Using generic cell libraries to improve the automatic sizing of an operational amplifier with input and output rail-to-rail

Nuno M. R. Machado, Instituto de Telecomunicacoes Instituto Superior Tecnico, Email: mac93nuno@gmail.com
Nuno C. G. Horta, Instituto de Telecomunicacoes Instituto Superior Tecnico, Email: nuno.horta@lx.it.pt
Jorge M. C. Guilherme, Instituto de Telecomunicacoes Instituto Politecnico de Tomar, Email: jorge.guilherme@ipt.pt

***Abstract*—The complexity grow in Systems-on-Chips (SoC), brings the necessity to automate their design process in the analog world. This thesis presents a new tool to ease automate analog IC design and design reuse. It was used an existent genetic algorithm optimization-based method (AIDA) with a new structure layer. The new structure layer contains a generic cell library and a circuit class database to facilitate the migration of different technologies and topologies. To validate the topology independence in the proposed solution, testbenches were built for a specific class of circuits - Operational Amplifier with rail-to-rail input and output. The transfer knowledge through different topologies can be accomplished by reusing an existent database of testbenches. In addition, technology migration was also validated using two different technologies: XFAB 350 nm and ATMEL 150 nm SOI. Finally, this solution allows the designer to develop easily a technology/topology independent system, which can be executed in multiple designs, improving the automation process in the analog world.**

***Index Terms*—Analog integrated circuit design, Automatic sizing, Generic cell library, Genetic optimized-based method, Operational amplifier, Input and output rail-to-rail.**

I. Introduction

THE need for high-advanced and adaptable equipments, forces the microelectronic world to grown at exponential rate. Moving the markets for application-specific integrated circuits (ASIC), which generates a higher level of integration complexity systems, also known as systems-on-chip,[1]. A System-on-Chip (SoC) generic structure is composed by 80% of digital design against 20% of analog. Despite the effort to replace analog with digital structures, some functions will always remain analog. This raises a problem - design productivity gap. In design productivity gap, more functionalities are integrated on the circuit, increasing the complexity and reducing the productivity design rate [2]. The reuse of intellectual property (IP), can improve this design productivity.

The concept, reuse IP, is well established in the digital world, using generic cell libraries in memories and CPUs to automate and simplify the reuse of each structure. On the other hand, the analog world is much more dispersed with a higher degree of freedom and different design rules. These prevent the creation of a generic cell libraries. However, with the use of Electronic Design Automation (EDA), several solutions have been developed to facility design reuse.

A. Current methodologies to optimize analog circuit design.

The traditional way to design analog circuits requires significant amount of manual work done by experts in the field. As consequence, the quality and design are subject dependent, making this process unreliable, tedious and not robust. In addition, the analog design constraints are specific and restricted to each circuit.

Many approaches have already been developed and they basically can be divided into two major methodologies: procedural generator-based method or optimization-based method [3].

Procedural generator-based method consists in a ”bottom-up” sequential descriptions of analog blocks with a simple dedicated structure. It’s dependent on a database knowledge created by experts in the fiel. Several examples of procedural generator-based method are:

- The gPCDS is an interactive tool that generates conventional PCell for basic functions, giving to the designer a familiar environment [4];
- The technology abstracted layer is a generator programmer that uses a generic code of the circuit, converting to an abstract representation [5];

The optimization-based methods uses numerical algorithms to solve analog circuits design with a higher degrees of freedom, while optimizing the performance of the circuit under the given specification constraints. Several examples of optimization-based methods are:

- Based Geometric Programming solutions, the optimization is proceeded using a list of design equations to improve the results [6];
- Based Genetic Algorithm, simulate the natural process of nature [7];

Using this type of approach the designer does not require an extensive database of knowledge. When a group of constrains are defined the system will generate/calculate several solutions until reach the desired goal. It allows the designer to extend and modify the system in a easy way.

In this project it will be used a genetic algorithm optimization-based method with a generic cell libraries to be applied as a reuse IP concept.

B. Background on operational Amplifier with I/O Rail-to-Rail structures

First, a conventional Op-Amp is divided into two stages, input and output.

1) Input Stage: A conventional input stage is build using a single differential pair, it will behave properly on a limited range of common-mode input voltage. This limitation is solved using a more versatile input stage that works for a rail-to-rail common mode input range. A complementary differential amplifier is the most common method to reach the full common-mode input range. The necessity of using two complementary differential pairs solves the problem of a rail-to-rail input common-mode range, but comes with a drawback of non-constant input transconductance, as example, if this input stage is applied to an operational amplifier in a non-inverting feedback configuration the gain varies about 100% over the common-mode input range, which leads to distortion. There are several solutions to overcome these drawbacks of rail-to-rail input stages:

- The tail current control consists of maintaining a constant input transconductance by keeping the sum of the square roots of the tail current of each complementary input pair steady [8];
- Voltage Control is another technique to keep a constant input transconductance. Biasing the input transistors in strong inversion in order to control the input transconductance by controlling the gate source voltages of each differential pair [8].
- W over L based gm Control is a technique that uses the aspect ratio of the complementary differential pair to achieve a constant input transconductance [8].
- DC level shift is a technique that uses only one input differential pair and a DC shift structure to control the input common-mode voltage with the goal of putting the differential pair always in the saturation region[9].

A study of several topologies using these techniques was done, in order to selected the best method and topology choosing the lowest variations of transconductance. Selecting a topology that uses the tail current control method, [10] and another topology with a DC level shift, [9].

2) Output Stage: The need of adding an output stage to an operational amplifier is to transmit the signal to the load with enough lower to reach a low level of signal distortion. In order to achieve a better efficiency, the output voltage need to have a wide range, i.e., a rail-to-rail output. Therefore, a typical output stage of an operational amplifier can be accomplished with a class-AB [8].

This abstract is organized as follows. In section II, the proposed solution architecture is discussed. In section III, the generic cell library is described. In section IV, the proposed approach is validated. Finally, in section V the conclusions are drawn.

II. Solution

The solution proposed uses several software tools with a particular emphasis on the AIDA-C optimizer to expand the reuse of analog designs. Presenting the developed structures in order to understand the design-flow. An example will be presented for the circuit class in study, Operational Amplifier with Input and Output Rail-to-Rail stages.

A. Software used

The Electronic Design Automation(EDA) represent the category of software tools used on the design of integrated circuits. Therefore, choosing the correct tool is extremely crucial to the project's success. The first step will be define the main tasks on the EDA.

Design and Simulation reflect into a topology selection and circuit sizing by a designer [11].

The first step to understand the proposed solution will be to know the tools chosen.

- **Cadence** - The chosen software is Virtuoso Schematic Composer from Cadence. It was used to implement the circuit design of each typology by using the generic design kit of CMOS components [12].
- **Mentor Eldo** - Eldo offers a unique partitioning scheme, allowing the use of different algorithms on differing parts of the design. The user has a flexible control of the simulation accuracy using a wide range of device model libraries and a high accuracy yield, combining the high speed and performance [13].
- **AIDA** - AIDA-C is a tool that target sizing of the analog devices using state-of-the-art multi-objective multi-constraint optimization techniques, addressing robust design requirements in a worst case approach. In order to reach a accurate circuit's performance, this tool use several industrial grade circuit simulators, enabling the designer to choose his own simulator [14].

B. Solution architecture

The solution developed is structured into three central blocks described below.

- **CircuitClass**, database of a topology independent test-benches with the aim of extracting measures to characterize a specific circuit class;
- **GenericCellLibrary**, database of generic symbols which can allow the creation of technology independent circuit netlists, due to their ability of ported into a specific technology;
- **Building Script + Configuration file**, files that join the previous databases, CircuitClass and GenericCellLibrary, into a project folder for AIDA-C, in order to optimize the circuit using all the input given by the configuration file;

For a better understanding of the solution architecture, a overview of the tool with the scheme depicted is presented in the figure 1.

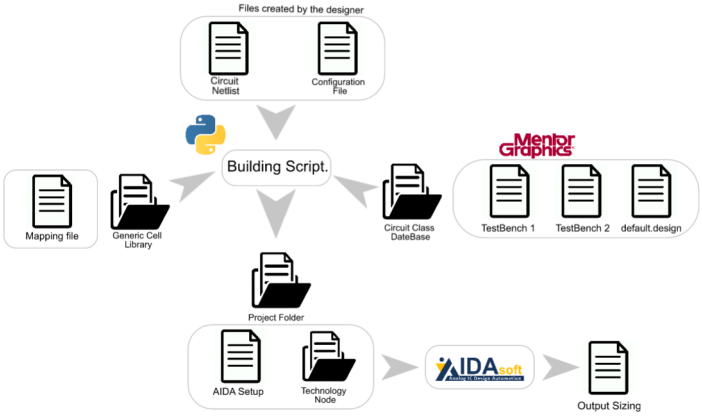

Fig. 1. Overview of the project architecture.

The project starts with two important blocks, created by the designer: Configuration file and the circuit netlist.

The files that make up the topology choose, correspond to the circuit netlist. The schematic need to be outlined using the devices in the generic cell library.

The other important block is the configuration file, the designer can choose the technology, the topology and dependent options. The following sub-chapter represents all the information about the current file.

The output generates an AIDA-C complete project, located in a folder with the same name of the configuration file, containing the files:

- Testing files, which have the technology independent testbenches with the topology dependent measure files.
- Directory files, which have all the directories need to choose the exact mapping file technology, specified in the configuration file.
- AIDA-C configuration file is generated to the project's demands by the designer, obtaining the file `design.xml` (it can be edit form default values, like, constraint, objectives, variables ranges, corners, etc).

1) Configuration file to build the project: In order to facilitate and make this system more user-friendly a configuration file was created. In this file the designer only needs to specify four main parameters.

The structure of this file was define through the analysis of a project. The first step is the selection of a `TECHONOLOGY`, followed by the characterisation of the circuit class, `CLASS`, making reference to the path of the circuir netlist, `CIRCUIT`. At the end, restrictions can be added to reach all the designer specifications, `MEASURES`.

The configuration file uses the `.config` filename extension to be identify by the script.

```
TECHNOLOGY: "Tech name"
CLASS:      "Circuit class"
CIRCUIT:    "Path to netlist"
MEASURES:   "Transistor region measures"
```

2) Script to build the project: To build the script was used the programming language python. Python brings several advantages, it is an interpreted high-level language and interpreters are available for Windows, Mac OS X, and Lunix. Thus the same code can be executed on these different platforms with no changes to the code. It hosts thousands of third-party modules, helping to overcame all challenges. And it is open-source.

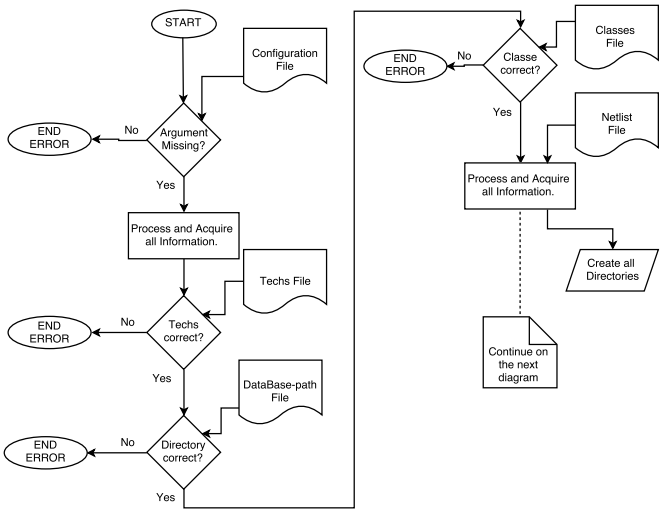

Fig. 2. Code Flow Diagram of the initial part of the Build Script.

The script is divided into two stages, it starts with the checking of all the input information and then pass to the creation of all the specific project files.

The first stage of the script represents a verification of an input configuration file and all its parameters. An essential step to reach the success of the final setup. The verification process starts with an **Argument Validation** by reading a configuration file.

The configuration file is defined by several parameters needed to be validate. The script executes a sequential verification:

1) **Technology Validation**, the script will check if the chosen technology is located on the file `techs`;
2) **Directory Validation**, the script will check if the designer set a directory on the file `database_path` to the class of circuit and the generic cell library;
3) **Class of circuit Validation**, the script will check if the designer specify an existing class on the circuit class database, the list of each class is located on the file `classes`;

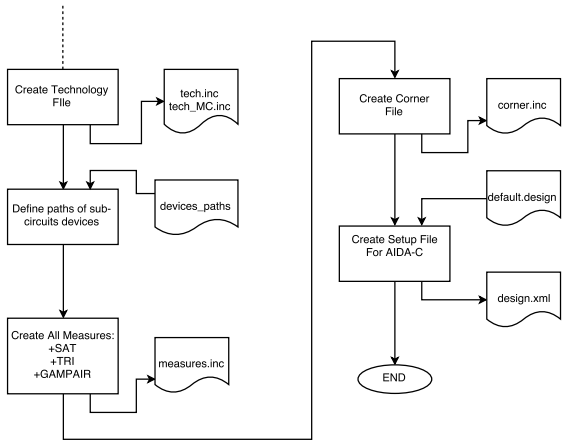

Fig. 3. Code Flow Diagram of the final part of the Build Script.

The second stage represents the **Creation Files** for the project.

1) **Technology File creation**, knowing which technology will be use, the script choose a tech libraries accordingly. Two files are create a typical and worst case analyses and other file for Monte Carlo analysis, `techs.inc` and `techs_MC.inc`;
2) **Device path search**, device path is mandatory in order to create the next files. A search algorithm was create to read and formulate the device path from the `.cir` files of the project. This algorithm uses a base file path, `devices_paths`;
3) **Measures File creation**, the script perform all measures functions defined on the configuration file. At the moment the script only perform three types of measures, mosfet saturation, mosfet triode and input transconductance, `+SAT`, `+TRI` and `+GMPAIR`;
4) **Corner File creation**, a `corner.inc` file is create using a pre-create file `corner_.inc` with the technology specification defined on the configuration file. *AIDA-C Setup File creation*, the script uses a pre-structure file, `default.design` and the project configuration file information to preform the final setup, `design.xml`;

C. Generic Cell Library

In order to meet the objective proposed, a small group of generic devices are created. It is described the detailed list of devices that can be used by different software, CADENCE, AIDA-C and Mentor ELDO.

The aim was the establishment of a platform to facilitate the technology transferability of the same circuit/topology. The solution consists in having a generic library to map the technology chosen for all the devices.

The generic library had a collection of 10 devices, mosfet transistor type N and P, resistors, capacitors and one bipolar transistor. It is important to refer that not all devices have a match for the technologies in study, as can be seen in table I.

1) Component Description Format CDF: All the devices are designed in CADENCE environment.

In order to use the CADENCE environment, each device had to be represent by a complete element cell. Which can be accomplished through the Component Description Format, CDF.

To create a cell element using CDF, it is important to chose what type of application intend to be used. The following list presents all the information which was selected on CDF.

- Application:
  - Design Entry;
  - Simulation;
- Requirements views:
  - Desing Entry: Symbol with G, S, B, and D pins for the gate, source, bulk and drain;
  - Simulation: Type of Simulator(Eldo);
- Requirements parameters:
  - Simulation: Width, Length and Multiplicity (W,L,M);

A library was created with the name of `Generic_lib`, where all the cell elements are presented.

TABLE I
THE LIST OF GENERIC CELL LIBRARY.

Generic Name	Description	Parameters	Port order	AT77K	XH035
nmos1v8	$1.8V$ NMOS Transistor	`W,L,Mul`	Drain, Gate, Source, Bulk	nmos	N/A
nmos3v3_elt	ELT $3.3V$ NMOS Transistor	`W,L,Mul`	Drain, Gate, Source, Bulk	`nfetox3_ring`	`nmos_elt`
pmos1v8	$1.8V$ PMOS Transistor	`W,L,Mul`	Drain, Gate, Source, Bulk	`pmos`	N/A
pmos3v3	ELT $3.3V$ PMOS Transistor	`W,L,Mul`	Drain, Gate, Source, Bulk	`pfetox3`	`pmos`
rpoly	Poly resistor	`W,L,Mul`	Plus, Minus, Sub	`rplow`	`rp1`
rpolyhr	Poly resistor	`W,L,Mul`	Plus, Minus, Sub	`rphigh`	`rpp1`
rpolyxr	Poly resistor	`W,L,Mul`	Plus, Minus, Sub	N/A	`rhp1`
mimcap	MIM capacitor	`W,L,Mul`	Plus, Minus, Sub	`cmim34`	`cmm`
dmimcap	Double MIM capacitor	`W,L,Mul`	Plus, Minus, Sub	N/A	`cdmm`
pnp_vertical	Vertical PNP transistor	`Mul`	Collector, Base, Emitter	`pnp_vert`	`qp4`

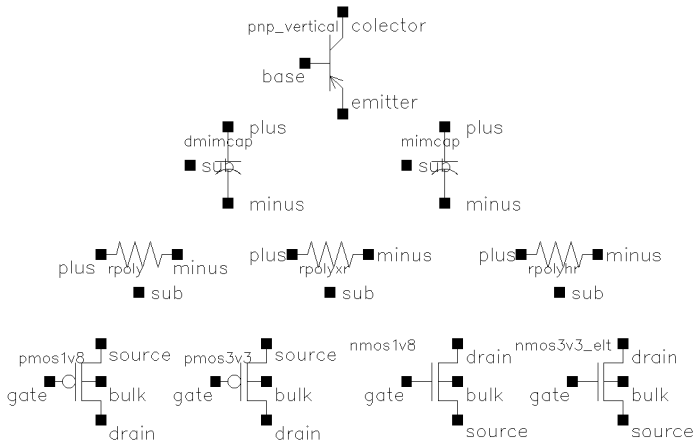

Fig. 4. Devices symbol views of the full Generic Library.

D. Mapping File Creation

It is important to understand what level the project it will be preformed. As mentioned before, the idea of the solution was to reach the best performance in SPICE level.

The SPICE level consists of a netlist which describes the various circuit devices and their connections, a set of models for the circuit devices and a specification of the type of analysis to be performed for a given circuit.

Due to the large variety of manufactures, each model had his own cell view, being necessary to rebuild every time the circuit when the designer change the technology. A generic cell library has been created for standardise the listed devices. Now the second stage is to map this cell views with the models of each manufacture.

The mapping file solution needed to be done one time only for each manufacturer technology using the following method. First each generic device is defined as a subcircuit, using the `.SUBCKT` command, this feature consists on represent a circuit in a fashion similar way to devices models, see the next general example:

```
.SUBCKT SubName Connection1 <Connection2
   Connection3 ...>
*Technology models*
.ENDS

```

Before presenting the final structure of the mapping file for each technology, it is crucial to understand the process corners and Monte Carlo Analysis and how to related with different technologies.

1) Mapping of process corners and Monte Carlo analysis: Two types of simulation are done on this thesis, process corners and Monte Carlo. Like the previous problem, there were not a standardisation for the two types of simulation by the manufactures. Each one had its own way to perform the simulation combining different libraries.

Each manufacture had a different way to present their processes corners. Therefore, create a generic definition to each process corner is the first step. Table II provide these information, must more combination are possible:

TABLE II
DEFINITION OF PROCESS CORNERS MODELS.

Corner	Mosfet		Resistor	Capacitors
	Nmos	Pmos		
TM: Typical Mean	*Nominal*	*Nominal*	*Nominal*	*Nominal*
WO: Worst Case One	*Fast*	*Slow*	*Nominal*	*Nominal*
WS: Worst Case Speed	*Slow*	*Slow*	*High*	*High*
WP: Worst Case Power	*Fast*	*Fast*	*Low*	*Low*
WZ: Worst Case Zero	*Slow*	*Fast*	*Nominal*	*Nominal*

In the design of an IC is important to simulate the mismatch and process variations, a Monte Carlo analyses is used to fulfil these simulations. Like the process corner, each manufacture had its own method to represent this analyses.

In closing, table III shows the matching of library's corners and Monte Carlo analyses between the two technologies in study..

TABLE III
DEFINITION OF PROCESS CORNERS MODELS.

Generic Name	Lib: XH035	Lib: AT77K
`typ`: Typical Mean	*tm*	`mos_nom, rnom, cnom`
`wo`: Worst Case One	`wo`	`mos_fsc, rnom, cnom`
`ws`: Worst Case Speed	`ws`	`mos_wcs, rhigh, chigh`
`wp`: Worst Case Power	`wp`	`mos_bsc, rlow, clow`
`wz`: Worst Case Zero	`wz`	`mos_sfc, rnom, cnom`
`mc`: Monte Carlo	`mc_g`	`mc, matching`

2) Mapping file: `at77k.lib` & `xh035.lib`: After defining the generic devices and the process analyses, including the corners and Monte Carlo. The next step was performing a match structure for each technology in study. The structure needed to include a device library and the process analyses libraries.

To facilitate the designer finding all the information needed to a specific technology, each matching structure is located in a single file. For the case of `AT77K` technology, the file was the name `at77k.lib` and for the `XH035` technology, the file `xh035.lib`.

III. OPERATIONAL AMPLIFIER DATABASE

As mentioned before, there are two main database required for the solution. The generic library, already cited, and a circuit class database, created by the designer when a new class of circuit is added to the database. His creation can be explained using as example the class of circuit Operational Amplifier Input and Output Rail-to-Rail, **`R2ROpAmp`**.

A generic circuit class database is defined by three groups. The first one is identified by the name of **Netlist**, all the topologies examples are locate here. The second group establishes the tests and measures files, it is identified by the

name **TestBench**. The last group is a single configuration file, with the name **default.design**, which has all the class circuit's characteristics but independent of the circuit topology. The follow image represented the graphic diagram for a better understanding.

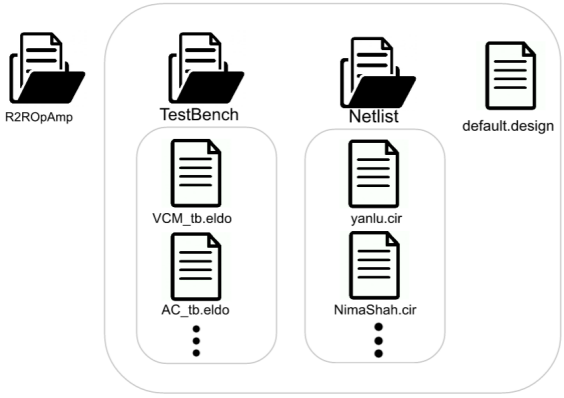

Fig. 5. Structure of the database `R2ROpAmp`.

A. Testbenches

A class of circuit is characterized by a list of performance parameters. Each parameter is measured using a data set of simulations files.

The table IV resumes all the association for each parameter in the circuit class Op-Amp.

TABLE IV
ASSOCIATION OF WHICH MEASURE TO THE TESTBENCH FILE.

Testbench file Name	Performance Parameter
AC_tb.cir.eldo	Gain;
	Gain Bandwidth
	Phase Margin
	PSRR
VCM_tb.cir.eldo	Variation of transconductance;
AC_CMRR_tb.cir.eldo	CMRR
TRAN_tb.cir.eldo	Slew rate;
OutSwing_tb.cir.eldo	Output-voltage swing;
ICMR_tb.cir.eldo	ICMR;

1) AC_tb.cir.eldo: The testbench presented is one of the methods used to measure the AC performance. This configuration give the possibility to obtain several performance parameters in one analysis which would normally be required two different testbench. One for measure the PSRR and other to measure the open-loop characteristics.

The circuit uses an AC small signal source inserted in series with the power supply V_{DD}, a voltage common mode source is added to the non-inverting input "+" and a loop stability source, LSTB,is link between the output and the inverting input "−".

The open-loop characteristics are obtained using a specific commands from the LSTB. The remaining measures are done using conventional functions from the simulator program.

The parameters calculated are the Open-Loop Gain, `GDC`, Gain Bandwith, `GBW`, the phase margin, `PM`, and PSRR, `PSRR`.

2) VCM_tb.cir.eldo: The variation of the input stage transconductance is an important measure for this type of circuit class. The transconductance affects a numbers of parameters on the Op-Amp, so, this value needed to have the minimal variation possible.

The previous testbench is used to perform the needed analysis. The simulation uses the same structure, the change only occurs on the type of analysis. It is perform a DC Sweep analysis of the common mode voltage source, VCM, from 0 to VDD.

The parameters calculated are the positive variation classified by the name `VARGMP` and the negative by the name `VARGMN`.

3) AC_CMRR_tb.cir.eldo: There is the need to eliminate the common signal of both inputs for a correct functioning of an Op-Amp. Therefore, the common mode rejection ratio, CMRR, is an important requirement, defining the standard measurement factor when comparing differential circuits. The equation 1 is used to calculate the CMRR.

$$CMRR = \frac{A_d}{A_c} \tag{1}$$

There are several solutions to perform this measurement using different configurations, such as *Matched Sources Measurement Setup*, *Power Supply Measurement*, *DC CMRR Measurement Setup* and *Matched Resistor Measurement Setup* [15].

The solution selected represents the method *Matched Resistor Measurement Setup*. The circuit is composted by a differential amplifier configuration with the two inputs link together, using matching resistor. The problem associate with this circuit is the non direct measure of CMRR. So, it is important to reach a relation to the input and the output in order to reaches the value of CMRR.

The parameter calculated is the CMRR.

4) TRAN_tb.cir.eldo: Another crucial parameter to characterize an Op-Amp is the Slew Rate. First, it is not frequency response but instead it is the ability of the output change from one value to other within a certain time, typically measured by $V/\mu s$.

An unit-gain configuration can be use to perform this measure. This configuration presented a largest feedback, resulting in the largest values of loop gain. Therefore, it should only be used to worst-case measurement. That is why the unit-gain configuration testbench was selected for this work.

The parameters calculated are the positive Slew Rate, `SLWUP`, and the negative Slew Rate, `SLWDOWN`.

5) OutSwing_tb.cir.eldo: The class of circuit in study represents an Op-Amp rail-to-rail output. Therefore, the output voltage range of the amplifier can span one supply to the other. But in reality, the use of complementary CMOS to preform the output stage limit the voltage range due to the internal resistance of the transistors, so the output voltage swing will be reduced.

The testbench selected puts the Op-Amp in an inverting configuration, where it is applied a DC Sweep input signal to perform a transfer curve output signal. The output-voltage swing is located on the linear part, so, when the linearity is lost the values of `VOH` and `VOL` can be obtained.

The parameters calculated are the output voltage high, `VOH`, and the output voltage low, `VOL`.

6) ICMR_tb.cir.eldo: The creation of the current testbench aims to validate if the topology selected has the characteristic needed to the class of circuit in study, Op-Amp with an input rail to rail.

The definition of this parameter consists to identify the range of input voltage where the Op-Amp will operate correctly. So, the idea is to sweep the common mode input voltage in order to identify where the normal operation output will no longer be possible. Identifying the voltage edges, where the low voltage is named by `VIL` and the high voltage with `VIH`.

The testbench selected place the Op-Amp in a differential configuration. The goal was to obtain a constant value in the output voltage, in this case, the constant is equal to the common mode voltage, `VCM`. When this constant do not exist its reach the values of the voltage edges, on lower and higher voltage.

The parameters calculated are the low voltage, `VIL`, and the high voltage, `VIH`.

B. Netlists circuits

Each Op-Amp topologies employ a traditional two gain stages, differential pair and a folded cascode stage, in order to achieve high impedance and gain followed by a class-AB output stage, providing an output rail-to-rail.

The followers Op-Amp topologies are designed in virtuoso's environment with Generic Library process.

Three topologies use to test the proposed solution.

- Yanlu topology: create a rail-to-rail input characteristic by having two dummies input differential pairs in order to control the tail current [10];
- NimaShahpari topology: a DC level shifter to perform a constant transconductance as V_{cm} varies from rail to rail [9];
- CurrentFlow topology: similar to *yanlu* topology, it is a two dummies input differential pairs structure with a different current control flow;

C. Configuration file `default.design`

The final step to complete the circuit database consists on the creation of `default.design` file. The needed came form the necessity to build the AIDA-C configuration file `design.xml`.

The independent class-characteristic topology merge all the tesbenchs previous presented, the list of corners and independent constrains. Defining all the default requirements of the circuit class on the `default.design` file. In order to the script understand all the requirements some key words are defined, as can be see on the list bellow.

- `START_CORNER` : Command for the script, defining the beginning of the default corner;
- `END_CORNER` : Command for the script, defining the ending of the default corner;
- `START_TESTBENCH`: Command for the script, defining the beginning of the default testbench;
- `END_TESTBENCH` : Command for the script, defining the ending of the default testbench;
- `START_CONSTRAINS` : Command for the script, defining the beginning of the default constrains;
- `START_CONSTRAINS` : Command for the script, defining the beginning of the default constrains;

The dependent class-characteristic topology merge all the topology dependent requirements created by the script. This type of requirements are defined on the `default.design` file by the key word `MEASURE`.

The defaults values used in this project can be resume on the following tables. Starting with the default corners presented on the table V, was chosen four cases of PVT, defining the two typical cases, Worst Speed (Worst Speed process, high temperature and low voltage supply) and Worst Power(Worst Power process, low temperature and High voltage supply). The other two PVT cases represent the Worst One case and Worst Zero case combining the variation of temperature and supply voltage.

TABLE V
DEFAULT CORNER PROCESS.

Name	Process		Supply Voltage	Temperature
`wp_TempLow_VDDHigh`	Worst Power	**WP**	VDD*1.05 = 3.465 [V]	-55 [°C]
`ws_TempHigh_VDDLow`	Worst Speed	**WS**	VDD*0.95 = 3.135 [V]	125 [°C]
`wo_TempLow_VDDHigh`	Worst One	**WO**	VDD*1.05 = 3.465 [V]	-55 [°C]
`wz_TempLow_VDDLow`	Worst Zero	**WZ**	VDD*0.95 = 3.135 [V]	-55 [°C]

IV. RESULTS

The final procedure, execution and collection of results in order to validate the solution presented. Presenting the decision procedure to choose the best topologies. And all post-optimization analyses possible, showing the versatility of this solution.

As it is known, the solution gives the designer an easy and fast migration of different topologies and technology. Therefore, it needs to be validated.

All the results that will be presented are based on the AIDA-C automatic generator.

The table VI presents the default ranges used in the optimization process for the variables W, L and Multiplicity.

TABLE VI
DEFAULT VARIABLE RANGES USED.

Type of Variable	Min	Max	Step
W	1	30	0.1
L	0.4	10	0.1
Mul	1	1000	1

A. Selection of the best topology

The validation need to prove two essential characteristics:

- Technology independent system;
- Topology independent system;

The circuit netlist needed to be created using the generic cell library for all three topologies.

The configuration file of each topology and its respondent technology will merge all the information to build the correct setup for AIDA-C. The class of circuit, `CLASS`, is the same for all the configuration files with a code name `R2ROpAmp`.

The circuit topology, `CIRCUIT`, is specific for each topology, the parameter will receive the relative path to a netlist file.

The correct operation will be accomplish by defining the circuit measures, `MEASURES`. Setting all transistors needed to be in linear zone, using the command `+SAT` plus the name of each device. It will inform the script to perform the calculation for the overdrive and saturation margin. The variation of the transconductance could be measured using the command `+GMPAIR` identifying the input differential pair transistors. The list of transistors change for each topology.

The circuit technology, `TECHNOLOGY`, defines the two technologies in study, `at77k` and `xh035`. So each topology will have a configuration file to each technology.

In order to validate the description it is presented an example of a configuration file.

```
************************
*** Author: Nuno Machado
*** Year: 2016
*** Local:  IT Lisbon, IST
*** Comment: Circuit of Yanlu  with the technology
    at77k
************************

TECHNOLOGY: at77k

CLASS: R2ROpAmp

CIRCUIT: .../R2ROpAmp/Netlist/yanlu.cir

MEASURES: +SAT: {ALL, MP1, MP2, MN1,
MN2}  VOV=0.1   DELTA=0.15
+SAT: {MP1, MP2, MN1,
MN2}        VOV=0.05  DELTA=0.1
+GMPAIR: {MN1, MP1}
```

Listing 1. Configuration file to Yanlu on the tech: at77k.

The traditional presentation of the results in this type of optimization circuit process are performed by a *Pareto Front*. The enhancement of an objective performance is impossible to obtain without compromise others. One example, it is the engagement of the power consumption with gain bandwidth or dc gain. However the objectives chosen for this thesis did not create a *Pareto Front*. Therefore, the results presented here refer to the best achieved in each technology.

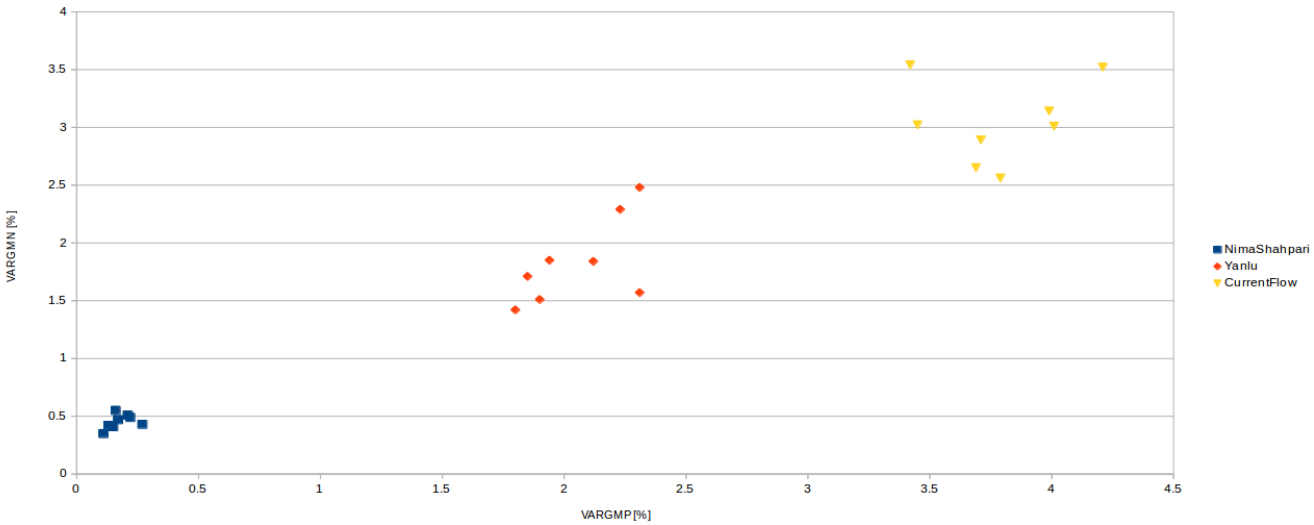

Fig. 6. The results of objective optimization in tech `at77k` for all three topologies used.

Using the results of the graphic 6, it is possible to see the best topology for the objective selected in the technology `at77k`. As can be seen, the topology *NimaShahpari* presents the best achieved solution. The table VII give a complete parametrization.

TABLE VII
VALUES OF THE PARAMETERS FOR THE CIRCUIT *NimaShahpari* WITH THE TECH `AT77K`.

		AT77K					
Parameter		**Typical**	**C[0]**	**C[1]**	**C[2]**	**C[3]**	**Seek**
IDD	[mA]	30.86	31.7	30.3	31.6	30.6	
GDC	[dB]	101.9	97.8	102.3	97.8	103.9	[≥] 70
GBW	[MHz]	72.9	95.2	55.6	83.9	95.4	[≥] 60
PM	[°]	51.8	51.7	53.2	56.1	46.6	[≥] 45
PSRR	[dB]	-94.4	-77.6	-101.8	-78.9	-104.4	[≤] -70
VARGMP	[%]	0.11	0.45	0.19	1.05	0.51	[≤] 4
VARGMN	[%]	0.35	0.69	0.93	1.63	0.76	[≤] 4
CMRR	[dB]	-86.6	-90.0	-74.5	-92.3	-92.0	[≤] -70
SLWUP	[V/μs]	69.9	73.9	61.4	67.7	74.0	[≥] 5
SLWDOWN	[V/μs]	66.4	69.2	65.4	67.6	68.3	[≥] 5
VOH	[mV]	55.6	68.6	71.1	68.6	49.3	[≤] 100
VOL	[mV]	9.6	8.1	34.1	8.1	3.0	[≤] 100
VIH	[V]	3.57	3.77	3.36	3.77	3.57	[≥] 3.3
VIL	[V]	-0.26	-0.29	-0.18	-0.4	-0.31	[≤] 0.0

B. Post-Optimization analysis and Conclusions

The system build give to the designer a verifiability and freedom by using the post-optimization analysis. This feature speed up the process of optimization, overpassing non-essential simulations to the desired performance, adding a more complete characterization to the circuit.

The idea is to choose a smaller list of tests and verifications to be perform with AIDA optimizer. The compute power needed will be reduce and hence the number of generations increase faster, given a more diverse solutions to test.

V. CONCLUSION

The increasing complexity in systems-on-chip demands a well defined structures and reliable processes. Therefore, an automated process is desired to have a consistent and higher productivity. However, a drawback is imposed by the analog world, due to a more disperse design rules.

In this work it is presented a new tool to ease automate analog circuit design and design reuse. The new structure layer contains a generic cell library and a circuit class database to facilitate the migration of different technologies and topologies.

In order to validate the generic cell library two distinct technologies were used. One of them was the foundry XFAB and the other one the ATMEL. A mapping file was applied, which link each technology to the developed generic cell library.

The structure layer has a second advantage - topology independence. This concept is a crucial step to the IP reuse. The transfer knowledge through different topologies can be accomplished by reusing an existent database of simulation files.

The proposed solution is not complete without the AIDA-C framework, so the new developed structure layer needs to merge with AIDA-C.Based in our knowledge we believe that this approach is the most suitable to reduce the productivity gap problem in analog circuit design.

Finally, in this work it was showed the advantages of using this proposed solution- a genetic optimization-based method merged with the new technology and topology abstraction. This solution allows the designer to develop easily a technology/topology-independent system, which can be executed in multiple design. Based in our knowledge we believe that this approach is the most suitable to reduce the productivity gap problem in analog circuit design.

VI. FUTURE WORK

The proposed solution in this thesis showed a good results in overcoming the productivity gap between analog and digital world. However, the results only apply to a limit small circuit class, operational amplifier. Therefore, to improve the proposed solution, further work is required to create a robust and extensive class database, adding more circuit classes. Furthermore, the simulations analyses only were applied to typical and corners processes in the optimization process, then parametrized testbenches and future developments on AIDA framework could help to more complete analysis, for example. The random offset, a performance characteristic on operational amplifier, are presently done in a post-optimization analysis, using a Monte Carlo analysis.

Layout technology independence is a more complex process, than the netlist level perform in this thesis. So, with the use of the AIDA-L, a module from AIDA framework that generates the complete layout for sized circuits, in order to create a generic cell library integrating layout level to reach a full technology independence database of devices.

REFERENCES

[1] G. G. E. Gielen and R. A. Rutenbar, "Computer-aided design of analog and mixed-signal integrated circuits," *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1825–1854, Dec 2000.

[2] M. Barros, J. Guilherme, and N. Horta, *Analog Circuits and Systems Optimization based on Evolutionary Computation Techniques*. Springer, 2010.

[3] G. G. E. Gielen and R. A. Rutenbar, "Computer-aided design of analog and mixed-signal integrated circuits," *Proceedings of the IEEE*, vol. 88, no. 12, pp. 1825–1854, Dec 2000.

[4] M. Greif, D. Marolt, and J. Scheible, "gpcds: An interactive tool for creating schematic module generators in analog ic design," in *2016 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, June 2016, pp. 1–4.

[5] B. Prautsch, U. Eichler, T. Reich, A. Puppala, and J. Lienig, "Abstract technology handling for generator-based analog circuit design," in *ZuE 2015; 8. GMM/ITG/GI-Symposium Reliability by Design*, Sept 2015, pp. 1–6.

[6] M. d. Hershenson, S. P. Boyd, and T. H. Lee, "Optimal design of a cmos op-amp via geometric programming," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, no. 1, pp. 1–21, Jan 2001.

[7] S. Sridhar and R. S, "A comparative study of design of analog cmos circuits using numerical optimization strategies," in *2015 International Conference on Trends in Automation, Communications and Computing Technology (I-TACT-15)*, vol. 01, Dec 2015, pp. 1–6.

[8] R. Hogervorst and J. Huijsing, *Design of Low-Voltage Low-Power Operational Amplifier Cells*. Boston: Kluwer Academic Publishers, 1996.

[9] N. Shahpari, R. Dehghani, and P. Rabani, "A process and temperature robust constant- input/output rail-to-rail op-amp," *Microelectronics Journal*, vol. 46, no. 6, pp. 506–512, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0026269215000749

[10] Y. Lu and R. H. Yao, "Low-voltage constant-gm rail-to-rail operational amplifier input stage," *Solid-State Electronics*, vol. 52, no. 6, pp. 957–961, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0038110108000543

[11] D. Marolt, M. Greif, J. Scheible, and G. Jerke, "PCDS: A new approach for the development of circuit generators in analog IC design," in *Microelectronics (Austrochip), 22nd Austrian Workshop on*, Oct 2014, pp. 1–6.

[12] "Top-Down Design Methods Bring Back The Useful Schematic Diagram," 2011. [Online]. Available: http://electronicdesign.com/boards/top-down-design-methods-bring-back-useful-schematic-diagram

[13] Mentor Graphics Corporation , in *Eldo Users Manual*.

[14] R. Martins, N. Lourenço, S. Rodrigues, J. Guilherme, and N. Horta, "Aida: automated analog ic design flow from circuit level to layout," in *Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), 2012 International Conference on*. IEEE, 2012, pp. 29–32.

[15] J. Zhou and J. Liu, "On the measurement of common-mode rejection ratio," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 52, no. 1, pp. 49–53, Jan 2005.