EXPERIENCES IN USING CADENCE – THE INDUSTRY STANDARD FOR INTEGRATED CIRCUITS DESIGN

Marko Dimitrijević, Borisav Jovanović, Bojan Anđelković, Milan Savić, Miljana Sokolović,
Faculty of Electronic Engineering Niš

Abstract – *Cadence design package is a powerful collection of programs for complete IC design including analog, digital and mixed-signal circuits. It provides the tools designers need from functional description of the system to the layout of the chip. This paper presents our first experiences in use of some of the programs for ASIC, as well as for full custom IC design.*

1. INTRODUCTION

Cadence is an industry leading software system for integrated circuit design, providing front-to-back design tools and services for all aspects of semiconductor design. The system is organized in packages performing particular functions. All the software components are supported for Solaris and HP-UX operating systems. Some of the tools are supported under Linux and NT/2000/XP operating systems.

The **ASIC Front End Package** provides a core set of design entry, simulation and synthesis tools for digital circuits. The package supports both behavioral and structural design descriptions in text and graphics. It contains the Logic Design Verification (LDV) tools – simulators (Verilog-XL, NC Verilog/VHDL), together with analysis tools for logic designers (HDL style, Code Coverage, Formal Checker), and the Synthesis, Place and Route (SPR**)** tool containing BuildGates family of logic synthesis tools and the new Physically Knowledgeable Synthesis tool (PKS).

IC Package provides high-performance tools based on Design Framework II software – Composer, Analog Artist, Virtuoso layout and Diva verification for each step of the IC design process from architectural definition through detailed structural implementation. The tools can be used for digital, analog and mixed-signal design and are based both on VHDL and Verilog HDL. The IC Package includes all the modules from the ASIC Front End tools: Deep Submicron place and route tools (Silicon Ensemble and Gate Ensemble), the Pearl Static Timing analyzer, the Design Planner software for semi-custom designs and the First Encounter software for very large designs.

The design flow of an ASIC begins with simulation of RTL (Register Transfer Level) description of the design (in VHDL or Verilog) in order to verify circuit functionality. This simulation is performed in NCSim simulator. Cadence PKS is used for logic synthesis, as well as for initial and final timing analysis. It uses VHDL or Verilog description of the design and appropriate technology libraries and generates standard cell netlist. That netlist is imported into Silicon Ensemble to perform floorplanning, cell placement and routing. Clock tree is inserted by CTPKS that is an integral part of Cadence PKS or by using a separate program CTGen. The layout is completed within Cadence Design Framework II and Virtuoso layout editor. Back annotation process can be performed for more accurate timing analysis based on the extracted parasitics from Silicon Ensemble. Pearl utility is used for static timing analyzes and it checks if the design meets timing constraints.

Virtuoso Composer and Layout editor can perform full custom design. NCSim is used for necessary simulations. Layout versus schematics and design rules check can be done in Diva.

2. ASIC DESIGN – LOGIC SYNTHESIS

The process of logic synthesis in Cadence package is performed by using the program Ambit PKS (Physically Knowledgeable Synthesis). The typical ASIC design flow is shown in Figure 1.

Ambit PKS can be executed either in ANSI (ASCII) mode – console application or in Graphical User Interface (GUI) mode. The first task to be done is to load in Ambit timing library files (ALF) containing timing parameters for standard cells. The target technology vendor provides these files. Then, RTL files describing the design in Verilog or VHDL should be read in.

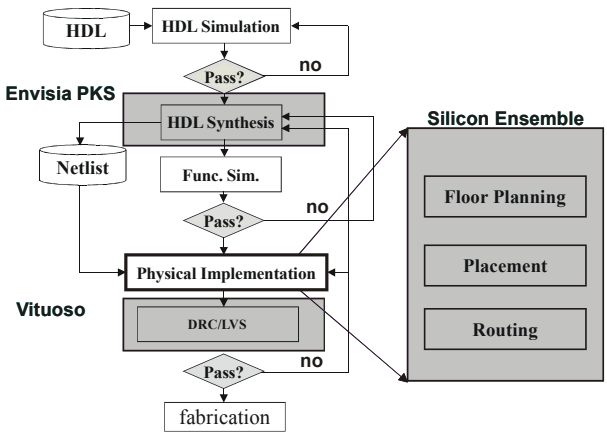

Figure 1. *ASIC design flow*

The logic synthesis begins with mapping of VHDL files into technology independent logic gates from Ambit Synthesis Technology library. When all VHDL files are translated, the top entity and constraints have to be specified. Constraints specify wire loads, operating conditions, port capacitances, fanout load limit etc.

In sequential designs an ideal clock signal should be specified including its name, period and logic 1 and 0 duration ratio. This ideal clock is binded to physical clock pin in the design. When constraints and clock are defined, the

design can be synthesized in the specified target technology. During the synthesis PKS performs cell mapping and optimizations. After the synthesis is finished reports on the size of the design and possible timing violations can be generated. The synthesized netlist can be saved in either Verilog or VHDL. Silicon Ensemble uses Verilog netlist in order to proceed back end part of ASIC chip design.

As an example the schematic of the synthesized 13-bit shift register is shown in Figure 2. The circuit is implemented by using AMS CMOS 0.8μm technology.

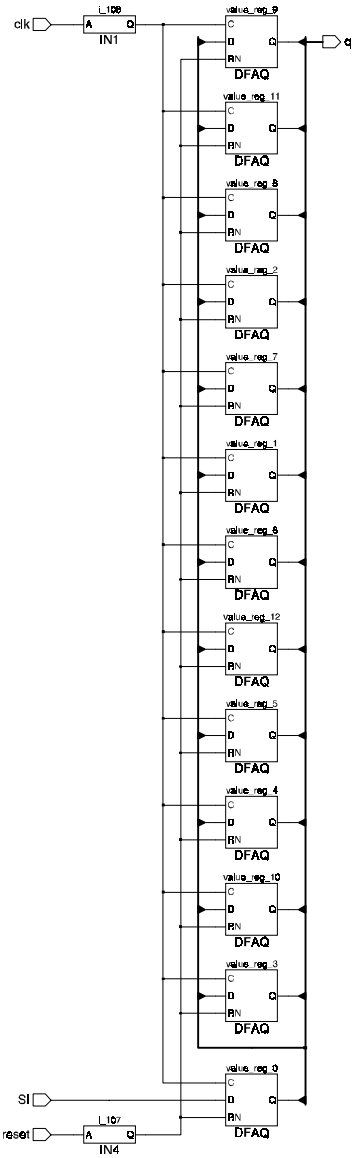

Figure 2. *13-bit shift register synthesized in Ambit PKS*

3. ASIC DESIGN – PLACE AND ROUTE

Silicon Ensemble is Cadence program for place and route standard-cell based designs with embedded blocks such as RAM, ROM or any other IP blocks. It can be run either in ANSI (ASCII) mode from the command line or in the X-windows GUI mode. A typical Silicon Ensemble design flow is shown in Figure 3.

Before the process of physical layout generation can be started, an appropriate library database must be created. This library describes the target technology process for chip implementation and the macro cells required to generate the physical description of the design. Importing one or several Library Exchange Format (LEF) files creates the library. These files are usually supplied by the silicon vendor and contain layer definitions, via definitions, via generation rules, some design rules and information about cell shapes and connections. If timing driven placement is used Compiled Timing Library Format (CTLF) files with standard cell timing parameters must be read in. The technology vendor also provides these files and it is necessary to create General Constraints Format (GCF) file that tells Silicon Ensemble the paths to the CTLF files. After the library database has been generated, importing netlist files, which can be either Verilog files produced by the synthesis tool or Design Exchange Format (DEF) files, creates the design database. Along with the Verilog design file(s), a stub library must be read in to tell Silicon Ensemble which cells are the basic level standard cells (Figure 4).

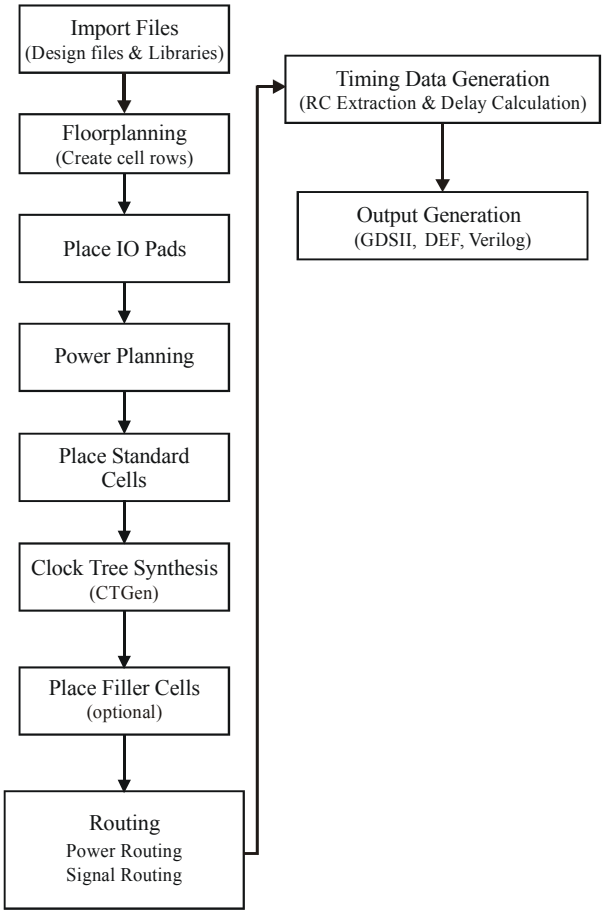

Figure 3. *Silicon Ensemble design flow*

Corner and supply cells can be added to the design database from a DEF file. Corner cells are used for continuity of power and ground rings at corners.

The first step in place and route process is floor planning. During this phase the parameters relating to the size and shape (square or rectangular) of the area where cells and blocks can be placed have to be specified. Before placement any of the blocks and standard cells, the I/O cells must be placed. They can be placed either automatically or by using an I/O placement file. That file is an ASCII readable file, which specifies the pin locations in terms of their positions on the chip. After placing I/O cells, power rings should be constructed by specifying their location, size and layer. For

large cell arrays power stripes are also added across the core area. They are needed to distribute power throughout the array.

The next step is to place the standard cells in the rows. For this purpose Quick Place (QPlace) utility invoked from Silicon Ensemble is used.

Clock Tree Generator (CTGen) is an optional feature in Silicon Ensemble, which is used to generate buffered clock trees according to various time constraints. It constructs an optimized clock tree and minimizes clock skew. The tree is built from the designated root pin to the clocked leaf pins.

Next, we must add filler cells to the design. The purpose of filler cells is to maintain continuity in the rows by adding power lines and an n-well. The filler cells also contain substrate connections to improve substrate biasing.

After the clock tree is built and filler cells are placed, the routing of the design can be started. First special routing is performed and it connects the stripes and blocks to the power rings of the design. Next step is to connect power pins for the standard cells within rows to the power rings. This is done on the left and right of the rows. Finally, the power rings are connected to the power pads.

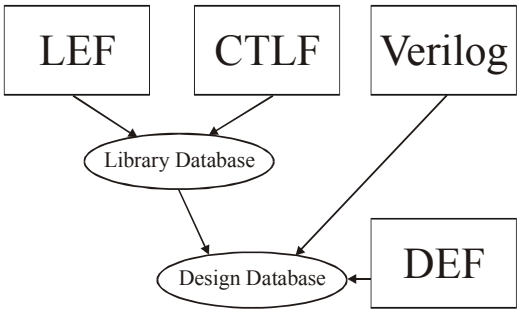

Figure 4. *File formats for library and design database in Silicon Ensemble*

The Warp Router (WRoute) utility in Silicon Ensemble version 5.2 is used for global and detailed routing of signal nets. Global routing creates routing channels and divides large nets that run across multiple channels into sub-nets. During the detailed routing the task of routing the signal nets in the chip channel by channel is performed. Automatic search and repair and optimization re-routes signal nets that have been violated design rules in the first phase of routing. During the several iterations WRoute optimizes the wire length and number of vias. After routing the design successfully, it can be verified that there are no antenna violations on the long nets, no shorts and opens and no design rules violations.

The final, placed and routed design can be saved in several different formats (GDSII, Verilog or DEF) for further processing. A vendor-supplied mapping file is used to convert data from Silicon Ensemble to GDSII. Importing the generated GDSII or DEF file in Virtuoso and invoking the physical verification tool Diva performs Layout Versus Schematic (LVS) and Design Rule Check (DRC). Also, parasitics can be extracted in RSPF (Restricted Standard Parasitic File) format for backannotation of the design later.

All design phases in Silicon Ensemble can be automated by using macros.

The layout of the 13-bit shift register from the previous section generated in Silicon Ensemble is presented in Figure 5. The design is implemented in AMS 0.8μm technology.

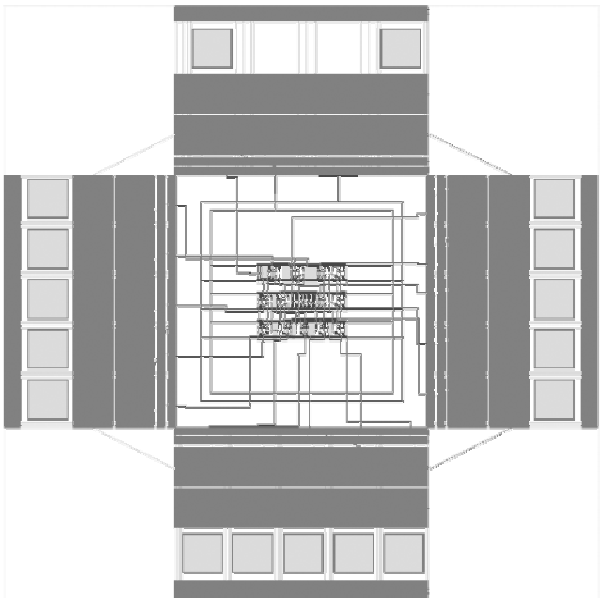

Figure 5. *Layout from Silicon Ensemble*

4. FULL CUSTOM IC DESIGN

Package IC 4.46 provides tools for full custom design. Figure 6. presents the basic full custom design flow. This flow starts with functional schematic, design with Virtuoso Schematic Composer. Simulations are performed using Affirma Analog Environment, which is the user interface to the simulator (Spectre or any other). OCEAN script language offers full automation of complex simulation tasks. Layout is designed using Virtuoso Layout Editor or Virtuoso XL. Verification is performed using Diva tools (DRC – Design Rules Check, ERC – Electric Rules Check, Extractor and LVS – Layout Versus Schematic). Using the extracted cell view, which contains the parasitics information, post layout simulation can be performed. Using the SKILL language, scripts can be designed to automate any stage of the design. After the design is verified it can be placed and routed with other parts of the design. Package ICC (IC Craftsman) provides tools for block placement and routing.

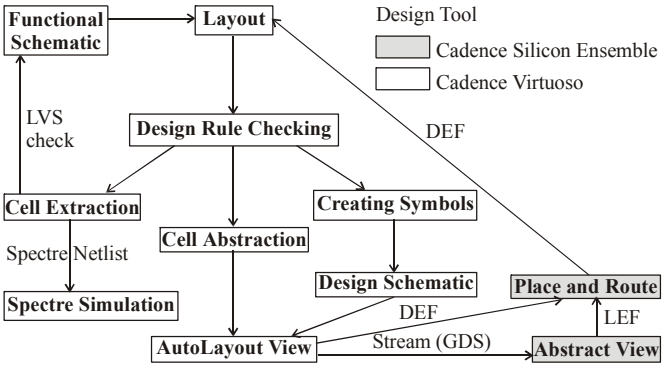

Figure 6: *Basic Full Custom Design Flow*

Using Abstract Generator, the design layout can be exported to LEF (Library Exchange Format), as a block, pad or standard core cell, and imported to Silicon Ensemble (SE). Designs can be exported from Schematic Editor into DEF (Design Exchange Format) and imported in SE. Silicon Ensemble supports both standard and block placement and routing. After placement and routing, design should be returned to Virtuoso Layout Editor, and the complete design should be verified. Throughout the full custom design flow, target technology support is essential. Being familiar with the technology is almost as important as being familiar with the Cadence environment.

5. SIMULATION AND VERIFICATION

Verification step in the IC design is preformed using one more Cadence tools from package LDV (Logic Design and Verification). This package consists of several very important tools. They are:

1. NC-Verilog, NC-VHDL, NC-Sim
2. Formal Check
3. Verifault XL (Verilog XL)
4. Verification Cockpit
5. Test Builder

NC-Verilog, NC-VHDL and NC-Sim are simulators for already described Verilog and VHDL digital circuits, and they are one of the most important tools in the LDV package. Formal Check performs a formal analysis from the block models point of view, without using any test vectors. This tool can detect faults that cannot be detected by using simulator. Verifault XL is a part of Verilog XL. It enables a fault simulation in order to make the test vectors generation easier. It performs a parallel simulation of the original circuit and a circuit with built in defects described also in Verilog. Available defects are: permanent zero, permanent one, opened connections and shortcuts. Verification Cockpit enables verification of the RTL (Register Transfer Level) designs. The test builder is a program that generates test vector for the digital circuits.

Usage of NC simulators will now be explained in more details. VHDL and Verilog simulators can be launched from "NC-Launch" environment as shown in Figure 7.

After accepting VHDL and Verilog descriptions, compilation and elaboration must be executed. Designs that pass these steps can now be simulated in NC-Sim program. This simulator consists of three important parts. First is SimControl that is used for simulation parameter adjustment. For graphical representation of simulation results Waveform Viewer is used. The last part is the Signal Flow Browser for tracing the flow of particular signals.

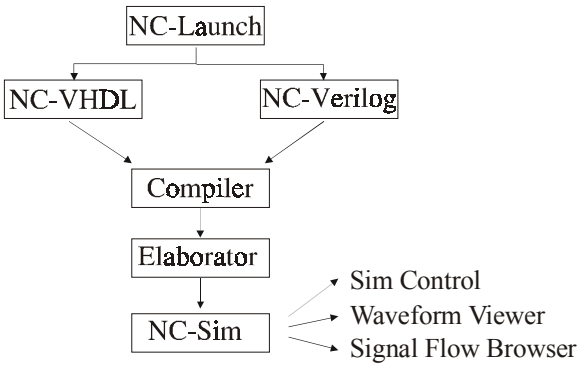

Figure 7. *NC-Launch environment*

Verification in NC-Launch consists of two simulations. First we simulate the VHDL or Verilog description of one circuit, and then, after the circuit is synthetized we extract its netlist and perform another simulation. If results of these two simulations are identical, than one can say that the verification is successful.

This work was supported in part by the Ministry of Science, Technology and Development of Serbia, through the Project IT.1.02.0075.A realized in Technology Development area.

REFERENCES

[1] V.B. Litovski, *Projektovanje elektronskih kola: simulacija, optimizacija, testiranje, fizičko projektovanje*, Vranje: Nova Jugoslavija, 2000.

[2] Cadence 2002 Documentation

[3] Linus Svensson, Åke Östmark, *Instructions for Lab 1: Quick Tool-Tutorial,* Luleå University of technology

Sadržaj – Paket za projektovanje Cadence je moćna kolekcija programa za kompletno projektovanje analognih, digitalnih i hibridnih integrisanih kola. Paket projektantima pruža neophodne alate za projektovanje od funkcionalnog opisa sistema do layout-a čipa. U ovom radu su predstavljena naša prva iskustva u upotrebi nekih programa za ASIC i full-custom projektovanje integrisanih kola.

CADENCE – INDUSTRIJSKI STANDARD ZA PROJEKTOVANJE INTEGRISANIH KOLA

Marko Dimitrijević, Borisav Jovanović, Bojan Anđelković, Milan Savić, Miljana Sokolović