Physical Design Automation

INTRODUCTION

- The process of converting the specification of an electrical circuit into a layout is called the physical design process.
  - physical design is an extremely tedious and error prone process.
- Objective
  - The objective is to investigate optimal arrangements of devices on a plane (or in three dimensions) and efficient interconnection schemes between these devices to obtain the desired functionality and performance.
  - algorithms must use the space very efficiently to lower costs and improve yield.
  - In addition, the arrangement of devices plays a key role in determining the performance of a chip.

NEW TRENDS

- New Trends in VLSI Design Cycle
  - Increasing interconnect delay
  - Increasing interconnect area
  - Increasing number of metal layers
  - Increasing planning requirements
  - Synthesis
    - Logic Synthesis
    - High Level Synthesis

PHYSICAL DESIGN CYCLE

- Partitioning
- Floorplanning and Placement
- Routing
  - Global Routing
  - Detailed Routing
- Compaction
- Extraction and Verification

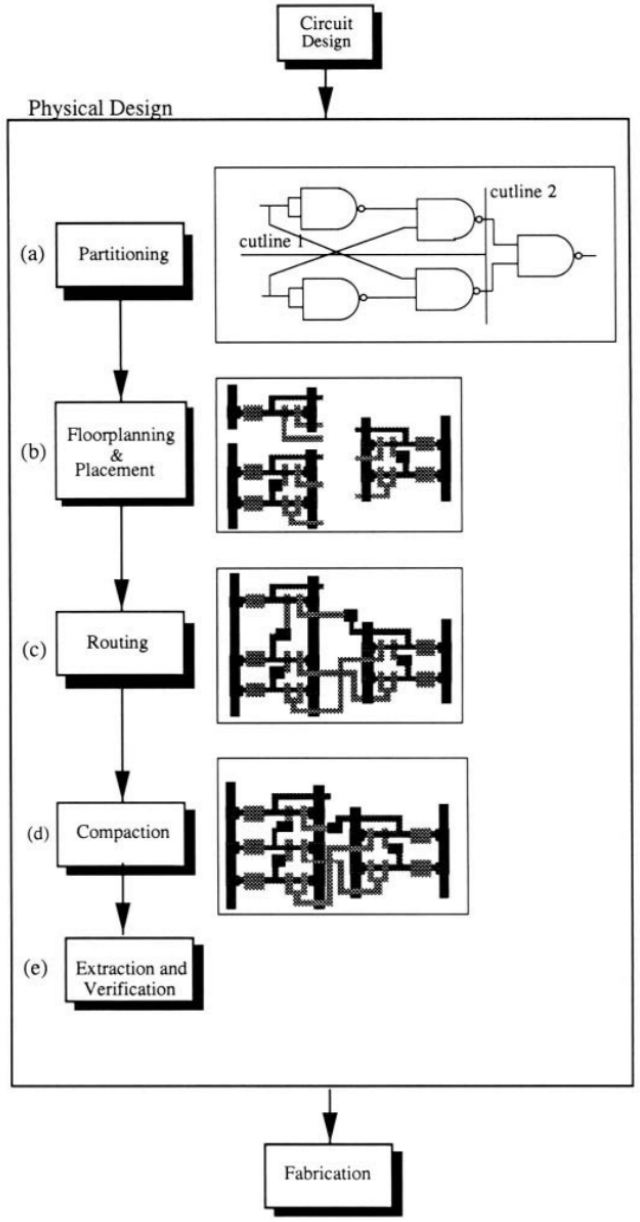


Figure 1.3: Physical design cycle.

NEW TRENDS

Interconnect delay is not scaling at the same rate as the gate delay.

- Chip level signal planning
  - routing of major signals and buses must be planned from early design stages, so that interconnect distances can be minimized.
- OTC routing
  - Over-the-Cell (OTC) routing is a term used to describe routing over blocks and active areas.
  - The OTC routing approach essentially makes routing a three dimensional problem.

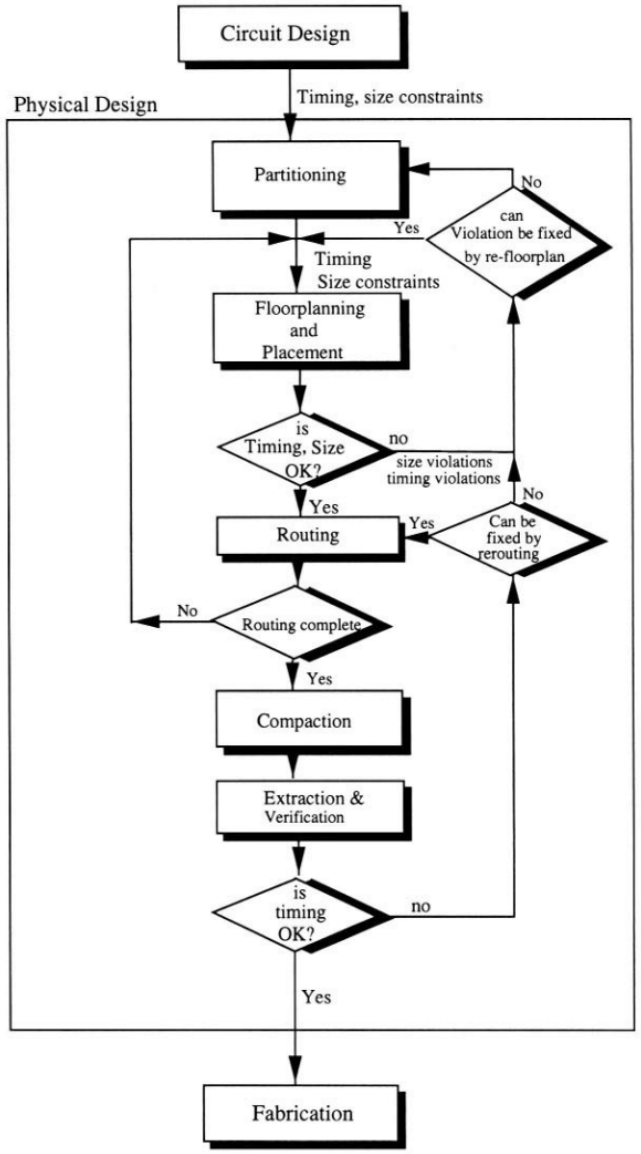


Figure 1.4: New physical design cycle.

DESIGN STYLES

- The design styles can be broadly classified as either full-custom or semi-custom.
  - In a full-custom layout, different blocks of a circuit can be placed at any location on a silicon wafer as long as all the blocks are non-overlapping.
  - On the other hand, in semi-custom layout, some parts of a circuit are predesigned and placed on some specific place on the silicon wafer.
  - Selection of a layout style depends on many factors including the type of chip, cost, and time-to-market.
  - On a large chip, each block may use a different layout design style.

DESIGN STYLES

- Full-Custom
- Standard Cell
- Gate Arrays
- Field Programmable Gate Arrays

FULL-CUSTOM

- The process is done hierarchically and thus full-custom designs have several levels of hierarchy. The chip is organized in clusters, clusters consist of units, and units are composed of functional blocks.
- The full-custom design style allows functional blocks to be of any size.
-

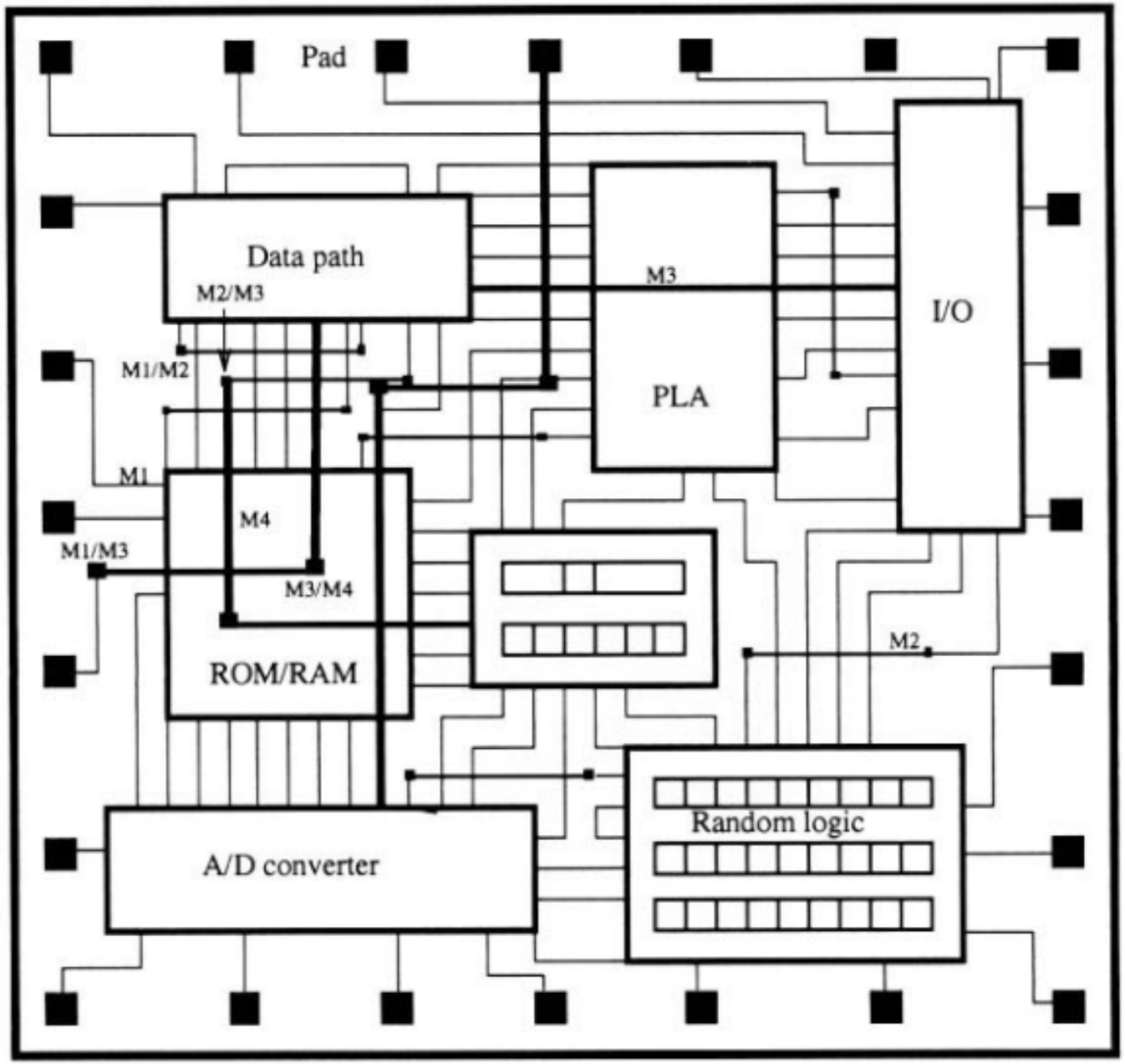


Figure 1.5: Full-custom structure

FULL-CUSTOM

- The process of automating a full-custom design style has a much higher complexity than other restricted models.
- Layout compaction is a very important aspect in full-custom design.
- The rectangular solid boxes around the boundary of the circuit are called I/O pads. Pads are used to complete interconnections between different chips or interconnections between the chip and the board.

STANDARD CELL

- Somewhat simpler than full-custom design style.
- Standard cell architecture considers the layout to consist of rectangular cells of the same height. Initially, a circuit is partitioned into several smaller blocks, each of which is equivalent to some predefined subcircuit (cell).
-

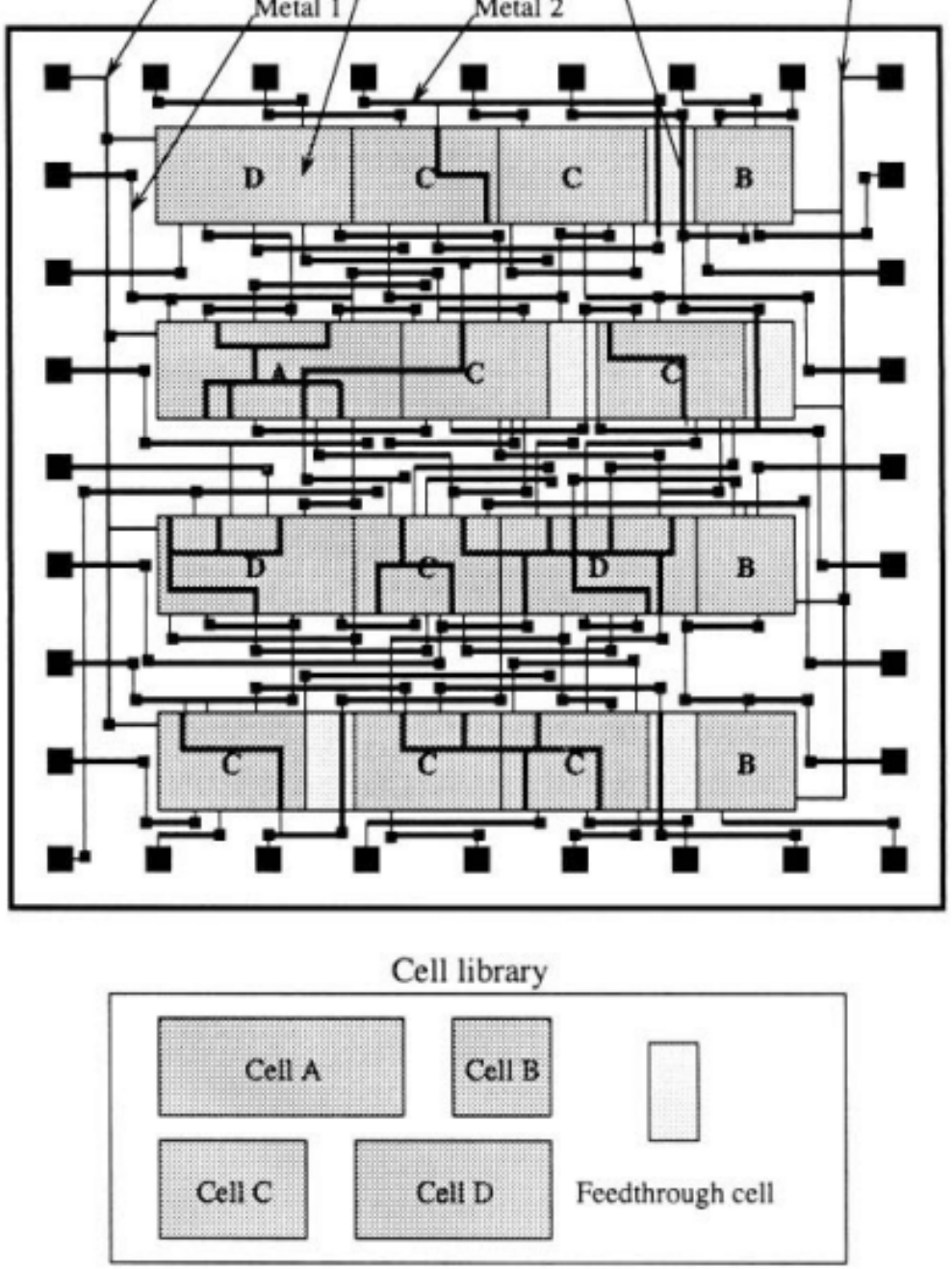
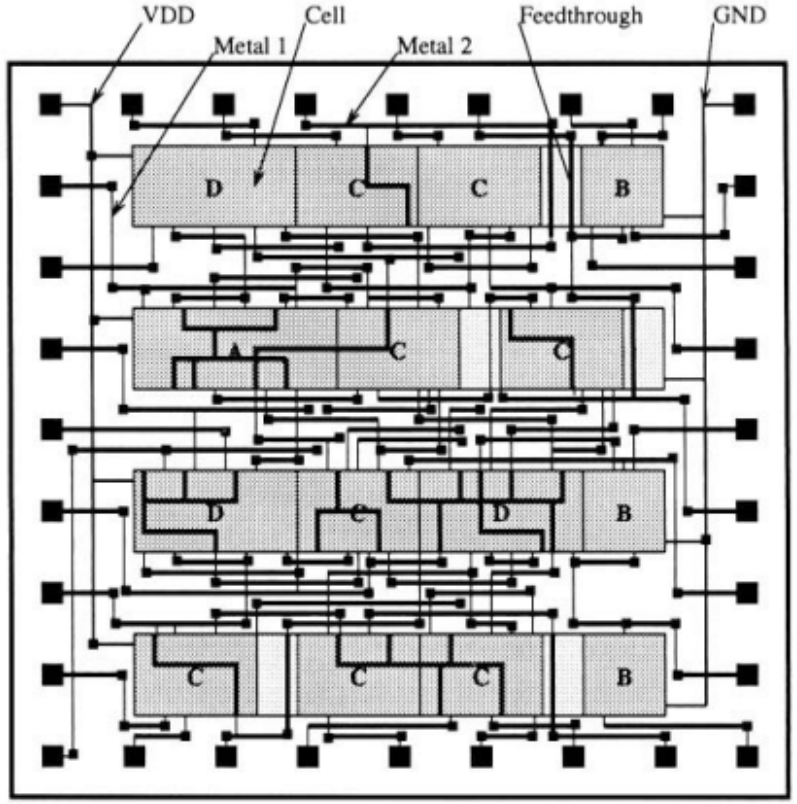


Figure 1.6: Standard cell structure.

STANDARD CELL

- A collection of these cells is called a cell library.
- Representation of a layout in the standard cell design style is greatly simplified as it is not necessary to duplicate the cell information.
- Cell library development is a significant project with enormous manpower and financial resource requirements.
- A standard cell design usually takes more area than a full-custom or a handcrafted design.



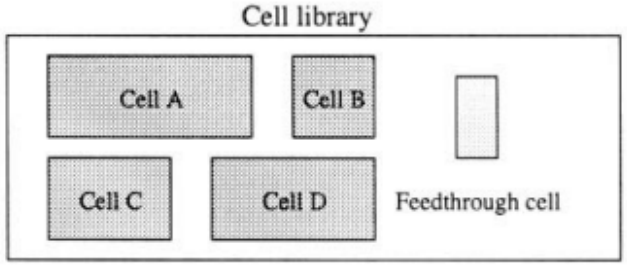


Figure 1.6: Standard cell structure.

REFERENCE

- Sherwani
  - 1.1
  - 1.2
  - 1.3
  - 1.4

Design Cycle-F2 (Slide 4)

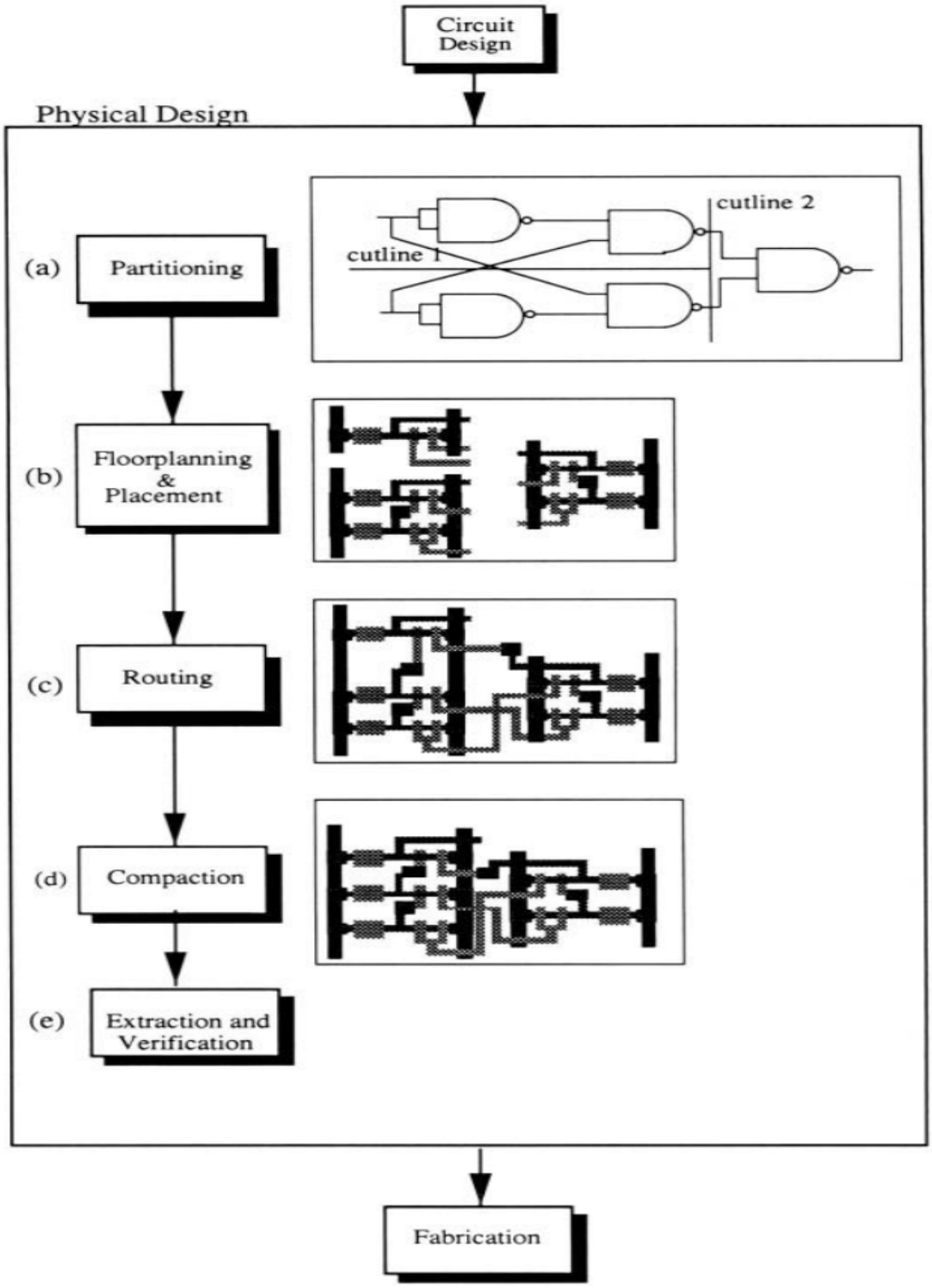


Figure 1.3: Physical design cycle.

Design Cycle-F2 (Slide 5)

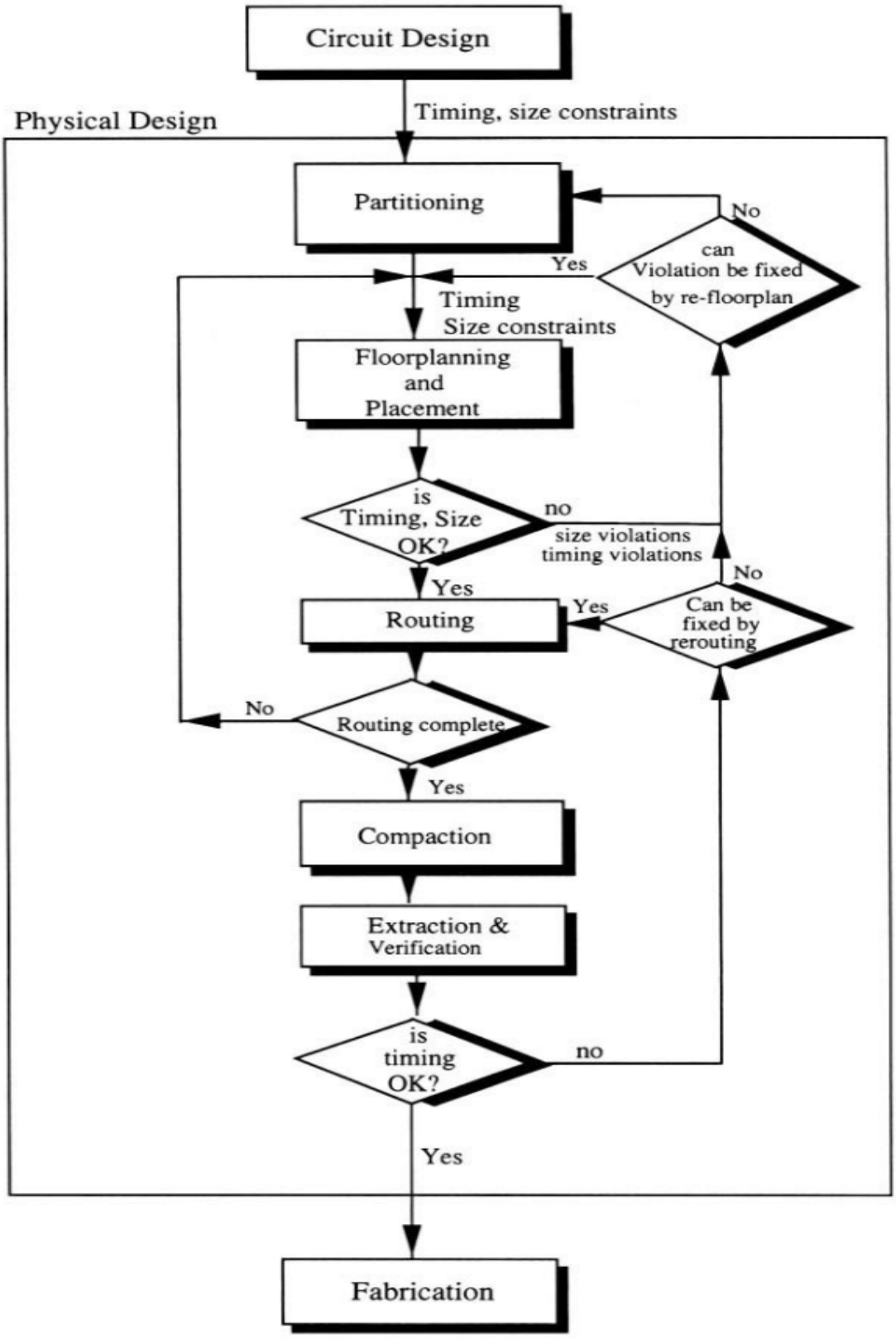


Figure 1.4: New physical design cycle.

2. **OTC routing:** Over-the-Cell (OTC) routing is a term used to describe routing over blocks and active areas. This is a departure from conventional channel and switchbox routing approach. Actually, chip level signal planning is OTC routing on the entire chip. The OTC approach can also be used within a block to reduce area and improve performance. The OTC routing approach essentially makes routing a three dimensional problem. Another effect of the OTC routing approach is that the pins are not brought to the block boundaries for connections to other blocks. Instead, pins are brought to the top of the block as a sea-of-pins. This concept, technically called the *Arbitrary Terminal Model* (ATM), will be discussed in a later chapter.

The conventional decomposition of physical design into partitioning, placement and routing phases is conceptually simple. However, it is increasingly clear that each phase is interdependent on other phases, and an integrated approach to partitioning, placement, and routing is required.

Figure 1.4 shows the physical design cycle with emphasis on timing. The figure shows that timing is estimated after floorplaning and placement, and these steps are iterated if some connections fail to meet the timing requirements. After the layout is complete, resistance and capacitance effects of one component on another can be extracted and accurate timing for each component can be calculated. If some connections or components fail to meet their timing requirements, or fail due to the effect of one component on another, then some or all phases of physical design need to be repeated. Typically, these 'repeat-or-not-to-repeat' decisions are made by experts rather than tools. This is due to the complex nature of these decisions, as they depend on a host of parameters.

1.5 Design Styles

Physical design is an extremely complex process. Even after breaking the entire process into several conceptually easier steps, it has been shown that each step is computationally very hard. However, market requirements demand quick time-to-market and high yield. As a result, restricted models and design styles are used in order to reduce the complexity of physical design. This practice began in the late 1960s and led to the development of several restricted design styles [Feu83]. The design styles can be broadly classified as either full-custom or semi-custom. In a full-custom layout, different blocks of a circuit can be placed at any location on a silicon wafer as long as all the blocks are non-overlapping. On the other hand, in semi-custom layout, some parts of a circuit are predesigned and placed on some specific place on the silicon wafer. Selection of a layout style depends on many factors including the type of chip, cost, and time-to-market. Full-custom layout is a preferred style for mass produced chips, since the time required to produce a highly optimized layout can be justified. On the other hand, to design an Application Specific Integrated Circuit (ASIC),

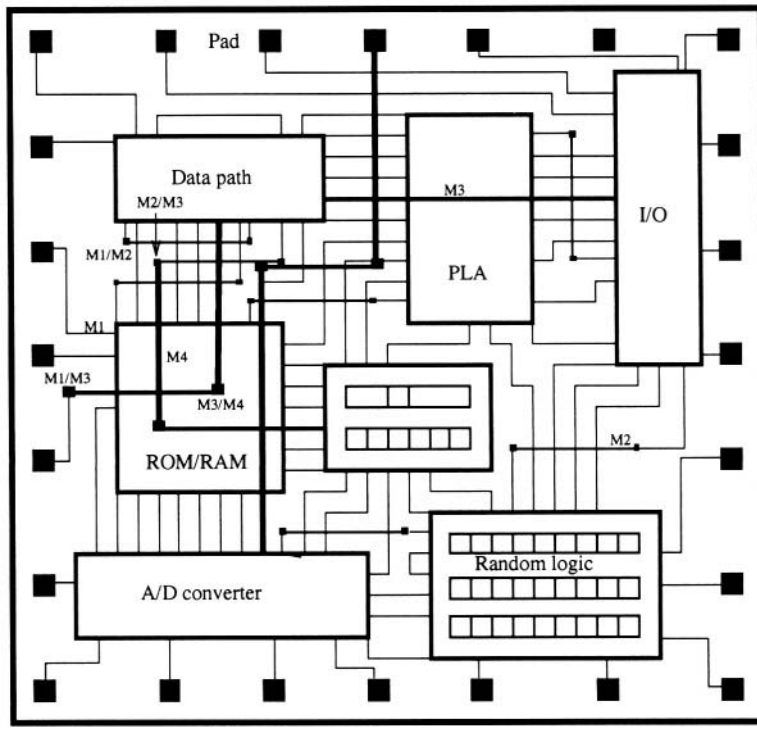


Figure 1.5: Full-custom structure.

a semi-custom layout style is usually preferred. On a large chip, each block may use a different layout design style.

1.5.1 Full-Custom

In its most general form of design style, the circuit is partitioned into a collection of sub-circuits according to some criteria such as functionality of each sub-circuit. The process is done hierarchically and thus full-custom designs have several levels of hierarchy. The chip is organized in clusters, clusters consist of units, and units are composed of *functional blocks* (in short, blocks). For sake of simplicity, we use the term blocks for units, blocks, and clusters. The full-custom design style allows functional blocks to be of any size. Figure 1.5 shows an example of a very simple circuit with few blocks. Other levels of hierarchy are not shown for this simple example. Internal routing in each block is not shown for the sake of clarity. In the full-custom design style, blocks can be placed at any location on the chip surface without any restrictions. In other words, this style is characterized by the absence of any constraints on the physical design process. This design style allows for very compact designs.

However, the process of automating a full-custom design style has a much higher complexity than other restricted models. For this reason it is used only when the final design must have minimum area and design time is less of a factor. The automation process for a full-custom layout is still a topic of intensive research. Some phases of physical design of a full-custom chip may be done manually to optimize the layout. Layout compaction is a very important aspect in full-custom design. The rectangular solid boxes around the boundary of the circuit are called *I/O pads.* Pads are used to complete interconnections between different chips or interconnections between the chip and the board. The spaces not occupied by blocks are used for routing of interconnecting wires. Initially all the blocks are placed within the chip area with the objective of minimizing the total area. However, there must be enough space left between the blocks so that routing can be completed using this space and the space on top of the blocks. Usually several metal layers are used for routing of interconnections. Currently, three metal layers are common for routing. A four metal layer process is being used for microprocessors, and a six layer process is gaining acceptance, as fabrication costs become more feasible. In Figure 1.5, note that width of the M1 wire is smaller than the width of the M2 wire. Also note that the size of the via between M1 and M2 is smaller than the size of the via between higher layers. Typically, metal widths and via sizes are larger for higher layers. The figure also shows that some routing has been completed on top of the blocks. The routing area needed between the blocks is becoming smaller and smaller as more routing layers are used. This is due to the fact that more routing is done on top of the transistors in the additional metal layers. If all the routing can be done on top of the transistors, the total chip area is determined by the area of the transistors. However, as circuits become more complex and interconnect requirements increase, the die size is determined by the interconnect area and the total transistor area serves as a lower bound on the die size of the chip.

In a hierarchical design of a circuit, each block in a full-custom design may be very complex and may consist of several sub-blocks, which in turn may be designed using the full-custom design style or other design styles. It is easy to see that since any block is allowed to be placed anywhere on the chip, the problem of optimizing area and the interconnection of wires becomes difficult. Full custom design is very time consuming; thus the method is inappropriate for very large circuits, unless performance or chip size is of utmost importance. Full custom is usually used for the layout of microprocessors and other performance and cost sensitive designs.

1.5.2 Standard Cell

The design process in the standard cell design style is somewhat simpler than full-custom design style. Standard cell architecture considers the layout to consist of rectangular cells of the same height. Initially, a circuit is partitioned into several smaller blocks, each of which is equivalent to some predefined subcircuit (cell). The functionality and the electrical characteristics of each

predefined cell are tested, analyzed, and specified. A collection of these cells is called a *cell library.* Usually a cell library consists of 500-1200 cells. Terminals on cells may be located either on the boundary or distributed throughout the cell area. Cells are placed in rows and the space between two rows is called a *channel.* These channels and the space above and between cells is used to perform interconnections between cells. If two cells to be interconnected lie in the same row or in adjacent rows, then the channel between the rows is used for interconnection. However, if two cells to be connected lie in two non-adjacent rows, then their interconnection wire passes through empty space between any two cells or passes on top of the cells. This empty space between cells in a row is called a *feedthrough.* The interconnections are done in two steps. In the first step, the feedthroughs are assigned for the interconnections of non-adjacent cells. Feedthrough assignment is followed by routing. The cells typically use only one metal layer for connections inside the cells. As a result, in a two metal process, the second metal layer can be used for routing in over-the-cell regions. In a three metal layer process, almost all the channels can be removed and all routing can be completed over the cells. However, this is a function of the density of cells and distribution of pins on the cells. It is difficult to obtain a channelless layout for chips which use highly packed dense cells with poor pin distribution. Figure 1.6 shows an example of a standard cell layout. A cell library is shown, along with the complete circuit with all the interconnections, feedthroughs, and power and ground routing. In the figure, the library consists of four logic cells and one feedthrough cell. The layout shown consists of several instances of cells in the library. Note that representation of a layout in the standard cell design style is greatly simplified as it is not necessary to duplicate the cell information.

The standard cell layout is inherently non-hierarchical. The hierarchical circuits, therefore, have to undergo some transformation before this design style can be used. This design style is well-suited for moderate size circuits and medium production volumes. Physical design using standard cells is somewhat simpler as compared to full-custom, and is efficient using modern design tools. The standard cell design style is also widely used to implement the 'random or control logic' part of the full-custom design as shown in Figure 1.5.

Logic Synthesis usually uses the standard cell design style. The synthesized circuit is mapped to cell circuits. Then cells are placed and routed.

While standard cell designs are quicker to develop, a substantial initial investment is needed in the development of the cell library, which may consist of several hundred cells. Each cell in the cell library is 'hand crafted' and requires highly skilled physical design specialists. Each type of cell must be created with several transistor sizes. Each cell must then be tested by simulation and its performance must be characterized. Cell library development is a significant project with enormous manpower and financial resource requirements.

A standard cell design usually takes more area than a full-custom or a hand-crafted design. However, as more and more metal layers become available for routing and design tools improve, the difference in area between the two design styles will gradually reduce.

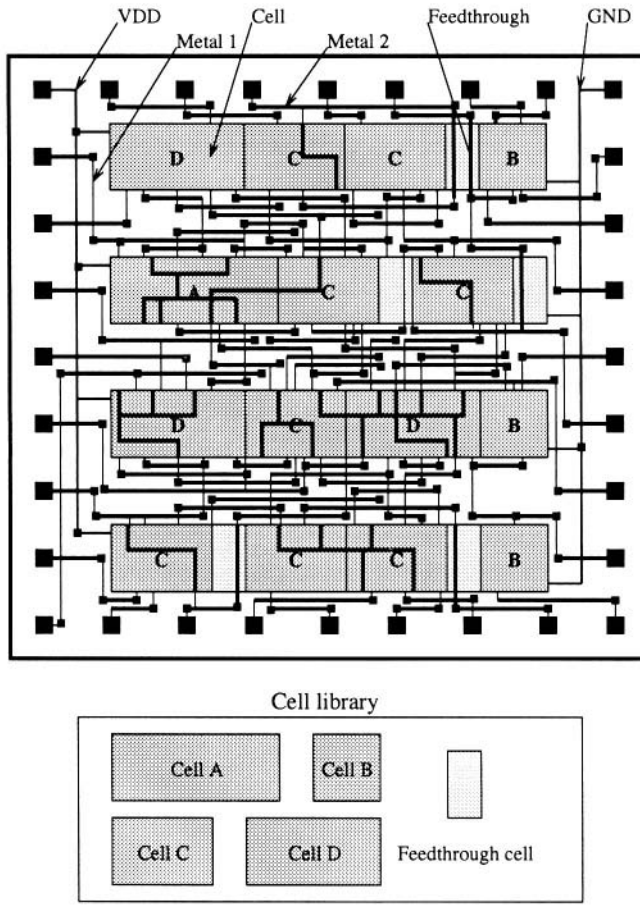


Figure 1.6: Standard cell structure.

1.5.3 Gate Arrays

This design style is a simplification of standard cell design. Unlike standard cell design, all the cells in gate array are identical. Each chip is an array of identical gates or cells. These cells are separated by both vertical and horizontal spaces called vertical and horizontal channels. The circuit design is modified such that it can be partitioned into a number of identical blocks. Each block must be logically equivalent to a cell on the gate array. The name 'gate array' signifies the fact that each cell may simply be a gate, such as a three input NAND gate. Each block in design is mapped or placed onto a prefabricated cell on the chip during the partitioning/placement phase, which is reduced to a block to cell assignment problem. The number of partitioned blocks must be less than or equal to the total number of cells on the chip. Once the circuit

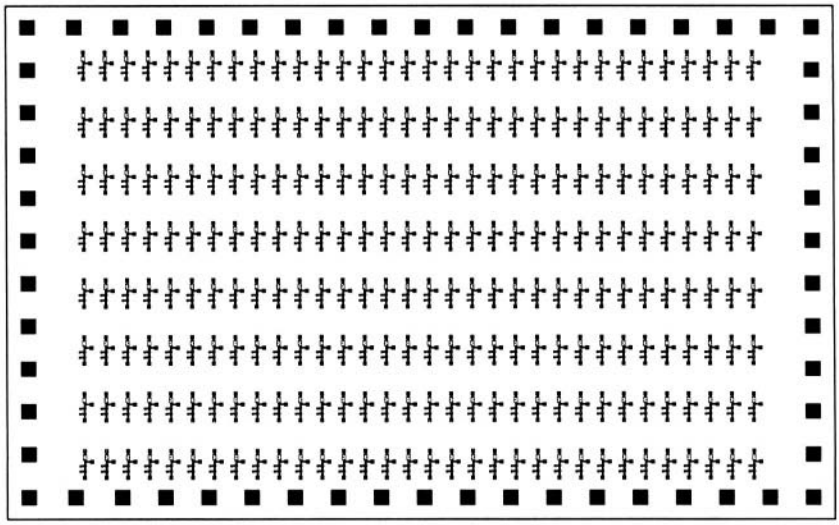


Figure 1.7: A conceptual uncommitted gate array.

is partitioned into identical blocks, the task is to make the interconnections between the prefabricated cells on the chip using horizontal and vertical channels to form the actual circuit. Figure 1.7 shows an 'uncommitted' gate array, which is simply a term used for a prefabricated chip. The gate array wafer is taken into a fabrication facility and routing layers are fabricated on top of the wafer. The completed wafer is also called a 'customized wafer'. It should be noted that the number of tracks allowed for routing in each channel is fixed. As a result, the purpose of the routing phase is simply to complete the connections rather than minimize the area. Two layers of interconnections are most common; though one and three layers are also used. Figure 1.8 illustrates a committed gate array design. Like standard cell designs, synthesis can also use the gate array style. In gate array design the entire wafer, consisting of several dozen chips, is prefabricated.

This simplicity of gate array design is gained at the cost of rigidity imposed upon the circuit both by the technology and the prefabricated wafers. The advantage of gate arrays is that the steps involved for creating any prefabricated wafer are the same and only the last few steps in the fabrication process actually depend on the application for which the design will be used. Hence gate arrays are cheaper and easier to produce than full-custom or standard cell. Similar to standard cell design, gate array is also a non-hierarchical structure.

The gate array architecture is the most restricted form of layout. This also means that it is the simplest for algorithms to work with. For example, the task of routing in gate array is to determine if a given placement is routable. The routability problem is conceptually simpler as compared to the routing

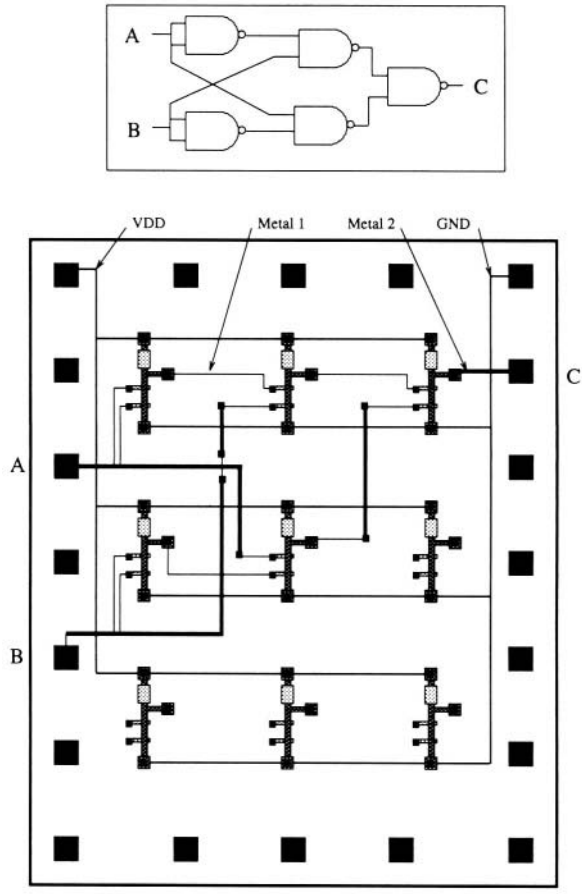
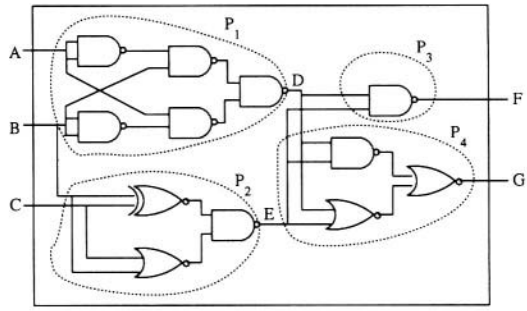


Figure 1.8: A conceptual gate array.

problem in standard cell and full-custom design styles.

1.5.4 Field Programmable Gate Arrays

The Field Programmable Gate Array (FPGA) is a new approach to ASIC design that can dramatically reduce manufacturing turn-around time and cost for low volume manufacturing [Gam89, Hse88, Won89]. In FPGAs, cells and interconnect are prefabricated. The user simply 'programs' the interconnect. FPGA designs provide large scale integration and user programmability. A FPGA consists of horizontal rows of programmable logic blocks which can be interconnected by a programmable routing network. FPGA cells are more complex than standard cells. However, almost all the cells have the same layout. In its simplistic form, a logic block is simply a memory block which can be pro-



(a)

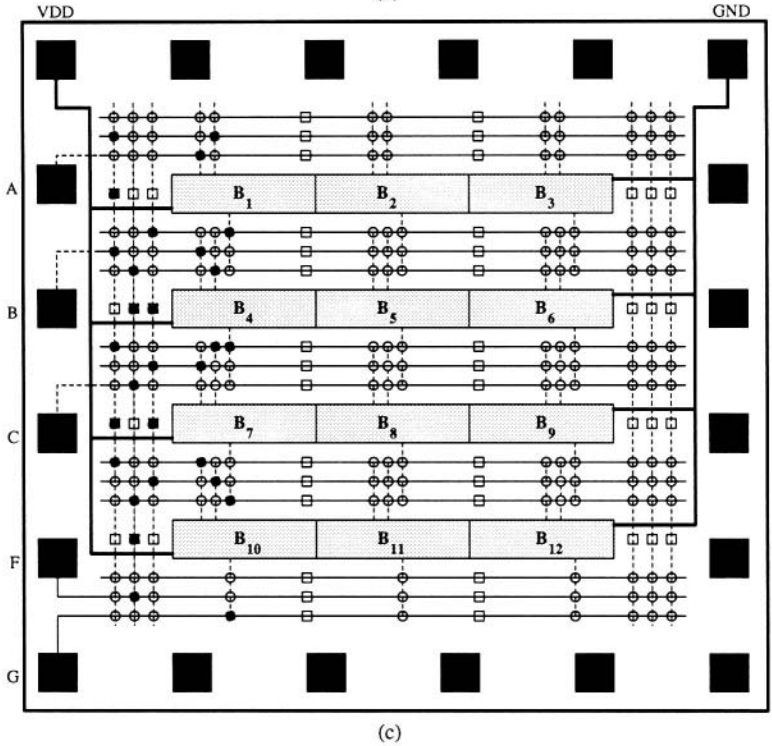
P1		
A	B	D
0	0	0
0	1	1
1	0	1
1	1	0

P2		
B	C	E
0	0	0
0	1	1
1	0	1
1	1	1

P3		
D	E	F
0	0	1
0	1	1
1	0	1
1	1	0

P4		
D	E	G
0	0	0
0	1	0
1	0	0
1	1	1

(b)



(c)

Figure 1.9: A committed FPGA.

grammed to remember the logic table of a function. Given a certain input, the logic block 'looks up' the corresponding output from the logic table and sets its output line accordingly. Thus by loading different look-up tables, a logic block can be programmed to perform different functions. It is clear that 2^K bits are required in a logic block to represent a K-bit input, 1-bit output combinational logic function. Obviously, logic blocks are only feasible for small values of K. Typically, the value of K is 5 or 6. For multiple outputs and sequential circuits the value of K is even less. The rows of logic blocks are separated by horizontal routing channels. The channels are not simply empty areas in which metal lines can be arranged for a specific design. Rather, they contain predefined wiring 'segments' of fixed lengths. Each input and output of a logic block is connected to a dedicated vertical segment. Other vertical segments merely pass through the blocks, serving as feedthroughs between channels. Connection between horizontal segments is provided through *antifuses,* whereas the connection between a horizontal segment and a vertical segment is provided through a *cross fuse.* Figure 1.9(c) shows the general architecture of a FPGA, which consists of four rows of logic blocks. The cross fuses are shown as circles, while antifuses are shown as rectangles. One disadvantage of fuse based FPGAs is that they are not reprogrammable. There are other types of FPGAs which allow re-programming, and use pass gates rather than programmable fuses.

Since there are no user specific fabrication steps in a FPGA, the fabrication process can be set up in a cost effective manner to produce large quantities of generic (unprogrammed) FPGAs. The customization (programming) of a FPGA is rather simple. Given a circuit, it is decomposed into smaller subcircuits, such that each subcircuit can be mapped to a logic block. The interconnections between any two subcircuits is achieved by programming the FPGA interconnects between their corresponding logic blocks. Programming (blowing) one of the fuses (antifuse or cross fuse) provides a low resistance bidirectional connection between two segments. When blown, antifuses connect the two segments to form a longer one. In order to program a fuse, a high voltage is applied across it. FPGAs have special circuitry to program the fuses. The circuitry consists of the wiring segments and control logic at the periphery of the chip. Fuse addresses are shifted into the fuse programming circuitry serially. Figure 1.9(a) shows a circuit partitioned into four subcircuits, P_1, P_2, P_3, and P_4. Note that each of these four subcircuits have two inputs and one output. The truth table for each of the subcircuits is shown in Figure 1.9(b). In Figure 1.9(c), P_1, P_2, P_3, and P_4 are mapped to logic blocks B_1, B_4, B_7, and B_{10} respectively and appropriate antifuses and cross fuses are programmed (burnt) to implement the entire circuit. The programmed fuses are shown as filled circles and rectangles. We have described the 'once-program' type of FPGAs. Many FPGAs allow the user to re-program the interconnect, as many times as needed. These FPGAs use non-destructive methods of programming, such as pass-transistors.

The programmable nature of these FPGAs requires new CAD algorithms to make effective use of logic and routing resources. The problems involved in customization of a FPGA are somewhat different from those of other design

styles; however, many steps are common. For example, the partition problem of FPGAs is different than partitioning the problem in all design style while the placement and the routing is similar to gate array approach. These problems will be discussed in detail in Chapter 11.

1.5.5 Sea of Gates

The sea of gates is an improved gate array in which the master is filled completely with transistors. The master of the sea-of-gates has a much higher density of logic implemented on the chip, and allows a designer to fabricate complex circuits, such as RAMs, to be built. In the absence of routing channels, interconnects have to be completed either by routing through gates, or by adding more metal or polysilicon interconnection layers. There are problems associated with either solution. The former reduces the gate utilization; the latter increases the mask count and increases fabrication time and cost.

1.5.6 Comparison of Different Design Styles

The choice of design style depends on the intended functionality of the chip, time-to-market and total number of chips to be manufactured. It is common to use full-custom design style for microprocessors and other complex high volume applications, while FPGAs may be used for simple and low volume applications. However, there are several chips which have been manufactured by using a mix of design styles. For large circuits, it is common to partition the circuit into several small circuits which are then designed by different teams. Each team may use a different design style or a number of design styles. Another factor complicating the issue of design style is re-usability of existing designs. It is a common practice to re-use complete or partial layout from existing chips for new chips to reduce the cost of a new design. It is quite typical to use standard cell and gate array design styles for smaller and less complex Application Specific ICs (ASICs), while microprocessors are typically full-custom with several standard cell blocks. Standard cell blocks can be laid out using logic synthesis tools.

Design styles can be seen as a continuum from very flexible (full-custom) to a rather rigid design style (FPGA) to cater to differing needs. Table 1.1 summarizes the differences in cell size, cell type, cell placement and interconnections in full-custom, standard cell, gate array and FPGA design styles. Another comparison may be on the basis of area, performance, and the number of fabrication layers needed. (See Table 1.2). As can be seen from the table, full-custom provides compact layouts for high performance designs but requires a considerable fabrication effort. On the other hand, a FPGA is completely pre-fabricated and does not require any user specific fabrication steps. However, FPGAs can only be used for small, general purpose designs.

	style			
	full-custom	standard cell	gate array	FPGA
cell size	variable	fixed height*	fixed	fixed
cell type	variable	variable	fixed	programmable
cell placement	variable	in row	fixed	fixed
interconnections	variable	variable	variable	programmable
design cost	high	medium	medium	low

Table 1.1: Comparison of different design styles.
* uneven height cells are also used.

	style			
	full-custom	standard cell	gate array	FPGA
Area	compact	compact to moderate	moderate	large
Performance	high	high to moderate	moderate	low
Fabricate	All layers	All layers	Routing layers only	No layers

Table 1.2: Area, Performance and Fabrication layers for different design styles.

1.6 System Packaging Styles

The increasing complexity and density of semiconductor devices are the key driving forces behind the development of more advanced VLSI packaging and interconnection approaches. Two key packaging technologies being used currently are Printed Circuit Boards (PCB) and Multi-Chip Modules (MCMs). Let us first start with die packaging techniques.

1.6.1 Die Packaging and Attachment Styles

Dies can be packaged in a variety of styles depending on cost, performance and area requirements. Other considerations include heat removal, testing and repair.

1.6.1.1 Die Package Styles

ICs are packaged into ceramic or plastic carriers called Dual In-Line Packages (DIPs), then mounted on a PCB. These packages have leads on 2.54 mm centers on two sides of a rectangular package. PGA (Pin Grid Array) is a package in which pins are organized in several concentric rectangular rows. DIPs and PGAs require large thru-holes to mount them on boards. As a result, thru-hole assemblies were replaced by Surface Mount Assemblies (SMAs). In SMA,