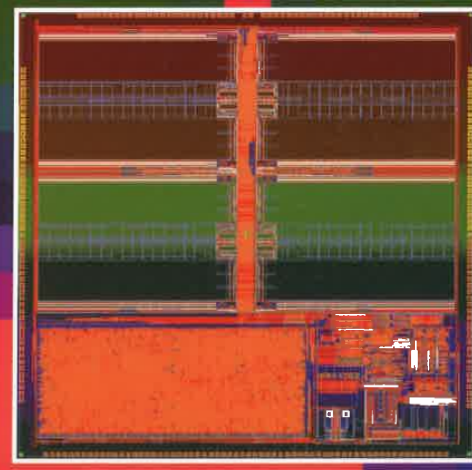
DAN CLEIN

CMOS IC LAYOUT

Concepts, Methodologies, and Tools



Newnes

Gregg Shimokura, Technical Contributor

CMOS IC LAYOUT

Concepts, Methodologies, and Tools

Dan Clein
Technical Contributor: Gregg Shimokura

Boston Oxford Auckland Johannesburg Melbourne New Delhi

Newnes is an imprint of Butterworth–Heinemann.

Library of Congress Cataloging-in-Publication Data

Clein, Dan, 1958–
CMOS IC layout : concepts, methodologies, and tools / Dan Clein; technical contributor, Gregg Shimokura.
p. cm.
ISBN 0-7506-7194-7 (pbk. : alk. paper)
1. Metal oxide semiconductors, Complementary—Computer-aided design. 2. Integrated circuits—Computer-aided design. I. Title.
TK7871. 99.M44C485 1999
621.39'732—dc21 99-44934
CIP

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

The publisher offers special discounts on bulk orders of this book.
For information, please contact:
Manager of Special Sales
Butterworth–Heinemann
225 Wildwood Avenue
Woburn, MA 01801-2041
Tel: 781-904-2500
Fax: 781-904-2620

For information on all Newnes publications available, contact our World Wide Web home page at: http://www.newnespress.com

10 9 8 7 6 5 4 3

Printed in the United States of America

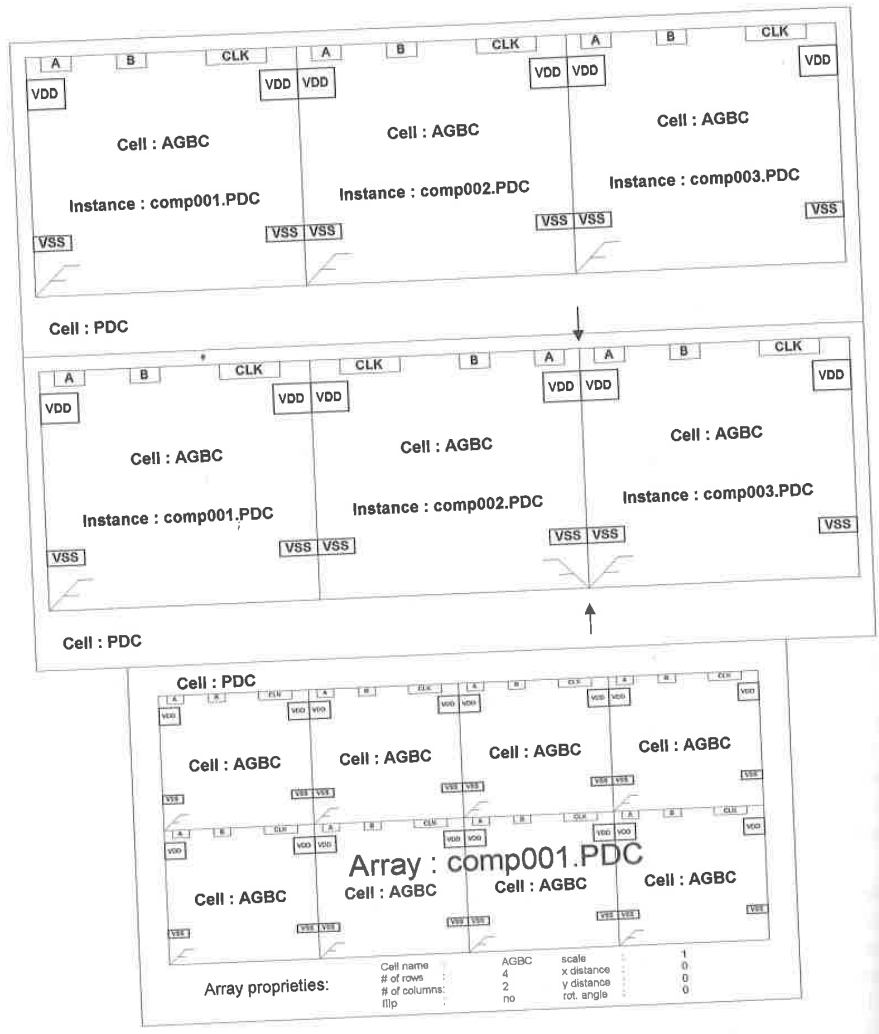


Figure 3.24 Examples of hierarchical designs.

3.8 GENERAL GUIDELINES

Now that we have considered a layout floorplan, it is time to implement the design. There are general guidelines that should be followed both in planning and implementing the design, and the fundamental ones are listed in this section.

3.8.1 Guidelines for the Layout of Power Lines

Power supply lines have to be determined before starting the layout of any cell. Specific guidelines for power lines include the following:

- Determine line width based on the following:
  Does the line provide power only for internal use, or is there a requirement for the line to feed other cells and be a part of the chip's power grid? This is critical information that can be determined from the layout floor plan.
  Use resistivity information of the different layers to determine the appropriate width of the lines.
- Use lowest level of metal for power for transistor level cells. It is important to consider that using higher layers of metal for power requires vias and local interconnect polygons to connect transistors to supplies. This will take space and limit the porosity of the cell. Generally use the lowest level of metal for power that the process and design allows.
- Avoid notching power lines. Power lines carry large amounts of current; therefore, it is important to make sure that they are routed with a consistent width and never notched. Any notches in a line create potential fuses that may break the power line under high current conditions.
- Avoid over the cell power routing. Unless the power is routed using automated tools, it is not advisable to run power supply lines over the cell. Keep the power lines inside the cell to ensure that the cell is correct by construction. Power routing outside the cell is ideally limited to connections between cells.

3.8.2 Guidelines for the Layout of Signals

The following is a list of guidelines specifically for routed signals. More details on each of these concepts are presented in later chapters.

- Choose routing layers based on process parameters and circuit requirements. For each process a standardized list of routing layers should be determined based on layer resistance and capacitance. Layers such as N-well, active, and high-resistance poly gate are not used for routing. Priorities between routing layers can also be standardized using the same criteria.
- Minimize the width of input signals. Minimizing the routing of the signals within the design is also important. This reduces the input capacitance for the signal. This is important for signals that are part of cells that are used many times. An example of this would be a clock signal within a cell.
- Choose routing width carefully. The choice of the width of a routing signal should be made judiciously. It is tempting to simply use the minimum design rule line width as the routing width. This is not practical in all cases because connections must be made to every line.
  These connection points require a via or contact, and as you can see in Figure 3.25, the space required for a via or contact is generally wider than the minimum width rule for the routing layer.

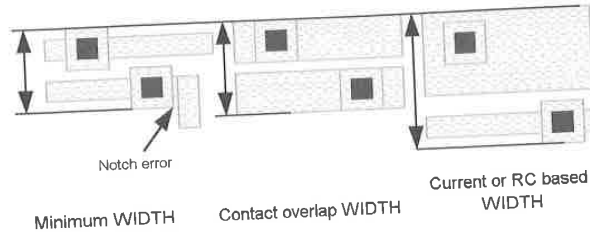


Figure 3.25 Routing line width considerations.

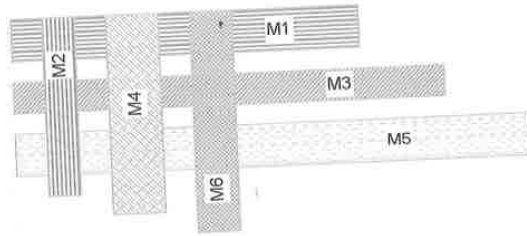


Figure 3.26 Routing direction.

It is perhaps more efficient to use a width of line that accommodates both a via and contact. This line would be wider but is much easier to manage and maintain as it avoids jogging other signals around the contact point.

- Maintain a consistent routing direction within a cell or block (Figure 3.26). Then, when a layout designer wants to change signal direction, it will take only one type of via and it will not interfere with the porosity of the cell. In general it makes sense to keep a consistent metal direction for each layer and alternate the direction from layer to layer. For example, if metal1, metal3, and metal5 are routed horizontally, then metal2, metal4, and metal6 should be routed vertically.
- Label all important signals. This is very important for the layout verification process, especially LVS. Error diagnosis, short isolation, and LVS run time are all easier when nodes are labeled.
- Determine the minimum number of contacts for every connection. Don't assume that a single contact or via for every connection is enough. Some memories, for example, are using double contacts everywhere possible to increase reliability.

3.8.3 Guidelines for the Layout of Transistors

The following is a list of guidelines specifically for a cell-level design environment.

- Use a predefined template for PMOS and NMOS transistor placement. The architecture of a cell should be defined beforehand, and this template should encapsulate the basic floorplan of a group of cells. Figure 3.27 shows an example of a cell template.

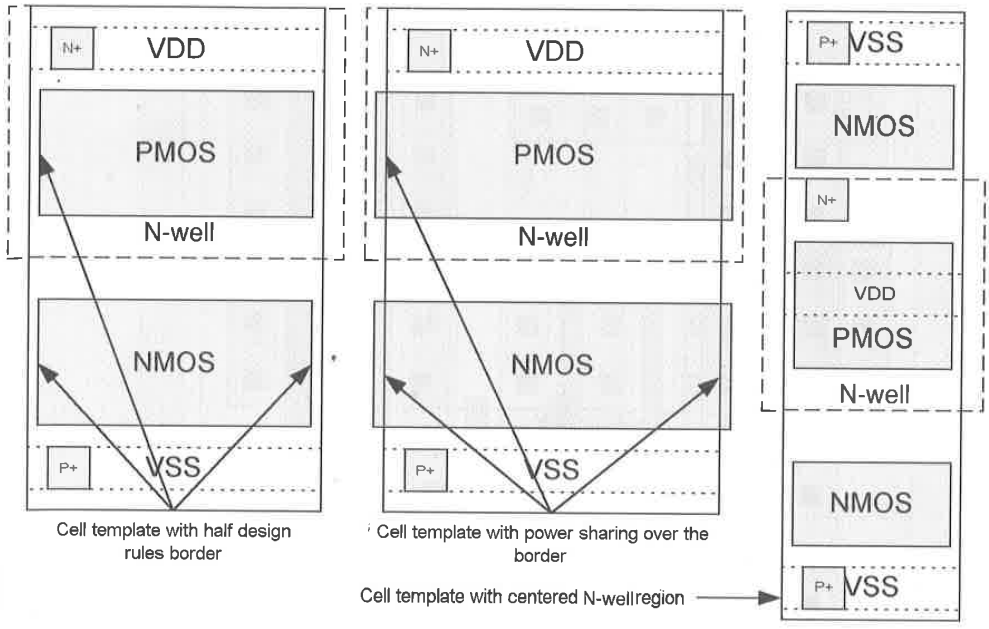


Figure 3.27 Example cell template.

- Use transistor fingering for large and critical transistors. A cell template similar to Figure 3.27 defines the maximum width of a transistor by the cell height. How do we lay out a transistor that exceeds this height? The solution is to "finger" the transistor into multiple transistors that are connected in parallel.

  Figure 3.28 shows three equivalent layout designs of a transistor that is 100 μm wide.

  There is an advantage with an even number of fingers: the active capacitance is less, because the drain region is surrounded with gate poly instead of field.

  Another reason to use fingering is to optimize the resistance of the gate poly along the width of the transistor. Since the gate poly is driven from one end and gate poly is resistive, there may be reason to have a guideline that states the maximum width of a single finger. Fingering is the only way to meet this guideline for large transistors.

  Fingering multiple transistors that are connected in series is trickier. Figure 3.29 shows an example of fingering a two-input NAND gate.

  Fingering the PMOS devices is straightforward; however, fingering the series NMOS devices is more difficult because the order of connectivity of the devices must be maintained.

- Share power supply nodes to save area. Sharing nodes whenever possible is a concept that is easy to understand. Power supply nodes are most easily

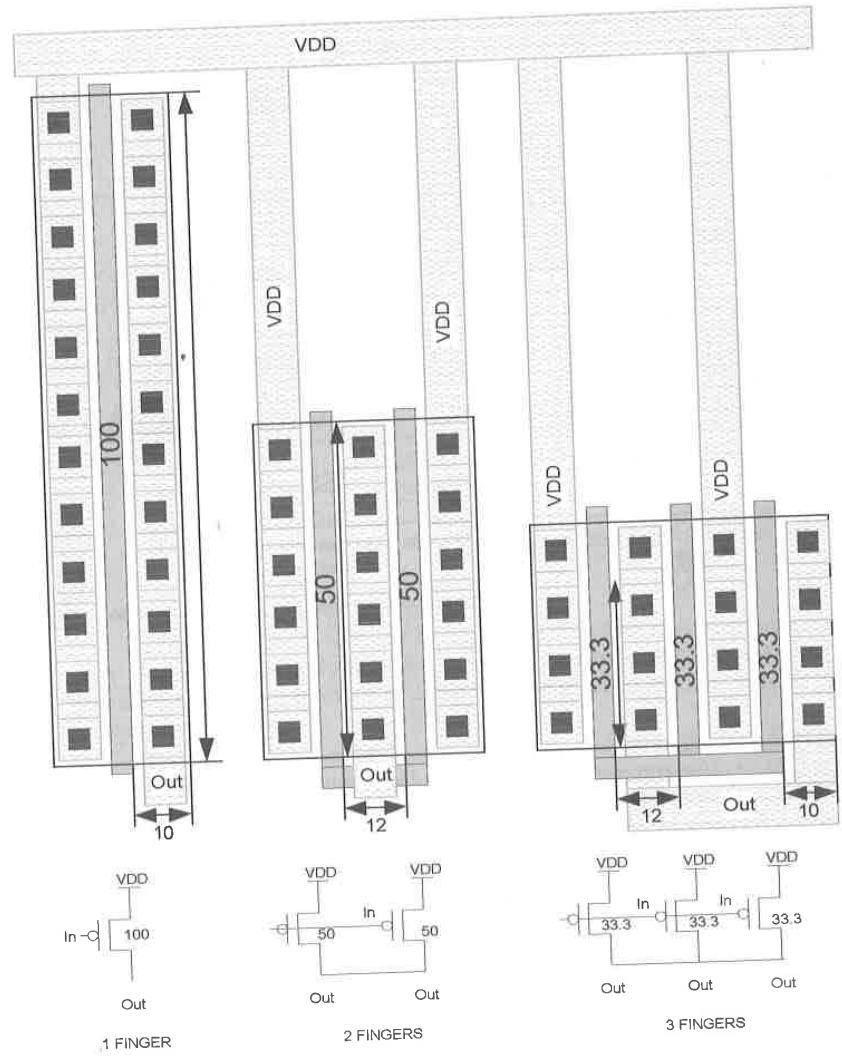


Figure 3.28 Fingering of transistors.

shared because they are very common and easy to connect. Very significant area savings can be achieved.

The main reason that area is saved is that both sides of a row of contacts are used and there is no need to space two active regions apart from each other. Figure 3.30 illustrates this technique.

Note that power nodes can be shared between transistors of different widths with a slight overhead of inserting a poly to active space at the end of the smaller transistor.

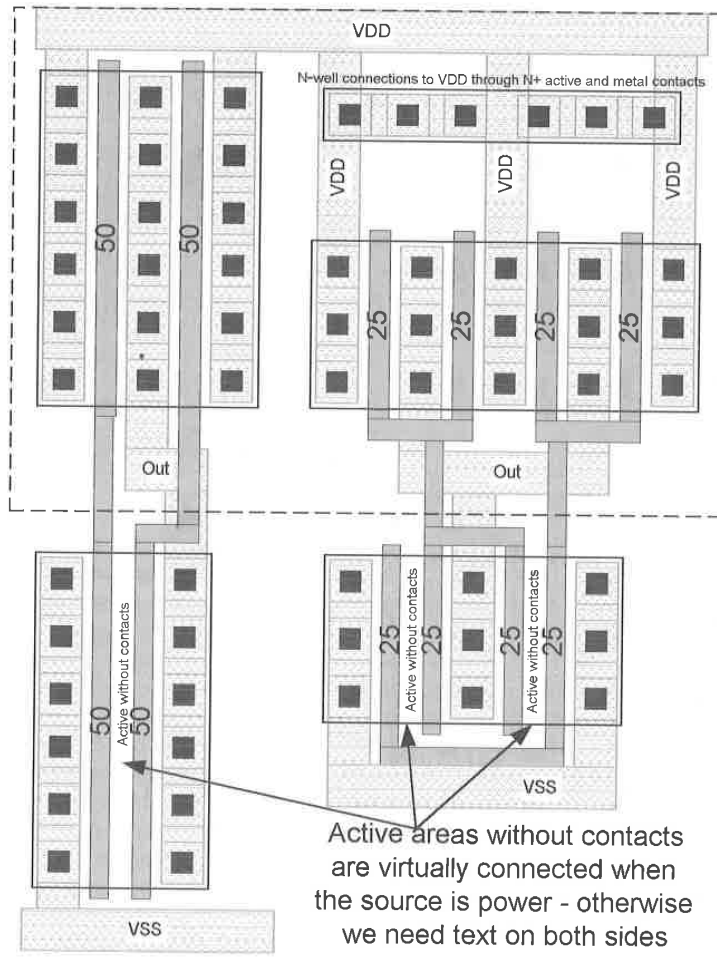


Figure 3.29 Example of fingering a NAND gate.

- Determine minimum number of contacts for source and drain connections. One simple rule might be to fit as many as you can using the minimum design rule between two contacts. This guideline is most reliable and maximizes the performance of the transistor. The downside of this approach is that the routability over the transistor is limited.

  If increased routability is required and accounted for in the circuit design, fewer than the maximum number of contacts may be optimal for the overall layout design. This approach must be considered carefully and accounted for in the circuit design process.

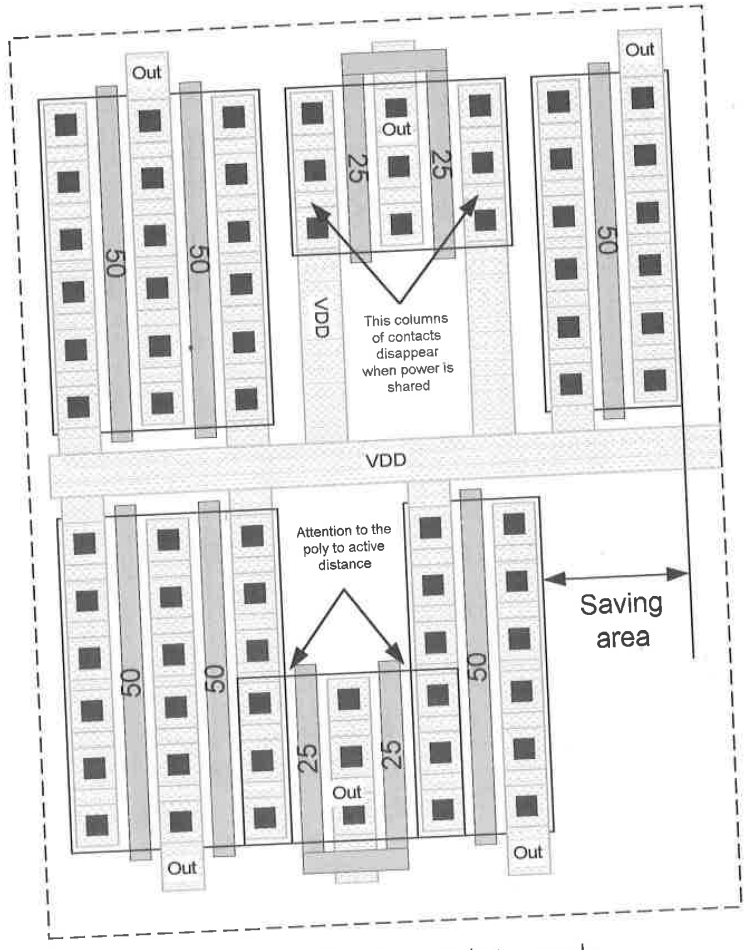


Figure 3.30 Transistor power sharing example.

In general, ASIC cell libraries use a minimum number of contacts within the cells, but for some high-frequency designs or analog parts, the transistors are fully contacted.

- Use 90-degree polygons or paths whenever possible. Most designs are done this way. The reason is that with orthogonal shapes the computer has to store a minimum amount of data and the layout process is much easier.

You should reserve 45-degree layout design for areas that have tight area and performance constraints. The reason is that 45-degree layout is more difficult to modify and maintain. The extra effort to use 45-degree layout design techniques is not worthwhile for the average layout cell.

Examples where 45-degree layout design is worthwhile are for memory cell and pitch limited layout, as well as datapath and large power supply

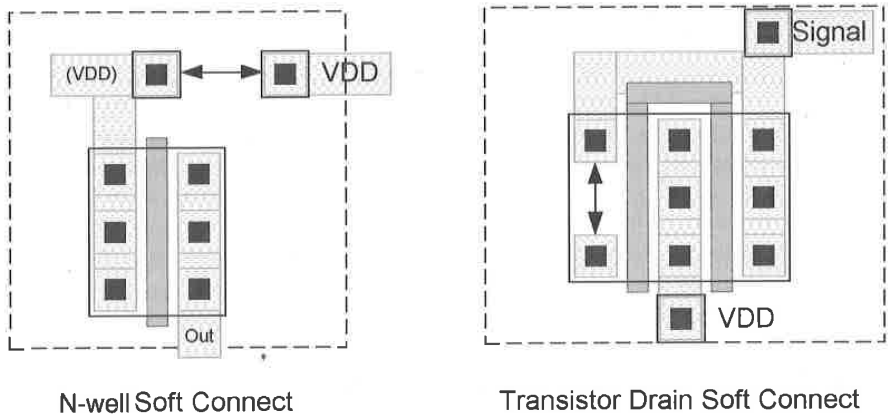


Figure 3.31 Example of soft connections.

lines. However, 45-degree contacts and vias cause many problems for many CAD programs, so they are to be avoided.

- Plan for and standardize tub and substrate connection locations. Plan and place connections of the N-well to the logical "1" power (VDD, VCC, VPP) and the P+ substrate to ground or VSS.

  The floorplan of the cell should include the general area where these contacts are to be placed. There are two basic methodologies: to place them between PMOS and NMOS transistors, or to place them on the outside of the transistor region. Between the transistors is better for latch-up protection (to be discussed later), but this complicates a cell layout.

- Avoid "soft connected" nodes. A "soft connected" node is one that has been connected through a nonrouting layer (Figure 3.31). Nonrouting layers are usually identified as such because they are highly resistive and result in poor circuit performance.

  Typically, active and N-well layers are not routing layers, but it is still possible to inadvertently use these layers to make electrical connections. The problem with this is that the DRC and LVS will pass, but the circuit performance will be poor. Only a very detailed layout extraction and simulation will find this type of "soft" error. Typically, this type of work is not practical, so a correct-by-construction approach is taken to avoid this effect.

  Figure 3.31 shows two examples of soft connections. The N-well example shows how a transistor is electrically connected to VDD, but the signal path flows through N-well as part of the connection. The second example shows how the transistor performance may be compromised by a connection to the drain that is not completed in metal. In this case the single contact does not help in any way, and an equivalent layout would be one without it.

  Special checks built into the layout verification process can help to identify these problems; however, they are difficult to debug, and it is best to simply avoid making the error in the first place.

3.8.4 Guidelines for the Layout of a Hierarchical Design

- Develop and use a floorplan plan. This cannot be emphasized enough, and it should be done at all levels of layout design.
- Define the hierarchy of a design in the planning stage. There are no hard-and-fast rules for defining the hierarchy of a design, but common sense is hard to beat. Common guidelines for determining different levels of hierarchy include the following:

  Circuits that are to be instantiated many times need to be cells

  Divide designs into functional or area-specific blocks

  Divide designs into blocks that allow multiple designers in parallel to work on them

  If symmetrical layout is desired, use a single half cell and mirror it to complete the design

  The use of hierarchy is discussed in many areas throughout the book.
- Develop and obey standards for layout near the boundary of a cell. The floorplan and the type of cell that is being implemented should define how the cell should interface to its neighbors. The interface requirements of any design should be known and understood in the planning stage.

  Here are some techniques and guidelines to properly design the interface for a cell:

  Use template cells to define global characteristics—cell dimensions and the placement of power supplies and wells are good candidates to define the interface of the cell. Consistent use of templates ensures that all cells conform to a standard and will integrate together smoothly.

  Assume that a boundary interface is fixed—if any polygon is required to cross the boundary of the cell, then the floorplan at a higher level in the design needs to be consulted before it is allowed. This avoids overlapping polygons with those that are unseen.

  Half design rule approach—if the cell is to abut to itself or other cells with similar boundary conditions, then a correct-by-construction approach would be to ensure that all internal polygons are spaced away from the boundary by a value that is half the specific design rule. In this way, when the cell abuts to another cell, spacing rules will not be violated.

  Verify the cell with its neighbors—this technique guarantees that the cell is correct in all cases.

3.8.5 Quality Metrics

Once the essential requirements have been met and/or we become experienced in doing layout, the following list outlines more subtle topics and metrics that will be covered later in the book. We only mention them here to introduce the topics so that we can anticipate and potentially plan ahead for these advanced requirements:

- Area
- Performance
- Porosity

- Manufacturability (i.e., is it all minimum design rules? If not, then it could be considered more manufacturable)
- Maintainability (i.e., will the layout be easy to change or optimize if the process changes?)
- Reliability over the long term (i.e., electromigration)
- Interface compatibility (i.e., does it abut and fit well in all instantiations?)
- Shrinkability (i.e., does the layout lend it self to future shrinks of the process?)
- Reuseability (i.e., does the layout lend itself to migration or retargeting to different processes?)
- Compatibility to layout flow (i.e., is the layout friendly for all downstream tools and methodologies such as P&R?)

3.9 IMPLEMENTING THE DESIGN

Let us now put all of the knowledge we have learned so far to work. At this point we have all of the fundamental knowledge to complete a basic layout design.

First, let's review the key concepts presented so far (Table 3.1). If you have mastered these fundamental concepts, you should be able to tackle almost any layout task.

We have learned that before we start any layout design, we must make a plan. Before we go ahead and execute this plan, it is always important to anticipate what the next steps are and keep these in mind as we implement our design. In this way we minimize or eliminate the amount of rework caused by our own ignorance.

Component placement (Step 2.1 from Figure 3.32) based on the floorplan is another area where we will always achieve a good return on the effort invested in doing a proper placement of components. The ability for the design to be completely routed is usually limited by the placement of components. These components can be other transistors, tub contacts, power supply lines, or interface

TABLE 3.1 Concept Review

Chapter	*Concept Summary*	*Comment*
1	What is layout design?	Also covers how layout design fits in the IC design flow
2	How do I read a schematic?	A schematic has more information than meets the eye!
3.2	Layer definition	An introduction to CMOS processes
3.3	CMOS transistor layout	A basic introduction
3.4	Design rules	These define the limits of what you are able to do
3.5	Layer connectivity	This defines what can be connected to what
3.6	A procedure to follow	General instructions
3.7	Developing a plan	Potentially the most important step for success!
3.8	General guidelines	Concepts to follow to do it right the first time