic synthesis
    - block size and, 203–204
    - clock tree synthesis (CTS), 350–353
    - hard IP macros, 353
    - hardware description language modeling, 319–320, 360–361
    - high-fan-out net synthesis (HFNS), 335–339
    - hold timing optimization, 348–349, 362
    - IP design methodology, 133–134
    - low-effort synthesis (LES), 204–205, 345, 354–356
      - design for testability analysis, 356–357
      - functional simulation, 358–359
      - path length-based model analysis, 357–358
    - performance optimization, 343–345
    - physically aware, 324–325
    - post-synthesis netlist characteristics, 339–340
    - power format file, 340–343
    - power optimization, 345–348
    - signal repowering
      - balanced repowering trees, 335–336
      - buffering networks, 336–339
    - technology mapping to cell library, 328–335
    - timing constraints
      - effective capacitance ( $C_{\text{eff}}$ ), 326–328
      - generation and verification of, 320–326
      - total capacitance ( $C_{\text{total}}$ ), 326–328
  - logical hierarchy. *See* hierarchy, logical and physical
  - logical macros, 27–29
  - logically symmetric inputs, 424
  - loopback functions, 135
  - low  $V_t$  (LVT) cells, 334–335
  - low-down victim noise transients, 487–488, 493
  - low-effort synthesis (LES), 204–205, 345, 354–356
    - design for testability analysis, 356–357
    - functional simulation, 358–359
    - path length-based model analysis, 357–358
  - low-resistivity gate material, 66–67
  - low-temperature operating life (LTOL) stress testing, 699
  - low- $V_t$  (LVT), 344
  - LPC (lithography process checking), 154–155, 618–620
  - LPE. *See* layout parasitic extraction (LPE)
  - LSI (large-scale integration), 13
  - LVS. *See* layout-versus-schematic netlist checking (LVS)
  - LVT (low  $V_t$ ) cells, 334–335, 344
- M**
- machine model (MM), 568
  - macro isolation, 639
  - macro-cells, 26–29
  - macros, logical, 27–29
  - main configuration (DDM), 173–174
  - Manhattan routing, 122
  - manufacturing release, preparation for
    - DRC waivers, 620–621
    - engineering change orders (ECOs)
      - design version management, 602–604
      - equivalency verification, 599–600
      - overview of, 595–597
      - physical design updates, 597–598
      - post-silicon cells for, 600–602
      - project tapeout planning, 603–604
    - physical design verification
      - design rule checking (DRC), 607–610
      - electrical rule checking (ERC), 616–618
      - job execution, 622–623
      - layout-versus-schematic netlist checking (LVS), 610–616
      - lithography process checking (LPC), 618–620
  - post-silicon debug
    - low-temperature operating life (LTOL) stress testing, 699
    - shmoo plots, 695–698
    - systematic test fails, 693–695
  - product qualification
    - highly accelerated temperature/humidity stress test (HAST), 700
    - high-temperature operating life (HTOL) testing, 698–699, 703
    - part sampling, 700–701
    - thermal cycling qualification testing, 700
  - tapeout
    - foundry interface release tapeout options, 678–684
    - project tapeout planning, 603–604, 689–691
    - tapeout checklist, 677, 684–689
- maps
- cell, 203
  - die thermal, 502, 507
  - route blockage maps, 22
  - state, 302–304
  - synthesis mapping, 32
  - thermal maps, 253
- marching patterns, 675
  - mask houses, 154
  - masking factors, 584
  - maximum likelihood, 181
  - Maxwell's equations, 405–406

- MBIST (memory built-in self-test), 163–164, 209, 638–645
  - MCMM (multi-corner multi-mode) analysis, 74, 88–89, 322
  - mean time to failure (MTTF), 529
  - median time to failure (MTTF), 530, 532
  - medium-scale integration (MSI), 13
  - memory array coverage, 292–293
  - memory array fault models, 674–675
  - memory built-in self-test (MBIST), 163–164, 209, 638–645
  - MEOL (middle-end-of-line), 679–680
  - merge operations, 173
  - metadata, 171
  - metal antennas, 396
  - metal cladding, 435
  - metal fill, 597–598, 606
  - metal grains, 536
  - metal hillocks, 536
  - metal personalization, 35
  - metal slots, 148–149
  - metal spacing, 52
  - metal trim masks. *See* cut masks
  - metal-insulator-metal (MIM), 50, 121, 522
  - metallization stack, 15, 31, 48–50, 176, 383–387
  - metal-oxide-semiconductor field-effect transistors (MOSFET), 59
  - metastability, 112, 130
  - methodology checks
    - electrical analysis flow status, 188–189
    - functional validation status, 187
    - overview of, 186
    - physical verification status, 188
    - tapeout audits, 189–190
    - testability status, 187–188
  - methodology flows, 1–6
  - methodology managers, 142, 165–168, 186, 677
  - middle-end-of-line (MEOL), 679–680
  - mid-life kickers, 697–698
  - migration, process
    - overview of, 92–93
    - parameterized layout cells (pCells), 93
    - relative cell placement, 94
  - MIM (metal-insulator-metal), 50, 121, 522
  - MIS (multiple-input switching), 244, 423–424
  - MISR (multiple-input signature register), 649–652
  - MM (machine model), 568
  - mock physical tapeouts, 621
  - model constraints, 39–40
  - model managers, 267–268
  - model order reduction (MOR) algorithms, 489–490
  - model properties, 39–40
  - model reduction, 374
  - model-based lithography process checking (LPC), 155
  - modeling. *See* cell and IP modeling; cell-level parasitic modeling; register-transfer level (RTL) modeling
  - moment matching algorithm, 489–490
  - moments of the transfer function, 448–451
  - Monte Carlo sampling, 92–93, 243–244, 255
  - Moore's law, 44
  - MOR (model order reduction) algorithms, 489–490
  - MPW (multi-project wafer), 44
  - MSI (medium-scale integration), 13
  - MTTF (mean time to failure), 529
  - MTTF (median time to failure), 530, 532
  - multi-bit errors, 584–589
    - circuit optimizations, 585–587
    - process optimizations, 584–589
    - system design optimizations, 587–589
  - multi-corner multi-mode (MCMM) analysis, 74, 88–89, 322
  - multi-corner optimization, 401
  - multi-cycle paths, 322, 458
  - multipatterning, 54–59, 376–377
    - cyclic decomposition error, 56
    - tapeout options, 678–679
  - multiple-input signature register (MISR), 649–652
  - multiple-input switching (MIS), 244, 423–424
  - multiplexing logic (MUX) cells, 329–330
  - multi-project wafer (MPW), 44
  - MUX (multiplexing logic) cells, 329–330
- N**
- names, cell instance, 339–340
  - natural yield, 645
  - NBTI (negative bias temperature instability), 252
  - NDR (non-default rule), 114
  - negative bias temperature instability (NBTI), 252
  - netlists, 8
    - definition of, 234–235
    - ECO mode netlist updates, 145–148
    - power and ground netlist (PG netlist), 151, 614–615
    - schematic/circuit, 140
  - nFET devices, 59, 251–253, 497
  - NIC (noise immunity curve), 479, 493
  - NLDM (Non-Linear Delay Model) tables, 419–421, 443
  - nodes
    - half, 46
    - nodal analysis formulation, 513
    - node transition faults, 662

- process, 44–45
  - process variants, 47–48
  - raised source/drain, 67
  - second sourcing, 47
  - shrink, 46
  - noise analysis
    - aggressors, 477
    - aggressor/victim model, 476–480
    - block-level, 493–494
    - cell input pin noise sensitivity, 247–249
    - detailed routing and, 391
    - direct aggressor coupling assumption, 477, 493
    - functional exclusions, 485–486
    - global net design planning and, 494
    - introduction to, 475–476
    - IP characterization data, 494
    - moment matching algorithm, 489–490
    - noise direction models, 485
    - noise immunity curve (NIC), 479, 493
    - noise impact on delay
      - I\*R voltage drop effect, 483–484
      - modeling, 481
      - static timing analysis, 482–483
    - parallel run length, 475–476
    - sensitivity windows, 487
    - small aggressor considerations, 485
    - template-based model, 490–491
    - timing windows, 479, 482, 486–488
    - victim low-down and high-up noise transients, 487–488, 493
    - victim nets, 477, 488–491
  - noise direction models, 485
  - noise immunity curve (NIC), 479, 493
  - non-blocking, 227
  - non-default rule (NDR), 114
  - Non-Linear Delay Model (NLDM) tables, 419–421, 443
  - non-orthogonal routes, 394–395
  - non-recurring expense (NRE), 42–44, 689–690
  - n-Sigma characterization, 241–245, 255
  - nucleation, 535
  - n-well, 59–60
- O**
- OASIS (Open Artwork System Interchange Standard), 19, 683
  - on-chip variation (OCV), 243, 630
  - OPAL (Open Pattern Analysis for Layout), 191
  - OPC (optical proximity correction), 618
  - Open Artwork System Interchange Standard (OASIS), 19, 683
  - Open Pattern Analysis for Layout (OPAL), 191
  - operating modes, 86–89
  - optical proximity correction (OPC), 618
  - optimizations. *See also* soft error rate (SER)
    - design, 185–186
    - IP design methodology, 132–133
    - iterative, 368–369
    - logic synthesis, 343–345
      - hold timing optimization, 348–349, 362
      - performance optimization, 343–345
      - power format file, 340–343
      - power optimization, 345–348
    - routing, 387–389
      - design-for-yield (DFY) optimizations, 395–399
      - electrical analysis optimizations, 389–393
      - multi-corner optimization, 401
      - non-orthogonal routes, 394–395
      - route segment and via topologies, 388–389
  - orientation, cell, 240
  - OSATs (outsourced assembly and test), 18, 692
  - out-of-range delay calculation, 422
  - outputs, registered, 473
  - outsourced assembly and test (OSAT), 18, 692
  - overdrive, 422–423
  - overflows, 33, 292–293, 382
  - overlay cell, 161
- P**
- package substrate data release, 692
  - parallel run length (PRL), 52, 475–476
  - parameter drift, 251–253
  - parameter sampling, 255
  - parameterized layout cells (pCells), 92–93
  - parametric faults and tests, 629–631
  - parasitic extraction. *See* layout parasitic extraction (LPE)
  - parasitic interconnect estimation, 366–367
  - parasitic netlist reduction, 436–437
  - part sampling, 700–701
  - partial inductance, 515
  - partition-level model for dynamic I\*R analysis, 519–521
  - partners, trusted, 136
  - path coverage, 293
  - path delay faults, 662
  - path length-based model analysis, 357–358
  - path-based timing analysis, 472
  - paths, false. *See* false paths
  - paths, multi-cycle, 458
  - pattern diagnostics, 665–666

- diagnostic network analysis, 666–667
- diagnostic pattern-to-fault correlation, 668–670
- failing dies, 672, 675
- overview of, 665–666
- test response compaction, 670–672
- pattern matching, 191, 608
- pattern-based lithography process checking (LPC), 154
- pattern-to-fault correlation, 668–672
- PCBs (printed circuit boards), 15–16
- PCT (pressure cooker test), 700
- PDK (process design kit), 18, 40–42
- PDN (power distribution network), 34, 509. *See also* power rail voltage drop analysis
- PDV. *See* physical design verification (PDV)
- peak current density (jPEAK), 541–542
- peak power calculation, 500–501
- performance boost operating mode, 87
- performance improvement at constant leakage power, 25
- performance optimization. *See* optimizations
- performance-sensing ring oscillator (PSRO), 525–526, 629, 697
- personalization, 35, 601
- pFET devices, 59, 251–253, 497
- P/G. *See* power and ground (P/G) distribution rails
- phase shifter, 655
- phase-locked loop (PLL), 124–125, 629–630
- physical design. *See also* hierarchy, logical and physical; physical design verification (PDV)
  - block size, 205–206
  - clock distribution, 124–126
  - design data management (DDM) and, 174–175
  - floorplanning
    - concept of, 363–365
    - early chip-package floorplanning, 95–96
    - floorplan channels, 363
    - optimization decisions, 113–114
    - pin assignment, 114–119
  - global route planning, 122–124
  - IP design methodology, 136–140
    - flip/mirror placement, 137–138
    - IP boundary, 137
    - pin data, 138–139
    - routing blockages, 137
    - schematic/circuit netlist, 140
  - logic equivalency checking (LEC) flow, 193–195
  - logical-to-physical correspondence, 193–196
  - skew management, 124–126
  - System-on-Chip (SoC)
    - chip finishing, 148–149
    - ECO mode, 143–148
    - floorplanning, 141
    - physical integration, 141–143
    - signoff flow, 149–165
- physical design verification (PDV)
  - design rule checking (DRC), 607–610
  - DRC waivers, 620–621
  - electrical rule checking (ERC), 616–618
  - job execution, 622–623
  - layout-versus-schematic netlist checking (LVS). *See* layout-versus-schematic netlist checking (LVS)
  - lithography process checking (LPC), 618–620
- physical integration, 141–143
- physical synthesis placement flow, 371
- physically aware logic synthesis, 324–325
- pillars, 377, 389
- pin assignment, 114–119
- pin data, 94, 116, 138–139
- pin groups, 117, 240
- pin property, 115
- pin shape, 115, 120
- pin-to-output pin arcs, 241–242
- pitch
  - contacted gate, 68
  - forbidden, 191, 394–395
  - matching, 200
- itches, 188
- placement, 94, 367–369
  - clock tree local buffer, 369–370
  - constructive, 368–369, 371
  - device-level, 400–401
  - Elmore delay, 366
  - estimated wire length, 370–371
  - global floorplanning of hierarchical units, 363–365
  - parasitic interconnect estimation, 366–367
  - relative cell placement, 27, 200–201, 325
- placement-and-routing boundary (prBoundary), 137
- platforms, 2
- PLL (phase-locked loop), 124–125, 629–630
- PMIC (power management integrated circuit), 101
- point tool integration, 169
- Portable Test and Stimulus Standard (PSS), 299
- post-silicon debug
  - low-temperature operating life (LTOL) stress testing, 699
  - shmoo plots, 695–698
  - systematic test fails, 693–695
- post-synthesis netlist characteristics, 339–340
- post-tapeout engineering change orders (ECOs), 688–689

- power, performance, and area (PPA), 14–15
- power abstract, 97
- power analysis
  - cell model characterization data, 245–247
  - design-for-power feedback from, 504–505
  - device self-heat models, 502–504
  - die thermal maps, 502, 507
  - global clock power dissipation, 501–502
  - hot spots, 496
  - introduction to, 495–497
  - IP power models, 501–502
  - peak power calculation, 500–501
  - power dissipation, 296–297, 497–500, 507
  - power stressmark testcases, 496, 506
  - PVT corners, 500, 503–504, 506
  - summary of, 505–506
  - switching factors, 495
  - temperature inversion, 501
- power and ground (P/G) distribution rails
  - ASIC (application-specific integrated circuit)
    - direct release, 510–512
  - conservative versus aggressive P/G grid design, 510
  - dynamic P/G voltage drop analysis, 513–526
  - sleepFET devices, 512
  - static I\*R rail analysis, 512–513
- power dissipation, 296–297, 497–500, 507
- power distribution network (PDN), 34, 175–177, 509. *See also* power rail voltage drop analysis
- power domain, 105–107, 206–207
- power format description, 40
- power format file, 340–343
- power grid, 119–122
  - power grid admittance matrix, 516–519
  - power grid conductance matrix, 516–519
- power intent files, 104, 340–343
- power management integrated circuit (PMIC), 101
- power map abstract, 96
- power optimization
  - detailed routing and, 391–392
  - ECO mode netlist updates, 145–148
  - logic synthesis, 345–348
- power rail electromigration analysis (powerEM), 545–548
- power rail voltage drop analysis, 509
  - ASIC (application-specific integrated circuit)
    - direct release, 510–512
  - conservative versus aggressive P/G grid design, 510
  - dynamic P/G voltage drop analysis
    - analysis results, 521–522
    - global and partition-level models for, 519–521
    - global power delivery frequency response, 523
    - matrix solution, 516–519
    - P/G rail capacitance, 514–515
    - simultaneous switching output (SSO)
      - analysis, 523–526
    - sleepFET devices, 512
    - static I\*R rail analysis, 512–513
  - power state sequence validation, 266
  - power stressmark testcases, 496, 506
  - power supply sequence validation, 266
  - powerEM, 545–548
  - power-gating design, 365
  - power-state leakage, 499
  - PPA (power, performance, and area), 14–15
  - pragmas, 133
  - prBoundary, 137
  - preprocessors, 361
  - pressure cooker test (PCT), 700
  - PRIMA calculation, 375
  - printed circuit boards (PCBs), 15–16
  - PRL (parallel run length), 52, 475–476
  - process, voltage, and temperature (PVT) corners, 86–89, 321, 500, 503–504, 506
  - process design kit (PDK), 18, 40–42
  - process migration, 92–93
    - parameterized layout cells (pCells), 93
    - relative cell placement, 94
  - process nodes, 44–45
  - process retargeting, 92–93
    - parameterized layout cells (pCells), 93
    - relative cell placement, 94
  - process variation-aware design
    - array design, 91–92
    - best-case (BC) process, 89–90
    - high-sigma Monte Carlo, 92–93
    - worst-case (WC) process, 89–90
  - process windows, 620
  - product qualification
    - highly accelerated temperature/humidity stress test (HAST), 700
    - high-temperature operating life (HTOL) testing, 698–699, 703
    - part sampling, 700–701
    - thermal cycling qualification testing, 700
  - production test patterns, validation of, 284–288
  - project scoreboards, 677
  - Property Specification Language (PSL), 265, 299
  - prototyping (simulation acceleration), 272–274, 299
  - PRPG (pseudo-random testcase pattern generator), 282–283, 650–652
  - pseudo-logical gate, 664

- pseudo-random pattern generation (PRPG),  
282–283, 632, 650–652
  - PSL (Property Specification Language), 265, 299
  - PSRO (performance-sensing ring oscillator),  
525–526, 629, 697
  - PSS (Portable Test and Stimulus Standard), 299
  - PVT (process, voltage, and temperature) corners,  
86–89, 321, 500, 503–504, 506
- Q**
- Qcrit, 580–581
  - quality of results (QoR), 185, 191
- R**
- raised source/drain nodes, 67
  - random pattern resistant fault, 648
  - random sampling, 255
  - random testcase generation, 263–264
  - range, 582
  - RAT (required arrival time), 455
  - RDL (redistribution layer), 395
  - RDRs (recommended design rules), 152–153
  - recognition operations, 409
  - recommended design rules (RDRs), 152–153
  - redistribution layer (RDL), 395
  - redundant vias, 396
  - reference EDA tools, 40–41
  - reference flows, 168–169
  - register files, 198
  - registered input/output pins, 473
  - register-transfer level (RTL) modeling, 133,  
232–234, 302–306
    - design optimization, 185–186
    - linting, 185
    - netlist signal name correspondence, 339–340
    - power stressmark testcases, 496, 506
    - PVT corners, 86–89, 321, 500, 503–504, 506
    - RTL combination model equivalency, 301–302
    - of scans, 285–288
    - simulation model statement coverage
      - analog mixed-signal coverage, 294–295
      - bad machine path coverage, 294
      - expression and conditional coverage, 291–292
      - finite state machine (FSM) coverage, 294
      - memory array coverage, 292–293
      - model coverage analysis, 290–291
      - path coverage, 293
      - signal toggle activity, 291
      - strengths and weaknesses of, 295–296
  - regression, 281
  - relative cell placement, 27, 94, 200–201, 325
  - release flow management, 165–168
  - release gate, 263
  - reliability function, 531
  - Rent's Rule, 203, 219
  - reporting, 460–461, 472–473
  - required arrival time (RAT), 455
  - required time, 147
  - reset, 179, 265–266
  - resistance increase, 543
  - resistive fails, 557
  - resistive Joule heating, 540–541
  - resistive shielding, 326, 375
  - resolution function, 227
  - resource planning, 281–284
  - return current loop, 439
  - ring-oscillator frequency variation, 102
  - rip-up and re-route, 141–142, 399
  - river routes, 394–395
  - RLC modeling, 439
  - robust design, 529
  - routing, 122, 373–374
    - blockages, 137, 377
    - C effective model reduction for delay calculation,  
374–375
    - congestion, 33
    - cut masks, 378
    - design-for-yield (DFY) optimizations,  
395–399
      - antenna diode insertion, 396–399
      - line extensions past vias, 395–396
      - redundant vias, 396
      - wire spreading, 395
    - detailed, 378–383
    - device-level placement, 400–401
    - DRC rule complexity, 375–376
    - electrical analysis optimizations, 389–393
      - electromigration, 392–393
      - noise optimization, 391
      - power optimization, 391–392
      - timing optimization, 390
    - estimates, 383
    - global, 122–124, 378–383
    - hold time fixes, 401
    - Manhattan, 122
    - metallization stack, 383–387
    - multi-corner optimization, 401
    - multipatterning, 376–377
    - non-default rule (NDR), 114
    - non-orthogonal routes, 394–395
    - overflows, 382
    - pillars, 377
    - rip-up and re-route, 399
    - route blockage maps, 22

- route segment and via topologies, 387–389
  - “balanced” routing, 388
  - route shielding, 388
  - via optimizations, 388–389
  - via pillars, 389
- route segments, 387–389
- router shielding, 388
- sidewall coupling capacitance, 374–375
- track jobs, 122
- RTL. *See* register-transfer level (RTL) modeling
- runset, 40, 607
- S**
- SADP (self-aligned, double-patterning), 57–58
- SAIF (Switching Activity Interchange Format), 299
- SAQP (self-aligned, quad-patterning), 59, 130
- scaled-sigma sampling, 255
- scaling factor, 44–45
- scan
  - boundary, 180, 635
  - RTL modeling of, 285–288
  - scan chains, 179
  - scan dumps, 695
- scanning electron microscopy (SEM), 703
- scenic path, 122, 348–349
- schematic/circuit netlist, 140
- scoreboards, 142, 186, 472–473, 677
- SCR (silicon-controlled rectifier), 69–70
- scribe channels, 124
- SDKs (software development kits), 30
- second sourcing, 47
- second-pass design, 35, 43
- segments, route, 387–389
- “selected net” extraction, 438–439
- self-aligned, double-patterning (SADP), 57–58
- self-aligned, quad-patterning (SAQP), 59, 130
- self-aligned contacts, 130
- self-heating, 82–83, 253, 502–504
- SEM (scanning electron microscopy), 703
- Semiconductor Equipment and Materials International (SEMI), 19
- semiconductor foundries, 18. *See also* fabrication technology
- semi-custom designs, 17
- sensitivity analysis, 181, 487
- sequential cells, modeling for, 249–251
- sequential circuit characterization, 424–425
- sequential logic equivalence, 307–309
- SER. *See* soft error rate (SER)
- serial shift-register scan chain, 632
- series connection equivalence, 613–614
- setup and hold time definition, 38, 130
- SEU (single-event upset), 70–71
- severity parameters, 238
- shallow trench oxide isolation (STI), 59–60
- shielding
  - resistive, 326, 375
  - route, 388
- shipped project quality level (SPQL), 164–165
- shmoo plots, 695–698
- short channel  $V_t$  effect, 347
- shrink nodes, 46
- shuttles, 44
- sidewall coupling capacitance, 374–375
- sidewall spacers, 57
- signal interconnect electromigration analysis (sigEM), 548–555
- signal repowering
  - balanced repowering trees, 335–336
  - buffering networks, 336–339
- signal toggle activity, 291
- signature registers, 209–210
- signatures, 641
- signoff flow, 42
  - definition of, 149–150
  - design rule checking (DRC), 152
  - design-for-manufacturability (DFM), 153–156
    - chemical-mechanical polishing (CMP) analysis, 155–156
    - lithography process checking (LPC), 154–155
  - design-for-yield (DFY), 153
  - DRC errors/waivers, 156–157
  - EDA DFM services, 156
  - electrical analysis flows for, 160–163
  - electrical rule checking (ERC), 157–160
    - electrostatic discharge (ESD) protection, 158–159
    - voltage-dependent design rule checks, 159–160
  - layout-versus-schematic netlist checking (LVS), 150–151
  - recommended design rules (RDRs), 152–153
  - test analysis for, 163–165
- silicon-controlled rectifier (SCR), 69–70
- silicon-on-insulator processes, 253
- simulation. *See also* fault diagnosis; testbenches
  - behavioral co-simulation, 275
  - event trace logging, 288–289
  - hardware-accelerated, 268
    - emulation, 268–272, 274
    - prototyping, 272–274
  - Portable Test and Stimulus Standard (PSS), 299
  - resource planning, 281–284
  - software

- delay-based functional simulation, 474
- event-driven, 259–261
- HDL source/configuration cross-reference, 262
- interactive mode, 261
- waveform display, 261
- switch-level, 276
- symbolic, 277–281
- throughput, 281–284
- time, 225–226
- simulation model statement coverage
  - analog mixed-signal coverage, 294–295
  - bad machine path coverage, 294
  - expression and conditional coverage, 291–292
  - finite state machine (FSM) coverage, 294
  - memory array coverage, 292–293
  - model coverage analysis, 290–291
  - path coverage, 293
  - signal toggle activity, 291
  - strengths and weaknesses of, 295–296
- Simulation Program with Integrated Circuit Emphasis (SPICE), 19
- simultaneous switching output (SSO), 98, 523–526
- single-event upset (SEU), 70–71
- sinkless pins, 143
- skew management, 124–126
- slack, timing, 454–457
- sleep mode, 87
- sleepFET devices, 106, 341
  - I\*R voltage drop effect, 559–560
  - power rail voltage drop analysis, 512
  - sleep-to-active state transition, 561–562
  - substrate noise injection, 562–563
- sleep-to-active state transition, 561–562
- slots, 148–149
- small aggressors, 485
- small-scale integration (SSI), 13
- SMO (source-mask optimization), 618
- SMT (surface-mount technology), 94
- snapback mode, 574
- snapshots, 142, 174
- SoC. *See* System-on-Chip (SoC)
- soft error rate (SER), 69–70, 163, 529–530, 576–577
  - alpha particles, 577–578
  - circuit optimizations, 585–587
  - combinational logic, 584
  - cosmic rays, 578
  - flops, 584
  - linear energy transfer (LET), 581–583
  - multi-bit errors, 584–589
    - circuit optimizations, 585–587
    - process optimizations, 584–589
    - system design optimizations, 587–589
    - process optimizations, 584–589
  - Qcrit, 580–581
  - SER diagnosis, 579–580
  - system design optimizations, 587–589
- soft fuses, 182
- soft IP, 32, 207
- software development kits (SDKs), 30
- software simulation
  - delay-based functional simulation, 474
  - event-driven, 259–261
  - HDL source/configuration cross-reference, 262
  - interactive mode, 261
  - waveform display, 261
- sourceless pins, 143
- source-mask optimization (SMO), 618
- spacer dielectric, 67–71
- spare logic cells, 600–602
- SPEF (Standard Parasitic Exchange Format), 19
- SPICE (Simulation Program with Integrated Circuit Emphasis), 19
- split lot processing, 43
- SPQL (shipped project quality level), 164–165
- SRAFs (sub-resolution assist features), 618
- SRAM bit cell design, 197
  - array compilers, 197
  - methodology flows, 197–198
  - register files, 198
- SSI (small-scale integration), 13
- SSO (simultaneous switching output), 98, 523–526
- STA. *See* static timing analysis (STA)
- stacked via topologies, 375–376
- standard cells
  - drive strength, 22–24
  - layout of, 21–22
  - logic, 36–37
  - threshold voltage, 24–25
- Standard Parasitic Exchange Format (SPEF), 19
- standard  $V_t$  (SVT) cells, 334–335
- state mapping, 302–304
- state-dependent delays, 420–421, 459
- statements
  - assertion, 132, 238–239
  - immediate assignment, 225
  - wait, 223
- states
  - inferred, 229–230
  - trap, 252
- static I\*R power grid voltage analysis, 483–484, 512–513
- static leakage tests, 180
- static noise analysis. *See* noise analysis

- static timing analysis (STA). *See* timing, static timing analysis (STA)
  - Steiner tree route, 122, 379
  - stepping window, 608
  - STI (shallow trench oxide isolation), 59–60
  - stop lists, 610
  - stopping power, 581
  - straggle, 582
  - streamed layout data, 683
  - stress testing
    - burn-in, 104, 208–211, 625–626, 703
    - highly accelerated temperature/humidity stress test (HAST), 700
    - high-temperature operating life (HTOL), 698–699, 703
    - low-temperature operating life (LTOL), 699
  - strong inversion, 61
  - structural cell-level netlists, 319
  - stuck-at fault, 241, 626–629
  - sub-resolution assist features (SRAFs), 618
  - substrate noise injection, 562–563
    - analog IP tests, 563–564
    - I/O pad circuit design, 563–565
    - latchup analysis, 562–563
    - latchup qualification, 565–567
  - subthreshold leakage currents, 62–63
    - subthreshold current sensors, 102
    - subthreshold slope, 72, 76–77, 79–80
  - sum of failure rates, 534–535
  - surface-mount technology (SMT), 94
  - survival function, 531, 533
  - sustaining voltage, 564
  - SVT (standard  $V_i$ ) cells, 334–335
  - swaps, cell, 203
  - switching activity factor estimates, 296–297
  - Switching Activity Interchange Format (SAIF), 299
  - switching factors, 495
  - switch-level simulation, 276
  - symbolic simulation, 277–281
  - synthesis, 8
    - clock tree synthesis (CTS), 336
    - high-fan-out net synthesis (HFNS), 335–339
    - logic. *See* logic synthesis
    - mapping, 32
    - pragmas, 133
    - test insertion, 134
  - System Verilog, 19, 32
  - systematic test fails, 693–695
  - SystemC, 254
  - System-on-Chip (SoC), 17–21, 212–214.
    - See also* design implementation;
    - electrical analysis; formal equivalency verification
  - error capture/recovery, 179
  - hierarchy, logical and physical. *See* hierarchy, logical and physical
  - physical design methodology
    - chip finishing, 148–149
    - ECO mode, 143–148
    - floorplanning, 141
    - physical integration, 141–143
    - signoff flow, 149–165
  - reset, 179
- T**
- tapeout, 42–44
    - foundry interface release tapeout options
    - back-end-of-line (BEOL) layer data, 679–680
    - cut metal masks, 682–684
    - FinFET data, 681–682
    - front-end-of-line (FEOL) layer data, 679–680
    - multipatterning decomposition color assignment, 678–679
    - tapeout data volume, 680–681
  - mock physical tapeouts, 621
  - planning, 603–604
  - project tapeout planning, 603–604, 689–691
  - tapeout audits, 189–190
  - tapeout checklist, 677
    - overview of, 684–685
    - post-tapeout engineering change orders (ECOs), 688–689
    - tapeout waivers, 685–688
  - tapeout signoff, 42
- TAT (turnaround time), 600
- techfiles, 40
- technology mapping to cell library, 328–335
- TEM (transmission electron microscopy), 703
- temperature inversion, 86–87, 501
- temperature sensing, 102–104
- template (standard cell design image), 21
- template-based model, 21
- template-based noise analysis, 490–491
- test compression, 209–210, 652–658
- test escape, 164–165, 625
- test insertion, 134
- test models, 134–136, 235–236
- test modes, 88
- test pattern diagnostics
  - automated test pattern generation (ATPG)
    - limitations, 208–211
  - IP design methodology, 134–136
  - ordering, 179–182
  - overview of, 197
  - pattern generation, 179–182

- test syndrome, 669
- testability, design for. *See* design for testability (DFT)
- testbenches, 30, 346
  - definition of, 259–260
  - pseudo-random pattern generation (PRPG), 282–283
  - stimulus development
    - assertion-based validation (ABV), 264–265
    - bad machine error injection testcases, 267
    - directed testcases, 263
    - model managers, 267–268
    - power state sequence validation, 266
    - power supply sequence validation, 266
    - random testcase generation, 263–264
    - reset validation, 265–266
    - transaction-based, 262
    - uninitialized signal propagation, 265–266
- testcases
  - assertion-based validation (ABV), 264–265
  - bad machine error injection testcases, 267
  - definition of, 259–260
  - directed, 263
  - error injection testcases, 267
  - model managers, 267–268
  - power state sequence validation, 266
  - power supply sequence validation, 266
  - pseudo-random pattern generation (PRPG), 282–283
  - random testcase generation, 263–264
  - reset validation, 265–266
  - transaction-based, 262
  - uninitialized signal propagation, 265–266
- test-mode equivalence verification, 309–310
- testpoints, 302
- text analysis, 163–165
- thermal cycling qualification testing, 700
- thermal gradients, 535
- thermal hot spots, 206–207
- Thermal Interface Material layers (TIM), 502
- thermal interface material (TIM), 96, 183
- thermal management, 101–104
- thermal maps, 253, 502, 507
- thermal resistance, 82–83
- thermal sensing, 206–207
- thermal shock, 700
- thermomigration, 535
- thermsense, 102, 496
- thick gate oxide devices, 26, 48
- three-dimensional (3D) field-solver algorithms, 405–406
- threshold voltage, 24–25, 67
- throughput, 281–284
- TIM (thermal interface material), 96, 183
- time borrowing, 251
- Time to Profit (TTP), 691
- timing analysis
  - block-level timing constraint budgeting, 473–474
  - cell delay calculation, 241–245, 443–446
  - delay-based functional simulation, 468, 474
  - derating delay multipliers, 451, 462–466
  - electrical design checks, 452–453
  - graph- versus path-based, 472
  - interconnect delay calculation, 446–451
  - static timing analysis (STA), 424, 453–454, 482–483
    - delay adjust, 458–459
    - delay-based timing verification, 467–469
    - false paths, 458–459, 472
    - inertial signal delay, 468–469
    - pin constraints, 457
    - reporting, 460–461, 472–473
    - timing analysis modes, 459–460
    - timing “don’t care,” 458–459
    - timing slack, 454–457
    - transport signal delay, 468–469
    - variation-based timing, 461–466
  - timing constraints, 361, 457
    - effective capacitance ( $C_{\text{eff}}$ ), 326–328
    - generation and verification of, 320–326
    - total capacitance ( $C_{\text{total}}$ ), 326–328
  - timing optimization
    - buffering networks, 336–339
    - detailed routing and, 390
    - ECO mode netlist updates, 145–148
    - hold timing optimization, 348–349, 362
  - timing slack, 24, 335, 454–457
  - total negative slack (TNS), 188, 461
    - worst negative slack (WNS), 188, 461
  - timing windows, 479, 482–483, 486–488
- TLP (transmission line pulse), 568–569
- TMR (triple modular redundancy), 588
- TNS (total negative slack), 188, 344–345, 461
- tool evaluation, 169–170, 191–192
- “top off” method, 652
- top-level flat analysis, 214–216
- total capacitance ( $C_{\text{total}}$ ), 326–328
- total negative slack (TNS), 188, 344–345, 461
- trace files, 233, 298–299
- trailblazing, 11, 170–171, 191–192
- transaction-based testcases, 262
- transfer function, 448–451
- transfer gates, 329–330
- transistor fingers, 22–23
- transistor-level parasitic modeling, 411
  - cell characterization, 419–422

- cell extraction, 411–413
- cell power characterization, 430–431
- extraction corners, 417–419
- input pin noise characterization, 426–430
- layout-dependent effects (LDEs), 414–417
- logically symmetric inputs, 424
- multiple-input switching (MIS), 423–424
- out-of-range delay calculation, 422
- sequential circuit characterization, 424–425
- voltage values, 422–423
- transition time, 61
- transmission electron microscopy (TEM), 703
- transmission line pulse (TLP), 568–569
- transport signal delay, 468–469
- trap states, 252
- trench isolation, 585
- triggering input, 69–70
- triple modular redundancy (TMR), 588
- triple-well process option, 49, 563
- trusted partners, 136
- truth tables, 360–361
- TTP (Time to Profit), 691
- turnaround time (TAT), 600
- twin-well process, 585

## U

- UDP (user-defined primitive), 360
- ultra-low- $V_t$  (ULVT) cells, 344
- unateness property, 454
- underfill, 694
- underflow, 292–293
- undetectable faults, 631
- unidirectional DC current density ( $j_{DC}$ ), 539–540
- uninitialized signal propagation, 265–266
- United States Department of Defense, 699
- useful clock skew, 351
- user-defined primitive (UDP), 360
- U-value propagation, 185

## V

- vacancy, 535
- Value Change Dump (VCD), 298
- variation-based timing, 461–466
- vectored data, 346

- vectorless data, 346
- Verilog, 19
- very-large-scale integration (VLSI), 13–14
- VHSIC Hardware Description Language (VHDL), 19, 32
- via optimizations, 388–389
- via pillars, 153, 389
- via resistance, 442
- via/contact resistance, 442
- victim nets, 477, 488–491. *See also* noise analysis
- viewports, 267
- views, cell, 38–39
- virtual filesystem interfaces, 171
- VLSI (very-large-scale integration), 13–14
- voids, 535
- voltage. *See* electrical analysis
- voltage regulator models (VRMs), 266
- voltage-dependent design rule checks, 159–160

## W

- wafer probe-level testing, 44
- wait statements, 223
- waivers
  - DRC, 156–157, 620–621
  - tapeout, 685–688
- waveform display, 261
- WC (worst-case) process, 87, 89–90, 243
- weak inversion, 61
- wearout region, 530, 533
- wire length, estimation of, 370–371
- wire spreading, 395
- wire temperature increase due to resistive Joule heating ( $J_{RMS}$ ), 540–541
- wireload per fan-out estimates, 325
- wiring tracks, 9
- worst negative slack (WNS), 188, 344–345, 461
- worst-case (WC) process, 87, 89–90, 243, 255
- wrap test, 134–136, 163–164, 209, 635
- wrong-way segments, 122, 380

## X-Y-Z

- yield, design for. *See* design-for-yield (DFY)
- yield with repair, 645