EDA Technology Roadmap Toward 2002

Cyber-Giga-Chip Design Technology

March 31st 1998

Electronic Industries Association of Japan (EIAJ)

EDA Technical Committee

EDA Vision Working Group

Preface

The technology innovation in the semiconductor industry has recently made tremendous progress so that system LSI's with compound functionalities integrated on a single chip tends to be widely used in electronic equipments. The small-variation-mass-product type business, such as general-purpose DRAM's, was a main driven force in the semiconductor industry so far, however, the market now is in strong demand of shifting to an application specific type LSI's business. Thanks to the solid progress of process technology, the manufactureability of high-performance and high-density LSI's has been established. To this end, the trend in the semiconductor industry is moving to "what to make" from "how to make".

The innovation of semiconductor technology brings changes of the business structure of the semiconductor industry. For example, business interface between semiconductor makers and system makers will change in terms of which semiconductor makers shift from supplying specific components to system solution to system makers.

The system LSI design in future would be focused on an entire design procedure from algorithm through physical layout design. Revolutionary design methodologies are required from various aspects, such as software design accomplished with hardware design, built-in analogy/sensor functionalities of human interface, shorter time-to-market design style for shorter product life cycle, and so on.

Under the above situation recognition, EDA Technical Committee in EIAJ (Electronic Industries Association of Japan) organized EDA Vision Working Group for investigation of design technologies of system LSIs in 2002. Started in September, 1996, the working group has defined the design profile/flow of system LSI's, and analyzed existing technical problems for LSI design by detailed interviews with advanced experts of LSI designers and EDA engineers. The working group also named those system LSIs used in consumer portable information terminal equipments, which are the major products of the domestic semiconductor industry, as "Cyber-Giga-Chip" and took it as analysis basis in its activities. Furthermore, the working group set up the targets for design engineering based not only on these existing technical problems in EDA industry but also changes of business/design environment.

"EDA Technology Roadmap Toward 2002" is summarized from analysis results of the paradigm shift in the semiconductor industry from the viewpoint of LSI design. The authors hope this report would be a guideline for the future system LSI design, and a starting point for discussion on this theme not only from the industrial side but also from universities as well as research institutes.

March 1998,
EDA Vision Working Group
Chair Yoshiharu Furui

Copyright

Electronic Industries Association of Japan (EIAJ)
Electronic Devices Department
Tokyo Chamber of Commerce and Industry Bldg.
2-2, Marunouchi 3-chome
Chiyoda-ku, Tokyo 100-0005, Japan
TEL : +81-3-3213-1065
FAX : +81-3-3211-0993

Document Access and Updates

Online Access

The electric file of this document is downloadable from the following home page.

"http://eda.ics.es.osaka-u.ac.jp/eiaj/eda/project/eda-vision.html"

Comments

Comments concerning this document are accepted by

"feedback-vision@EDA.ics.es.osaka-u.ac.jp"

History

03/31/1998	First Edition

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Background

With the advancement of semiconductor technology, the possible number of transistors on a silicon chip is doubling every three years. Soon, it is estimated that over 10 million transistor circuits will be realized on a silicon chip of only 1 cm^2. Now, with this progress in LSI manufacturing technology, it is time to consider what kind of system should be designed, which is an important issue for the future semiconductor industry.

Formerly, the principal products of the semiconductor industries, such as memories and microprocessors, were just "parts" of a system that make up the final product. Since the size of required by specification was over limitation of productivity, the main concern was how to design, produce and test the LSI, rather than what to design. In the next decade, however, it is expected that existing microprocessors with the highest performance and a large memory will be integrated on a single chip, and the whole system consisting of processors, memories and other logic could be implemented as a single LSI. The major problems on the design of LSIs will then shift to what kind of systems should be designed and specified. As we enter the era of System-On-Silicon (SOS) or System-On-Chip (SOC), new design methodologies are requested, which should be quite different from that for designing the “parts” and not the “system”.

In the case of standard parts, such as memories, process technologies for manufacturing and internal circuit designing are more important than specification of the products. In the case of microprocessors, although it is a small-variation-mass-product type business, specification and design technologies are more important. However the number of designs was quite limited and much time and cost was spent on the design of the microprocessors. In ASIC (Application Specific Integrated Circuit) business, specifications of LSIs, which are extremely important, are given by the system designers. However, in design of system LSI, the design for the final product and LSI are inherited, as all functions will be implemented ultimately on a single chip. The specification of system LSI is also the specification of the final product. Therefore, it is important in semiconductor business as well as the final product business.

There are three principal changes of environment in system LSI design technology. First, as process technology advances, there is an exponential increase of investment in production lines and demand for new design technologies to resolve many problems caused by scaling down and improving performance. Second, design technologies to handle large scale, complicated systems are important concerns, since superiority of an LSI greatly depends on the quality of the system and its circuit design. Third, there is pressure from the market to shorten the time of system development and to improve design efficiency.

Therefore, technology for the fast and efficient design of system LSIs is urgently required.

The future of LSI business will depend on how these requirements are dealt with. The LSI businesses may be divided into the following three species:

1) Fabrication business based on the advanced process technology,

2) Vertically integrated system business including both system design and fabrication,

3) Fabless System business.

In Japan, many companies have both system design and fabrication technology divisions. However, some of them will have no choice but to become simply fabricators or else fabless designers, if they can not resolve upcoming problems. To keep the style of vertically integrated structure and to enjoy the advantage of it, there is a need to develop a new design technology and construct a new style of business for the era of system LSIs. It is an urgent research and development task to establish a new design method, within which system designers and LSI designers may collaborate efficiently, with EDA tools supporting the collaboration.

This roadmap aims to clarify the direction of EDA technology to support design methods for system LSIs. It summarizes research and development targets of EDA technology in 2002, which may be reasonably easy to predict, and proposes scheme to reach them. We intend to give a foundation on which to start discussions on new design methods and the restructuring of the vertically integrated industries for a new technological environment.

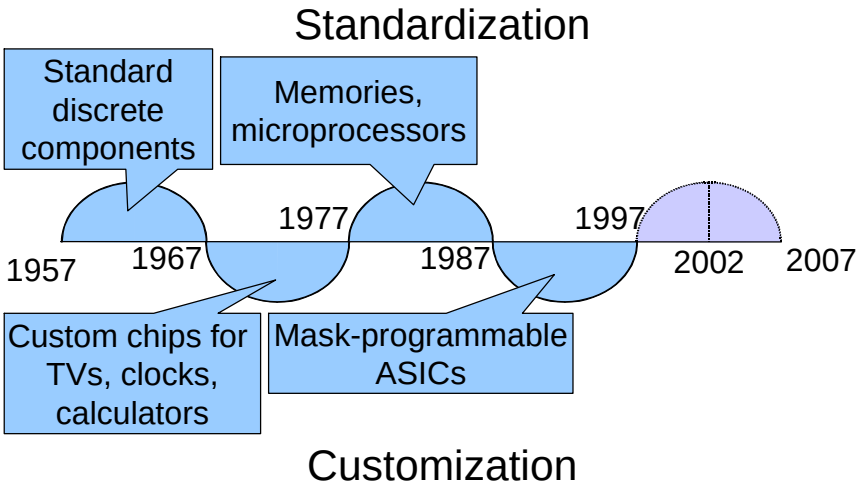


Figure 1 Makimoto's Wave (Source: IEEE Spectrum Jan. 1992)

Technical revolutions in semiconductor business are illustrated as Makimoto's wave (Figure 1). Technology innovation (customization) and competition (standardization) have been repeated with a period of 20 years. After the innovation period of developing a new market, the technology is standardized and many products are available. When the market becomes saturated, new customized technologies are invented for product differentiation. From the macroscopic point of view, the decade from 1997 to 2007 will be the era of standardization, in which semiconductors are designed and produced

using combinations of standardized technologies. 2002 will be the summit of the wave of standardization and the entrance of the era of practical system LSIs with several tens of millions of transistors. To manage the variety of specification of system LSIs, design reuse will be an important technology. The system LSI will be designed as a combination of well-design cores. Cores, interfaces between cores and interfaces between system LSIs will be standardized and large, high performance and complicated system LSIs will be designed flexibly in 2002. In other words, this is the era of custom system design utilizing standard components such as cores. Therefore, the next decade will mark a new phase for design technology, in which the customization of system design and the standardization of LSI design should be treated simultaneously.

From the viewpoint of EDA technology, standard tools are used in the era of customization, and superiority of an LSI strongly depends on its conception and planning. On the other hand, in an era of standardization, superiority of an LSIs is determined by design technologies and tools. In the next decade, design methodologies and tools will be a key to success in the semiconductor business.

Discussing the direction of semiconductor technologies and drawing a roadmap are important measures for the sound development of the semiconductor industry. In U.S., SIA published roadmaps [1] including various technologies related to semiconductor industries; and CFI published a roadmap [2] on EDA technology mainly for standardization. In Japan, ATLAS project of the Semiconductor Industry Research Institute Japan examines the ability of Japanese designers to create applications in 2010 [3]. However, there is no roadmap for EDA technology for the system LSI era. This roadmap summarizes the discussion in EIAJ on the future of EDA technology, in the hope that it will act as a guide for the next EDA technology innovations. This roadmap summarizes the discussion in EIAJ on the future of EDA technology, in the hope that it will act as a guide for the next EDA technology innovations.

1.2 Objectives

The objective of the roadmap is to show the following by the 21st century:

- The target for system LSI in 2002, which we have called the Cyber-Giga-Chip (CGC),
- Design and test methodology to be used for system LSI,
- EDA technology to assist design and test the system LSI,
- An EDA roadmap for each important application area in 2002.

In order for the roadmap to be realistic and practical, we have focussed on:

1) Investigation of necessary technology by 2002,
2) Requirement analysis on EDA technology from the viewpoints of LSI designers.

The reason for setting 2002 as the deadline in the former policy is as follows:

- Core-based design will be the main stream in the age when the design rule becomes 0.18 to 0.13 μm,
- 2002 is the summit of “Standardization Age”,

- 5 years is a reasonable period to draw up a realistic timetable for the various EDA topics.

The latter policy is adopted so that the roadmap will link the seed (i.e. the technology EDA engineers can offer) to the need (i.e. the requirements LSI designers must meet) for the proposed system LSI design.

1.3 Definitions

The terms listed below are used frequently from now on.

Cyber-Giga-Chip:	A chip which forms the kernel of a piece of electronic equipment, and is composed of various functional blocks.
Core:	A sub circuit with certain function.
IP:	Intellectual property, such as design property.
System Design:	To design the specification of a system LSIs, and establish the means to realize target functions in the hardware and software.
Architecture:	To decide the hardware composition of the function that needs to be realized, under some restrictions.
RTL/Logic Design:	To select the hardware circuits and logic circuits of the function to be realized, subject to constraints.
Circuit Design:	To describe each circuit in terms of its basic parts, such as cell, analog and memory, subject to constraints.
Layout Design:	To position basic parts on a silicon chip, and to wire between them based on connection information, in order to realize an electronic circuit. Placement and routing of the basic parts will directly influence electronic behavior in deep submicron era, so it is important to perform place and route with electronic behavior in mind.
Test Design:	To generate test data for an LSI tester to check whether the electric circuit implemented on a silicon chip realizes the required functions. Design for test (DFT), for example, adding a circuit to make measurement convenient, may also be included under this heading.
Test:	To make measurements to see whether manufactured LSI actually performs its desired function, using a LSI tester etc.

1.4 Audience for the Roadmap

The roadmap is suitable for all managers and/or engineers who are concerned with system design/test, semiconductor design/test, and EDA technology. Researchers in universities are also targeted.

1.5 Making of the Roadmap

The roadmap is made by EDA Technical Committee / EDA Vision Working Group of Electronic Industries Association of Japan (EIAJ), in cooperation with Institute of Systems and Information Technologies/KYUSHU (ISIT/KYUSHU), shown in Table 1 and Table 2, respectively.

Table 1 Member List of EDA Vision Working Group

Name	Representing	Participation
Yoshiharu Furui	Sony Corporation	Chair
Takashi Kambe	Sharp Corporation	Co- Chair
Tsutomu Someya	IK Technology Co., Ltd.	Working Member
Ichirou Yamamoto	OKI Electric Industry Co., Ltd.	EDA TechnoFare 98 Publication
Kazuya Morii	SANYO Electric Co., Ltd.	Working Member
Akihisa Yamada	Sharp Corporation	Working Member
Takayuki Yamanouchi	Sharp Corporation	Working Member
Tetsuya Fujimoto	Sharp Corporation	Working Member
Masato Ikeda	Zuken Incorporated	Working Member
Nobuto Ono	Seiko Instruments inc.	Working Member
Masaru Kakimoto	Sony Corporation	Working Member
Kenji Yoshida	Toshiba Corporation	Advisory Member
Tamotsu Hiwatashi	Toshiba Corporation	Leader of LSI Design Needs WG
Akihiro Yamada	Toppan Printing Co., Ltd.	EDA TechnoFare 98 Publication
Naoya Takahashi	NEC Corporation	Working Member
Hiroko Yamamoto	ViewLogic Japan	Working Member
Yoshio Ohshima	Hitachi, Ltd.	Leader of EDA Requirement WG
Noriyuki Itou	Fujitsu Ltd.	WWW Publication
Michiaki Muraoka	Matsushita Electric Industrial Co., Ltd.	Leader of Cyber-Giga-Chip Profile WG
Mitsuyasu Ohta	Matsushita Electric Industrial Co., Ltd.	Working Member
Hideyuki Hamada	Mitsubishi Electric Corporation	Working Member
Mitsuhiro Kitta	Mitsubishi Electric Corporation	Leader of EDA Roadmap WG
Itsuo Suetsugu	Mentor Graphics Japan Co. Ltd.	Working Member
Kousuke Shiba	Mentor Graphics Japan Co. Ltd.	Working Member
Masaharu Imai	Osaka University	Advisory Member
Tokinori Kozawa	STARC	Advisory Member
Sagorou Hazama	Fujitsu Ltd.	Advisory Member
Satoshi Kojima	Hitachi Ltd.	Advisory Member
Tsuneo Shibazaki	EIAJ	Secretariat
Koji Kitada	EIAJ	Secretariat

Table 2 Institute of Systems & Information Technologies / KYUSHU

Name	Representing	Participation
Hiroto Yasuura	First Research Laboratory Director Also Prof. of Kyushu University	Investigation Editing Roadmap
Hiroshi Date	First Research Laboratory Researcher Doctor of Engineering	Investigation Editing Roadmap

Figure 2 shows the roadmap development concept. In the EDA Vision Working Group, Cyber-Giga-Chip Profile WG and EDA Requirement WG are organized. Cyber-Giga-Chip Profile WG forecasted

the profile and the design flow of Cyber-Giga-Chip in 2002. EDA Requirement WG analyzed the requirements of LSI designers from the viewpoint of EDA technology, and made a detailed roadmap toward 2002. Entrusted by EIAJ, ISIT/KYUSHU collected information from LSI designers, investigated the trend of EDA technology, and edited the roadmap. Specifically, ISIT/KYUSHU interviewed for LSI designers working for companies, which belonged to EDA Vision Working Group, and summarized the results as requirements of designers. This was followed by interviews with EDA engineers and investigation on EDA technology. Finally, EIAJ and ISIT/KYUSHU edited the content of the activity as a roadmap.

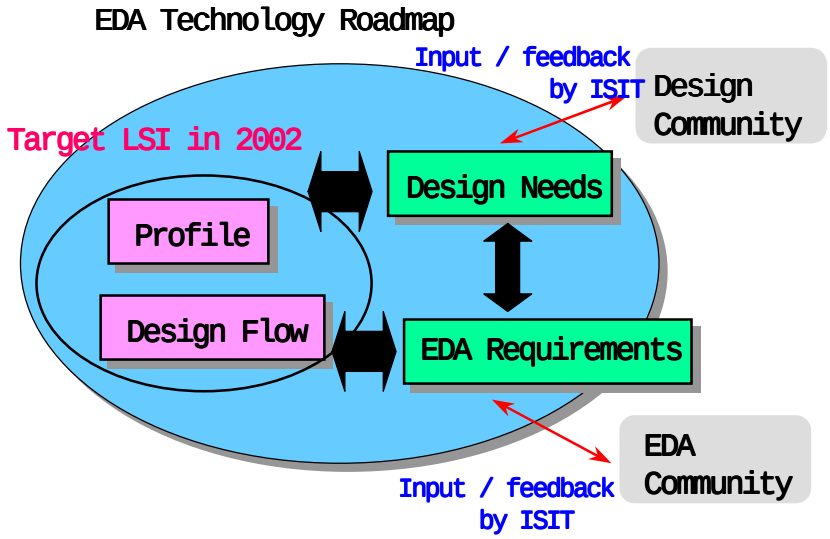


Figure 2 Roadmap Development Concept

1.6 How to Utilize the Roadmap

Some suggested uses of the roadmap are:

- For system LSI designers: To determine the most suitable design environment for their system LSIs. The roadmap will suggest a design environment for each category of system LSI.
- For semiconductor manufacturers: To determine business strategies for system LSIs in their organizations. An LSI design tool will act as an important interface between users and semiconductor manufacturers. Each semiconductor manufacturer must decide its business tactics; whether to specialize in fabrication only; proceed with vertically integrated industries from system design to fabrication, or to take an intermediate course.
- For EDA engineers: To determine the most profitable EDA tool for system LSIs.
- For EDA researchers: To discuss what is the fundamental technology for future EDA.

2. Executive Summary

In this chapter, we present the basic policies of the EDA Technology Roadmap toward 2002. The main purpose of the roadmap is to predict:

1) The objects for the LSI design and their environment in 2002,
2) EDA technology in 2002.

2.1 Semiconductor Industry in 2002

In this section, we summarize interviews concerning, firstly what kind of LSI will be designed (the objects for the LSI design) in 2002 and, secondly how they will be designed (LSI design environment) in 2002.

2.1.1 Overview of Design Objects in 2002

The technique for designing a system LSI could be considered equivalent to that for designing a target system itself. The main question is what sort of system should be built by making use of several ten million transistors, and this decision may influence the future direction of the semiconductor industry. system LSIs will be used in a wide spectrum of applications, such as computers, electric appliances, toys, cars, optical machinery, industrial robots, and social systems. In this section, we will predict the potential application areas for system LSIs and examine the requirements of system LSIs specific to each application.

1) Embedded systems

Embedded systems are widely used in electric appliances, cars, industrial robots, communication devices and even toys (entertainment gadgets), all of which are basic commodities in modern day life. System LSIs will play a central role in embedded systems. Unfortunately, they cannot be designed and manufactured effectively enough in the current framework, and there is a call for a breakthrough in design methods to overcome the problems. The new design methods will be required to coordinate individual component technologies in software, hardware, package, display parts and machine parts and optimize the entire system. Each application has a different set of requirements. Safety and reliability are indispensable factors to medical equipment. On the other hand, promptness and productivity are important in electric appliances and entertainment gadgets, as their lifecycles have shortened dramatically in recent years. The design of system LSIs should reflect the needs of each application area.

2) Network devices

A variety of communication forms such as image, sound and text are used in network. This revolutionary move has created a new social structure sometimes termed as multimedia society. In order to send a mixed form of image, sound, and text across a network, data needs to be transformed. Such processing includes data compression and restoration, coding, encryption and decryption, etc. The same procedure must be performed at sending and receiving ends, since the standard protocols are defined in network communication. If the connected systems contain some human errors and cause malfunction, the entire network (in the worst case, the Internet which spans the globe) may behave abnormally. Communication circuits should be standardized and used as fixed parts in system LSIs. This will promote the reuse of communication circuits as well as stabilization of the network. With a view to such development, the reuse of integrated circuit design, the definitive procedure to claim integrated design as intellectual property, and suitable criteria for macro library are being discussed. In addition, analog circuits are thought to be important in radio/wireless communication.

3) Information technology for social infrastructure

Information technology for building the social infrastructure will be a potential market for system LSIs. What we call “information technology for social infrastructure” has two missions. One is an information system that monitors and controls the social infrastructure (buildings, traffic, railways, electricity, water, gas and telecommunication) This helps a smooth running of social activities. The other is the information system that deals with unpredictable catastrophes. This will assist effective data acquisition, assimilation, transmission, processing and controls in disasters. Real-time processing and safety are essential of the information system and embedded system LSIs. In addition, there is a desire for those systems to operate using reusable and environmental-friendly energy such as solar energy. Electric money is an application of system LSI in the economic system. This is an electric substitute for paper notes and metal coins and is usually stored in an IC card (Smartcard). Smartcard and its associated equipment are expected to create a big market for system LSI. The system LSI for smartcards must, above all, guarantee high reliability to protect the privacy of cardholders against deliberate attacks, fraud and forgery. This means that cryptographic technology must be incorporated in system LSI design. In addition, R&D efforts should be put into submicron technology, to increase the security level.

4) High performance computer systems

The computer system industry is the biggest marketplace for the semiconductor industry and has motivated the semiconductor industry to make epoch-making progress in high performance processors. The demands from supercomputing will continue to guide the development of high performance system LSIs. In particular, scientific computation in fields of genome analysis, molecular dynamics, fluid dynamics, and space development will put pressure on the improvement of high quality performance system LSIs. Massively parallel system will serve as a good design model for the forthcoming system

LSI design. Mixed technology of DRAM and logic, and functional memory technology are expected to show enormous progress.

5) Personalized digital equipment

Personal computers have become a widespread commodity in the modern life due to their compactness and portability, and have contributed to the expansion of the semiconductor industry. This spread of personalized digital equipment will continue in aid of the Internet and multimedia technology. The requirements for those systems will not only be high performance, but also cost reduction and low power consumption. Moreover, the progress in sensor technology, digital/analog merged integrated circuits technology and high performance technology is important to meet the demands in multimedia and HCI (human-computer interaction) technology.

Having considered the above mentioned potential applications, system LSIs can be classified into the three categories.

1) Process-oriented LSI

Like the former DRAM, this LSI is relatively stable in terms of product specification, and can compete with others to achieve a world record by making good use of the latest process technology. It is designed for mass-product, and several years are usually spent in designing. Its performance is moderately high, combining the existing logic and circuits and aiming at mass-product.

2) Performance-oriented LSI

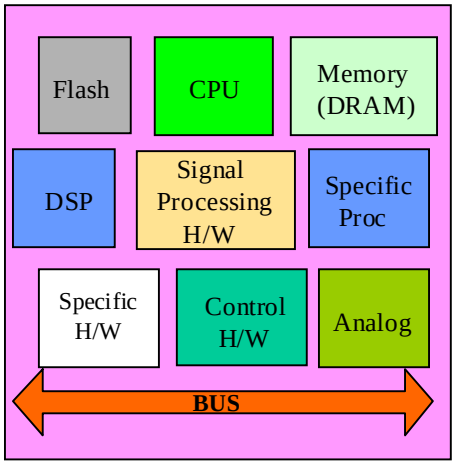
The LSI's specification, including those at the architecture level, are at the discretion of the design side, although its framework is fixed just like the microprocessor. It is designed by making good use of advanced design technology at every design level, such as architecture, logic, circuits and layouts. However, the latest processes are not necessarily used. So the difference in the design technology becomes quite apparent. Sometimes, several years are spent in designing. Such fields are called leading edge system LSI.

3) Market-oriented LSI

This is a widely used LSI, because many systems adopt it. Many LSIs that are used in various built-in systems and information telecommunication equipment, etc. are classified in this field. Most of them are being developed in a short period because their market value is decided by the time function. System LSI included in this classification should deal with various specifications of the system in addition to short-term development. It should be based on the technology which optimizes the performance of LSI by using a standard core for a specific usage. The system LSI which will become the main stream in the future will be included in this classification. Because of the unification of system and LSI design, the traditional design method cannot be applied. This LSI could be considered to have a

form suitable for the vertical and integrated industrial structure in Japan.

In this roadmap, we focus on a market oriented LSI, especially, a system LSI which has a core based on the leading edge technology in 2002. We named such a core-based system LSI as "Cyber-Giga-Chip(CGC)". Figure 3 shows the specification of CGC. CGC has several such cores, and is used as a kernel of a system. Chapter 3 explains the specification and summarizes the results from the interviews with LSI designers from the viewpoint of "what will be needed in EDA technology for the design of CGC".



- **CPU Performance**
  500MIPS, 400MHz
- **Process Technology**
  0.18 ~ 0.13 μ m
- **Logic Gates**
  5M Gates
- **Mixed Technologies**
  DRAM/Flash/Analog
- **DRAM**
  256M ~ 1Gbits

Figure 3 Profile of Cyber-Giga-Chip in 2002

2.1.2 Overview of Design Environments in 2002

With the progress and spread (wide acceptance) of network and database technology, various collaborated works between different organizations will be possible. The platform for parallel processing has advanced, which can be a potential candidate for the use of EDA tools in distributed environments.

In the case of process-oriented LSI, it is important to develop a good process and to utilize the characteristics of the process. Therefore, it is thought that the relation between the process technology and the EDA technology becomes stronger. As in the case of supercomputing, design methodology and EDA technology for the performance oriented system LSIs will be limited to a group of organizations. In market oriented system LSIs, design methodology and EDA technology will be supported by various industrial demands and EDA technology will therefore continue to develop. In application-specific system LSIs, CAD technology is seen as a fundamental technology for a two-way (upward and downward) silicon compiler between system level and process level. As application-specific system LSI design is involved in multi-level system descriptions, different styles of IP descriptions, and a

hardware/software mixed system, and verification of design becomes important. Methods to distinguish "good" IPs from "bad" IPs are also required.

2.2 EDA Technology in 2002

With a spread of system LSIs, LSI design method based on core processors will become main stream, as shown in Figure 4, and the LSI design flow will be changed accordingly. The design workload is partitioned into system designers, silicon designers and process designers. The current practice of LSI design is that several LSIs are used, and the job of a system designer is merely to combine these LSIs to produce an "optimal" system.

Therefore, LSI and process design are proceeded independently of each other. However, in order to design an "optimal" system LSI for "optimal" system, the system designer must consider the best means to achieve the "optimal" system at hand under the given circumstances. CAD tools for system designers are desirable to assist with this job. Once the specifications of the system LSI are fixed, silicon designers take part in designing each processor core using the existing CAD tools. At present, design rules are provided prior to system design, prohibiting the possibility of process design changes. Design rules may have to be modified to produce an optimal system LSI. Hence, new process-dependent CAD tools which can deal with modifications of the process technology are desirable.

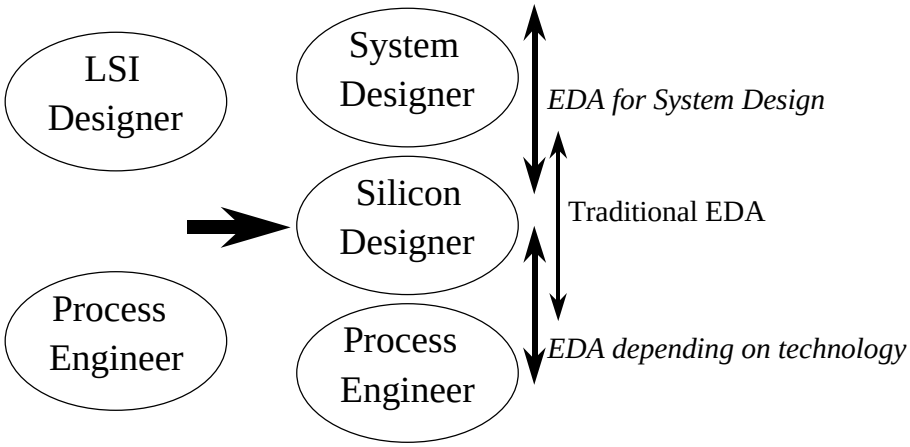


Figure 4 Design Flow of Core Based System LSIs

The application fields of system LSIs are various. Therefore, the priority of each EDA technology needed in 2002 changes according to the design target of the system LSI. Chapter 4 explains each item concerning EDA technology problems. Chapter 5 forecasts the design flow of CGC in 2002, and shows what kinds of EDA technology are necessary to design CGC. The following paragraphs show the outline of EDA technology problems in 2002. The details are described in Chapter 4.

The problems and the counter measures concerning EDA technology are summarized along the design

flow of hardware and software. As for hardware design, we have divided it into digital circuits and analog circuits. Moreover, in digital circuit design, problems of EDA technology are classified into specification, estimation, verification, synthesis, and test.

Table 3 EDA Technology Problems of Digital Circuit Design

(System Design to Architecture Design)

<table>
<tr><td></td><td colspan="5">Digital</td></tr>
<tr><td></td><td>A Specification</td><td>B Estimation</td><td>C Verification</td><td>D Synthesis</td><td>E Test</td></tr>
<tr><td rowspan="3">1.
System</td><td>1A(1)
Standard System Level Modeling</td><td rowspan="3">1B(1)
Performance Estimation of System (Application Software/ Compiler/ Hardware)</td><td>1C(1)
System Level Simulation</td><td>1D(1)
Hardware/Software Partitioning</td><td rowspan="3">1E(1)
Test Strategy Decision for System (Hardware /Software)</td></tr>
<tr><td rowspan="2">1A(2)
Standard System Description Language (SLDL etc.)</td><td>1C(2)
System Level Emulation</td><td rowspan="2">1D(2)
System Level Library (IP Core, Middleware)</td></tr>
<tr><td>1C(3)
Formal Verification (System Spec. vs. System)</td></tr>
<tr><td rowspan="2">2.
Architecture</td><td>2A(1)
Standard Architecture Level Modeling (Including Domain Specific Models)</td><td>2B(1)
Architecture Level Estimation (Area, Timing, Power, Floorplan)</td><td>2C(1)
Formal Verification (-System vs. Architecture)</td><td>2D(1)
Architecture Synthesis</td><td>2E(1)
Test Strategy Decision for Architecture</td></tr>
<tr><td>2A(2)
Standard Architecture Description Language (VerilogHDL, VHDL etc.)</td><td>2B(2)
Budgeting (Area, Timing, Power, Floorplan) for RTL</td><td>2C(2)
Validation Simulation</td><td>2D(2)
Co-Synthesis</td><td>2E(2)
Architecture Level DFT</td></tr>
</table>

Table 3 shows EDA technology problems in system design and architecture design of a digital circuit. In system design, it is necessary to deal with Standard System Level Modeling, Standard System Description Language (SLDL etc.), System Level Simulation, Performance Estimation of System (Application Software/ Compiler/ Hardware) , System Level Emulation, Formal Verification (System Spec. - System), Hardware/Software Partitioning, System Level Library (IP Core, Middleware), and Test

Strategy Decision for System (Hardware /Software). In architecture design, the items are Standard Architecture Level Modeling (Including Domain Specific Models), Standard Architecture Description Language (Verilog HDL, VHDL etc.), Architecture Level Estimation (Area, Timing, Power, Floorplan), Budgeting (Area, Timing, Power, Floorplan) for RTL, Formal Verification (System - Architecture), Validation / Simulation, Architecture Synthesis, Co-Synthesis, Test Strategy Decision for Architecture, and Architecture Level DFT.

Table 4 shows the technical issues of EDA ranging from RTL / logic design to manufacture interface in a digital circuit design. In RTL/logic design, it is necessary to consider problems, such as, Standard RTL (Synthesizable) Modeling (Including Graphical Model), Standard RTL Description Language (Verilog HDL, VHDL etc.), RTL Estimation (Area, Timing, Power, Floorplan), Budgeting (Area , Timing, Power, Floorplan) for Logic Synthesis, Power-rail Estimation, False path free Timing Analysis, Formal Verification (Architecture - RTL, RTL - Gate), Function/Timing Verification beyond Gate Level Simulation, Test Pattern Generation for Mixed IP Chip on Function and Timing Verification, Timing Driven Synthesis, Reverse Synthesis (from Gate to RTL), RTL Synthesis, Incremental Design Methodology, Logic Synthesis with Parameterized Cell/Macro Library (Vdd and Vth), Timing Budgeting and Function Porting for Mixed IP chip, Standard DFT Interface for both IPs and inter IPs, Design for Fault Diagnosis, and Multiple Fault Models & Test Methods. In the circuit, it is necessary to take account of the Process Variation Model (Accurate Circuit Simulation), Power/Noise/Electro-magnetic Analysis, Accurate Model for DSM Process, Parameter Extraction, Timing Verification on Real Environment (consider Package, Board etc), Transistor Circuit & Layout Synthesis (from RT/Gate Level), Parameterized Cell/Macro Library Generation and Simulation with Vdd and Vth Variation, and Performance Constraint Driven Process Migration. In a layout design, we have to consider Standard Physical Description Language, High-speed Verification, Power-rail Routing, Simulation based Layout Synthesis, High speed Mask Data Processing, Decision Support for Multi-Layer Routing, Layout Synthesis with Parameterized Cell/Macro Library (Vdd and Vth), and Layout Design for Test. Speeding-up of mask processing is a big problem concerning the interface to the manufacturing equipment.

Table 4 EDA Technology Problems of Digital Circuit Design (RTL to Manufacture Interface)

	Digital				
	A Specification	B Estimation	C Verification	D Synthesis	E Test
3. RTL/Logic	3A(1) Standard RTL (Synthesizable) Modeling (Including Graphical Model)	3B(1) RTL Estimation (Area, Timing, Power, Floorplan)	3C(1) False path free Timing Analysis	3D(1) Timing Driven Synthesis	3E(1) Standard DFT Interface for both IPs and inter IPs
		3B(2) Budgeting (Area , Timing, Power, Floorplan) for Logic Synthesis	3C(2) Formal Verification (-Architecture vs. RTL) (-RTL vs. Gate)	3D(2) Reverse Synthesis (from Gate to RTL)	
				3D(3) RTL Synthesis	
	3A(2) Standard RTL Description Language (Verilog HDL, VHDL etc.)	3B(3) Power-rail Estimation	3C(3) Function/Timing Verification beyond Gate Level Simulation	3D(4) Incremental Design Methodology	3E(2) Design for Fault Diagnosis, Multiple Fault Models & Test Methods
			3C(4) Test Pattern Generation for Mixed IP Chip on Function and Timing Verification	3D(5) Logic Synthesis with Parameterized Cell/Macro Library (Vdd and Vth)	
				3D(6) Timing Budgeting and Function Porting for Mixed IP chip	
4. Circuit		4B(1) Process Variation Model (Accurate Circuit Simulation)	4C(1) Power, Noise, Electro-magnetic Analysis	4D(1) Transistor circuit & Layout Synthesis (from RT/Gate Level)	
			4C(2) Accurate Model for DSM Process, Parameter Extraction	4D(2) Parameterized Cell/Macro Library Generation and Simulation with Vdd and Vth Variation	
			4C(3) Timing Verification on Real Environment (consider Package, Board etc)	4D(3) Performance Constraint Driven Process Migration	
5. Layout	5A(1) Standard Physical Description Language		5C(1) High-speed Verification	5D(1) Power-rail Routing	5E(1) Layout Design for Test
				5D(2) Simulation based Layout Synthesis	
				5D(3) High speed Mask Data Processing	
				5D(4) Decision Support for Multi-Layer Routing	
				5D(5) Layout Synthesis with Parameterized Cell/Macro Library (Vdd and Vth)	
6. Manufacture Interface				6D(1) Interface Technology for Mask Manufacturing	

Table 5 EDA Technology Problems of Analog Circuit Design

	F Analog
1. System	
2. Architecture	
3. RTL/Logic	3F(1) Standard Analog Modeling (for Mixed Signal Simulation)
	3F(2) Analog/Digital Mixed Signal Simulation
	3F(3) System Level Analog Modeling
	3F(4) Synthesis from AHDL
4. Circuit	
5. Layout	5F(1) Analog Cell Generation
6. Manufacture Interface	

Table 5 shows EDA technology problems in the analog circuit design. It is necessary to deal with Standard Analog Modeling (for Mixed Signal Simulation), Analog/Digital Mixed Signal Simulation, System Level Analog Modeling, Synthesis from AHDL, and Analog Cell Generation.

Table 6 Problems of EDA Technology in Software Design and Entire design

G Software	H Entire design
G(1) OS Generation/ Customization	H(1) Design Re-use
G(2) Software Core Generation/ Customization	H(2) Design Flow Management
G(3) Software Compiler Generation	H(3) Asynchronous Circuit Design
G(4) Unified Software Development Platform for various IP Cores	
G(5) Software/Hardware Co-Simulation	

Table 6 shows technological problems of EDA in the software design as well as the entire design. In the software design, it is necessary to deal with OS Generation/ Customization, Software Core Generation/ Customization, Software Compiler Generation, Unified Software Development Platform for various IP Cores, and Software/Hardware Co-Simulation. It is necessary to deal with Design Re-use, Design Flow Management, and Asynchronous Circuit Design over the entire design.

2.3 Outline of the Roadmap

This roadmap employs two basic policies. Firstly, the roadmap is intended to be effective only for 5 years. The reason for this rather short period of time is a roadmap showed to be as realistic and practical as possible. 5 years is a good period to draw up a realistic timetable for various EDA foundation technologies. Secondly, the roadmap is intended to address the issues raised from not only EDA engineers but also from LSI designers. Thus we hope that the roadmap will encompass the seed (i.e. the technology EDA engineers can offer) and the need (i.e. the requirements LSI designers must meet) for future LSI design.

We employed the following research method to write the roadmap. Firstly, we interviewed active LSI designers on site and summarized their requirements in LSI design. Secondly, we proposed a model of system LSI for the next generation, named “Cyber-Giga-Chip”. Lastly, We classified each system LSI into categories according to its business model. The roadmap suggests the ideal design method for each category and points out problems associated with the design method. The problems are examined on the basis of specification, performance estimate, verification, implementation, and testing from EDA technology perspective. We have also included our comments on each problem. The following describes the expected picture of the semiconductor industry in 2002 and analyzes the requirements of the semiconductor industry for EDA technology.

3. Requirements for EDA Technology

The purpose of this chapter is to specify the environment that surrounds the EDA technology. To be more specific, we define a system LSI called Cyber-Giga-Chip as a design target in 2002 and analyze the demand for EDA technology.

3.1 Profile of Cyber-Giga-Chip

In this roadmap, our target is the system LSI which is designed with the latest technology in its period. In order to provide basic information for defining the system LSI, the investigation result concerning the characteristic of the system LSI in 2002 is shown.

(1) Merged memory and logic technology: Memories such as DRAM and SRAM, a processor, and a DSP are merged into a single chip. System LSI with 32-bit microcomputer and DRAM would be in the mainstream.

(2) Mixed digital and analog technology: A digital circuit and an analog circuit are combined. The system realizes each function by a program, and analog technology is applied only at a minimum level. However, analog technology should be regarded as important for design, since it becomes the key of differentiation.

(3) Combination of a sensor and an analog circuit: By combining a sensor and an analog circuit, a process unifying data compression and recognition is conducted within a chip.

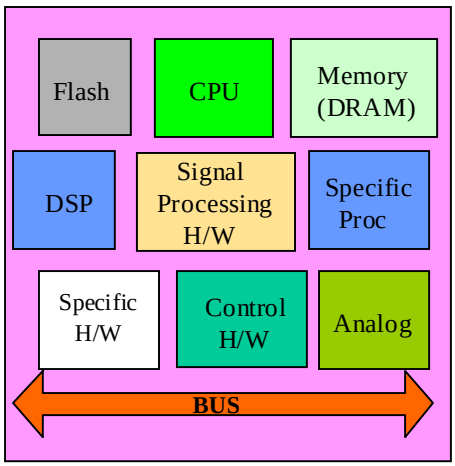
(4) Parallel Processing: On-chip RISC multiprocessor (MIMD) is the main type of CPU for system LSI. It is applied to a portable equipment for multimedia. In the system, the hardware is standard whereas the software is more diversified.

(5) Fault-tolerant: Technology such as duplex control in a chip is used in the field where high reliability is required.

(6) Wireless system core: A wireless system core is built in the system LSI for communication.

(7) Specification of high-end processor: Listed below are the specifications of a high-end processor which were investigated to predict the spec of the system LSI in 2002. The results: clock frequency, 1GHz; the number of transistor, 500M-1G (including a logic part consisting of 50M-100M transistors); power consumption, 50-60W; power-supply voltage, 1-1.5V; and process technology, CMOS. It is worth nothing that the number of I/O pins of high performance processors for the super computer is 1000-2000 because of adoption of the multiprocessor.

The specification of the system LSI for specific uses in 2002 is defined based on these investigation results. It is called *Cyber-Giga-Chip*, abbreviated as CGC. CGC, whose profile is shown in Figure 5, consists of such cores as CPU, DSP, memory, and so on. Merged memory/logic technology and mixed digital/analog technology are important too.



- **CPU Performance**
  500MIPS, 400MHz
- **Process Technology**
  0.18 ~ 0.13 μ m
- **Logic Gates**
  5M Gates
- **Mixed Technologies**
  DRAM/Flash/Analog
- **DRAM**
  256M ~ 1Gbits

Figure 5 Profile of Cyber-Giga-Chip in 2002

Table 7 shows the specification of the core in CGC. In 2002, the semiconductor manufacturing technology will use 0.13 to 0.18μm rule; the clock frequency of CPU cores will be 400MHz; the performance of DSP cores, 3GOPS; the transmission speed of the internal bus, 4 G bytes per second; the display performance for 3-dimensional graphics, 5GFLOPS; the power consumption of an analog / digital converter, 2 mW; the number of logic gates, 5M; and the size of DRAM, 256M to 1G bits. Process lines of CGC and high-end processors are different, because CGC consists of a various kind of cores. Therefore, the clock frequency of CPU cores is a half as compared with high-end processors.

Table 7 Specification of Cores for Cyber-Giga-Chip

Embedded Cores	1997	2002	2007
Process	0.25 μ m	0.18 ~ 0.13 μ m	0.1 ~ 0.07 μ m
CPU	100 MIPS 100 MHz	500 MIPS 400 MHz	3 GIPS 650 MHz
DSP	1 GOPS	3 GOPS	10 GOPS
BUS	3 GB	4 GB	10 GB (Asynchronous)
3 D Graphic (Display)	1 G Flops	5 GFlops	60 GFlops
Analog(ADC)	5 mW (10bit 20MS/s)	2 mW	-
DRAM	64 Mbits	256 M ~ 1Gbits	1 ~ 4 Gbits

MS/s = Mega Sample / second

3.2 Requirements for EDA Technology

This section summarized those requirements for the EDA technology that were obtained via the interview of LSI designers. The design process consists of system design, architecture design, RTL design, logic design, circuit design, layout design, test design, mask design, package design, and testing. It is classified into one group for each design phase and one more group for the entire design.

Table 8 summarizes the EDA requirements obtained from LSI designers. In the table, rows show the design targets and column show the design phases.

Table 8 LSI Design Requirements Table

Category / Design Process	Size	DSM	Speed	Power	Function	TAT	Reliability	IP	Others	Sub-total
Product Spec.	**9**		**3**	**4**	**3**	**2**	**2**	**7**	**5**	**35**
System/Algorithm	**7**	**2**	**6**	**7**	**2**	**19**	**2**	**1**	**4**	**50**
Architecture	**4**	**2**	**6**	**3**	**2**	**8**	**1**	**1**		**27**
RTL	**2**	**1**	**4**	**3**		**5**				**15**
Logic		**2**	**5**	**5**		**4**				**16**
Circuit	**2**	**6**	**5**	**4**		**5**	**2**	**1**	**1**	**26**
Test			**1**	**1**		**1**	**4**	**2**		**9**
Layout	**1**	**7**	**4**	**1**		**1**				**14**
Mask	**1**					**1**				**2**
Package										
Testing				**4**		**3**	**5**		**1**	**13**
Others	**4**		**4**	**1**		**12**	**1**	**4**	**8**	**34**
Sub-total	**30**	**20**	**38**	**33**	**7**	**61**	**17**	**16**	**19**	**241**

The table shows the number of items pointed out as demands.

Requirements in System Design

(1) To date, software, analog circuits, digital circuits and memory are developed independently. Soon, an integrated system (system LSI) which implements all elements on a single chip will be on market. To respond to demands swiftly, optimization methods for those integrated systems will be necessary.

(2) A standard system design description will be necessary. Writing comprehensive specifications is an important problem in system LSI design.

(3) Semiconductor designers may have to develop software for the chip. In such circumstances, program development tools, such as compiler, on-chip debugger, or on-chip ICE (In Circuit Emulator) will be necessary.

(4) The abilities to describe accurately, understand, and verify the specification, are vital to improve the

efficiency of LSI design. In particular, It is important for silicon designers to communicate with system designers. Therefore, it is desirable to develop methods and tools that assist such communications.

(5) It is necessary to develop EDA tools that assist Hardware/Software co-design using hard-cores and soft-cores.

(6) The modeling for CPU cores becomes more important as more functions are implemented on a CPU core. Simulation models incorporating CPU cores or architecture-level simulator for pipeline will be necessary.

(7) Design methods for CPU architecture, to minimize the program size, and communication tools for OS, compiler and LSI engineers will be necessary. Parts for communication will be reused as standard modules. A Hardware/Software co-design environment will be necesssary, for instance high-speed emulator with FPGA that can debug on OS even if a chip is not completed.

(8) It is difficult to improve multi-level circuits and analog circuits, because Vdd mould drop down. The use of analog circuits will be limited as much as possible. Using DC-DC converter, the user interface will be implemented using analog circuits, while inside will be implemented using digital circuits.

(9) High-level design tool such as those for evaluating tradeoff between frequency and the number of gates will be necessary.

(10) A power management system, which controls the internal voltage, will be introduce at various levels. A design tool for power management, such as sleep control, will be necessary. For example, a statement meaning “Shut off the power" should be included in the HDL (Hardware Design Language).

(11) The discrepancy between architecture level and RT level must be resolved immediately. An automatic synthesis tool from architecture level to RT level, or verification tool within the architecture level will be necessary. Moreover tools that automatically inserts latches for pipeline processing will be necessary.

(12) Layout tools, such as a placement and routing tool based on timing profiling simulator, a multi-layer placement and routing modeling and tool , and a tool for analyzing coupled wires, will be necessary.

Performance Estimation

(13) Currently, elements such as software, analog, digital, or memory are designed separately. However, in system LSI design, these are integrated into a chip as a consolidation system. Therefore, the performance estimation technology for consolidated system becomes important.

(14) The performance of a high-speed processor depends heavily on the quality of its implementation. Therefore, performance estimation tools for the pipeline and the cash are necessary.

(15) Power estimation technology in system level CAD (analog/digital total simulation) is also necessary.

(16) Activation rate analysis tools which uses dynamic test patterns are necessary to estimate the power consumption of RAM and ROM.

(17) In embedded system, the demand for product performance is severe such as fanless or finless. Accurate modeling of power consumption is demanded.

Analog Design

(18) Efficiently designing a analog circuit leads to the efficiency improvement of system LSI design. Partitioning technology of analog and digital, total verification of analog/mixed circuit, and analog HDL become important.

(19) Design verification techniques for analog/mixed circuit, design for test, auto test pattern generation, and test pattern description are in new research fields. The development of a new design technology is expected. In particular, testing of the analog/mixed circuits including active components is a research and development object rife with difficult problems.

(20) In analog/mixed circuit, digital noise is a big problem for the analog circuit. It is necessary to handle the noise as a system problem, including the substrate and the package. Analog/digital total simulation to analyze the noise and the noise simulation technology are necessary. Moreover, it is necessary to discuss how much knowledge the system designer should have.

(21) How to speed up the external clock is also important. Speed-up of the interface, design and power supply of the board, and uniting of the internal chip technology and the board technology, are needed. Total simulation technology, including the board, is needed to achieve freedom from skew problems.

(22) The techniques and tools are necessary to lead low power consumption design by appropriately using analog/digital/sensor circuit.

Design Property Reuse

(23) The design reuse technology of the digital circuit and the analog circuit is extremely important. It is necessary that the reused circuit need not be used as it is, but designers are able to modify it.

(24) The enhancement of the modeling technology for CPU core becomes important, because CPU cores will become more multi-functional.

(25) The reuse technology of the processor core will be important. The core of CPU and DSP will be standardized for efficient software development. Property designed in a language can be reused when it is well verified.

(26) Core based design for digital circuit is inevitable. As for the analog circuit, it depends on whether the analog circuit is described in a language. 80% of analog cores can be standardized. Analog circuits, which can not be described in a design language, might not be IP. The standardization of the interface between modules is also indispensable.

(27) It is necessary to standardize the IP interface for consolidation of LSI with DRAM. Correspondence with various memories (DRAM, ROM, Flash) is also important.

(28) The cores of analog circuits cannot be practically reused, unless there is a technology which

automatically generates test patterns to verify the core functions.

(29) Verification technique becomes essential in the case that the core is bought in from outside. Moreover, it is necessary to standardize the inspection method for modules.

(30) It is necessary to promote data standards for EDA tools.

Design Flow Control (Version Control in Team)

(31) A version control mechanism in design and manufacturing management will be necessary.

(32) Large-scale circuits are normally designed by a team of engineers who may be located in different sites. In such circumstances, simulation tools and verification tools for the large-scale circuit assembled from team members' contributions will be necessary. In addition, communication tools and version control tools for team designs will be necessary.

Edit Design/ECO

(33) The prompt response to changes of design is important. Analysis tools and synthesis methods taht focus on the changes, such as incremental simulation and incremental netlist generation, will be necessary.

System Emulation (Including Software Verification)

(34) Prototyping, for example with FPGA, will be necessary to detect software bugs found in device drivers.

(35) A co-simulation tool and a co-verification tool for boards, packages, and controllers (CPU + software) will be necessary.

(36) Chip design and board design should be integrated. An integrated simulator, including printed circuit boards, will be necessary to achieve skew-free design.

Device Model and Characteristic Extraction

(37) With the progress of deep submicron technology, the impact of margin on performance and cost cannot be neglected. Therefore, design methods, resulting a small margin and with accurate performance, estimate will be necessary.

(38) At present, LSI designer must consider constraints on power/voltage supply, temperature, and packages. LSI design method that takes automatic care of those constraints will be necessary.

(39) Synchronous design is limited by the ability to estimate the discrepancy in interconnection delay. Estimate for interconnection delay needs to improve for quality synchronous design.

(40) More control over margin in LSI design will be necessary.

(41) A new breed of methods and technologies will be necessary for LSI running at more than 500 MHz. Tools are required that can handle voltage drop, impact on inductance, and in-line capacity. Verification tools for discrepancy introduced in manufacturing process, models for multi-layer

routing delay, and integrated tools for placement, routing and timing analysis will be necessary.

(42) Precise modeling will be important.

(43) Extensive libraries will be necessary. Libraries will be parameterized (e.g. by Vth and Vdd).

Timing Analysis and Delay Calculation

(44) Timing verification methods to identify locality in behavior will be necessary.

(45) As with large-scale and high-speed LSI, effective description and verification for timing constraints at a system level will be necessary. It must be possible to make each block synchronous, even if the whole system is asynchronous. New methods and tools for such timing constraints will be necessary.

(46) Reduction on power, wiring delay and noise will be the important issues. CAD for wiring delay and for multi-layer interconnection, and a simulation tool that measures wiring after layout will be necessary. Large current switching, parasitic resistance, and inductance should be considered in measuring the effect of noise.

(47) A placement and routing tool based on timing profiling simulator, a multi-layer placement and routing model and tool, an analysis tool for coupling wires, and a layout tool to guarantee AC behavior will be important.

(48) The bottleneck in reading data from DRAM in Logic/Memory merged LSI needs to be resolved. Design methods to control timing depending on the distance form the controller or CPU will be necessary.

(49) Synchronous design is limited by the ability to estimate the discrepancy in interconnection delay. Estimate for interconnection delay needs to improve for quality synchronous design.

(50) A new breed of methods and technologies will be necessary for LSI with more than 500 MHz. Tools are required that can handle voltage drop, impact on inductance, and in-line capacity. Verification tools for discrepancy introduced in the manufacturing process, models for multi-layer routing, and integrated tools for placement, routing and timing analysis will be necessary.

(51) Improved efficiency and more accuracy are required in delay checking. Bug detection for timing-related logic bugs will be necessary.

Asynchronous Circuit Design Automation

(52) Design methods to cope simultaneously with a local synchronous system but a global asynchronous system will be necessary.

(53) A demand for skew-free design is increasing. Design methods not only are skews in a chip handled, but the positions of chips on a printing board are also determined will be necessary .

(54) As with large-scale and high-speed LSI, effective description and verification for timing constraints at a system level will be necessary. It is possible to make each block synchronous, even when the whole system asynchronous. New method and tool for such timing constraints will be necessary.

(55) Verification methods for those circuits that behave synchronously within a block but asynchronously between blocks will be necessary.

(56) An environment to support a synchronous circuit design is urgent.

(57) A chip size of 15 cm x 15 cm is rarely designed in a complete synchronous manner, since the chip would take more than one clock cycle to transmit a signal from one edge to another. Design aid tools for circuits that behave synchronously within a block but asynchronously between blocks will be necessary. Synthesis and verification tools for asynchronous circuits will be necessary.

Formal Verification

(58) Formal verification should be incorporated into simulation.

(59) Large-scale circuits are normally designed by a team of engineers who may be located in different sites. In such circumstances, simulation tools and verification tools for the large-scale circuits assembled from team members' contributions will be necessary. In addition, communication tools and version control tools for a team design will be necessary.

(60) Distributed simulation as well as hardware engine will be necessary for large-scale circuits. Formal verification at function design level will be necessary.

Verification and Test of Analogue/Digital Mixed Circuits

(61) Verification, BIST, automatic test pattern generation, and test pattern description for analog/digital mixture circuits have not been researched. In particular, testing for analog/digital mixture circuit including active components will be one of the most challenging problems.
Design environment and tools to support BIST in analog/digital mixture circuit will be necessary.

(62) New testing and better coverage rate of BIST will be important as the number of pins increases.

Verification and Test in System on Silicon (SOS)

(63) New test methods will be necessary for new implementation methods such as multi-chip module and chip bonding. There remains a fundamental problem in testing before and after bonding. Boundary scanning that covers inter-chip scanning will be necessary.
Design environment and tools to support BIST in analog/digital mixture circuit will be necessary.

(64) New testing and better coverage rate of BIST will be important as the number of pins increases.

High-speed Logic Verification (Speeding Up Logic Simulation)

(65) Logic simulation takes longer as the size of circuits increases. High-speed logic verification will be necessary for large-scale LSI.

User Interface for Function Design (Effective High-level Design)

(66) The use of GUI in LSI design will be important.

(67) Organized documentation for IP will be necessary. Skills and supporting CAD tools for documentation will be necessary.

Voltage Management

(68) Optimization methods and tools for multi-level Vth, multi-level Vdd, and so on will be necessary. These technologies will not be standardized but will be developed independently by manufacturers.

(69) Design methods and tools for voltage management circuits that control the required power/voltage for the given clock frequency are needed.

(70) Tests for voltage management circuits will be necessary.

(71) Extensive libraries parameterized on Vth and Vdd will be necessary.

Clock Control

(72) Data will be read form memory at the rate of 1-2 G byte/sec by 2002. ASIC design to manage interface between memories and skew-free design for the data-bus will be necessary. The tools to support those designs will be necessary, too.

(73) Clock distribution will be complex as LSI technology progresses. Better skew-management methods and tools for clock distribution will be important. CAD that quantitatively analyze crosstalk noise will be necessary.

(74) In clock design, CAD that supports gated clocks will be necessary. CAD that can evaluate the effect of noise in low power design will also be necessary, as well Layout CAD that can handle clock skew.

Noise

(75) Noise is a big problem in digital/analog mixture circuits. A design environment that can deal with noise between analog circuits and digital circuits, noise due to bulk current, or to coupling of high frequency circuits will be necessary. Crosstalk should be considered on chip as well as between chips. Noise simulation will also be necessary.

Power Design

(76) Measurement for power supply drop will be necessary.

Layout (Placement and Routing)

(77) A planar DRAM may be the only practical candidate for DRAM core. A high-speed interface for memory should be standardized. Layout tools that deal with cell capacity and bit-line capacity in

memory will be necessary for DRAM cores.

Process Migration

(78) Technology migration that guarantees behavior of analog circuits will be necessary.

(79) Process-independent design for memory will be necessary. Blocks implemented using old technology should be re-implemented using the new technology in a short period and with a minimum effort.

(80) Memory generation technology and automatic generation technology for peripheral circuits (analog/digital) will be necessary.

(81) Layout CAD that automatically adjusts the size of transistors will be necessary.

(82) Productivity improves significantly if the following methods are established; the method to control errors produced by libraries, library generation that takes account of errors in CAD, and the reuse of analog circuits library. In particular, method to respond promptly to changes of process/devices.

Test

(83) The design that takes account of tradeoff between testing cost and testing reliability will be possible. It is important to devise methods for an increase in testing criteria, for testing of digital/analog mixture circuit by a digital tester, and for self-testing of A/D and D/A. It is necessary to establish the technology to decrease the test cost like the test using fast Fourier transform.

Fault Analysis

(84) Analysis for fault operations will be necessary.

On-board System Verification

(85) The support for board design and tester will be important in improving efficiency.

(86) A processor with 1000-5000 pins on which analog circuits, digital circuits and wireless are all implemented will be on market. Unified noise analysis within a chip or a board for many pins, multi-layers and MCM will be necessary.

(87) Packages and interfaces will be standardized. At least three companies/vendors will be necessary for such standard packages and interface.

(88) A standard interface among macro blocks will be necessary.

Mask Process

(89) Shortening mask processing will be important. A parallel processing is one of key technologies.

Software Design

(90) There remains a problem of development of OS for multi-processors.

(91) Software parts will be important. Software deals with demands for more functions like modem or driver.

Design Description Model

(92) The abilities to describe accurately, understand, and verify the specification, are vital to improve the efficiency of LSI design. In particular, It is important for silicon designers to communicate with system designers. Therefore, it is desirable to develop methods and tools that assist such communications.

Chapter 4 summarizes the above-mentioned design requirements from the viewpoint of the EDA technology. Table 9 shows the correspondence of design requirement items (Chapter 3) and EDA technology problems (Chapter 4). Each item in EDA technology problems of Table 9 corresponds to that in from Table 3 to Table 6.

Table 9 Correspondence of design requirement items and EDA technology problems

	Chapter 3 Design Requirements	Chapter 4 Problems of EDA Technology
1	Requirements in System Design	1A(2),1B(1),1C(1),1D(1),1F(1),1D(3),2B(1),2D(1),2E(2), G(3),G(4),G(5)
2	Performance Estimation	1B(1),2B(1),2B(2),3B(1),3B(2),2D(1)
3	Analog Design	3F(1),3F(2),3F(3),3F(4),5F(1)
4	Design Property Reuse	H(1)
5	Design Flow Control (Version Control in Team)	H(2)
6	Edit Design/ECO	3D(2),3D(4)
7	System Emulation(Including Software Verification)	1C(2)
8	Device Model and Characteristic Extraction	4B(1),4C(2),4C(3)
9	Timing Analysis and Delay Calculation	3B(2),3C(1)
10	Asynchronous Circuit Design Automation	H(3)
11	Formal Verification	1C(3),2C(1),3C(2)
12	Verification and Test of Analog/Digital Mixed Circuits	3F(5)
13	Verification and Test of System on Silicon (SOS)	1E(1),3D(6),3E(1),3C(4)
14	High Speed Logic Verification(Speeding up Logic Simulation)	3C(2),3C(3)
15	User Interface of Functional Design (Effective High-level Design)	H(1)
16	Power Management	3D(5),4D(2),5D(5)
17	Clock Control	3D(1),3D(3)
18	Noise	4C(1),4C(3),5D(2),5D(4)
19	Power Design	3B(3),5D(1)
20	Layout(Placement and Routing)	5A(1),5C(1),5D(2),5D(4),5D(5)
21	Process Migration	4D(1),4D(2),4D(3)
22	Test	1E(1),2E(1),2E(2),5E(1)
23	Fault Analysis	3E(2)
24	On-board System Verification	4C(1),4C(3),
25	Mask Process	5D(4),6D(1)
26	Software Design	G(1),G(2),G(4),G(5)
27	Design Description Model	1A(1),2A(1),2A(2),3A(1),3A(2)

4. Problems and Targets of EDA Technology

In this chapter, we will present the problems and targets of individual EDA technology. We will analyze the requirements from LSI designers, as described in chapter 3, from the aspect of EDA technology. Then we will summarize problems and targets of EDA technology. In the following, we classify them into four categories: digital circuit design, analog circuit design, software design, and the entire design, and how the EDA technology will change from current (1997) to 2002, and after-2002 is described.

4.1 Problems and Targets of EDA Technology in Digital Circuit Design

In this section, we will describe the problems and targets of EDA technology at each stage of system design, architecture design, RTL/logic design, circuit design, layout design, and manufacture interface in digital circuit design.

4.1.1 System Design

Table 10 Problems of EDA Technology in System Design

<table>
<tr><td></td><td colspan="5">Digital</td></tr>
<tr><td></td><td>A Specification</td><td>B Estimation</td><td>C Verification</td><td>D Synthesis</td><td>E Test</td></tr>
<tr><td rowspan="3">1.
System</td><td>1A(1)
Standard System Level Modeling</td><td rowspan="3">1B(1)
Performance Estimation of System
(Application Software/ Compiler/ Hardware)</td><td>1C(1)
System Level Simulation</td><td>1D(1)
Hardware/Software Partitioning</td><td rowspan="3">1E(1)
Test Strategy Decision for System (Hardware /Software)</td></tr>
<tr><td>1A(2)
Standard System Description Language (SLDL etc.)</td><td>1C(2)
System Level Emulation</td><td rowspan="2">1D(2)
System Level Library (IP Core, Middleware)</td></tr>
<tr><td>1C(3)
Formal Verification (System Spec. - System)</td></tr>
</table>

Item	Standard System Level Modeling Position : 1-A(1) Outline : At system level, a model is standardized to express systems. The specification, behavior, function, composition and constraint of a system are described in these models.
Current (1997)	There is no standard model. Specification, behavior, function, composition, and constraint of a system are vaguely expressed using such disunited level and method as the notation, table, description language, and natural language. There is no standard model, though algorithm and behavior of some parts of a system are expressed by VHDL, Verilog HDL, and C language.
2002	A basic model will be established using notation and language to express and define specification, behavior, function, composition, and constraint of a system at system level.
After 2002	A model will be established to express macros and the IP (software and hardware) completely. A system will be expressed completely using an IP model and a basic model. The model corresponds exactly to the system description language and its implementation.

Item	Standard System Level Description Language (SLDL etc.) Position : 1-A(2) Outline : The system level description language should be standardized to describe the specification of a system, including both the hardware and the software. The language is used for performance estimation, verification (including co-simulation), and behavior synthesis of the system.
Current (1997)	There is no standard specification. Both of the programming language such as C and C++, and the hardware description language such as Verilog HDL and VHDL are used in the development process.
2002	Some candidates for the system description language will appear, and tools to compile the language will be further developed.
After 2002	The system specification description language will be standardized, and several venders produce tools to compile the language. Macros with high function will be standardized. IP business for these macros will become activate.

Item	Performance Estimation of System (Application software/Compiler/Hardware) Position: 1-B(1) Outline: Performance analysis technology and automatic optimization technology are necessary for the environment to develop the software for the processor core. A technique to evaluate the performance of algorithms is one of the ways to evaluate tradeoffs in the architecture design phase.
Current (1997)	Simulators and debuggers for general-purpose processors exists, but the level of performance analysis and optimization varies depending on the tools. In general, the simulation environment is not sufficient, though there are speed-up techniques of simulators for a specific processor. The design environment for architecture is not sufficient, and the quality depends strongly on designer's skills. C language is generally used for algorithm evaluation. As for signal processing LSIs, the architectural design environment is put into practical use. There is an active research on a technology to evaluate the performance and efficiency of the architecture in the application software. EDA tools for Hardware/Software codesign (partition and estimation), are in the research level, and they are not yet practical.
2002	The processor architecture will become standardized further and the level of optimization in a compiler will be improved. A parallel processor will be in common use, and its software development technology advances. When we use a core processor, which is either fixed or changeable in terms of the number of registers, a estimation technique will be put to practical use. The estimation technique estimates what kind of user-designed hardware to add to the core processor to satisfy the performance constraint of a system The compiler-compiler technology will be put to practical use for processor and custom hardware.
After 2002	Performance estimation techniques for automatic partition of hardware/software will improve. A parallel processor will be widely used. The optimization of a parallel compiler will become more progress. The technology, which automatically generates the best architecture for a given application, is put to practical use.

Item	System Level Simulation Position: 1-C(1): Outline: System level simulation proceeds one of functional verification technologies assuming that the system level description language is standardized.
Current (1997)	The algorithm verification using C language is main stream. The system behavior simulation with VHDL appears.
2002	A simulator is developed for a system description language. The development of a model for simulators advances. HDL and C are mainly used at a practical level.
After 2002	A simulator for a system description language spreads. The techniques for optimum partition and synthesis of hardware/software are realized.

Item	System Level Emulation Position: 1-C(2) Outline: System level emulation is a technology at system level to emulate the entire system at high speed.
Current (1997)	An emulator based on FPGA is practically used. Although it is more high-speed than a simulation, it is later than a real tip.
2002	The emulator can verify the entire system.
After 2002	A simulator simulates the entire system at high speed by automatic starting a emulator.

Item	Formal Verification (System Specification - System) Position: 1-C(3) Outline: Behavior and constraint of a target system are expressed based on a basic model. A formal verification tool inputs the model and verifies whether it satisfies the condition of specification.
Current (1997)	A formal verification tool is practically used at RT level.
2002	Standardization of a method (language) starts to express behavioral constraints at system level.
After 2002	Formal verification starts to be used at development level.

Item	Hardware/Software Partitioning Position: 1-D(1) Outline: A Hardware/Software partitioning tool partitions hardware and software for a system in consideration of constraints such as area and performance. The tool targets a general-purpose processor widespread as IP.
Current (1997)	Estimation and partitioning are manually performed.
2002	Partitioning of hardware/software are manually performed, and a specification is described in a system level description language. The development of tools for performance estimation from a specification starts. Also, the development of automatic partitioning tools starts.
After 2002	The best solution is obtained by partitioning hardware/software from the system description. Automatic partitioning tools appear.

Item	System Level Library (IP Core, Middleware) Position: 1-D(2) Outline: General-purpose functional blocks are prepared as a library by using a basic model for system expression.
Current (1997)	There is no system level library.
2002	Making of system level library starts based on a system description language.
After 2002	A standard model is prepared for functional blocks frequently used.

Item	Test Strategy Decision for System (Hardware/Software). Position: 1-E(1) Outline: A strategy to confirm behavior of a system and a mechanism to support the decision of policy to confirm its validity are necessary at system level.
Current (1997)	There is no mechanism of test strategy decision for system
2002	A basic test strategy is prepared as a library for each basic component of the system. Formal verification technology is used to confirm the best combination of the components while referring to the library.
After 2002	The library is enhanced.

4.1.2 Architecture Design

Table 11 Problems of EDA Technology in Architecture Design

	Digital				
	A Specification	B Estimation	C Verification	D Synthesis	E Test
2. Architecture	2A(1) Standard Architecture Level Modeling (Including Domain Specific Models)	2B(1) Architecture Level Estimation (Area, Timing, Power, Floorplan)	2C(1) Formal Verification (System - Architecture)	2D(1) Architecture Synthesis	2E(1) Test Strategy Decision for Architecture
	2A(2) Standard Architecture Description Language (Verilog HDL, VHDL etc.)	2B(2) Budgeting (Area, Timing, Power, Floorplan) for RTL	2C(2) Validation / Simulation	2D(2) Co-Synthesis	2E(2) Architecture Level DFT

Item	Standard Architecture Level Modeling (Including Domain Specific Models) Position: 2-A(1) Outline: A standard model is established to express hardware at architecture level. Specification, behavior, function, composition, and constraint of architecture are expressed using this model.
Current (1997)	There is no standard model. There is no unified method to express architecture. Notation and HDL are used in describing architecture of CPU, and data flow graphs and block diagrams are used in describing architecture of signal processing. There exists a research instance to express architecture on a unified level by HDL or GUI. Simulator and synthesis tools at architecture level are also researched.
2002	At architecture level, a model is established using notation and language, and to express specification, behavior, function, composition, and constraint of hardware.
After 2002	A model (pipeline, data flow, component, behavioral description method) is established to express completely various kinds of architecture (CPU, signal processing, and controller) and interfaces become possible from this model to verification and to synthesis.

Item	Standard Architecture Description Language (Verilog HDL, VHDL etc.) Position:2-A(2) Outline: A description language is established to express specification, behavior, function, composition, and constraint for "architecture model"
Current (1997)	There is no standard description method. The description levels are not unified though there is often a case, which describes the architecture of CPU using notation or HDL, and simulates it. There exists a research instance to standardize "architecture model" to unify behavioral description level. Moreover, simulator and synthesis tools at the behavioral description level are researched.
2002	Architecture description method is standardized corresponding to "architecture model" using a sub-set of HDL (VHDL and Verilog HDL).
After 2002	"Hardware model" is established to express various kinds of architecture, and the architecture description based on this model is standardized. Moreover, tools for simulator, synthesis, and formal verification, are developed corresponding to the model.

Item	Architecture Level Estimation (Area, Timing, Power, Floorplan) Position:2-B(1) Outline: Performance, area, and power are estimated at architecture level. This technology enables designers to evaluate tradeoffs of performance, area (floorplan), and power.
Current (1997)	At architecture level, designers estimate performance, area, and power of hardware, based on experience and intuition. Therefore, estimation accuracy depends on the skill of each designer, and the error is large.
2002	Expression of architecture by "architecture model" enables designers to evaluate performance, area, and power. As a result, they can evaluate various tradeoffs of performance, area (floorplan), power, and design the best architecture. They can evaluate relatively the architecture, though the absolute error is from +50% to -30%.
After 2002	Using physical information like synthesis and layout from low level design, they can estimate accurately design objective considering constraints at architecture level. Absolute error is +-20%.

Item	Budgeting (Area, Timing, Power, Floorplan) for RTL Position:2-B(2) Outline: Design constraints of RTL design (performance, areas, and powers) are decided at architecture level. Design constraints are generated using estimation technology.
Current (1997)	A designer decides manually design constraints (performance, area, and power) from a specification. The designer iterates synthesis based on constraints, that is, designs by trial and error.
2002	Design constraints (performance, area, and power) are generated based on estimation technology. As a result, the number of iteration in logic synthesis decreases drastically.
After 2002	Optimum design constraints are generated by improving the accuracy of estimation technology. It becomes unnecessary to iterate logic synthesis.

Item	Formal Verification (System - Architecture) Position:2-C(1) Outline: The equivalence of system and architecture is verified.
Current (1997)	There is no technology to verify the equivalence of system and architecture
2002	Formal verification technique enables designers to verify the equivalence of system and architecture under constraints (or for only a part of the system). In this case, software part of the system is excluded.
After 2002	Formal verification technology enables designers to verify the equivalence of system and architecture. In this case, software part of the system is included.

Item	Validation / Simulation Position:2-C(2) Outline: Validation and simulation technologies are needed at the architecture design stage. System specifications and various design conditions can be validated with the target architecture by a sort of architecture level simulation. Also, the performance of the architecture such as number of execution cycles and pipeline scheme in needed to be evaluated.
Current (1997)	C language based or HDL based simulation is being applied for some restricted systems such as CPU embedded systems. However, it is still not widely applied.
2002	Simulators are available for various kinds of architectures to validate them in terms of checking consistency with the specification and to simulate hardware performance aspects.
After 2002	Validation and simulation technologies applied in architecture design level are improved, so that many design related conditions such as specification, behavioral conditions, consistency with test data, also performance related matters are available to be evaluated at the architecture design stage.

Item	Architecture Synthesis Position:2-D(1) Outline: Architecture synthesis technologies synthesize architectures from behavioral description, algorithm description, or system description of the target system.
Current (1997)	There are research activities to synthesize architectures from behavioral description or algorithm description. However, assumed behaviors or algorithms are very narrowly limited, then there are no practically available tools.
2002	Architecture synthesis technologies become practical level, which synthesize from behavioral description or algorithm description. Also, a synthesis technology from system description is developed for some restricted "System Models".
After 2002	Synthesis from system description becomes available for wide varieties of "System Models".

Item	Co-Synthesis Position:2-D(2) Outline: Co-synthesis tools analyze hardware/software tradeoff for the target system specification description, and generate an architecture under the given parameters for CPU embedded system as well as software compilers.
Current (1997)	Though there are on going research level activities with co-synthesis technologies, those still not reach in practical level.
2002	Co-synthesis technologies become available which cover limited architectures for the system with having some specific CPUs.
After 2002	Co-synthesis technologies cover wide variety of architectures for the system with having any type of CPUs.

Item	Test Strategy Decision for Architecture Position:2-E(1) Outline: The cost for testing is estimated at the architecture design stage. The estimation is taking tradeoff relations among area, power, and speed into consideration. The estimation results help designers to decide an appropriate test strategy in terms of the cost for testing. Further, the estimation also considers tradeoff relation between the cost for testing and how much it is reliable and confident enough. Finally, the estimation needs to become an integrated decision support technology for the test strategy addressing not only selection of DFT techniques but including IP selection.
Current (1997)	Estimation of the cost for testing and the decision for test strategy is relied on the designers with having experienced skill.
2002	The cost for testing is automatically estimated at architecture level with exploring candidate DFT techniques and information of IP specification. A low-cost combination of a DFT technique and an IP is automatically selected in the possible candidates. The selected DFT technique and IP is applied as one of the design tradeoff aspects in an automatic architecture generation.
After 2002	The selected DFT technique and IP during the architecture generation stage are automatically transferred into each following design stage. As a result, test patterns are automatically generated at the end of design stage deriving from information coming down from the former design stages. The estimated cost for testing is applied as one of the tradeoff aspects at hardware/software partitioning.

Item	Architecture Level DFT Position:2-E(2) Outline: The most suitable DFT technique for the target architecture is determined taking tradeoff versus the required time for testing into account. So that, the cost for testing can be optimally minimized by means of some estimation technique at the architecture design stage.
Current (1997)	Currently, a test strategy is decided and associated DFT technique is determined at the gate level design stage, not at architecture design level. Hence, the applied test strategy may not be the most suitable one.
2002	The cost for testing can be saved, because an architecture design level decision supporting technology is established for DFT methodology alternatives such as full scan, partial scan, and BIST.
After 2002	New DFT methodologies such as another self-testing technique for instance are developed and applied. Wide varieties in choosing a test methodology at architecture level further contributes saving the cost for testing.

4.1.3 RTL/Logic Design

Table 12 Problems of EDA Technology in RTL/Logic Design

<table>
<tr><td></td><td colspan="5">Digital</td></tr>
<tr><td></td><td>A Specification</td><td>B Estimation</td><td>C Verification</td><td>D Synthesis</td><td>E Test</td></tr>
<tr><td rowspan="4">3.
RTL/Logic</td><td rowspan="2">3A(1)
Standard RTL (Synthesizable) Modeling (Including Graphical Model)</td><td>3B(1)
Area, Timing , and Power Estimation at RT level</td><td>3C(1)
False path free Timing Analysis</td><td>3D(1)
Timing Driven Synthesis</td><td>3E(1)
Standard DFT Interface for both IPs and inter IPs</td></tr>
<tr><td>3B(2)
Budgeting (Area , Timing, Power, Floorplan) for Logic Synthesis</td><td>3C(2)
Formal Verification (Architecture - RTL)
(RTL - Gate)</td><td>3D(2)
Reverse Synthesis (from Gate to RTL)
3D(3)
RTL Synthesis</td><td>3E(2)
Design for Fault Diagnosis, Multiple Fault Models & Test Methods</td></tr>
<tr><td>3A(2)
Standard RTL Description Language (Verilog HDL, VHDL etc.)</td><td>3B(3)
Power-rail Estimation</td><td>3C(3)
Function/Timing Verification beyond Gate Level Simulation</td><td>3D(4)
Incremental Design Methodology</td><td rowspan="2"></td></tr>
<tr><td></td><td></td><td>3C(4)
Test Pattern Generation for Mixed IP Chip on Function and Timing Verification</td><td>3D(5)
Logic Synthesis with Parameterized Cell/Macro Library (Vdd and Vth)
3D(6)
Timing Budgeting and Function Porting for Mixed IP chip</td></tr>
</table>

Item	Standard RTL (Synthesizable) Modeling (Including Graphical Model) Position:3-A(1) Outline: An RTL Model, including graphical model, is standardized.
Current (1997)	Standardization activity is in progress, but not so aggressive.
2002	Standardization is achieved.
After 2002	A standardized RTL model is widely used.

Item	Standard RTL Description Language (Verilog HDL, VHDL etc.) Position:3-A(2) Outline: A generic RTL standard model other than for logic synthesis is needed.
Current (1997)	There is no standard RTL models except for logic synthesis. On the other hand, standardization of an RTL model is desired as an input for cycle-base simulation, formal verification and so on.
2002	The standardization of the RTL model is achieved.
After 2002	A standardized RTL model is widely used.

Item	Area, Timing , and Power Estimation at RT level Position:3-B(1) Outline: For performance evaluation and optimization of a system, a highly accurate area, timing, and power estimation at RT level is required.
Current (1997)	Although area, timing, and power estimation technique at RT level is developed and applying practically to some extent, estimation results accuracy is not satisfactory because of fast logic synthesis based technique.
2002	With the spread of IP based design style, performance information such as area, timing, and power associated each IP is managed and maintained in a database. As a result, estimation accuracy becomes around 10% or hopefully less.
After 2002	Design stages to apply performance estimation of a system moves upward from RT level. Particularly, estimation techniques at architecture design level progress.

Item	Budgeting(Area, Timing, and Power) for Logic Synthesis Position:3-B(2) Outline: A set of constraints for logic synthesis is generated to meet target performance and specification of a system based on area, timing, and power estimation results at RT level.
Current (1997)	An RT level floorplanner is being practically used. However, currently available RT level floorplanner only analyzes timing constraint by doing fast logic synthesis and following placement, and backannotates the information to RTL design stage. There is still no technique to take care information other than timing, and automatically generating constraints for logic synthesis at the timing of getting entered logic synthesis design stage.
2002	RTL floorplanner becomes practically applied level, and it is partly automated.
After 2002	RTL floorplanner is completely automated.

Item	Power-rail Estimation Position:3-B(3) Outline: Power-rail estimation is an RT level power supply estimation technology. Estimated results are transferred to the layout design stage through floorplan.
Current (1997)	An estimation technology for power consumption at RT level is being practically applied.
2002	Power consumption estimation technologies to be used in early design stages before RT level are developed. The estimation technologies have good enough accuracy to provide selection guidelines or major premise in terms of package selection, power line distribution scheme, and floorplan constraint. Also, the estimation technologies support if the chip has IP cores with multiple Vdd and custom blocks with parameterized clock frequency. Once a candidate power-rail planning is applied, the power estimation technologies estimate power consumption, peak current, joule heat and do thermal analysis. As a result, critically hot spots are indicated and identified. The technologies also support the system performance analysis corresponding to the power planning related design specification changes such as voltage changes and power supply separation schemes changes. Finally, power related constraint conditions are generated for the following design stages.
After 2002	In addition to what the power estimation technologies reach in 2002, automated synthesis becomes available which synthesizes power-rail physical specifications and conditions based on the constraints obtained by power estimation at RTL or higher design stages. Thus, multiple power supply configuration, its supplying scheme to each block, and global power route with tapering width planning are automatically determined. Furthermore, by virtue of getting high accuracy of power estimation in the early design stages, power related design iteration between layout design stage and the early design stages such as RTL or higher one becomes not necessary in most cases.

Item	False path free Timing Analysis Position:3-C(1) Outline: A timing analysis tool may generate paths that do not exist in fact. These paths are called false path. A tool for static timing analysis is improved to remove false paths as much as possible. Consequently, the efficiency of static timing analysis is improved.
Current (1997)	A tool for static timing analysis has already been used practically, and removing a pseudo-error is a problem. There is no means to solve the problem, except that a designer judges in detail. This is a factor to decrease the efficiency of static timing analysis.
2002	False paths are deleted using test pattern at high level. About 80% of false path are deleted.
After 2002	A timing analysis without false path is realized. (False path free timing analysis)

Item	Formal Verification (Architecture - RTL) (RTL - Gate) Position:3-C(2) Outline: Formal verification is still on the stream of quiet extension of the currently available technology. It means the formal verification technique used in practical level today is limited to equivalence checking under several assumptions. Essential improvement of a formal verification tool is desired, such as RTL to gate and RTL to RTL powerful equivalence checking without restrictions.
Current (1997)	Formal verification for gate to gate equivalence checking is now in practically applied level, and commercially available tools are being used. However, it is still difficult to do equivalence checking for RTL to gate, hence no commercially available tools can well support RTL to gate equivalence checking.
2002	RTL to gate equivalence checking becomes practically available level, and therefore widely applied under some limitation, which means the design should be digital and synthesizable HDL. In case of IP core embedded design, the equivalence checking can be available between RTL expression of IP core and gate level expression of peripheral circuits by means of the boundary interface information attached to the IP core.
After 2002	RTL to gate equivalence checking is fully available, and the similar accomplish level is realized for RTL to RTL.

Item	Function/Timing Verification beyond Gate Level Simulation Position:3-C(3) Outline: New logic verification technology beyond the current logic simulation is necessary to significantly shorten the gate level simulation time.
Current (1997)	The "verification crisis" has been coming along with the rapid growth of number of gates (design size), because the gate level simulation time increases in proportion to the size. Coping with the problem, cycle-based simulation and formal verification are being applied for some limited circuitry models and in gate level usage, respectively. However, those approaches can not take timing issues into account, so that static timing analysis tools have to be complementarily used.
2002	As IP based design fashion is getting popular, hard IP evaluation purpose test LSI chips are widely spread. Then, behavioral verification on prototyping boards is applied. This helps verification efforts. On the other hand, RT level formal verification can be applied for large-scale circuit with high speed.
After 2002	IP is widely used. At system level, formal verification and behavioral simulation proceeds.

Item	Test Pattern Generation for Mixed IP Chip on Function and Timing Verification Position:3-C(4) Outline: Inside of the IP is usually difficult to be visible to designers for those functions and performance. Then, it is generally not so easy to validate chip level function and timing validation for the designs with having IPs.
Current (1997)	Although it is difficult and tedious job, but designers have to unavoidably make chip level test patterns, so that functions and timing among IPs can be barely validated.
2002	A simulator is developed which analyzes and then calculates how much functional coverage of IP is guaranteed by the prepared chip level test patterns. Next, by virtue of standardizing interface among IP is established, static timing analysis among IPs is realized and chip level timing validation for IP based chip integration becomes available.
After 2002	Chip level functional verification test patterns can be automatically generated by using standardized functional and also system level descriptions of IPs.

Item	Timing Driven Synthesis Position:3-D(1) Outline: Timing driven synthesis means logic synthesis technique which is able to generate very high-speed circuit, and also technique to control and optimize timing constraints during the logic synthesis.
Current (1997)	100K gate or less is maximum circuit size for logic synthesis within acceptably reasonable turnaround time. Expected reachable timing performance can not be certainly understood without doing logic synthesis. This results lengthy iterations from and to RTL modifications and timing optimization exploration, which is an outstanding design productivity issue.
2002	The logic synthesis and timing optimization tools treat circuit of 1M gate level in practical time. Integrating a logic synthesis and timing optimization enables us to estimate timing with high accuracy. To achieve speed-up, an asynchronous circuit is generated automatically.
After 2002	Logic synthesis and P&R are tightly linked together or combined to a unified design tool, so that timing driven synthesis becomes reality.

Item	Reverse Synthesis (from Gate to RTL) Position:3-D(2) Outline: A design sometimes needs to be changed or modified after the RTL design stage, the feedback for the changes and modifications should be taken back to the RTL design particularly for the purpose of making it easier to reuse. This item is closely related to 3-D(4) "Incremental Design Methodology".
Current (1997)	It is efficiently applied from transistor level to gate level reverse conversion for make it easier for simulation and verification, while reverse conversion from gate level to higher level abstractions are still not in practical use level.
2002	It is still difficult to do reverse conversion from gate level to RTL level without having additional particular information. However, by adding certain information in terms of specific difference between them, the reverse synthesis becomes limited in practical application level.
After 2002	The most important purpose for the reverse synthesis is to be defined, so that problems to be solved become clear.

Item	RTL Synthesis Position:3-D(3) Outline: RTL synthesis points RTL generation and synthesis from architecture or algorithm description.
Current (1997)	There are some commercially available tools which support RTL synthesis for some specific applications such as CPU, DSP and so on.
2002	Modeling technologies for architectural or algorithm description are developed for associated specific applications, then corresponding RTL synthesis tools for each application is to be developed.
After 2002	Generic purpose modeling technologies for architectural and algorithm description are developed for wide variety of applications. RTL synthesis follows that generic purpose modeling be synthesized.

Item	Incremental Design Methodology Position:3-D(4) Outline: An Incremental Design Methodology indicates an organized collections of techniques which reflect design condition changes, resulting by logic synthesis and design analysis etc., into some design stages such as logic synthesis. There is underlying observations in the designers community that a design process can not be always straightforwardly done in top down design fashion even in the future. Though some sort of the current ECO technique extension is considered as a possible solution to cope with the issue, more generic and advanced methodologies are expected.
Current (1997)	For the purpose of a design change order, it is widely applied that after manual corrections or changes in module (or block) level, entire (or chip level) verification follows. However, manual modifications and changes in layout design stage may cause functional or performance related inconsistency with the original higher level design. Many Synthesis tools such as logic synthesis and P&R support an ECO feature, so that above mentioned inconsistency issues can be resolved to some extent.
2002	An ECO synthesis becomes in practical use level and is widely applied. The ECO synthesis completely takes care of a design change order under certain condition, also it supports two-way interface regarding design change information with layout tools, delay calculation and verification tools, and layout verification tools.
After 2002	An ECO synthesis assists designers by providing information on tradeoff conditions for re-synthesis, and re-synthesize the design optimally under the specified tradeoff condition. The tradeoff conditions ranges in two-dimensional exploration, one is area and performance optimization, and the another is TAT for doing re-synthesis. Moreover, each condition determines the size of the design to be re-synthesized. Further, design flow management tools (relating H(2)) manages and controls design changes, version information of the design, and required operation for the changed part tightly linked together with the result of ECO synthesis.

Item	Logic Synthesis with Parameterized Cell/Macro Library (Vdd and Vth) Position:3-D(5) Outline: Logic synthesis supports multi Vdd/Vth.
Current (1997)	There are no well-arranged models which explain interrelation among Vdd/Vth, power consumption, and system frequency. Thus, how to model libraries for logic synthesis and how to synthesize and optimize for multi parametric Vdd system is unresolved.
2002	A model corresponding to parametric Vdd/Vth at high level is defined, developed, and standardized. Logic synthesis technology corresponding to parametric Vdd/Vth is necessary.
After 2002	A logic synthesis tool and a simulator control and optimize power, speed, frequency, and area by dynamic Vdd changing. Technology, which controls Vdd at the OS and programming language level, advances.

Item	Timing Budgeting and Function Porting for Mixed IP chip Position:3-D(6) Outline: The IP based Chip integration design methodology is becoming widely applied, and modeling technique for IPs and its standardization are also in progress. In the future, in order to optimize chip level performance and improve flexibility on IP based integration, various tradeoff controls among IPs are needed. Those are adjustment or budgeting for function and timing in chip level, and generating tradeoff of the function versus timing, also putting resulting changes to each IP.
Current (1997)	The technology that models the function and timing and makes it to the black box is achieved.
2002	Budgeting function among IP cores will be realized. A technology will be developed which supports importing and exporting a part of modules in IP among IPs.
After 2002	A tool will be developed which traces a bug across IPs, in debugging with patterns for verification. The tool judges automatically which IP is cause of the bug.

Item	Standard DFT Interface for both IPs and inter IPs Position:3-E(1) Outline: In the future, LSI will be developed by combining two or more IPs, as the use of IP increase. Therefore, the standardization of the inside of IP or the test specification interface between IP is needed to facilitate test design of the entire LSI.
Current (1997)	Each IP provider has included an original test specification in IP. An IP integrator plans an test specification of LSI from the test specification of each IP. The standardization of the design for test of IP is advanced with VSIA and IEEE P1500.
2002	An interface specification of the design for test of IP is standardized. A test strategy of each core and across cores is automatically decided, and test circuit and test pattern are automatically generated, from the interface specification of the design for test of IP.
After 2002	A test circuit specification for testing a target IP is standardized, and the IP is tested (plug and play) automatically.

Item	Design for Fault Diagnosis, New Fault Models & Test Methods Position:3-E(2) Outline: A fault analysis becomes difficult increasingly as the circuit scale grows. In addition, it is necessary to consider a new fault model. Therefore, the demand for not only technology for detecting the new fault models, but technology for facilitating a fault analysis, rises increasingly.
Current (1997)	A fault analysis tool adopts 0-1 stuck-at fault model, and the fault model covers a part of actual fault on a chip. As for the DC parametric testing, the AC parametric testing and the IDDQ testing, it is difficult to specify the fault location on a chip, though they can judge GO/NOGO. Specifying the fault location is more difficult for test using BIST.
2002	Fault analysis, DFT and ATPG for fault analysis will be established for 0-1 stuck-at fault in a circuit with scan flip-flops. Testing models for the delay/bridge fault are generalized. DFT and ATPG are automatically done. New fault models such as power supply bus, crosstalk and substrate noise will be defined, and a fault simulator will be developed based on the fault models.
After 2002	DFT and ATPG will be automated for a new fault model.

4.1.4 Circuit Design

Table 13 Problems of EDA Technology in Circuit Design

	Digital				
	A Specification	B Estimation	C Verification	D Synthesis	E Test
4. Circuit		4B(1) Process Variation Model (Accurate Circuit Simulation)	4C(1) Power, Noise, Electro-magnetic Analysis	4D(1) Transistor circuit & Layout Synthesis (from RT/Gate Level)	
			4C(2) Accurate Model for DSM Process, Parameter Extraction	4D(2) Parameterized Cell/Macro Library Generation and Simulation with Vdd and Vth Variation	
			4C(3) Timing Verification on Real Environment (consider Package, Board etc)	4D(3) Performance Constraint Driven Process Migration	

Item	Process Variation Model (Accurate Circuit Simulation) Position: 4-B(1) Outline: As design margins become smaller, it is necessary to consider process fluctuation and design tool modeling errors in circuit design technology.
Current (1997)	In general, to consider process fluctuation, the circuit simulation is done at 3 conditions, that is, best, typical, and worst. The parameter fitting on the process evaluation chip (TEG: Test element Group) is done to create the delay models for the 3 conditions. Therefore, it is difficult to simulate the circuit under certain combinations of the process conditions.
2002	A design system will become practical, which links with TCAD (Technology CAD). The system will execute a circuit simulation under various process conditions. This lead to concurrent design of process and circuit.
After 2002	TCAD and circuit simulation will be systemized based on statistical analysis of process fluctuation. Then, the system will be widely used.

Item	Power, Noise, Electro-magnetic Analysis Position: 4-C(1) Outline: The accuracy of analysis on circuit characteristic change by internal heat of a chip, crosstalk, IR drop, and switching noise is improved. Electromagnetic field noise (EMC,EMI) estimation technology proceeds.
Current (1997)	The value and distribution of heat begin to be estimated practically from the power consumption. Power supply voltage drop can be simulated. Crosstalk noise can be extracted for only a part of circuit, because of TAT restrictions. The simulation for static analysis of simultaneous switching noise can be done considering the package. As for the simulation of the electromagnetic field noise, it is possible at board design level, but is impossible at chip level.
2002	The simulation of the temperature distribution for a chip becomes possible from the power consumption, and the simulation for the circuit design to include the effect of heat becomes possible. IR drops and simultaneous switching noise for the whole chip will be simulated. Chip level simulation of electromagnetic field noise will become possible.
After 2002	The crosstalk noise will be analyzed for the whole chip. Chip design methodology to decrease the electromagnetic field noise will be established. The speed and accuracy of simulators for the simultaneous switching noise and IR drop will improve.

Item	Accurate Model for DSM Process, Parameter Extraction Position: 4-C(2) Outline: Parameter extraction is a technology for performing accurate extraction of parasitic elements and to compress the extracted circuit in realistic time while maintaining simulation accuracy. The accurate model for DSM process a circuit modeling technology with high accuracy, which corresponds to process technology in deep submicron era. A technology for the accurate modeling of delay and circuit characteristics of IP cores is necessary to reuse the cores effectively.
Current (1997)	The extraction of a parasitic element in two-dimensional space is put to practical use, and that in three-dimensional space begins to be put to practical use. However, both accuracy and TAT are not satisfied. The design error is decided based on min-max policy. As deep submicron technology proceeds and the ratio in an uncertain part increases, it is difficult to realize the design satisfying the constraint. Highly accurate device models (BSIM3) are becoming standard. Many SPICE simulators have started supporting BSIM3. Compression technology achieves practical accuracy and practical TAT.
2002	Three-dimensional parasitic elements will be extracted with a high degree of accuracy. Device modeling will be based on actual devices, and the parameter of the model will be extracted from actual process, corresponding to deep submicron technology.
After 2002	The accuracy of verification and extraction will be improved utilizing estimation technology. Evolution of process and model will be done. The design of process and product will be done concurrently.

Item	Timing Verification on Real Environment (consider Package, Board etc) Position: 4-C(3) Outline: The clock frequency inside and outside a chip will be speeded up, total simulation technology including the simulation of elements outside the chip will be necessary.
Current (1997)	There are many demands for a board level simulation, but it is not possible to execute board level simulation many times, because of execution time limitations. Board level noise simulation is becoming possible because of the standardization of IBIS.
2002	Chip level behavioral modeling will be widely used, and the IBIS model will be standardized. This will make board level behavioral and noise simulation possible.
After 2002	The behavioral verification including software will become possible.

Item	Transistor circuit & Layout Synthesis (from RT/Gate Level) Position: 4-D(1) Outline: The circuit and layout at transistor level are synthesized from the RT/logic level description. EDA technology for handling dynamic circuits is required. The circuit and layout at transistor level are synthesized directly from description at RT/logic level.
Current (1997)	Currently a designer performs transistor level design by trial and error. As for layout, automatic generation of basic cells and generation with templates of the specific macro cells such as the memory are already in practical use.
2002	A technology for automatically generating a suitable circuit and performing cell layout for the application and target specification, such as pass-transistor logic, will be put to practical use.
After 2002	A technology for an optimized circuit and cell layout for the application and target specification from the aspect of the chip design will be put to practical use.

Item	Parameterized Cell/Macro Library Generation and Simulation with Vdd and Vth Variation Position: 4-D(2) Outline: Generation technology to create a cell library that is optimized for the target Vdd and Vth. Simulation technology to verify a circuit that contains multi Vdd and Vth combinations.
Current (1997)	A designer judges tradeoff such as the speed and the power, and selects proper Vdd. A technology that combines the timing analysis with the selection of proper voltage appears. There is no EDA technology, which handles the effect of Vth on leak power.
2002	A simulation model will be standardized which handles parametric Vdd/Vth High speed characterization of cell libraries with parameterized of Vdd/Vth. Simulation technology which enables validation of power consumption and substrate bias even when the circuit has been stopped. Simulation technology which will execute even if the Vdd/Vth parameters are not fixed.
After 2002	Dynamically variable Vdd will enable control power, speed, frequency and area.

Item	Performance Constraint Driven Process Migration Position: 4-D(3) Outline: Design reuse technology to migrate the existing circuit and layout into the target process, which maintaining the performance constraint.
Current (1997)	The synthesis of layout cells from the SPICE netlist, the design rules, and the cell generation specification, such as height and topology, is beginning to be put into practical use. The cells are migrated to another process by changing the design rules.
2002	Integration of layout pattern conversion and extraction technology. During cell or module migration, the oversizing and shrinking of the layout pattern and characterization will be done concurrently.
After 2002	Links to TCAD will enable optimum device design rules to be automatically generated from the process parameters.

4.1.5 Layout Design

Table 14 Problems of EDA Technology in Layout Design

	Digital				
	A Specification	B Estimation	C Verification	D Synthesis	E Test
5. Layout	5A(1) Standard Physical Description Language		5C(1) High-speed Verification	5D(1) Power-rail Routing 5D(2) Simulation based Layout Synthesis 5D(3) High speed Mask Data Processing 5D(4) Decision Support for Multi-Layer Routing 5D(5) Layout Synthesis with Parameterized Cell/Macro Library (Vdd and Vth)	5E(1) Layout Design for Test

Item	Standard Physical Description Language Position: 5-A(1) Outline: Physical description language including transistor level and gate level, which is to be common interface between logic synthesis system and layout system, is required.
Current (1997)	PDEF and DEF are becoming standard interfaces of the floorplan and layout information respectively. Description language depends on each layout tool. A few systems use transistor level description language.
2002	The standardization of the layout description interface advances. It enables us to describe common input information for logic synthesis system and layout system, and to feedback layout information to logic synthesis.
After 2002	Common interface for layout specification and information of cell generation, logic synthesis, and logic simulation is standardized. The effort of the re-simulation is greatly reduced at ECO.

Item	High-speed Verification Position: 5-C(1) Outline: A more high-speed verification method is requested to solve the problem of the increased verification time according to the LSI scale in number of gates.
Current (1997)	There was no dramatic change of verification method itself. Tuning of boolean algorithm and improvement of machine performance has been restrained the explosion of verification time. Recently hierarchical processing is put to practical use for the solution to speed and memory problem.
2002	Parallel processing is put to practical use. New algorithm, which corresponds to the area division, is developed, and computer environment for efficient parallel processing is prepared.
After 2002	The data compression technology, which focuses on the characteristics of data (regularity etc.), advances.

Item	Power-rail Routing Position: 5-D(1) Outline: EDA technology which automates layout of power supply line in consideration of voltage drop and electro migration, etc.
Current (1997)	The technology which observes the voltage drop and the peak current by extracting and simulating the RC network after the layout is completed, is being put to practical use.
2002	The voltage drop and the peak current are presumed based on the power-supply voltage and the operation frequency of each functional block decided by the planning, and an actual power supply wiring route candidate is generated.
After 2002	The power line is generated automatically under the restriction mentioned above, optimizing layer/width/branch and via. Other signals are routed avoiding power line automatically.

Item	Simulation based Layout Synthesis Position: 5-D(2) Outline: Layout technologies, which consider not only speed and power but also noise, crosstalk, inductance, etc. Since accurate prediction of noise and crosstalk before actual layout is impossible, consideration of them is needed in the layout phase.
Current (1997)	Delay is estimated based on logical wire length in logic synthesis, and constraints generated from it are used in layout tool when placement and routing. Logic is optimized based on backannotated actual wire length after layout. Buffer insertion and gate sizing which are constraint-basis are available in layout. A routing tool, which improves delay and reduces noise for specified signals, is emerging.
2002	Layout in which timing, power, and noise are completely assured is generated by controlling gate size, wire width, space between wires based on the concurrent analysis of timing, power, and noise in a layout tool.
After 2002	A necessary part of layout is optimally modified to solve problems of noise, and etc. by interpreting the simulation result. Circuit structure is partly modified keeping the function the same, if necessary. In addition, the consideration of a statistical process variation model becomes possible.

Item	High speed Mask Data Processing Position: 5-D(3) Outline: Speed-up technology of processing and generation of mask data which is becoming huge in size. Since the time required for processing and generation of mask data becomes huge according to decreasing feature size and increasing chip scale, it is necessary to solve the problem by some new EDA technology.
Current (1997)	Hierarchical, distributed, and parallel processing techniques are now used, but improvement far beyond them is necessary.
2002	A tool, which processes data incrementally and optimally in sequence, will emerge.
After 2002	The repetition processing etc. are taken by the characteristics of data (regularity, etc.).

Item	Decision Support for Multi-Layer Routing Position: 5-D(4) Outline: The manufacturing technology of the multi layer routing has already been established, and the strategic layer assignment will become important in the future.
Current (1997)	Designer specifies some layer assignment for special signals such as power and clock.
2002	Application programs select best layer/width/spacing for each logical net group automatically.
After 2002	The optimal number of wiring layers is selected in consideration of noise, heat, speed, current, and material, etc. based on estimation of cost, manufacturing TAT, performance, and reliability, etc. in the first stage of the design, and the layer assignment is planned.

Item	Layout Synthesis with Parameterized Cell/Macro Library (Vdd and Vth) Position: 5-D(5) Outline: Physical EDA technology for circuit with parameterized Vdd/Vth and voltage converter.
Current (1997)	Placement and routing with parameterized Vdd library is fundamentally possible. But it does not become the general technology as regards the restriction of chip structure and the relation to timing driven layout.
2002	The EDA tool assumed to be able to use the same layout module with different Vdd/Vth for the low power consumption is developed.
After 2002	The system described above is put to practical use with the circuit synthesis and the simulation library generation.

Item	Layout Design for Test Position: 5-E(1) Outline: Layout technology, which supports the improvement of fault coverage and takes care of problems caused by DFT. Moreover, the optimal layout technique in the DFT part is included.
Current (1997)	Scan path re-ordering method, which reduces the wire length of scan path, is applied in layout phase.
2002	The layout technique, which takes care of electro migration and voltage drop caused by scanning operation, is established. In addition, the technology, which supports the fault coverage improvement of the IDDQ test by dividing the power supply, is put to practical use.
After 2002	Various DFT technologies are established and restrictions are imposed on placement and timing. The automatic layout technologies, which take care of such restrictions, are put to practical use.

4.1.6 Manufacture Interface

Table 15 Problems of EDA Technology in Manufacture Interface

	Digital				
	A Specification	B Estimation	C Verification	D Synthesis	E Test
6. Manufacture Interface				6D(1) Interface Technology for Mask Manufacturing	

Item	Interface Technology for Mask Manufacturing Position: 6D(1) Outline:
Current (1997)	A detailed analysis is put off this time.
2002	
After 2002	

4.2 Problems and Targets of EDA Technology in Analog Circuit Design

Table 16 Problems of EDA Technology in Analog Circuit Design

	F Analog
1. System	
2. Architecture	
3. RTL/Logic	3F(1) Standard Analog Modeling (for Mixed Signal Simulation)
	3F(2) Analog/Digital Mixed Signal Simulation
	3F(3) System Level Analog Modeling
	3F(4) Synthesis from AHDL
4. Circuit	
5. Layout	5F(1) Analog Cell Generation
6. Manufacture Interface	

Item	Analog Modeling (for Mixed Signal Simulation) Position: 3-F(1) Outline: Modeling for verification of analog circuits.
Current (1997)	CMOS device models BSIM1: model for digital circuits. BSIM2: model in consideration of analog behavior. However, the accuracy is insufficient because the model is expressed as piece-wise linear functions. BSIM3: improved model in which first-order derivative discontinuity problems have been drastically reduced. It covers up devices down to about 0.2μm.
2002	Radio frequency and short channel effects will be included in the model. New modeling methodology for fast simulation would be developed in a bottom-up fashion.
After 2002	System for post-layout extraction of substrate parasitic devices will emerge for noise simulation between digital and analog parts of the circuit.

Item	Analog/Digital Mixed Signal Simulation Position: 3-F(2) Outline: Analog/Digital Mixed Signal Simulation Technology.
Current (1997)	The upper limit for SPICE-based simulation is about 100,000 transistors with 10 to 100 clocks. There is no practical verification method of analog circuits except for the SPICE-based method. In an analog simulator for a digital circuit, waveform relaxation and circuit reduction technologies will be realized for practical use.
2002	The analog mixed signal circuit simulation environment is indispensable for SOC devices. There will be a move to a mixed analog/digital behavioural language and simulation environment, in order to perform system level verification. In the areas of radio frequency and sub-micron devices, the feedback of layout information becomes indispensable, consequently the number of analytical elements and nodes increase explosively. Substrate parasitic device modeling methodology or circuit reduction technique will be developed.
After 2002	Brand new mixed signal design environments will emerge, which surpasses a patchwork of conventional tools. For instance, the simulator automatically partitions a mixed signal circuit and applies the most appropriate algorithm to each part of the circuit.

Item	System Level Analog Modeling Position: 3-F(3) Outline: An analog behavior description language is standardized for the synthesis and verification of analog circuits.
Current (1997)	Two kinds of language standardization for AHDL (Analog Hardware Description Language) are under way: (AMS: Analog Mixed Signal) VHDL-AMS: IEEE Standardization is achieved. Verilog-AMS: Standardization drafts are being discussed in OVI (Open Verilog International). The following simulators are available at the end of 1997: VHDL-AMS: Only parser is available. Verilog-AMS: Only analogue part is supported.
2002	Commercial simulators will be developed by EDA venders. Each vender will develop a distinct analytical technique for the standard analog behavior description languages.
After 2002	There is no definition for the frequency domain in Verilog-AMS. The expressions of frequency domain, radio frequency and Z space will be standardized in the analog behavior description language. Simulators, which can handle these expressions, will be developed.

Item	Synthesis from AHDL Position: 3-F(4) Outline: A circuit synthesis tool supports an analog hardware description language (AHDL).
Current (1997)	There is no practical tool for analog circuit synthesis. In the area of specific circuit design, such as filters and operational amplifiers, circuit optimization tools are used. The designer is responsible for selecting a topology, and the optimizer optimizes the parameters.
2002	Use of analog IP will become practical. A digital IP is supplied and synthesized automatically at RT level, however, the analog IP will be provided as a flexible hard macro. The data for analog IP, including layout information, will be optimized according to the user specification.
After 2002	An expert system for analog synthesis using knowledge-based technology will be developed.

Item	Analog Cell Generation Position: 5-F(1) Outline: An analog cell generator generates layout cells from a netlist at transistor level. The layout cells can be used in digital circuits. Opinion among designers regarding analog cell synthesis is split into: "It is absolutely necessary to use it" and "Use should be kept to an absolute minimum".
Current (1997)	An analog cell synthesis is being used in some operational amplifier designs. There are still some issues in performance and layout size.
2002	Analog cell synthesis used in specific circuits such as operational amplifiers, filters, and AD/DA converters. The performance and area size of analog cells will attain practical levels.
After 2002	An analog cell generator will be used in cooperation with analog synthesis tools. As a result, analog cells with hundreds of elements will be optimized and generated automatically.

4.3 Problems and Targets of EDA Technology in Software Design

Table 17 Problems of EDA Technology in Software Design

G Software
G(1) OS Generation/ Customization
G(2) Software Core Generation/ Customization
G(3) Software Compiler Generation
G(4) Unified Software Development Platform for various IP Cores
G(5) Software/Hardware Co-Simulation

Item	OS Generation/ Customization Position: G(1) Outline: Implementation and optimization technique for OS that runs on a processor core are necessary.
Current (1997)	The processor core on a chip restricts the available OS to be installed on it. There is a variety of “available” OS that is offered by processor vendors and free software such as ITRON. It is very rare to customize OS for the processor core, and the customization is only done by manual.
2002	OS customization becomes easy as the object-oriented OS becomes available. OS performance tuning becomes feasible with the progress of Software/Hardware co-simulation techniques.
After 2002	The relationship between OS and hard IPs will be clearly defined through the standardization of OS for system LSI. This leads to an efficient customization of the system for a specific application by eliminating unnecessary hardware and software functions.

Item	Software Core Generation/ Customization Position: G(2) Outline: Development and optimization method for application that runs on a processor core advanced.
Current (1997)	Applications are written in C language or even in assembly language. Programmers must deal with any changes in a program necessary for a processor core.
2002	Software functions common in multimedia applications will be made middleware and be realized on LSI. A part of middleware is supplied with core and OS. Customization for the software becomes easy as the software is well classified and made with object-oriented techniques. Most of applications will be written in high-level languages, then the object codes are optimized by compiler rather than manual. Verification, performance evaluation and customization for applications will be easy through a simulator for hardware, OS and software altogether.
After 2002	A system will be specified in a system description language and methods to generate and optimize the application from the system specification will be researched. A general-purpose middleware can be customized so that only includes necessary functions.

Item	Software Compiler Generation Position: G(3) Outline: A compiler is generated automatically for a processor core with a “standardized” architecture.
Current (1997)	Each processor requires its own brand of compiler, simulator debugger and ICE. More applications are written in C than in assembly language. Portability of a processor core is difficult because of differences in expression of its I/O. Software development tools with static or dynamic analysis has been introduced. However, optimization for a particular processor depends on the design environment such as compiler performance.
2002	A retargetable compiler will be on market. It will make the processor designs specific to application as well as the software optimization for a processor easy.
After 2002	A parallel compiler at thread level will be widely used and the large-scale parallel system will be developed.

Item	Unified Software Development Platform for various IP Cores Position: G(4) Outline: An integrated framework for software development platform will support various hardware cores.
Current (1997)	A different software development platform is used for each IP core. A unified software development platform that supports several IP cores has been proposed.
2002	The unified development platform will be standardized as the OS for system LSI will be widely used and standardized.
After 2002	Software development platform will be integrated into a system development platform. The automatic translation from a system description to software will be researched.

Item	Software/Hardware Co-Simulation Position: G(5) Outline: Co-Simulation technique for verifying the operations of hardware, OS and software on a system LSI.
Current (1997)	There are some simulators for interface design between hardware written in C and software. Logic simulators link to software parts' simulation by passing signal values such as 0 and 1. It takes long time to prepare behavioral description. The time range that the simulator can simulate is too narrow.
2002	There will be a simulation model for hard IP and co-simulation will be much more feasible. The communication between software and hardware will be much faster.
After 2002	A system will be specified in a system description language. Top-down design of both software and hardware will be used. Co-simulation will be quite popular in system LSI designs.

4.4 Problems and Targets of EDA Technology in Entire Design

Table 18 Problems of EDA Technology in Entire Design

H Entire design
H(1) Document generation from RTL
H(2) Design Flow Management
H(3) Asynchronous Circuit Design

Item	Document generation from RTL Position: H(1) Outline: Reuse of design property is an important problem. If we add document, which is generated from RTL in consideration of standardization, to a specification, the specification becomes clearer and corresponds with the real design. This makes reuse of design property easy.
Current (1997)	There exists a tool to generate flow chart or state machine description from RTL. However, the flow chart and the state machine description are unclear as a document for a designer. The original RTL description itself lacks the high level information method from RTL code is necessary.
2002	An ESDA tool spreads, and supports a designer making a document by the improvement like the interactive function for the documentation.
After 2002	When a system level description is given, RTL is used to verify consistency and to add information of making it detail. The technology of a reverse synthesis (Relate to 3-D (2)) is used limitedly.

Item	Design Flow Management Position: H(2) Outline: A design flow management tool displays graphically a design flow, progress situation of design flow, and situation of making design data, and prohibits process violating the design flow. The tool manages the version of design data.
Current (1997)	The process and style of design depend on a design team. There are some commercial tools for managing a design flow. The tools are not utilized because they are not feasible with relation to the process and style of design. An ESDA tool manages design property and links to a synthesis tool and a simulator. A tool, which runs on PC, for controlling schedule and design flow begins to be used. However, the tool does not link directly to EDA tools.
2002	A tool for managing a schedule and a design flow links to the design environment. An ESDA tool will be a kernel of the design environment, and will control the entire design flow. The management of design data will be standardized, though the interface of an EDA tool is customized individually.
After 2002	EDA tools and version control tools will be prepared as a component, which are combined for each project. The standardization of data formats is necessary for realizing the combination of the component. The interface of an EDA tool will be standardized. An EDA tool judges the type of input data (design progress), and executes the process corresponding to the type. EDA tools will be easily built in a standard control system.

Item	Asynchronous Circuit Design Position: H(3) Outline: EDA tools for asynchronous circuit design is necessary.
Current (1997)	There exists a research instance of an automatic synthesis based on data flow model for an asynchronous circuit.
2002	A tool for multiphase clock design appears. A circuit implementation method is established by combination of synchronous and asynchronous circuits. As for a design method, synchronous design is adopted for each block, and asynchronous design is adopted for a part of chip such as an interface part.
After 2002	Technological development advances to use properly synchronous and asynchronous circuit according to the usage.

5. EDA Technology Roadmap

We described the transition of each fundamental technology and the environment surrounding EDA technology in previous chapter. Here, we present a global EDA technology roadmap. First, we show the design flow of Cyber-Giga-Chip, and problems and objectives of EDA technology in each design phase. Next, we present EDA technology roadmap for Cyber-Giga-Chip in consumer electronics field.

5.1 EDA Technology Roadmap for Cyber-Giga-Chip

5.1.1 Design flow of Cyber-Giga-Chip

If we design Cyber-Giga-Chip at the present days (1997), the design flow is as shown in Figure 6. EDA tools are practically used at lower level than RT level design. Some commercial tools appear for hardware / software codesign. System design and architecture design are being done manually. The designers can use EDA tools downstream from RT level design. As for high level design, some commercial tools for hardware / software codesign appear.

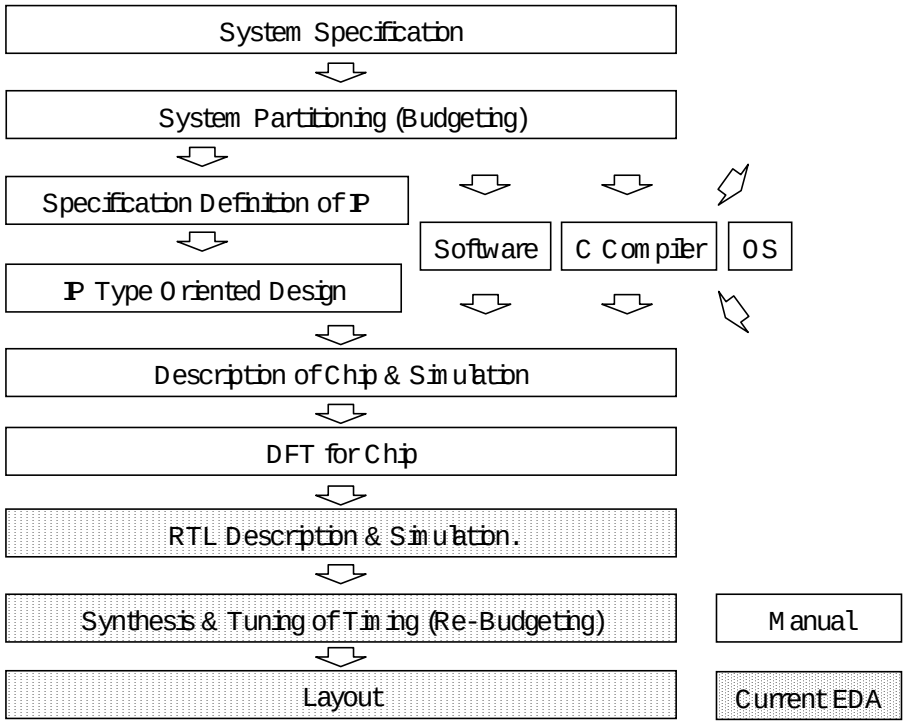


Figure 6 Design Flow of Cyber-Giga-Chip in 1997

Figure 7 shows a typical design flow of CGC in 2002. We add indices to the flowchart to identify the problems of EDA technology, described in chapter 4. New EDA tools are introduced in the design phase of system level and architecture level. A system level description language is practically used, and hardware/software codesign technology advances.

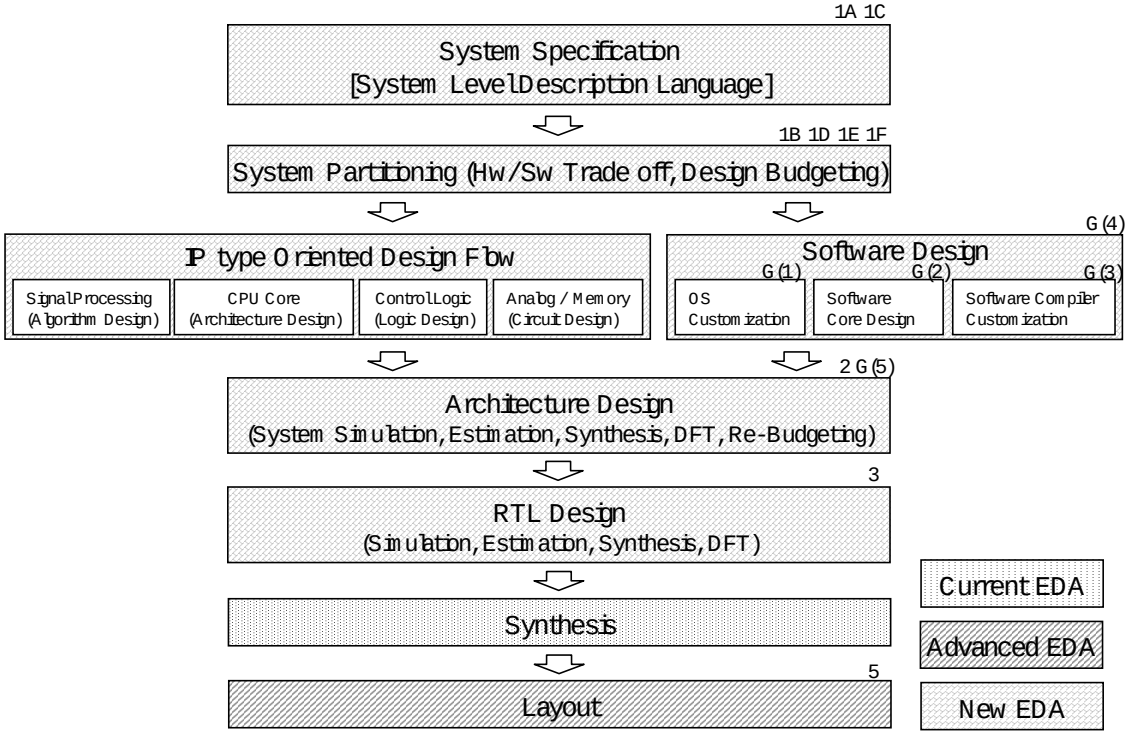


Figure 7 Design Flow of Cyber-Giga-Chip in 2002

In the following, we show the design flow of each core in Cyber-Giga-Chip in 2002. Three cores (processor core, digital signal processing core, and control logic core) are especially discussed.

A design flow of CPU cores is shown in Figure 8. EDA tools are newly developed for hardware/software co-simulation. A tool simulates hardware and software simultaneously based on architecture information described at instruction set level. Also, tools supporting architecture synthesis, verification, performance estimation, and design for test, are developed.

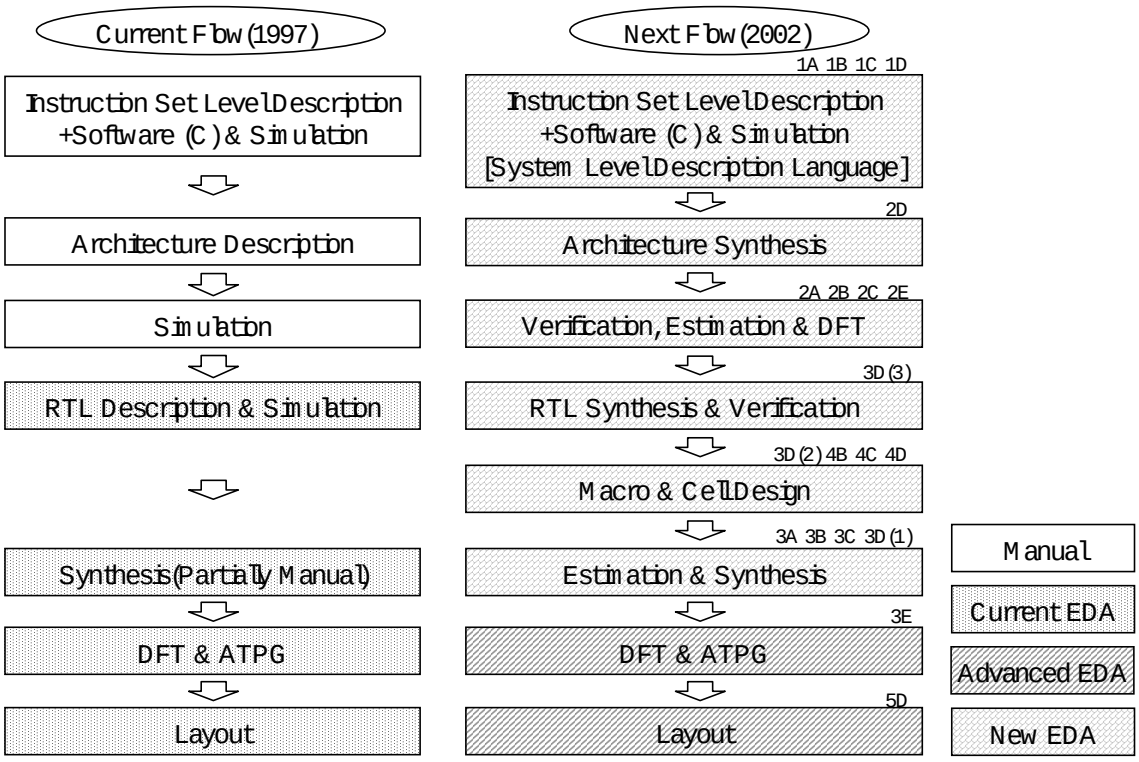


Figure 8 Design Flow of CPU Cores

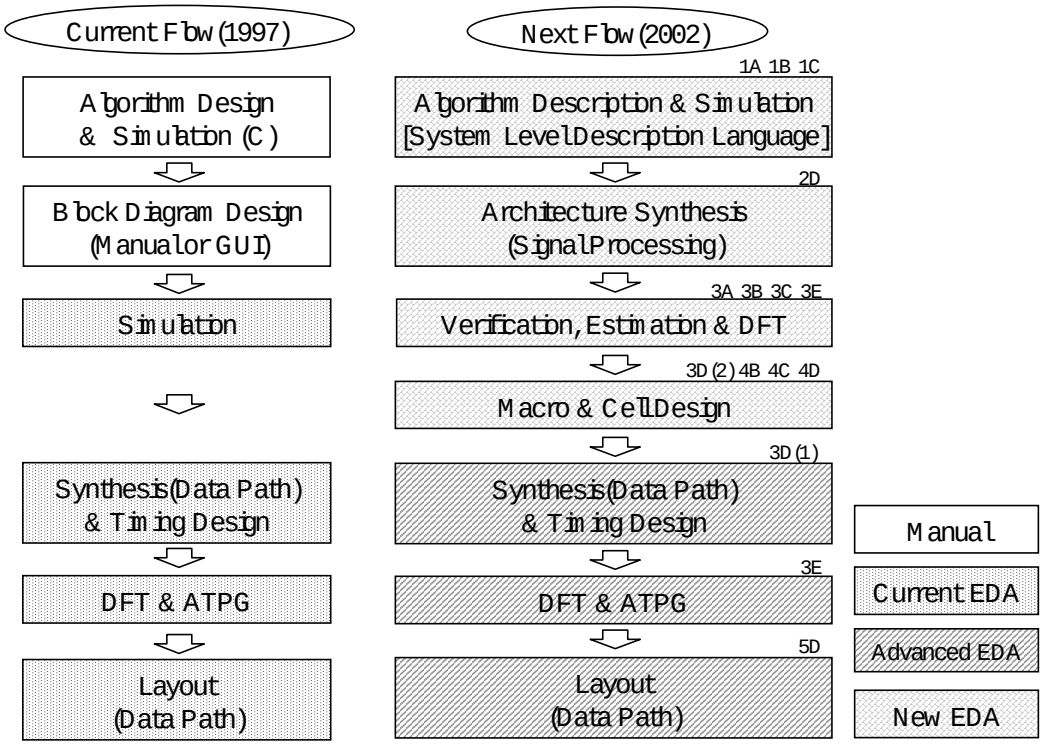


Figure 9 Design Flow of Digital Signal Processing Cores

Figure 9 shows a design flow of digital signal processing cores. EDA tools will be developed to support system design. An algorithm for signal processing is described in a system level description language. Then, the architecture is synthesized to execute efficiently the algorithm. EDA tools for verification, estimation, and design for test, are put to practical use in RTL design.

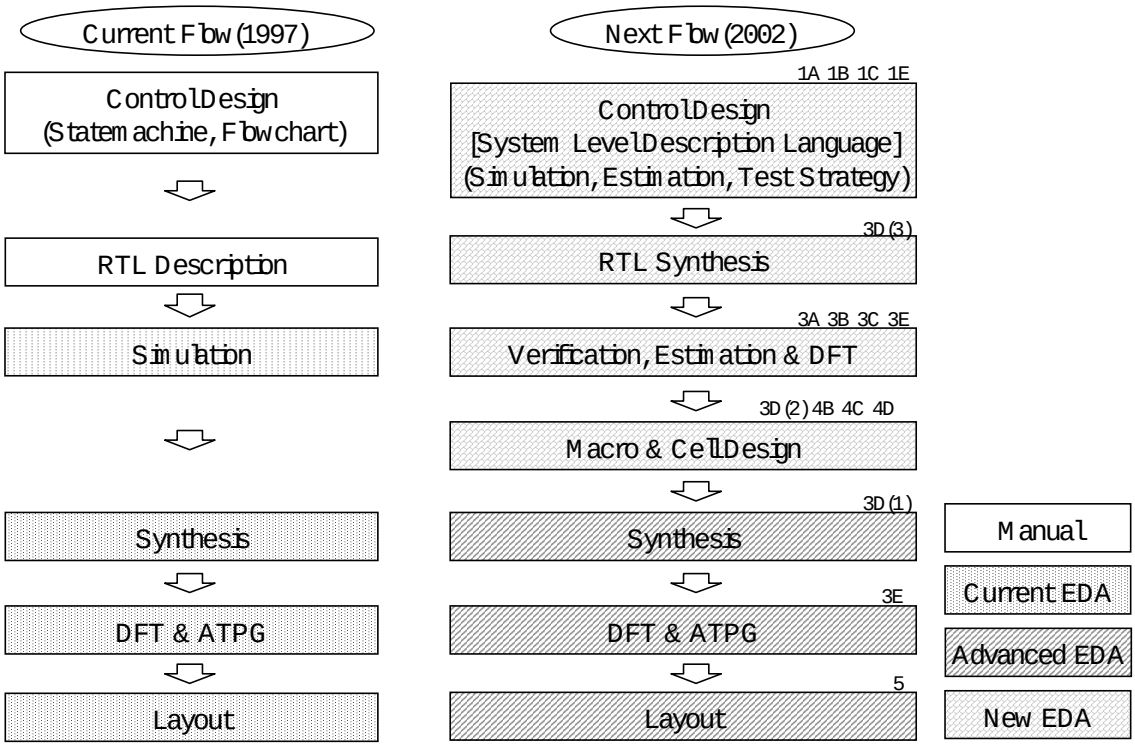


Figure 10 Design Flow of Controller Cores

We show a design flow of controller logic cores in Figure 10. At system level design of controller logic cores, EDA tools advances as well as digital signal processing cores. A control flow such as a state machine is described in a system level description language. EDA tools for simulation, estimation, and test strategy decision, verify whether the controller logic works correctly. Then logic synthesis and layout are done based on design for test.

5.1.2 EDA Technology Roadmap for Cyber-Giga-Chip

Table 19 and Table 20 show EDA technologies to make the design of Cyber-Giga-Chip efficiently and how the technology changes in next five years and more.

Table 19 Transition of EDA Technology (1)

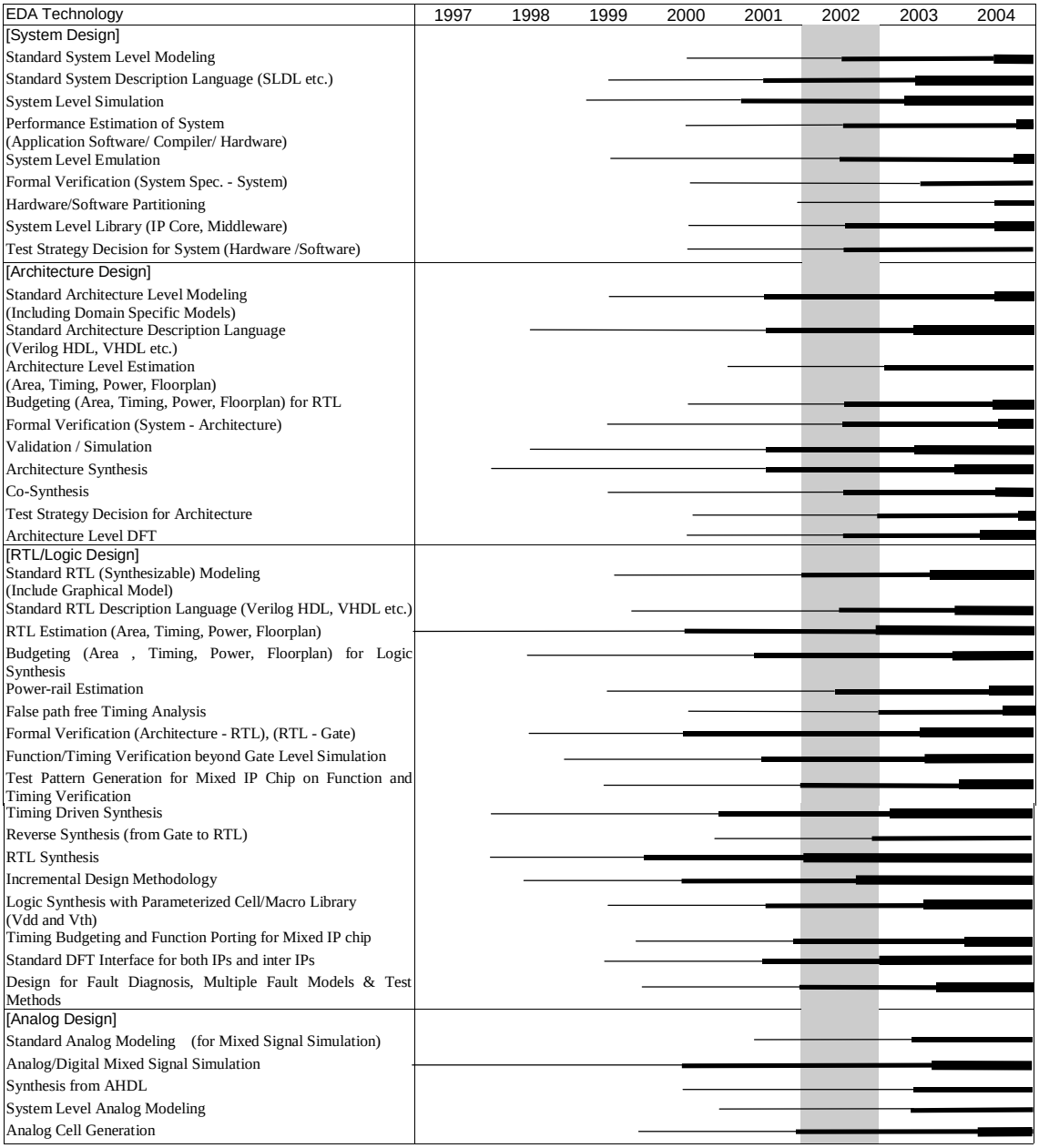
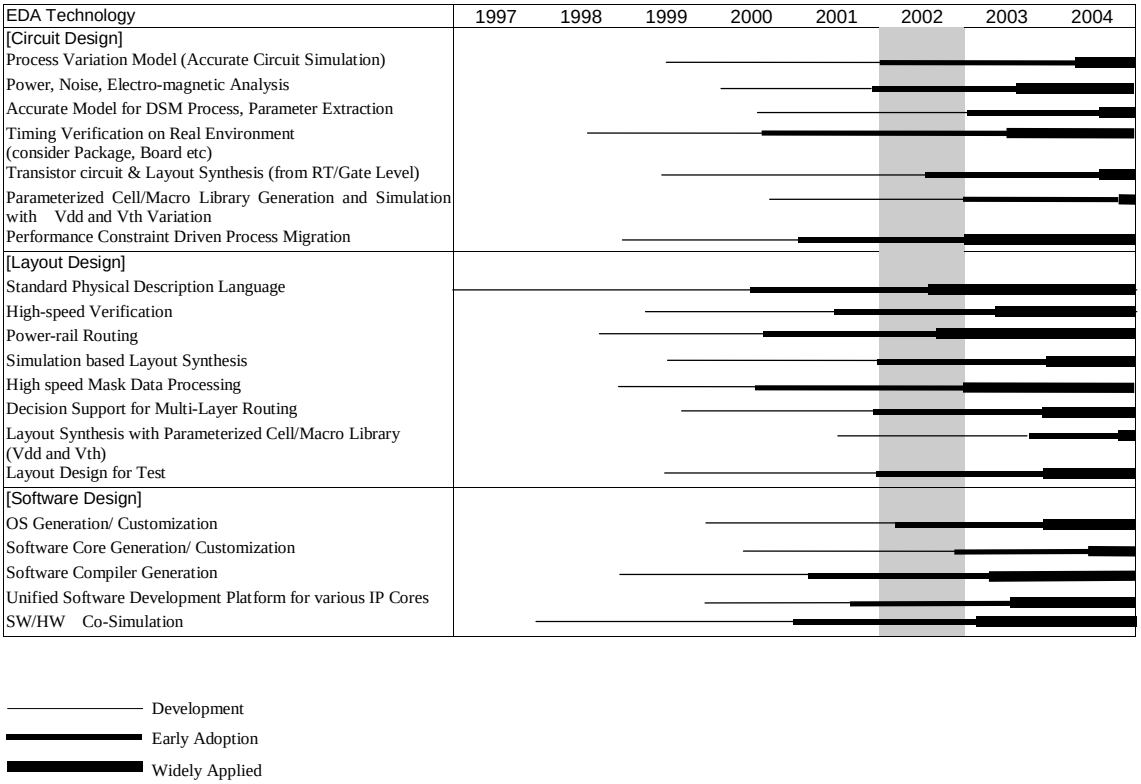


Table 20 Transition of EDA Technology (2)



5.2 Cyber-Giga-Chip for Consumer Electronics

Here, we target consumer electronics as a designing product, and we draw a roadmap of EDA technology assuming some conditions. If the conditions are satisfied, this roadmap may be realized.

1) Assumed conditions

We target information system consumer electronics, which need multi-function and are frequently changed to meet the demands. Details of the specification are not fixed until late in design phase and even after introduction to the market, minor modification may be required frequently. The time between design start and introduction to the market is usually only three to six months. To assign experienced LSI designers as development staffs is difficult. Process technology being used is by and large stable. Technology of mixing memory and logic can be used. Demand to cost reduction and power consumption is much stricter than that to performance. Similar products are commonly developed.

2) Assumption for Design Technology

The system is assumed to be composed of processor and program. The circuit is designed by cores, which have already been designed, are reused if possible. Modification of the system is met with the system which changing software, (rewriting ROM) as much as possible.

3) EDA Technology Demanded at System Level

Processor architectures and programs should be completely independent, if programs are maintained and reused as design property. As EDA technology for realizing it is as follows:

(a) Technology and tools to extract various information regarding system LSI design from the results of prototyping programmed on a general-purpose processor,
(b) Tools to support development and improvement of algorithm to analyze program and to evaluate possibility of parallelization.

Moreover, rapid prototyping technology will become important such as FPGA.

4) EDA Technology Demanded at Architecture level

It becomes important how to offer processor cores, memory macros, and user-defined cores. Concretely, the following are needed as technology to offer processor cores:

(a) Technology to provide the parameterized processor architecture which is automatically synthesizable,
(b) Technology to customize processor subject to application programs,
(c) Technology to generate retargetable compiler,
(d) Technology to provide library/OS/debugger for programs,
(e) Technology to make programmers control cost/performance/power consumption.

As for memory macros, models are necessary for functional verification and performance evaluation. Moreover, as technology to provide user-defined cores, a technique is needed to express concisely and accurately the specification of each core.

The following will also become important, methodology and tools to verify and test the entire system, and combination of floorplan at the early stage and the compiler technique (change in the area of RAM and ROM).

5) EDA Technology Demanded in RTL/logic level

As a circuit generation technology for processor cores, memory macros, and user-defined cores, the following becomes important:

(a) Simulation models for verification that provide essential information for area/delay/power evaluation,
(b) Efficient logic synthesis,
(c) Test interface for each core and the entire chip.

6) EDA Technology Demanded at Circuit Level

There are two key technologies for constructing processor cores, memory macros, and user-defined cores. One is the circuit structure that is suitable to provide models for verification and test. The other is tools for developing the models.

7) EDA Technology Demanded at Layout Level

Layout synthesis technology become important for processor cores, memory macros, and user-defined cores.

8) EDA Technology Demanded at Manufacturing Device I/F Level

It becomes important to extract the process parameters for feeding back of the information of fluctuation in manufacturing to the design phase. In addition, utilization techniques of such information are key.

Acknowledgements

We would like to express our gratitude to everybody the following who has the cooperation in making of this Roadmap.

Matsushita Electric Industrial Co., Ltd.	Akira Matsuzawa, Hiroyuki Yamauchi, Yoshifumi Okamoto, Hirofumi Taguchi, and Hiroshi Mizuno
Fujitsu Ltd.	Shigeru Fujii, Kimiaki Satoh, Fumiyasu Hirose, and Yuusuke Matsunaga
Mitsubishi Electric Corporation	Yukihiko Shimazu, Harufusa Kondoh, Katsuhiko Seo, and Junji Ishikawa
Hitachi Ltd.	Takashi Hotta, Fumio Murabayashi, Kazumi Hatayama, and Toshihiro Hattori
Toshiba Corporation	Tadahiro Kuroda and Seijiro Moriyama
NEC Corporation	Youichi Yano and Tsuneo Kurobe
Sharp Corporation	Takayoshi Tanaka
Sanyo Electric Co., Ltd.	Isao Ogura

References

[1] **To the enhancement of LSI design capacity in the year 2010, Semiconductor Industry Research Institute Japan, Electronic Industries Association of Japan, March 1997.**

[2] **The National Technology Roadmap for Semiconductors 1994 (1997) Edition, Semiconductor Industry Association.**

URL=http://www.sematech.org/public/roadmap/index.htm

[3] **EDA Industry Standards Roadmap, CFI, EDAC, SEMATECH.**

URL=http://www.cfi.org/roadmap/

A Glossary

B

BIST :	Built In Self Test.
BSIM :	Berkeley Short-channel Insulated-gate FET (Field Effect Transistor) Model developed by UC Berkeley. Berkeley Short-channel Insulated-gate FET Model.

C

Cyber-Giga-Chip :	Cyber-Giga-Chip is abbreviated as CGC, and is used as a kernel of system. It is a system LSI of market-oriented type, and is composed of cores based on the latest technology.

D

DEF :	Design Exchange Format.
DFT :	Design For Test.

E

EMC :	Electromagnetic Compatibility.
EMI :	Electromagnetic Interference.

I

IBIS :	I/O Buffer Information Specification.
Intellectual Property :	Intellectual Property is abbreviated as IP.

P

PDEF :	Physical Design Exchange Format

S

SLDL :	System Level Design Language
SPICE :	Simulation Program with Integrated Circuit Emphasis. Circuit simulator.

T

TCAD :	Technology CAD.
TEG :	Test Element Group. Test chip to verify process technology