An Open Source approach for ASIC Design Flow

Kesari Ananda Samhitha
M.Tech
Holy Mary Institute of Technology and ScienceAffiliated to JNTU, Hyderabad.
Email: samhithaka@gmail.com

Y. David Solomon Raju
Associate Professor
Holy Mary Institute of Technology and Science Affiliated to JNTU, Hyderabad.
Email: davidsolomonraju.y@ gmail.com

Abstract:

In this paper, we are going to study the process of implementation of ASIC design flow with the help of open source EDA tools. In order to design our own circuitry without any other expensive licenced tools, we apt the open source tools. The design of electronic circuits can be achieved at many different refinement levels from the most detailed layout to the most abstract architectures.

Given the complexity of Very Large Scaled Integrated Circuits (VLSI) which is far beyond human ability, computers are increasingly used to aid in the design and optimization processes. It is no longer efficient to use manual design techniques, in which each layer is hand etched or composed by laying tape on film due to the time-consuming process and lack of accuracy.

Therefore, Computer Aided Design (CAD) tools are heavily involved in the design process. In today's market, there are plenty of VLSI CAD tools; however, most of them are expensive and require high performance platforms. Selecting an appropriate CAD tool for academic use is considered as one of the key challenges in teaching VLSI design courses.

In this paper, number of open-source and freeware CAD tools are presented and evaluated. Based on the objectives of the user, this paper furnishes guidelines that help in selecting the most appropriate open-source and freeware VLSI CAD tool for teaching a VLSI design course.

Keywords: VLSI, CAD Tools, Electric, Magic, Alliance, Comparative Study

Introduction:

ASIC design flow is a very mature process in silicon turnkey design. The ASIC design flow and its various steps in VLSI engineering that we describe below are based on best practices and proven methodologies in ASIC chip designs. This paper attempts to explain different steps in the ASIC design flow, startingfrom ASIC design concept and moving from specifications to benefits.

ASIC design flow:

ASIC design flow is a mature and silicon-proven IC design process which includes various steps like design conceptualization, chip optimization, logical/physical implementation, and design validation and verification. Let's have an overview of each of the steps involved in the process.

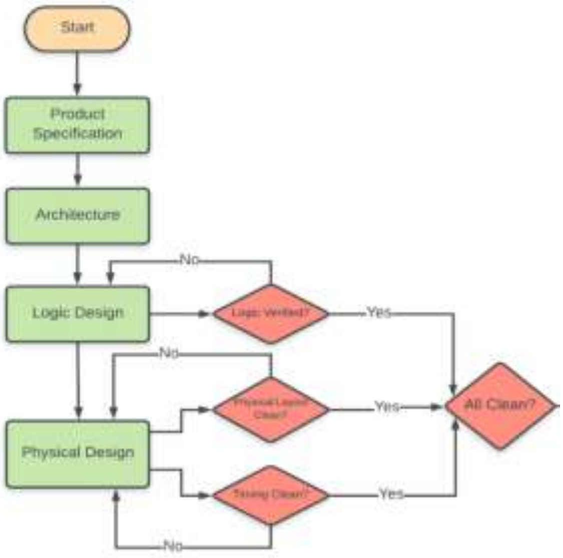

Figure 1: ASIC design flow

Steps involved in ASIC design flow:

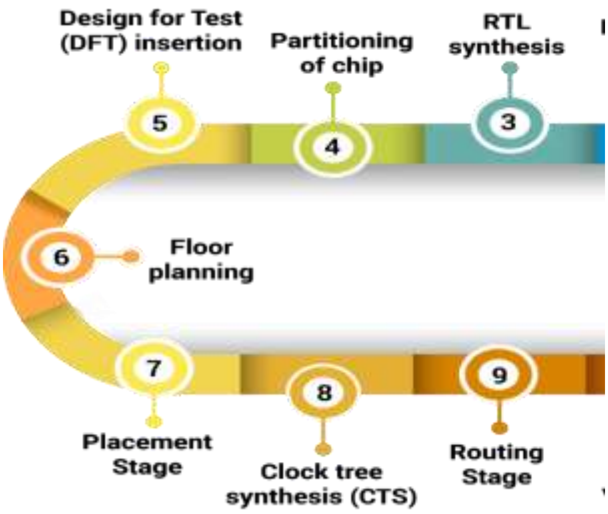

Figure 2: Steps involved in ASIC design flow

Step 1. Chip Specification:

This is the stage at which the engineer defines features, microarchitecture, functionalities (hardware/software interface), specifications (Time, Area, Power, Speed) with design guidelines of ASIC.

Two different teams are involved at this juncture:

- **Design team:** Generates RTL code.
- **Verification team:** Generates test bench.

Step 2. Design Entry / Functional Verification:

Functional verification confirms the functionality and logical behavior of the circuit by simulation on a design entry level. This is the stage where the design team and verification team come into the cycle where they generate RTL code using test-benches. This is known as behavioural simulation.

In this simulation, once the RTL code (RTL code is a set of code that checks whether the RTL implementation meets the design verification) is done in HDL, a lot of code coverage metrics proposed for HDL. Engineers aim to verify correctness of the code with the help of test vectors and trying to achieve it by 95% coverage test. This code coverage includes statement coverage, expression coverage, branch coverage, and toggle coverage.

There are two types of simulation tools:

- **Functional simulation tools:** After the testbench and design code, functional simulation verifies logical behavior and its implementation based on design entry.
- **Timing simulation tools:** Verifies that circuit design meets the timing requirements and confirms the design is free of circuit signal delays.

Step 3. RTL block synthesis / RTL Function:

Once the RTL code and testbench are generated, the RTL team works on RTL description – they translate the RTL code into a gate-level netlist using a logical synthesis tool that meets required timing constraints. Thereafter, a synthesized database of the ASIC design is created in the system. When timing constraints are met with the logic synthesis, the design proceeds to the design for testability (DFT) techniques.

Step 4. Chip Partitioning:

This is the stage wherein the engineer follows the ASIC design layout requirement and specification to create its structure using EDA tools and proven methodologies. This design structure is going to be verified with the help of HLL programming languages like C++ or System C.

After understanding the design specifications, the engineers partition the entire ASIC into multiple functional blocks (hierarchical modules), while keeping in mind ASIC’s best performance, technical feasibility, and resource allocation in terms of area, power, cost and time. Once all the functional blocks are implemented in the architectural document, the engineers need to brainstorm ASIC design partitioning by reusing IPs from previous projects and procuring them from other parties.

Step 5. Design for Test (DFT) Insertion:

With the ongoing trend of lower technology nodes, there is an increase in system-on-chip variations like size, threshold voltage and wire resistance. Due to these factors, new models and techniques are introduced to high-quality testing.ASIC design is complex enough at different stages of the design cycle. Telling the customers that the chips have fault when you are already at the production stage is embarrassing and disruptive.

It’s a situation that no engineering team wants to be in. In order to overcome this situation, design for test is introduced with a list of techniques:

- **Scan path insertion:**
  A methodology of linking all registers elements into one long shift register (scan path). This can help to check small parts of design instead of the whole design in one go.
- **Memory BIST (built-in Self-Test):**
  In the lower technology node, chip memory requires lower area and fast access time. MBIST is a device which is used to check RAMs. It is a comprehensive solution to memory testing errors and self-repair proficiencies.
- **ATPG (automatic test pattern generation):**
  ATPG is a method of creating test vectors / sequential input patterns to check the design for faults generated within various elements of a circuit.

Step 6. Floor Planning (blueprint your chip):

After, DFT, the physical implementation process is to be followed. In physical design, the first step in RTL-to-GDSII design is floorplanning. It is the process of placing blocks in the chip. It includes: block placement, design portioning, pin placement, and power optimization.

Floorplan determines the size of the chip, places the gates and connects them with wires. While connecting, engineers take care of wire length, and functionality which will ensure signals will not interfere with nearby elements. In the end, simulate the final floor plan with post-layout verification process.

A good floorplanning exercise should come across and take care of the below points; otherwise, the life of IC and its cost will blow out:

- Minimize the total chip area.
- Make routing phase easy (routable).
- Improve signal delays

Step 7. Placement:

Placement is the process of placing standard cells in row. A poor placement requires larger area and also degrades performance. Various factors, like the timing requirement, the net lengths and hence the connections of cells, power dissipation should be taken care. It removes timing violation.

Step 8. Clock tree synthesis:

Clock tree synthesis is a process of building the clock tree and meeting the defined timing, area and power requirements. It helps in providing the clock connection to the clock pin of a sequential element in the required time and area, with low power consumption.

In order to avoid high power consumption, increase in delays and a huge number of transitions, certain structures can be used for optimizing CTS structure such as Mesh Structure, H-Tree Structure, X-Tree Structure, Fishbone Structure and Hybrid structure.With the help of these structures, each flop in the clock tree gets the clock connection.

During the optimization, tools insert the buffer to build the CTS structure. Different clock structures will build the clock tree with a minimum buffer insertion and lower power consumption of chips.

Step 9. Routing:

- **Global Routing:**
  Calculates estimated values for each net by the delays of fan-out of wire. Global routing is mainly divided into line routingand maze routing.
- **Detailed Routing:**
  In detailed routing, the actual delays of wire is calculated by various optimization methods like timing optimization, clock tree synthesis, etc.

As we are moving towards a lower technology node, engineers face complex design challenges with the need for implanting millions of gates in a small area. In order to make this ASIC design routable, placement density range needs to be followed for better QoR. Placement density analysis is an important parameter to get better outcomes with a smaller number of iterations.

Step 10. Final Verification (Physical Verification and Timing):

After routing, ASIC design layout undergoes three steps of physical verification, known as signoff checks. This stage helps to check whether the layout working the way it was designed to. The following checks are followed to avoid any errors just before the tapeout:

- Layout versus schematic (LVS) is a process of checking that the geometry/layout matches the schematic/netlist.
- Design rule checks (DRC) is the process of checking that the

geometry in the GDS file follows the rules given by the foundry.

- Logical equivalence checks (LVC) is the process of equivalence check between pre and post design layout.

Step 11. GDS II – Graphical Data Stream Information Interchange:

In the last stage of the tapeout, the engineer performs wafer processing, packaging, testing, verification and delivery to the physical IC. GDSII is the file produced and used by the semiconductor foundries to fabricate the silicon and handled to client.

VLSI CAD TOOLS:

Based on the typical VLSI design work flow, a good VLSI CAD tool must support the following basic features: logical design, circuit schematic design, layout generation, and design check. In today's market, most VLSI CAD tools are based on Unix or Linux platforms. Only few of them have the ability to run in Windows or run independently of certain emulation/simulation software.

Considering the popularity of the VLSI CAD tools, three kinds of commonly used and freely available tools were selected for comparison in this paper. They are, as mentioned earlier, Electric, Magic, and Alliance. Both of Electric and Alliance are available under the GNU General Public License (GPL). Magic, which was developed back in 1980's, is available under Berkeley Open-Source License.

For the commercial software side, another popular tool used at the academic institutes is OrCAD PSpice. This tool is a variance of the SPICE tool family. It offers student demo version with full function but limited capacity. It is now a product of the Cadence Design Systems, Inc. Another tool from this company, Cadence Custom IC Design, is also very popular software in VLSI design.

It is capable to design, analyse and help to optimize an analog, radio frequency, or mixed-signal ICs. A basic copy of the Cadence Custom IC Design is sold for several hundred dollars. This paper mainly focuses on the comparison of open-source VLSI CAD tools for academic and educational use. Therefore, commercial software is not included in the comparison.

OPEN SOURCE CAD TOOLSCOMPARATIVE STUDY:

Tool	Description	Type	Function
Xcircuit	A general-purpose drawing program and also a specific-purpose CAD program for circuit schematic drawing and schematic capture.	Mixed signal	Schematic
Ktechlab	Is a schematic capture and simulator. It is a an IDE for microcontrollers andelectronics. It supports circuit simulation, program development for microcontrollers and simulating the programmed microcontroller together with its application circuit.	Mixed signal	Schematic, simulatorand microcontroller

gnucap	A general-purpose circuit simulator with its engine designed to do true mixed-mode simulation.The primary component is a general-purpose circuit simulator. It performs nonlinear dc and transient analyses, Fourier analysis, and ac analysis. Spice compatible models for the MOSFET (level 1-7), BJT, and diode are included in this release.	Mixed signal	Circuit simulator
ngspice	A mixed level/signal circuit simulator	Mixed signal	Circuit simulator
Qucs (Quite Universal Circuit Simulator)	Qucs is a circuit simulator with graphical user interface. The software aims to support all kinds of circuit simulation types, e.g. DC, AC, S-parameter and harmonic balance analysis. Pure digital simulations are also supported using VHDL and/or Verilog.	Mixed signal	Simulator, Verilog, VHDL
IRSIM	It is a switch-level simulator originally originating from Stanford.	Mixed signal	Circuit simulator
Icarus Verilog	Is a Verilog simulation and synthesis tool. It operates as a compiler, compiling source code written in Verilog (IEEE-1364) into some target format. For batch simulation, the compiler can generate an intermediate form called vvp assembly. This intermediate form is executed by the "vvp" command. For synthesis, the compiler generates netlists in the desired format.	Digital	Simulator & Synthesis
Verilator	Verilator is the fastest free Verilog HDL simulator. It compiles synthesizable Verilog, plus some PSL, SystemVerilog and Synthesis assertions into C++ or SystemC code. It is designed for large projects where fast simulation performance is of primary concern, and is especially well suited to create executable models of CPUs for embedded software design teams.	Digital	Verilog HDL simulator & synthesis
Ghdl	Is a tool to generate metal layers and vias to physically connect together a netlist in a VLSI fabrication technology. It is a maze router, otherwise known as an "over-the-cell" router or "sea-of-gates" router.	Digital	VHDL simulator (doesn't do synthesis)

ChipVault	It is a chip development program for organizing VHDL and Verilog designs. Chip Vault displays designs hierarchically and provides for rapid design navigation and editor launching. Chip Vault provides hooks for performing bottom-up tasks such as launching RTL compilers, synthesis, block generation and instantiation, and includes simple to use Revision Control and Issue Tracking systems to help facilitate large group design projects with multiple designers and hundreds of design files.	Digital	VHDL & Verilog RTL compiler & synthesis
GTKwave	Is a waveform viewer that can view VCD files produced by most Verilog simulation tools, as well as LXT files produced by certain Verilog simulation tools.	Mixed signal	Wave viewer
Gwave	It is a waveform viewer for the output of analog electronic circuit simulators such as spice. It displays the data as 2-D plots, and allows for interactive scrolling, zooming, and measuring of the waveforms	Analog	Wave viewer
LabPlot	It is a free software data analysis and visualization application built on the KDE Platform	Analog	Wave viewer
gEDA Suite	A full suite of Electronic Design Automation tools.	Mixed signal	Full suite (electrical circuit design, schematic capture, analog and digital simulation, prototyping, and production)
Alliance	It is a complete set of free CAD tools and portable libraries for VLSI design. It includes a VHDL compiler and simulator, logic synthesis tools, and automatic place and route tools. A complete set of portable CMOS libraries is provided, including a RAM generator, a ROM generator and a data-path compiler.	Mixed signal	Design flow from VHDL up to layout,
Electric	Electric is a sophisticated electrical CAD system that can handle many forms of circuit design, including custom IC layout (ASICs), schematic drawing, hardware description language specifications, and electro-mechanical hybrid	Mixed signal	From HDL to layout and some extra

	layout.		
Magic	Magic is a venerable VLSI layout tool. Magic VLSI remains popular with universities and small companies. Magic is widely cited as being the easiest tool to use for circuit layout, even for people who ultimately rely on commercial tools for their product design flow.	Mixed signal	Circuit Layout
Toped	It is a cross-platform IC layout editor supporting GDS, OASIS and CIF formats. It is an open source project licensed under the GNU General Public License. The project is under active development. focuses on rendering speed and quality of the screen output.		IC layout editor
Netgen	A tool for comparing netlists, in analog or mixed-signal circuits that cannot be simulated in reasonable time.	Mixed signal	LVS
Dragon	Dragon is a fast, effective standard-cell placement tool for both variable-die and fixed-die ASIC design. It was designed and implemented by NuCAD group in Dept. of ECE, Northwestern University, and ERLAB in Computer Science Dept., UCLA. Dragon does wirelength and routability optimization by combining powerful hypergraph partitioning package (hMetis) with simulated annealing technique. It is a university tool that produces high-quality placement comparable with commercial software such as ITools (formerly Timber Wolf) and Cadence QPlace.	Digital	Placement
Fairly Good Router (FGR)	FGR is free open-source software for global routing, based on Lagrange Multipliers --- an approach similar to what industry routers use, but with greater mathematical rigor and robust performance. Unlike most other academic tools, FGR is self-contained and does not rely on ILP or external Steiner-tree constructors.	Digital	Router
Qrouter	It is a tool to generate metal layers and vias to physically connect together a netlist in a VLSI fabrication technology. It is a maze router, otherwise known as an "over-the-cell" router or "sea-of-gates" router.	Digital	Detail router
OS-VVM	Open Source - VHDL Verification Methodology (OS-VVM) delivers advanced verification test methodologies, including Constrained and Coverage-driven Randomization, as well as Functional Coverage, providing advanced features to VHDL design engineers while enabling them to continue to develop using VHDL.	Digital	Verification
Teal	Helps you perform verification by providing a set of capabilities that access HDL signals and enable actions based on changes in the values of these	Digital	Verification

	signals.		
Jove	It is a set of Java APIs and tools to enable Verilog hardware design verification of ASICs and FPGAs using the Java programming language.	Digital	Verification
PCB	It is free software for designing printed circuit board layouts. It has many features and is capable of professional-quality output.	PCB	The printed circuit board layout editor

CONCLUSION:

VLSI design has become an important course at most of the electrical and computer engineering programs. However, buying licenses for commercial VLSI CAD is usually costly and requires high performance workstations which many academic institutes may not be able to afford. This paper provided some insight on the most popular open-source CAD tools that can be used in the academic field.

Electric tool shows the best compatibility with different platforms, and also have plenty of design and check functions in a friendly user interface. Magic tool, which has a long history in VLSI design, has the best technical support and documentation, and it provides both stable and development versions for different use. It is very attractive to the academic institute users. Alliance tool, which can only run in Unix/Linux platforms, has the best usage stability and good balance in functions. But its popularity is limited by its strict operation system requirement.

Even in Linux, it requires a re-compile with specific compiler. It also increases the complexity of the installation. In conclusion, Electric tool is highly recommended for Windows users.

Yet, in Unix/Linux systems, Electric doesn’t have significant advantage as it does in Windows. Magic and Alliance are also competitive for their own features. Magic may be more suitable for education that focuses on software development and Alliance may be good for education that focuses on design skill. This comparison is hopefully useful for those who are looking for a suitable VLSI CAD tool for academic purposes.

REFERENCES:

1. Intel Corporation, The Intel® Itanium Processor 9300 Series: A Technical Overview for IT Decision-Makers, White Paper, February. 2010.
2. Neil Weste and David Harris, CMOS VLSI Design, the Third Edition, Pearson Education, 2004.
3. Static Free Software. November 2009. http://www.staticfreesoft.com.
4. Alliance. November 2009. http://www-asim.lip6.fr/recherche/alliance.
5. Magic VLSI Resource. November 2009. http://www.opencircuitdesign.com/magic.
6. Cadence OrCAD Solutions. February 2010. http://www.cadence.com/products/orcad.
7. Custom IC Design, February 2010. http://www.cadence.com/products/cic.
8. Yohn K. Ousterhout, Gordon T. Hamachi, Robert N. Mayo,Walter

S. Scott, and George S. Taylor, "Magic: A VLSI Layout System", Design Automation, 1984. 21st Conference on, 25-27 June 1984.

9. Taylor, G. S. and Ousterhout, J. K. "Magic's Incremental Design Rule Checker", Design Automation, 1984.
10. Scott, W. S. and Ousterhout, J. K. "Plowing: Interactive Stretching and Compaction in Magic", Design Automation, 1984.
11. Alain Greiner and Francois Pecheux, "ALLIANCE: A complete Set of CAD Tools for teaching VLSI Design", Technical Report: Laboratoire MASI/CAO-VLSI, Institut de Programmation, Universite Pierre et Marie, 1992.
12. Amara Amara, Alain Greiner, Luis Lucas, Frederic Petrot, Franck Wajsburt, Laurent Winckel, "The Portable Cell Libraries of the Freeware Alliance CAD system".
13. BSIM3 – Introduction, February 2010. http://www-device.eecs.berkeley.edu/~bsim3.
14. IRSIM, February 2010. http://opencircuitdesign.com/irsim.
15. Topped – Open Source Layout Editor, February 2010. http://www.toped.org.uk.