Picking the appropriate model for designing electronics is an art and is learned over an extended time.

fication, and design management. Such tools perform their specific tasks on the output of each of the design stages of Fig. 1.1. For example, to verify the outcome of the data path design stage, the bussing and register structure is fed into a simulation program. Also, to generate tests for register transfer faults, a design automation tool can be used for processing this level of system description and producing tests that can be used by a test engineer. Other DA tools include a synthesizer that can automatically generate a netlist from the register and bus structure of the system under design.

HDLs provide formats for representing the outputs of various design stages. An HDL-based DA tool for the analysis of a circuit uses this format for its input description, and a synthesis tool transforms its HDL input into an HDL which contains more hardware information. In the sections that follow, we discuss HDLs, digital system simulation, and hardware synthesis.

1.2 The Art of Modeling

Core Concept

The dictionary[1] defines *model* and *modeling* as follows:

mod-el (mŏd'l) *noun.* **1.** A small object, usually built to scale, that represents in detail another, often larger object. **2.a.** A preliminary work or construction that serves as a plan from which a final product is to be made. **b.** Such a work or construction used in testing or perfecting a final product. **3.** A schematic description of a system, theory, or phenomenon that accounts for its known or inferred properties and may be used for further study of its characteristics.

mod-el-ing (mŏd'l-ĭng) *noun.* **1.** The act or art of sculpturing or forming in a pliable material, such as clay or wax. **2.a.** Representation of depth and solidity in painting, drawing, or photography.

We can even go further in defining model and modeling from the point of view of a hardware designer and digital system design environments. Modeling is an art, and a modeler is one who uses certain modeling tools for representing an event, an object, or an idea as best as possible. The modeling tools, however, may be different from one art to another. For a painter, the modeling tools are the paintbrush, easel, paint, and the paint palette. In addition to being different in the way things are represented by various modelings, the level of details of representation in a model may be different from one model to another. For example one painting may represent a mountain at the

[1]*The American Heritage® Dictionary of the English Language, Third Edition,* copyright © 1992 by Houghton Mifflin Company.

detailed level of rocks and plantation, while another painting of the same mountain may represent it from a distant view showing its peak and the hills around it.

Models are used for different purposes. A painting may be used for decorating a room, while another type of a model, for instance a piece of music, may be used to express a political view or an event. Shostakovich wrote his Symphony No. 7, the Leningrad Symphony, about the siege of his native city of Leningrad (St. Petersburg) by German troops. The symphony models the spirit of this city during the war, life in this city under siege, and the eventual victory over the invaders. In this model, he employs a certain level of details and uses his general modeling schemes for doing so.

A hardware designer models a circuit using any available tools. The level of abstraction for this modeling depends on the purpose for which the model is intended. If the model is to be used for documenting the functionality of a circuit at a very abstract behavioral level, a relatively simple abstract model is all that is necessary. On the other hand, if the model is to be used for verification of the timing of the circuit, a more detailed description is needed. A hardware engineer models his or her circuit such that it imitates the actual hardware component as closely and accurately as possible for its intended purpose. A good modeler uses available hardware modeling tools to produce an elegant and artistic model of the hardware part.

Available modeling tools to a hardware engineer include paper and pencil, schematic capture programs, breadboarding facilities, and hardware description languages. The newest and the most promising of all such tools are hardware description languages. These modeling tools enable a hardware designer to model his or her circuit at many levels of abstractions for various design, analysis, and documentation purposes. Although all hardware description languages may be regarded as such, the level of model elegance and artistic representation of hardware may be different from one language to another.

1.3 Hardware Description Languages

HDLs are used to describe hardware for the purpose of simulation, modeling, testing, design, and documentation. These languages provide a convenient and compact format for the hierarchical representation of functional and wiring details of digital systems. Some HDLs consist of a simple set of symbols and notations that replace schematic diagrams of digital circuits, while others are more formally defined and may present the hardware at one or more levels of abstraction. Available software for HDLs include simulators and hardware synthe-

From "VHDL" 2nd ed.
by Z. Navabi

From “Analysis and Design of Analog ICs” by Gray et al

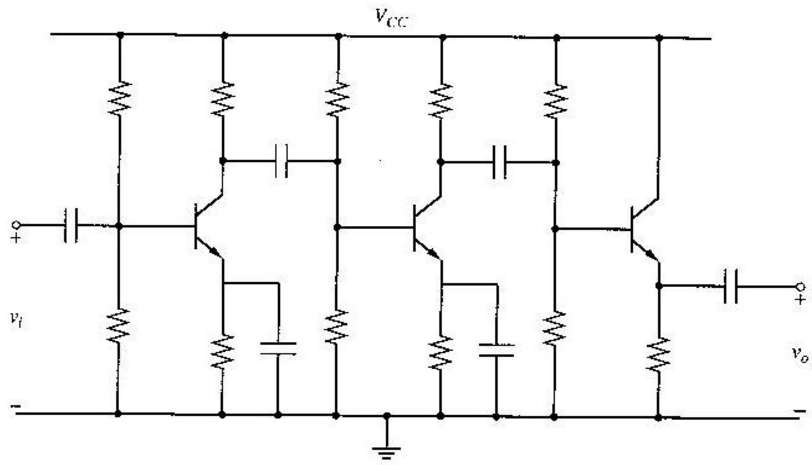

Figure 3.1 Typical discrete-component realization of an audio amplifier.

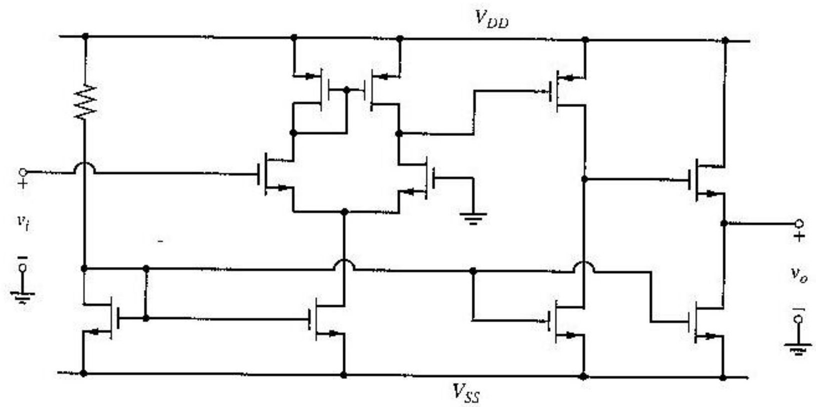

Figure 3.2 Typical CMOS integrated-circuit realization of an audio amplifier.

these multi-transistor circuits are the differential pairs, which are analyzed extensively in this chapter.

3.1 Device Model Selection for Approximate Analysis of Analog Circuits

Much of this book is concerned with the salient performance characteristics of a variety of subcircuits commonly used in analog circuits and of complete functional blocks made up of these subcircuits. The aspects of the performance that are of interest include the dc currents and voltages within the circuit, the effect of mismatches in device characteristics on these voltages and currents, the small-signal, low-frequency input and output resistance, and the voltage gain of the circuit. In later chapters, the high-frequency, small-signal behavior of circuits is considered. The subcircuit or circuit under investigation is often one of considerable complexity, and the most important single principle that must be followed to achieve success in the hand analysis of such circuits is *selecting the simplest possible* *model* for the devices within the circuit that will result in the required accuracy. For example, in the case of dc analysis, hand analysis of a complex circuit is greatly simplified by neglecting certain aspects of transistor behavior, such as the output resistance, which may result in a 10 to 20 percent error in the dc currents calculated. The principal objective of hand analysis, however, is to obtain an intuitive understanding of factors affecting circuit behavior so that an iterative design procedure resulting in improved performance can be carried out. The performance of the circuit can at any point in this cycle be determined precisely by computer simulation, but this approach does not yield the intuitive understanding necessary for design.

Unfortunately, no specific rules can be formulated regarding the selection of the simplest device model for analysis. For example, in the dc analysis of bipolar biasing circuits, assuming constant base-emitter voltages and neglecting transistor output resistances often provides adequate accuracy. However, certain bias circuits depend on the nonlinear relation between the collector current and base-emitter voltage to control the bias current, and the assumption of a constant V_{BE} will result in gross errors in the analyses of these circuits. When analyzing the active-load stages in Chapter 4, the output resistance must be considered to obtain meaningful results. Therefore, a key step in every analysis is to inspect the circuit to determine what aspects of the behavior of the transistors strongly affect the performance of the circuit, and then simplify the model(s) to include only those aspects. This step in the procedure is emphasized in this and the following chapters.

3.2 Two-Port Modeling of Amplifiers

The most basic parameter of an amplifier is its gain. Since amplifiers may be connected to a wide variety of sources and loads, predicting the dependence of the gain on the source and load resistance is also important. One way to observe this dependence is to include these resistances in the amplifier analysis. However, this approach requires a completely new amplifier analysis each time the source or load resistance is changed. To simplify this procedure, amplifiers are often modeled as two-port equivalent networks. As shown in Fig. 3.3, two-port networks have four terminals and four port variables (a voltage and a current at each port). A pair of terminals is a port if the current that flows into one terminal is equal to the current that flows out of the other terminal. To model an amplifier, one port represents the amplifier input characteristics and the other represents the output. One variable at each port can be set independently. The other variable at each port is dependent on the two-port network and the independent variables. This dependence is expressed by two equations. We will focus here on the admittance-parameter equations, where the terminal currents are viewed as dependent variables controlled by the independent terminal voltages because we usually model transistors with voltage-controlled current sources. If the network is linear and contains no independent sources, the admittance-parameter equations are:

$$i_1 = y_{11}v_1 + y_{12}v_2 \tag{3.1}$$

$$i_2 = y_{21}v_1 + y_{22}v_2 \tag{3.2}$$

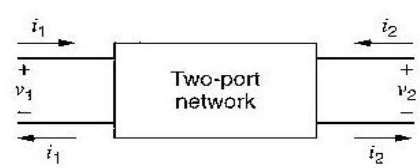

Figure 3.3 Two-port-network block diagram.