A Method to Implement Layout Versus Schematic Check in Integrated Circuits Design Programs

Radu Gabriel Bozomitu, Daniela Ionescu

Telecommunications Department
Faculty of Electronics and Telecommunications, "Gh. Asachi" Technical University
Iaşi, 700506, Romania
Phone +40-232-213737, Fax +40-232-217720, bozomitu@etc.tuiasi.ro, danait@etc.tuiasi.ro

Abstract

In this paper, a new method to implement the LVS check in different IC design software, without using specialized LVS software, is presented. The main advantage of the proposed method consists in the fact that all types of checkings (DRC and LVS) can be easily performed, without using specialized parameters extraction software, which is very expensive and difficult to implement. The method principle is based on the ability of certain IC design software (like L-Edit) to display in layout design the electric connections between basic cells (even at a transistor’s level) and also to permanently monitoring the routed wires and also of those remained to be routed. In order to export the electrical scheme (designed with specialized software, as OrCAD or SPICE) in layout, there have to be passed few well defined stages, which will be presented in detail in the paper. After a successful trespassing of those stages, all cells of the electric scheme will be imported in layout design and the electric connections between them will be also displayed. Connectivity is displayed as a network of routing guides for the pin-to-pin connections. Throughout the layout design process, the connectivity display is interactively updated as objects are moved and new wire is added. In this way, the LVS is checked in real time, when assisted manual routing is performed. Finally, all layout design stages are illustrated, using as example a circuit implemented into 0.8 μm BiCMOS technology.

1. INTRODUCTION

One of the most difficult problems that occur in integrated circuit (IC) design, using different specialized software, consists in checking the layouts design, for their validation.

These checkings may be subdivided in two main categories: 1) design rule check (DRC) - to check a layout for design rule violation; design rules, in their simplest form, are usually minimum allowable values for certain widths, separations, extensions, and overlaps of and between geometrical objects; the exact nature of design rules depends on specifications supplied by the foundry to which the design will be submitted for fabrication. To check a layout for design rule violation, then involves two basic steps:

- defining the rules that are acceptable for IC design;
- be made up entirely of primitives or a combination of primitives and instances of other cells.

2. LAYOUT DESIGN PRINCIPLE

The basic building block of the integrated circuit design in IC design software is a cell. Design layout occurs within cells. A cell can:
- contain part or all of the entire design;
- be referenced in other cells as a sub-cell, or instance;
- be made up entirely of instances of other cells;
- contain original drawn objects, or primitives;
- be cross referenced in other design files;
- be made up entirely of primitives or a combination of primitives and instances of other cells.

2.1. Design rules check (DRC)

Design rules, in their simplest form, are usually minimum allowable values for certain widths, separations, extensions, and overlaps of and between geometrical objects. The exact nature of design rules is dependent on specifications supplied by the foundry to which the design will be submitted for fabrication.

To check a layout for design rule violation, then involves two basic steps:
- defining the rules that are acceptable for IC design;
- running the design rule checker on the entire design or a portion of it.

When the layout is complete, the design rule violation should be checked before sending the layout to the chip foundry for fabrication. If a chip is fabricated with design rule violations, it may fail to function as designed.

2.2. Layout versus schematic (LVS)

LVS compares elements and connections for similar characteristics, not for functionality or purpose. Therefore, it is important that the netlists being compared have the same types of basic circuit elements.

LVS processes subcircuits by flattening them, then individually comparing their constituent elements and nodes. If the two designs resolve at the same hierarchical levels, however, one can explicitly define higher-level subcircuits as elements for comparison in an element description file [1].

LVS begins by reading in the two netlists and compiling a list of elements and a list of nodes. An element is any type of logic or circuit component, such as a transistor, resistor or capacitor. A node indicates a connection: a wire and anything directly attached to it.

LVS then sorts the elements and nodes into classes. A class is a set of elements or nodes with something in common: a transistor class, a resistor class, and so on. Further divisions might divide the transistor class into P transistor and N transistor classes [1].

Nodes are separated in a similar manner. They might be separated into classes according to the number of elements attached to them. Further separation might be based on the types of elements attached to the nodes.

The process of separating elements and nodes according to topological information is called topological matching [1]. Topological matching groups elements and nodes by types and connectivities, rather than by capacitance or element size, which are not used in the default matching process. Topological matching continues until no further fracturing is possible.

2.3. Hierarchy

IC design software supports fully hierarchical mask design, where cells may contain instances of other cells. An instance is a reference to a cell; if editing the instanced cell, the change is reflected in all the instances of that cell. There is no preset limit to the size or complexity of the hierarchy. Cells may contain instances of other cells that in turn contain instances of other cells, to an arbitrary number of levels.

Manufacturing constraints can be defined in L-Edit as design rules. Layouts can be checked against these design rules. IC design software can import and export two standard mask layout interchange formats: GDSII (“Graphic Data System II”) and CIF (“Caltech Intermediate Format”). Although CIF and GDSII are industry-standard formats, each design tool employs its own customizations and extensions, and enforces its own limitations within the standard format.

2.4. Connectivity

An important advantage of different IC design software (like L-Edit) consists in displaying connectivities between components in layout design as a network of routing guides using block place and route (BPR) process.

BPR initializes a design by reading the netlist for the blocks to use and for their connectivity. It compares the netlist with the blocks in the layout and places referenced blocks in the design within a top-level cell.

Connectivity is displayed as a network of routing guides for the pin-to-pin connections. Throughout the BPR process, the connectivity display is interactively updated as you move objects and add routing [1].

Once the design is initialized, one can use BPR to automatically or manually place blocks according to the relative importance of minimizing the total routing length or the total area covered by blocks. BPR can perform incremental placement relative to the netlist, where specific sections of the design can be replaced or rerouted while any unaffected placement or routing is preserved.

BPR supports fully automatic routing or assisted manual routing. During assisted manual routing, BPR guides the cursor from pin to pin as the nets are routed. Keyboard shortcuts allow to quickly selecting from the defined routing layers, and BPR automatically adds the appropriate via when changes occur from one routing layer to another.

Any BPR design must have one cell defined as the top-level BPR cell. This is the cell at the very top of the cell hierarchy for the file, which contains all blocks.

2.5. Extracting layout

Extraction is a method of verifying a layout. The extraction process produces a netlist that describes the circuit represented by the layout in terms of device and connectivity information.

The diagram shown in Fig. 1 illustrates the design flow for IC design with specialized software [1]. Thus, according to diagram shown in Fig. 1, the stages of an integrated circuit’s designing may be grouped as follows:

- electric scheme edit;
- electric scheme simulation;
- electric scheme importing in layout;
- layout designing;
- layout checking (DRC, LVS);
- layout extraction of parasitic elements;
- post-layout simulations;
- chip fabrication.
3. ELECTRICAL SCHEME PREPARATION TO EXPORT IN LAYOUT EDITOR

3.1. Electrical scheme design

Integrated circuits’ designing by specialized soft is made through hierarchic levels.

Thus, in the layout, the circuit top will contain more cells, placed on different hierarchic levels, made in previous stages.

A cell must contain the same pins name and number, both at electric scheme, symbol level, and at layout level.

In order to be correctly exported in layout, a cell must be realized both at electric and symbol views. At electric scheme level the pins name is specified by some ports in order to be easily recognized at the layout level.

Only pins name is specified (identical to that used for electric scheme designing) and not their number when an OrCAD cell symbol is made.

At layout level, cell’s pins name may be specified by certain ports put by designer, having the same name as pins used at electric scheme level.

In order to export the electric scheme in layout, the netlist designing in a special format is required, which can be recognized by the program used for layout designing.

Thus, to export an electric scheme in L-Edit program in order to design the layout, its netlist must be realized in a format recognized by most of the layout editor soft as “electronic design interchange format” (EDIF) or “Tanner place and rout” format (TPR).

3.2. Layout design

The electric scheme layout importing is made by its netlist, built by EDIF 2000 or TPR format, using block place and route process (BPR).

After the electric scheme netlist is imported by BPR process, the electric connections between components will be shown in the layout. For this, it is necessary that all cells at electric scheme level to have a correspondent at layout level which expresses the same pins number and the same name.

The display of the electric connections between components represent a major advantage of certain programs (as is L-Edit) which can be correctly exploited in order to implement the LVS type checking. This method will be presented in the following.

Wire routing can be made automatically (as is the case of integrate circuits’ designing by digital circuits) or manually (for analog integrated circuits, where, for a better matching between different components, their manual placement on the chip is mandatory, situation when the designers’ experience has a major role).

During routing, the program permanently displays the routed wires and also those remained to be routed. This program facility (“Netlist Navigator”) can be exploited to realize the LVS type checking, the program permanently displaying the percentage of the routed wires’ number. Thus, if after manual routing of all wires, a routing of 100% is displayed, this means that there is a total correspondence between electric scheme and its layout (“schematic and layout match”).

Fig. 1. Design flow for IC design
A disadvantage of the method is due to disappearance of the already routed wires (not also of the electric connections) which are afferent to a cell when it is reposed in order to realize a more compact layout. This aspect is specific for this type programs, which display the electric connections between cells, as happens in the case of the OrCAD program used in PCB-s design. The mentioned disadvantage can be ruled out by designing the more complex layouts by hierarchic levels, in order to decrease the number of connections inside a level. In this situation, the compounding cells of the inferior hierarchic level will be represented by elementary components as transistors, diodes, resistors, capacitors, etc. The number of hierarchic levels realized in a layout project is practically unlimited, this being settled by designer depending on the circuit’s complexity.

4. LAYOUT DESIGN EXEMPLIFICATION

4.1. Electrical scheme design

The electric scheme is made with specific editors as PSPICE or OrCAD.

To export the electric scheme in the layout it is important that the editors used for its designing allow the netlist built in a format recognized by the program used for the layout designing, as is “EDIF 2000” or “TPR” in the case of the L-Edit program.

For exemplification, in Fig. 2 is presented an electric scheme drawn with OrCAD editor, which will be exported in “EDIF 2000” format in order to design the layout with a specialized soft (as is L-Edit).

All the components of the electric scheme (cells), including pins, must have a correspondent at layout level, in order to be exported.

Thus, a cell must have three viewers: electric scheme, symbol and layout. For a successful export, all these viewers must have the same name.

Finally, after the electric scheme is drawn, its netlist is built in a format recognized by the layout editor (EDIF, TPR).

Thus the electric scheme is ready for being imported in the layout editor.

4.2. Importing the electrical scheme in layout editor

The import of the electric scheme in layout editor is realized by the netlist, in EDIF or TPR format, in the case of the L-Edit program.

If all the cells in the electric scheme have a correspondent in the layout, then the import of the electric scheme in the layout editor is successful, the electric connections between components being simultaneously displayed.

After importing, the layout editor will display the electric connections between components as shows Fig. 3.

In the following, the manual or automate wire routing is made. For manual routing, specific for analog integrated circuits designing, the program will display the routed wires and also those remained to be routed. Thus, the correspondence between electric scheme and the layout is permanently checked.

In Fig. 4 is presented the complete layout of the electric scheme shown in Fig. 2, realized in a 0.8 µm BiCMOS technology.

Fig. 2. Electrical scheme designed in OrCAD
After the layout is done, its accuracy is finally checked by DRC and LVS.

Some programs (L-Edit) allow the easy writing of technological files, required for implementing the DRC type checking. For this reason the information given by manufacturer of the used technology is used.

The program monitor permanently the routed wires and those remained to be routed.

So, if after the layout has been done, the program displays a 100% routing (as shows Fig. 5), this means that there is a perfect correspondence between the electric scheme and its layout (“schematic and layout match”). Thus, this facility may be successfully used for the LVS type checking.

By using this method, the specialized soft for the LVS type checking can be avoided, because of their inaccessibility for all designers of integrated circuits, due to high costs and difficult implementing.

![Fig. 3. Layout of the electric scheme shown in Fig. 2 before routing](image)

![Fig. 4. Layout of the electric scheme shown in Fig. 2](image)

After the layout has been finalized and passed through all checking (DRS and LVS type), the next designing level (exemplified in Fig. 1) is represented by the parasitic elements extraction from the layout in order to perform post-layout simulations.

![Fig. 5. Netlist Navigator used for LVS check](image)

Depending on post-layout simulations results one decides if the designed layout still requires or not modifications to improve performances.

Although the soft complexity and performances for integrated circuits’ designing have significantly raised in the last years, even the successful of post layout simulations do not always offer 100% guarantee that the fabricate chip will operate according to expected results.

5. CONCLUSIONS

In this paper, a new method to implement the DRC and LVS type checking in the integrated circuits designing with specialized soft is presented.

The method relies on the facility of certain layout editor programs to display in the layout the electric connections between the components (cells) of an electric scheme, and also their capacity of permanently monitoring the routed wires and also the ones remained to be routed.

To import in the layout, the electric scheme may be realized with any schematic editor but with respect for it to allow the netlist built in a format recognized by the used layout editor.

To exemplify the method, an electric scheme made in 0.8 μm BiCMOS technology was considered. The scheme was drawn in OrCAD and its import in L-Edit was made by the netlist built in the “EDIF 2000” format.

REFERENCES