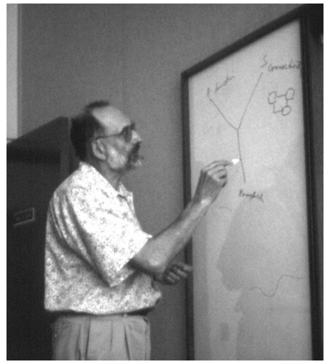
Foreword



Prof. Dr. Gajski demonstrates the Y-chart

When I attended college we studied vacuum tubes in our junior year. At that time an average radio had five vacuum tubes and better ones even seven. Then transistors appeared in 1960s. A good radio was judged to be one with more then ten transistors. Later good radios had 15–20 transistors and after that everyone stopped counting transistors. Today modern processors runing personal computers have over 10 million transistors and more millions will be added every year.

The difference between 20 and 20M is in complexity, methodology and business models. Designs with 20 transistors are easily generated by design engineers without any tools, whilst designs with 20 M transistors can not be done by humans in reasonable time without the help of automation. This difference in complexity introduced a paradigm shift which required sophisticated methods and tools, and introduced design automation into design practice.

By the decomposition of the design process into many tasks and abstraction levels the methodology of designing chips or systems has also evolved. Similarly, the business model has changed from vertical integration, in which one company did all the tasks from product specification to manufacturing, to globally distributed, client server production in which most of the design and manufacturing tasks are outsourced.

In general, the product creation process can be divided into several tasks including requirements gathering, specification generation, component selection, architectural exploration, component synthesis, physical design, verification, simulation, prototyping, and manufacturing. Furthermore, each product can be represented on different levels of abstraction defined by the complexity of components used in the design, such as transistors, gates, registers, processors, or general types of cores, or any other intellectual property. For each task and each level we have created many different methods and different tools, and thus have created the Electronic Design Automation (EDA) industry.

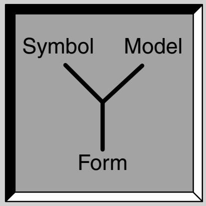
The set of tasks and tools which lead from product specification to manufacturing is called design methodology. Each system company uses a slightly different methodology in the creation of different products, depending on market needs, product requirements, quality, cost, and business model.

This application specific methodology combined with the advances in chip fabrication allow system companies to build complete systems on a single piece of silicon. That has introduced many changes in the way we used to think about building systems.

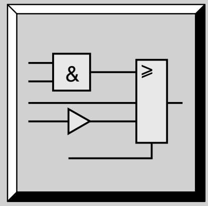
In this new design environment the system companies are defining products in their application domain whilst semiconductor companies are building systems on silicon. Design activities are usually outsourced to the third party vendors. The design activity is mostly focused on specification, architecture exploration, software, and verification. The cost of most SOCs is in software and not in hardware. Software and hardware are designed together, and we are forced to think about the whole product and not just about one block of design. Therefore designers must be aware of the whole design process and not just of one task on one level of abstraction.

This insight is also leading our university education to focus on system engineering in order to generate graduates with system knowledge and not just programmers or circuit designers.

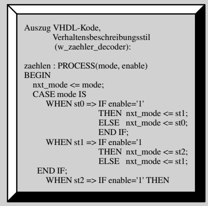
Overview EDA
1, 2



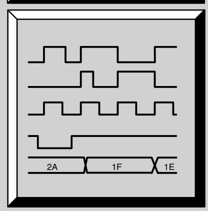
Symbolic Design
3



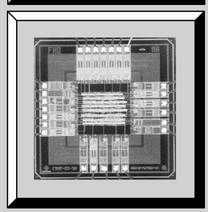
High Level Language Design
4, 5, 6, 7, 8



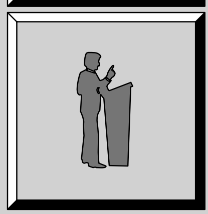
Modelling and Verifications
9, 10, 11, 12, 13, 14, 15



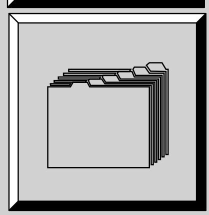
Implementation
16, 17, 18, 19, 20, 21, 22, 23, 24, 25



Tutorial
26



Appendix
A, B, C, D, E



1 Introduction

DIRK JANSEN

In 2000 the Semiconductor Industry Association (SIA), an American trade association of the semiconductor industry, announced a world-wide electronic industry market volume in excess of $200 billion. The average growth is anticipated to remain at a rate of 17 %/year [1.10]. Market volumes for the last 20 years can be seen in fig. 1.1. This figure shows the exponential growth of the curve; the first billion dollar was reached in 1961. By 2010 the SIA expects a trillion dollar market volume for an industry that was not in existence 50 years ago. No other industry in history has experienced this phenomenal growth rate.

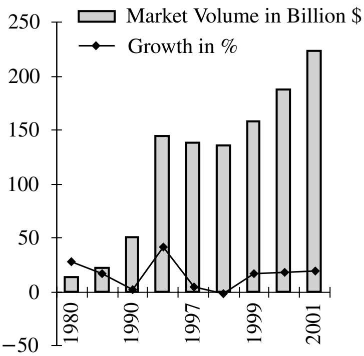


Fig. 1.1 Worldwide market volume and growth of the semiconductor industry [1.10]

The largest sales market is in the US. Japan, the second largest market, has sales on a par, with the rest of the European countries combined. The Asiatic-pacific nations of Taiwan and South Korea also have a great importance table 1.1. Europe has a subordinate role in the market; Europe is more or less a producer of semiconductors and electronics. Only one German company, Siemens, is amongst the 60 largest semiconductor manufacturers. Siemens has a ranking of 16 [1.9]. The consumption of German made semiconductors within Germany's own industries is infinitesimally small. The same is true within the PC-production sector, nevertheless a 100 million PC/YEAR production rate was observed in 1998. This is more than television production world-wide; a sector where European countries have been unable to conquer a considerable market share.

Table 1.1 Electronics market shares (consumption) according to data of the Semiconductor Industry Association SIA [1.10]

Market	1998	2001
USA	32.7 %	33.1 %
Japan	21.6 %	18.7 %
Asien/Pazifik	23.0 %	25.0 %
Europa	22.7 %	23.2 %

How it started

The transistor was originally invented by J. BARDEEN and W. BRATTAIN in 1947 at Bell Laboratories. Advance were quickly made by W. SHOCKLEY and the production of the first integrated circuits by KILBY at Texas Instruments began in 1958, and subsequently by Noyce at Fairchild Inc. [1.9]. The field of microelectronics had begun its triumphant advance. Before the creation of the transistor, electronic circuits were limited to simple designs relying heavily on vacuum tubes. Electronics were used almost exclusively in the area of radio and television broadcast and reception. Vacuum tubes were used on a large scale in these fields until the 60s. Even before transistors breached this vacuum tube stronghold, it was used as a discrete device or in simple integrated circuits in niche technological markets (e. g., medical electronics (hearing aids), in military equipment and space technology). This provided the means of funding the first useful transistors. Later Japan began to use the first semiconductors in consumer applications such as portable radios, tape recorders

12 Mixed Signal Simulation

MARTIN RIEGER

12.1 Overview

The term **mixed signal simulation** signifies a simulation of a circuit where parts of which are described at different levels of abstraction and are simulated simultaneously [12.6]. In this chapter particularly the simultaneous simulation of analog and digital parts of a circuit is discussed.

The *necessities and motivations* for applying mixed signal simulation to a system are listed below.

- Many important circuit blocks are intertwined analog and digital circuits, e.g., Analog Digital Converters (ADC), Digital Analog Converters (DAC) or Phase Lock Loops (PLL).
- More and more digital circuit blocks are described at an abstract level, e.g., in VHDL. If other digital and/or analog parts of the system have to be simulated, the result is again mixed signal simulation.
- Sometimes the different parts of a system have a different development maturity. The more elaborate parts are described at a logical level, for instance, whereas other parts are described only at a more abstract level.
- Another reason for using different abstract levels is given by complex components (e.g., a CPU) being available only as models at a high level of abstraction whereas other circuit parts have to be simulated at a more detailed level.
- One can reach the optimal simulation speed and accuracy only if the critical parts which determine the accuracy are described at a level of higher abstraction, and therefore can be simulated much faster.

12.2 Simulation on different levels of abstraction

The levels of abstraction discussed in this chapter are

- transistor (or circuit) level;
- logic level;
- register transfer level;
- system level (see fig. 12.1) [12.5].

The models in the different levels of abstractions represent different views of the parts of the system. The quantities used for description differ according to the level of abstraction (e.g., voltages or logic levels, respectively), also to the algorithms applied. Therefore one needs connecting **interfaces** when simulating at different levels of abstraction. When climbing up the hierarchy the complexity increases, the details are more or less lost, and the simulation effort decreases.

At the **transistor (or circuit) level** the models of the components are described by algebraic equations often in the form of differential equations. Time and value range are continuous, the quantities used at the analog nodes are voltage and current.

The circuit used as an example for transistor level description in fig. 12.1 represents a two input NAND gate. The performance of circuits of this kind can be checked by analog (or circuit) simulation. For that purpose the simulator has to solve the non-linear, coupled differential equation which describes the circuit. This process is very time consuming. The most popular analog simulator is SPICE.

At the **logic level** the models of the components are described by Boolean equations, truth tables, rise, fall, delay times, etc. Time and value range are discrete. The quantities used at the digital nodes are named ‘Bit’ and are logic levels, e.g., with the values 0, 1, X, R, F ... These values can be associated with different driving forces.

The circuit used as an example for logic level in fig. 12.1 represents a D flip flop consisting of NAND gates. It could be possible to simulate one of the NAND gates (e.g., the one leading to the output Q) at transistor level. Simulators for logic circuits mostly use event driven algorithms, which leads to faster simulations compared to analog simulators.