

# IP-Based Library for analog design reuse

Timothée Levi, Noëlle Lewis, Jean Tomas, Pascal Fouillat  
 IMS laboratory, University Bordeaux 1  
 Talence, France

## I. INTRODUCTION

The main objective of this work is to improve the design flow of analog and mixed signal integrated circuits. Purely digital systems profit from standardized tools and methods which allow automatic synthesis. In the analog domain, automatic synthesis has probably no sense and efforts must be done towards developing consistent and coherent CAD (Computer Aided Design) methods and tools.

The main idea is to re-use the accumulated design knowledge which could be illustrated by the IP (Intellectual Property) concept. In this paper, we give our experience on the following aspects: content of an analog IP, IP-based library and its integration in the system design flow, applied on a specific example.

## II. CASE STUDY

Engineering of neuromorphic integrated systems is a research field where microelectronics encounters biology. One issue is to have the adequate simulation system that implements the models of biological neural networks; that is the role of neuromorphic engineering [1], [2]. The so-called Spiking Neural Networks implement models from the cell-level to the system level and they are exploited in neuroscience as they emulate a part of brain activity.

From the microelectronic point of view, one solution is to design analog ASICs for the real-time computation of neurons activity. As it is often the case, the development of the analog part requires the largest amount of time, due to the lack of formalism and automation in that domain.

The analog ASICs integrate several biologically realistic neurons, following a mathematical model, known as Hodgkin-Huxley formalism [3]. All model equations can be implemented using classical analog building blocks (OTAs, etc.). An experience of about ten years and 19 designed ASICs [4] allow now to have an accurate idea of the system hierarchy and the recurrent analog blocks, which is the basis of IP-based design.

Figure 1 illustrates the typical hierarchy of our analog ASICs.

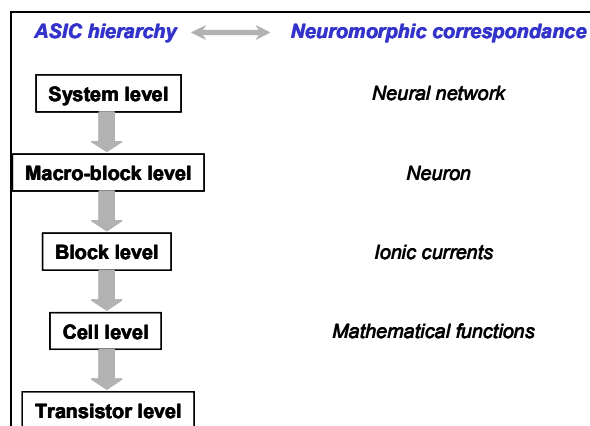


Figure 1: Typical ASIC hierarchy

## III. DEFINITION OF AN IP

Defining the IP content for analog applications is the first important task. Some works already pointed out this concept as the basis of optimization of analog design flow [5] [6] [7].

What should be the main properties of an IP block? Because the IP concept is the support of design reuse, it should give a precise characterization of already designed block; second, it has to be described with adequate representations or models, consistent with the design levels, to enable the easy re-use of the block, along the complete design flow. Table 1 shows the different descriptions (or *views*) that are embedded in our IP blocks. All views have the same terminals and the same symbol.

Table 1: IP hierarchical description levels

View name	Description / Role
<b>Symbol</b>	visualizes the function
<b>Connectical</b>	verifies connections between blocks (VerilogA)
<b>Functional</b>	models ideal electrical behavior (VerilogA)
<b>Behavioural</b>	models non-ideal electrical behavior, extracted from schematic (VerilogA)
<b>Schematic</b>	transistor-level schematic
<b>Layout</b>	chip masks generation
<b>Characterization</b>	See below ...

For the *connectical*, *functional* and *behavioral* views, we use VerilogA language. These views are useful for multilevel simulations especially in the verification phase of the design

process (Bottom-Up design flow). The *functional* view describes the ideal equations of the function to be implemented. The *behavioral* view is more detailed, composed of refined equations which fit the schematic behavior. One important point is to have a fluent and coherent design flow. That is why logical and mathematical links have been established between the different views.

The *characterization* view is the most important view for automatic exploration of IP-based architecture (Top-Down design flow). This view includes the main criteria of the re-use methodology. It contains information about the design and the re-use ability of the IP-block. The technology, the supply voltage, the terminals and their validity domains, the links between the functional model and the behavioral model, the area of the layout and the cost of the design (which may be estimated from the area) are defined in this file.

The most important criterium in our application is the validity domain which gives the range of input signals, where the circuit functionality is conserved. This information comes from systematic transistor-level simulation of the primitive *cells* (*cell level*, figure 1); then it is propagated to the *blocks* and *macro-blocks*. The set of validity domains attached to one IP defines the design space coverage by this IP.

#### IV. IP-BASED LIBRARY

All IP-blocks are collected in a data-base. This data-base must implement:

- The hierarchy of the system (figure 1)
- The IPs and their content (Table 1)

The primitive IP blocks correspond to the cell-level of the ASIC hierarchy (figure 1). However “macro” IP blocks are also available at the block and macro-block levels, in order to improve the re-use potential.

The data-base is implemented with MySQL [8] formalism. Six tables have been created: - one is for the link father-son between the different entities of the hierarchy, - one table defines each IP, - one table gives the number of IPs available for each function, - three other tables define the set of model parameters for each hierarchical level (*Macro-block level*, *Block level* and *Cell level*).

This library is integrated in the design cycle in two phases: the Top-Down phase and the Bottom-Up one.

##### A. Top-Down Phase

The Top-Down phase is the automatic design space exploration. The objective is to find one ASIC solution according to the initial specifications.

A Top-Down exploration is performed, from the *macro-block level* to the *cell level*. The exploration requests are made in Php. At the final step, a diagnostic on re-usable blocks is returned which quantifies the

use for a new ASIC project. In a case of re-use, IPs hierarchical netlists are automatically created for the designer.

The data-base automatic exploration process has been tested for many plausible specifications set. An example of a diagnostic is given in Table 2. It describes the IPs we can re-use.

Table 2: Final diagnostic

Neurons	Numbers of IPs we can re-use	Percentage of re-use for this circuit
Neuron 1	49/63	78 %
Neuron 2	51/51	100 %

##### B. Bottom-Up Phase

The Bottom-Up phase verifies if the proposed architecture really meets the specifications. Indeed the behavioral, connectical and functional models are used to perform a multi-level simulation of the entire ASIC.

All the blocks have been validated using the analog simulator *Spectre* under *Cadence* environment. Due to the huge number of components that compose the chip, it is not possible to perform transistor-level simulation of the final chip. We decided as a consequence to mix transistor level, behavioral, functional and connectical descriptions of the different blocks to validate their connectivity and ensure the functionality of the whole design. As an example, a simulation for only one neuron lasts 2’17” using *VerilogA* description and 14’56” using transistor level description.

#### V. CONCLUSION

In this paper, a case-study in analog IP re-use is exposed. The questions that have been developed are: analog IP definition, IP-based library and IP-based system design. To accelerate the time of design of our system and improve the design re-use, we define an IP-based library of analog elementary functions and a specific database of already designed blocks. This database allows us to be able to build more rapidly the next generations of ASICs (Top-Down Phase). An automatic database exploration allows obtaining a rapid estimation of the amount of re-usable circuits. The library associates behavioral and schematic views of all blocks at all hierarchical level, to be able to perform *Spectre* simulations of the whole chip (Bottom-Up Phase) in a reasonable CPU time.

#### REFERENCES

- [1] Neuromimetic ICs and System for Parameters Extraction in Biological Neuron Models, S. Saïghi, Y. Bornat, J. Tomas, S. Renaud, Proceedings ISCAS 2006, pp.4207-4210, Island of Kos, Greece, May 2006.
- [2] BiCMOS Analog Integrated Circuits for Embedded Spiking Neural Networks, Y. Bornat, J. Tomas, S. Saïghi, S. Renaud, Proceedings DCIS 2005, Lisbon, Portugal, November 2005.
- [3] A silicon neuron, M. Mahowald, R. Douglas, Nature, vol. 354, pp. 515-518, 1991.
- [4] <http://neuromorphic.ims-bordeaux.fr>
- [5] A Study on Analog IP Blocks for Mixed-Signal SoC, Zheyang Li, Li Luo, Jiren Yuan, Proceedings ASIC, pp.564-567, Beijing, China, October 2003.
- [6] UML/XML-based approach to hierarchical AMS synthesis, I. O’Connor, F. Tissafi-Drissi, G. Révy, F. Gaffiot, Proceedings FDL 2005, Lausanne, Switzerland, September 2005.
- [7] Analog/Mixed-Signal IP modelling for design reuse, N. Martínez Madrid, E. Peralías, A. Acosta, A. Rueda, DATE Conference, Munich, Germany, 2001
- [8] MySQL, P. Dubois, Sams Developer’s Library, 2005.