Exploring the use of Cadence IC in Education

Danijela Efnusheva, Josif Kjosev, and Katerina Raleva

***Abstract*—Microelectronics technologies and structures is electronics subfield, related to the study of integrated circuit design and fabrication. To familiarize with this widely applicable area, engineering students should gain practical experience, in addition to the theoretical knowledge attained on different microelectronics courses. Cadence IC is a world standard tool in this area, applicable not only in industry, but also in academic institutions. The academic environment of Republic of Macedonia has the ability to use this package through cooperation with foreign partners. Assuming that the access to the provided design kits is limited, the freely available NCSU CDK library from the North Carolina State University is integrated within the Cadence IC environment, and later used for educational purposes. The basic contribution of this paper is the systematization of the methodology for NCSU CDK library application by the students at FEEIT - Skopje, at both third cycle, and other levels of studies. Our general opinion is that the Cadence IC tool suite provides novel opportunities to academia and students in many educational and research activities.**

***Index Terms*—Cadence IC, CMOS inverter, integrated circuits, microelectronics, NCSU design kit, and ring oscillator.**

Original Research Paper
DOI: 10.7251/ELS1317089E

I. Introduction

The process of designing integrated circuits is very demanding, especially when a complex circuit that contains a large number of transistors is going to be designed. In order to enable automation of this process, Electronic Design Automation (EDA) and Computed Aided Design (CAD) tools are used [1]. These environments are very complex and usually integrate several tools required to support different phases of the integrated circuit design flow. Cadence IC is a tool suite, commonly used for the given purpose. This environment is not only a standard in the electronics industry, but is also widely applied in many educational and research institutions around the world, [2], [3]. Despite its expensive license, academic institutions gain access to this set of tools via cooperation with various electronics companies. Universities can use Cadence IC tool suite for several purposes including education, science and industry. Actually, this paper presents the experiences in using Cadence IC tool suite within the microelectronics technologies and structures course, at the Faculty of Electrical Engineering and Information Technologies in Skopje.

The Cadence IC environment is used in many wide world universities for educational purposes. Therefore, there are a numerous tutorials and documentation about this software usability and operation, such as many laboratory exercises from a number of courses in the field of microelectronics, [4]. There are also many papers in the area of education, [2], [3], which share their experiences and provide opportunities for improving the process of learning Cadence IC as a tool suite which is commonly used in microelectronics, analog circuit design and other related courses in the electronics field.

Cadence IC enables integrated circuits design and fabrication with various technology processes, specified by the manufacturer. When a particular technology is selected, the Cadence IC environment, [4] is customized with appropriate configuration files, known as design kit. Actually, this paper describes the use of Cadence IC 5.1.41 set of tools along with NCSU Cadence design kit 1.5.1. This design kit is available free of charge from the University of North Carolina and is used to provide full-custom CMOS IC design through MOSIS. It includes several technology files which define the mask layers and their appearances and properties, as well as parameters used at library creation time which set the value of lambda, the technology code, and the availability of process-dependent layers, along with the SCMOS (Scalable Complementary metal-oxide-semiconductor) rules [5] for the given MOSIS process.

The procedure of downloading NCSU CDK 1.5.1 includes free registration on the web site: http://www.eda.ncsu.edu/wiki/NCSU_CDK_download and obtaining a link to a free version of this software on the user's e-mail address. After that, the user has the opportunity to select one of the two available versions of NCSU CDK: 1.5.1 or 1.6.0.beta. The first NCSU CDK version can be used in combination with several different versions of Cadence IC ranging from 4.4 to 5.1, while the second version is unstable and is still in development, mainly used in combination with Cadence IC 5.2.51 or 6.1 versions, [6].

Technology processes supported by NCSU CDK 1.5.1

Manuscript received 28 November 2013. Received in revised form 16 December 2013. Accepted for publication 17 December 2013.

Danijela Efnusheva, Msc, is Phd student and teaching assistant at the Faculty of Electrical Engineering and Information Technologies, Skopje, R. Macedonia (e-mail: danijela@feit.ukim.edu.mk).

Josif Kjosev, Phd, is full professor at the Faculty of Electrical Engineering and Information Technologies, Skopje, R. Macedonia (e-mail: josif@feit.ukim.edu.mk).

Katerina Raleva, Phd, is associate professor at the Faculty of Electrical Engineering and Information Technologies, Skopje, R. Macedonia (e-mail: catherin@feit.ukim.edu.mk).

follow the SCMOS rules for design and verification, defined by the MOSIS IC fabrication service. The main idea of MOSIS is to provide a nearly process- and metric-independent interface to many CMOS fabrication processes available through MOSIS, in such a way that designers work in the abstraction of SCMOS layers and metric unit ("lambda"), [5], [6]. After the designer specifies the process and feature size of the design that should be fabricated, MOSIS maps the SCMOS design onto that process, generating true logical layers and absolute dimensions required by the process vendor. The designer can regularly submit exactly the same integrated circuit, but to a different fabrication process or feature size. In such case, MOSIS will alone handle the new mapping. On the other hand, using a specific vendor's layers and design rules will yield to a design which is less likely to be directly portable to any other process or feature size.

The general purpose of MOSIS is to combine designs from multiple customers (companies and universities), or diverse designs from a single company, on one mask set (Multi Project Wafer - MPW). This allows customers to share overhead costs associated with mask making, wafer fabrication, and assembly. The cost savings, along with the ability to use NCSU design kit and to support many technology processes, makes MOSIS attractive choice for integrated circuits design. According to [7], more than 50,000 integrated circuits have been processed through this service, since 1981.

The main idea of this paper is to present the experiences in using Cadence IC tool set and NCSU design kit for educational purposes. The paper is organized in five sections. In second section we describe the customizations of Cadence IC environment, which are required to enable the technology processes specified by NCSU CDK. In third section we explain the process of designing integrated circuits in Cadence IC, starting from schematic till layout. The results of the practical realization of CMOS inverter and ring oscillator in 0.25 µm TSMC MOSIS technology are presented in the fourth section. The paper concludes in the fifth section, which summarizes the experiences of using these tools within the microelectronics technologies and structures course at the Faculty of Electrical Engineering and Information Technologies in Skopje.

II. Cadence IC Environment Customization

Cadence IC is an environment which allows integrating in a single framework different applications and tools (both proprietary and from other vendors), allowing to support all the stages of IC design and verification. To invoke the Cadence IC environment, the command icfb& is entered at the command prompt in an UNIX based system (in this case Red Hat). This command opens the command interpreter window (CIW), which appears as a user interface to the complete Cadence IC environment. Through the CIW window the user can control the Cadence IC set of tools and as well perform several activities, including: opening new windows, tools startup, terminating sessions, reviewing the warnings, errors and other messages, changing the environment configuration etc., [8].

When the Cadence IC tool suite is initially invoked, several messages are written in the CIW window, ending up with the sentence: "Done loading NCSU_CDK customizations". This indicates that the Cadence IC environment loads all the configuration files specified by NCSU CDK, when it is launched. NCSU CDK 1.5.1 includes several technology processes that use special technology libraries. These libraries are named as NCSU_TechLib_xxxYY, where xxx is used for the manufacturer's name and YY indicates the minimal transistor length in microns (e.g. the library NCSU_TechLib_ami16 defines $\lambda = 0.8$ µm and minimal transistor length of 1.6 µm). NCSU CDK 1.5.1 includes several libraries for the following MOSIS processes, [6], [7]:

- NCSU_Techlib_ami06 - AMI 0.60u C5N (3M, 2P, high-res)
- NCSU_Techlib_ami16 - AMI 1.6u ABN (2P, NPN)
- NCSU_Techlib_hp06 - HP 0.60u AMOS14TB (3M, sblock, thin-ox cap)
- NCSU_Techlib_tsmc02 - TSMC 0.20u CMOS018 (6M, HV FET, sblock)
- NCSU_Techlib_tsmc02d - TSMC 0.18u CMOS018/ DEEP (6M, HV FET, sblock)
- NCSU_Techlib_tsmc03 - TSMC 0.30u CMOS025 (5M, HV FET)
- NCSU_Techlib_tsmc03d - TSMC 0.24u CMOS025/ DEEP (5M, HV FET)
- NCSU_TechLib_tsmc04_4M2P - TSMC 0.40u CMOS035 (4M, 2P, HV FET)

Cadence library manager displays the technology libraries for the MOSIS CMOS processes, included in the NCSU Cadence design kit. This window (shown in Figure 1) automatically opens, always when the Cadence IC environment is started.

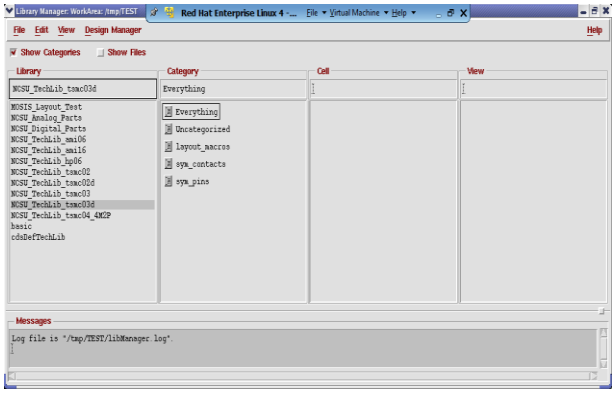

Fig. 1. Technology libraries in the Cadence Library Manager.

Figure 1 illustrates the library manager, showing all the libraries of the Cadence working directory, and their elements in different views (schematic, layout etc.), as well. For example, the NCSU_Analog_Parts library, which is used for analog circuits design includes basic electronic elements, such as: transistors, current sources, voltage sources, resistance, capacitance, etc.

III. Integrated Circuits Design Flow in Cadence IC

Cadende IC is an environment which includes a set of CAD tools, including: Composer Schematic, Composer Symbol, Spectre Simulator and Virtuoso Layout, [4], [8]. Each of these tools individually participates in a particular phase of the integrated circuits design flow in Cadence IC. To start a design in Cadence, one must first create a library where the design cells will be stored. Each library is associated with a technology file and it is the technology file that supplies color maps, layer maps, design rules and extraction parameters required to view, design, simulate and fabricate a circuit.

Cadence organizes its files in libraries, cells and cell views. A library (which actually appears as a directory in UNIX) contains cells (subdirectories), which in turn contain views (schematic, symbol, and layout). Each library contains a catalog of all cells, viewed along with the actual UNIX paths to the data files. Every cell in a library uses the same mask layers, colors, design rules, symbolic devices, and parameter values (which is the information contained in the technology file). A cell is the basic design object that forms an individual building block of a chip or system. Each cell has one or more views, which are files that store specific data for each cell. A cell view is the virtual data file created to store information in Cadence. A cell may have many cell views, signifying different ways to represent the same data of the cell (e.g. layout, schematic, etc).

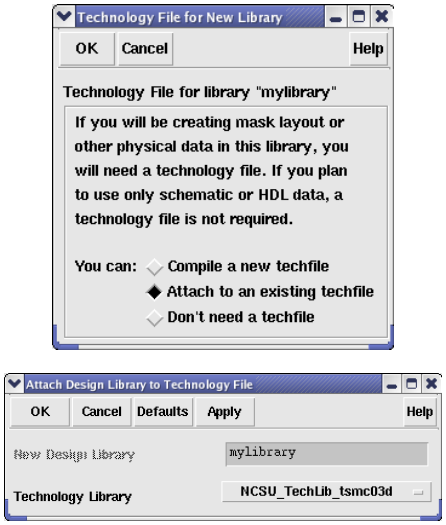

Fig. 2. Specification of technology file, while designing a library in Cadence IC.

The integrated circuit design flow begins by creating a schematic with the Composer Schematic tool. The created schematic is later used for automatic symbol creation with the Composer Symbol tool. The circuit schematic is then simulated with the Spectre simulator which allows different types of analyzes, such as transient, DC, AC etc. Once the circuit operation is verified, a layout is generated. Using the Virtuoso layout editor, the designer describes the detailed geometries and the relative positioning of each mask layer to be used in fabrication. The realized circuit layout has to match the actual circuit schematic.

The created mask layout must conform to a complex set of design rules, in order to ensure a lower probability of fabrication defects. A tool built into the Layout Editor, called Design Rule Checker, is used to detect any design rule violations during and after the mask layout design. The designer must perform DRC and make sure that all layout errors are eventually removed from the mask layout, before the final design is saved. The layout is then extracted and a layout vs. schematic (LVS) comparison is run to ensure the cell layout exactly matches the schematic. All these verification tools are included in the Cadence Diva software. If some problems during the checks appear the designer returns back to the layout design phase. On the other hand, a netlist is created and finally the extracted layout is simulated (with Spectre) in order to observe the effect of parasitics that will be present in the fabricated chip. The post layout simulation results are closer to reality and will show weather the created design would work if is fabricated.

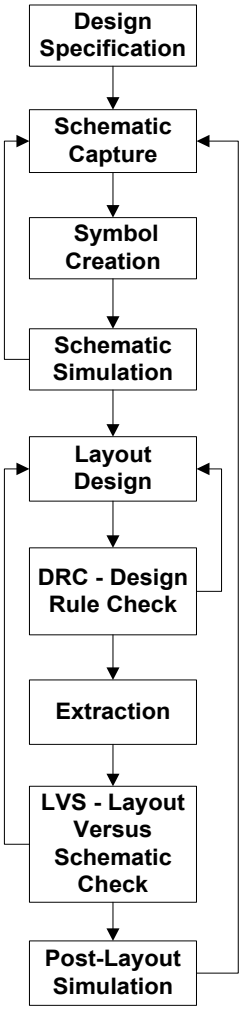

Fig. 3. Integrated circuit design flow in Cadence IC.

Due to the complexity of Cadence tools, approximately a month (eight hours weekly) is required for an engineering student to understand and learn their functionalities and capabilities for integrated circuits design with technologies supported by MOSIS.

IV. Integrated Circuit Design in 0.25 µm Technology for MOSIS

Best way to provide deeper understanding of Cadence IC environment capabilities is to start with integrated circuits design, using this tool. According to the numerous Cadence tutorials from many universities, [4], [8] it is almost a best practice to start designing a simple circuit, and then proceed with the creation of complex ones. This approach enables fast acquiring of the necessary knowledge about the Cadence tool and complete familiarization with the process of integrated circuits design. In order to design a particular circuit, one should first know the components it is consisted of, and then its characteristics, schematic, simulation capabilities etc.

In the continuation of this paper, we are describing all of the phases required for CMOS inverter design in Cadence IC. Using the NCSU_Techlib_tsmc03d library the CMOS inverter is realized within the 0.25 µm TSMC technology. This simple circuit is later used for ring oscillator design in the same MOSIS technology, [7]. Presenting this, we are describing the process of building a hierarchical design from already realized integrated circuits.

The experience with students at FEEIT-Skopje has shown that designing a simple integrated circuit lasts approximately a month (ten hours weekly). Therefore, we believe that in some cases is a good practice to build the basic logic circuits and then use them for developing hierarchical designs in the future. This approach can significantly accelerate the process of designing more complex integrated circuits.

A. CMOS Inverter

CMOS inverter is a simple circuit consisted of pmos and nmos transistors, input and output pins, and power sources (ground and high voltage), [8]. The process of designing an inverter integrated circuit begins with schematic creation, using the Composer Schematic tool. The pmos, nmos, vdd and gnd components are selected from the NCSU_Analog Parts library and then inserted into the scheme. These components are wired, as shown in figure 4, and thus complete schematic is created. The symbol of the circuit is automatically generated from its schematic, using the Composer-Symbol tool. Actually, for drawing the inverter symbol (triangle with a circle), Line and Rectangle tools are used.

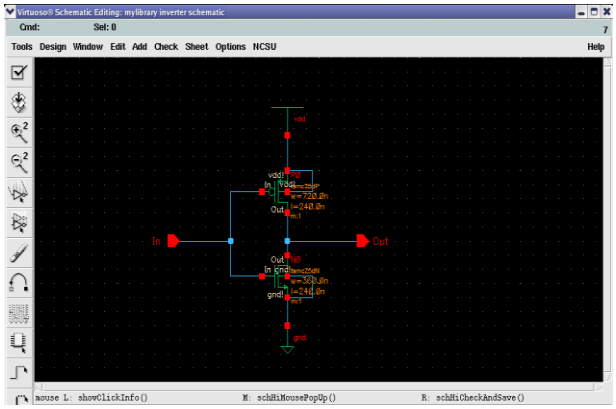

Fig. 4. CMOS Inverter schematic.

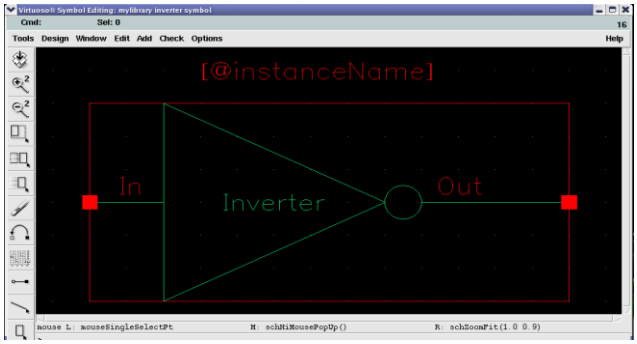

Fig. 5. CMOS Inverter symbol.

In order to simulate the CMOS inverter, a test scenario schematic is created into the Composer Schematic editor. The test scenario includes dc voltage generator and vpulse generator, selected from the NCSU_Analog_Parts library. For the vpulse generator, the voltages V1 and V2 are set to 0V and 2.5V, accordingly, while for the vdc generator the DC voltage is set to 2.5V. In CMOS digital circuits, the output nodes are typically loaded by purely capacitive loads, so a capacitor of 1pF is inserted in the schematic. The created test scenario is simulated using the Spectre simulator, which allows several types of simulations, including transient, DC, AC etc. Figure 7 shows the results of CMOS inverter transient analysis for duration of 6 µs.

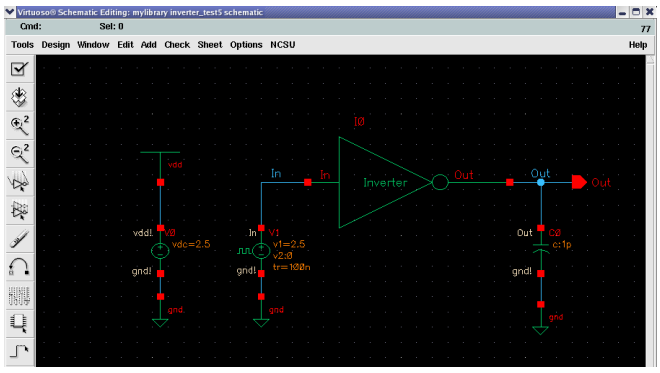

Fig. 6. CMOS Inverter test scenario.

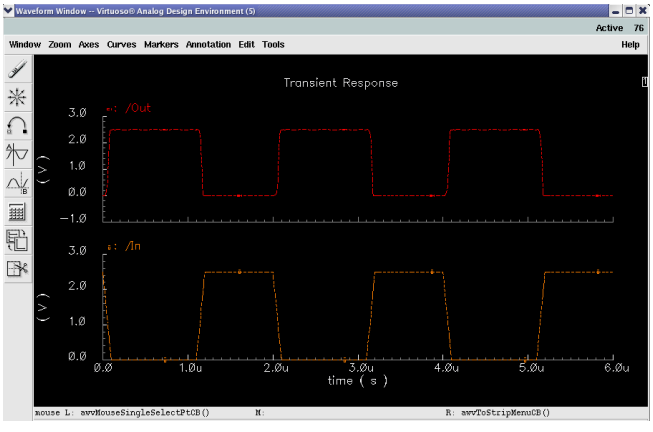

Fig. 7. Results of CMOS Inverter transient analyses.

Next thing to analyze are the characteristics of the output, such as rising and falling edge time, signal delay etc. All these signal properties are computed, using the embedded functions (RiseTime and Delay) of the Waveform calculator.

Additionally, DC and AC simulations are also done, but we are not discussing them here. However, the overall simulations results have shown that the CMOS inverter achieves the expected behavior.

Given that the functionality of the CMOS inverter is verified, the circuit layout design can start. The CMOS inverter layout will be designed in TSMC 0.25µm technology, whose specification is provided on the MOSIS website, [7]. The given web documentation states that this CMOS technology process has 5 metal layers and 1 poly layer, and is used for 2.5 volt applications.

In order to realize inverter layout with the TSMC 0.25µm MOSIS technology process, the designer must follow the SCN5M_DEEP rules, specified in [5]. These rules define the value of Lambda as 0.12µm, and thus limit the minimal length and width of a transistor to 0:24 µm and 0.36µm, respectively. In addition to this, the same rules identify the gridRes parameter (resolution of the grid layout) as Lambda/2, which equals to 0.06 mµ.

Considering the SCN5M_DEEP rules, the NMOS transistor length and width are specified as 0.24µm and 0.36µm, while the PMOS transistor length and width are set to 0.24µm and 0.72µm. The actual process of inverter layout design consists of: insertion of pmos and nmos transistors (pmos/nmos layout cell view, selected from the NCSU_Techlib_tsmc03d library), connecting the transistor layers (poly and metal1), and addition of I/O pins, voltage sources and contacts (metal1-poly) in the layout editor, as well.

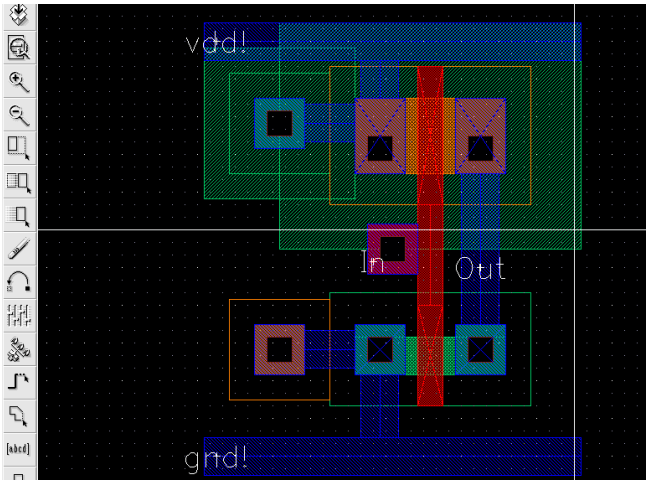

Fig. 8. CMOS Inverter layout.

Once the layout of the circuit is designed, the geometric rules check or DRC analysis can start. If this analysis is successful, the parasitic capacitances from the circuit can be extracted. The extracted circuit creates a novel cell view of the CMOS inverter, shown in Figure 9.

After the LVS check has finished successfully, it is considered that the layout and the schematic of the inverter circuit mach. However, it is still necessary to make additional circuit simulations in order to observe the parasitic effects that will remain in the manufactured chip. These analyses check whether the circuit will work properly after fabrication.

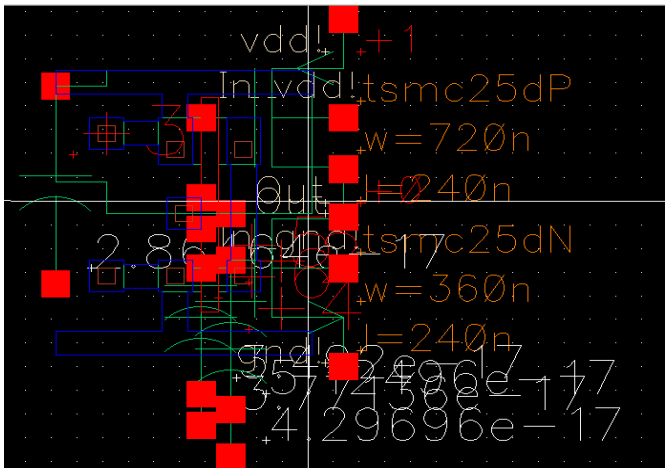

Fig. 9. CMOS Inverter extraction.

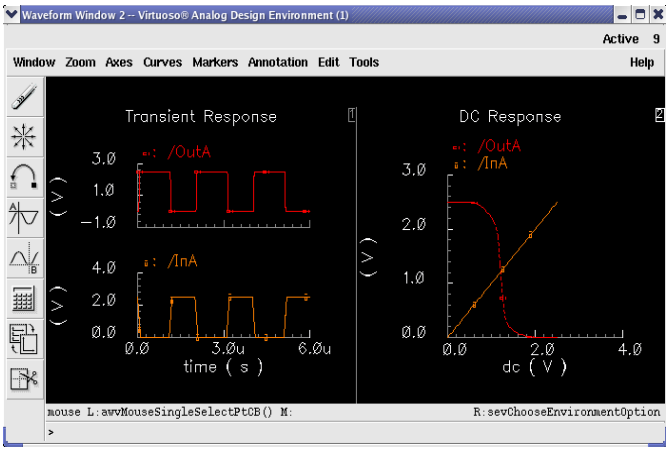

Fig. 10. Results of CMOS Inverter transient and DC post-layout simulation.

In order to perform post-layout simulations, some minor changes (removing the vdd and gnd contacts) in the CMOS inverter schematic and symbol are done, and several modifications in the test scenario schematic are also made. The results of these post-layout simulations (figure 10) verify that the extracted inverter circuit operates in similar way as the CMOS inverter schematic.

B. Ring Oscillator

The cell views of the CMOS inverter can be directly placed into other cell views of hierarchical designs. In order to show this, in the continuation we are presenting the process of designing a ring oscillator circuit, [9]. The ring oscillator is consisted of five CMOS inverters wired in a circular way, with coupled inputs and outputs. The complete ring oscillator schematic, realized in Cadence IC, is shown in figure 11. As can be seen on this picture, the ring oscillator output is connected to a noConn component, which is used to prevent any unconnected warnings. Additionally, some other components are also added in the Schematic editor (vdd, gnd and vdc - 2.5V voltage sources, selected from the NCSU_Analog_Parts library), and later used for simulating the ring oscillator circuit.

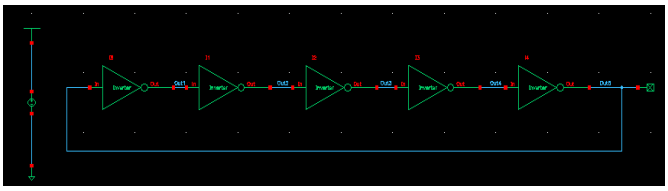

Fig. 11. Ring oscillator test scenario.

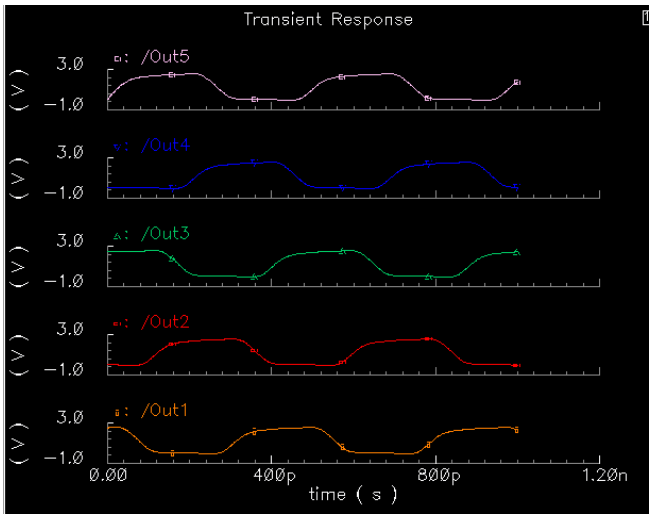

Fig. 12. Results of Ring oscillator transient analyses.

The previous figure presents the results of ring oscillator transient simulation for duration of 1ns. As shown in this figure, it can be considered that the given circuit achieves the expected behavior. This is additionally proven by computing the frequency and power consumption of the ring oscillator, using the special functions of the waveform calculator. However, the results of the last analyzes are not going to be discussed here.

The ring oscillator layout is designed in a similar way as its schematic. This process begins by adding the layout cell views of the five CMOS inverters in the Virtuoso layout editor. The inverters are then connected with metal1 layer, while the input of the first inverter and the output of the last inverter are connected with a metal2 path. The last connection is created, using M2_M1 contacts. Additionally, a RingOut PIN is added to serve as an output of the ring oscillator circuit.

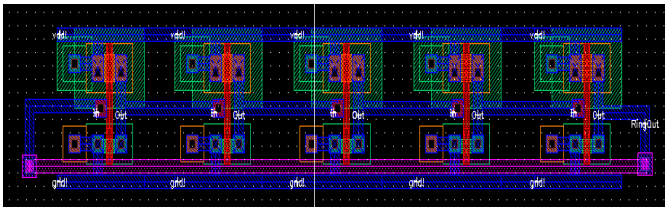

Fig. 13. Ring oscillator layout.

After the ring oscillator layout is created, DRC check, extraction, LVS check and post-layout simulations are performed. All of these design flow stages are successfully completed, with LVS analysis ending up with the message: "The net lists match". After that, post-layout simulations are performed, and the ring oscillator functionality is verified.

V. Conclusion

According to the complete flow of integrated circuits design in Cadence IC, shown by example, it can be said that this powerful environment is suitable for use not only by professionals but also by beginners. The structural approach of its tools organization and the availability of many written tutorials enable engineering students to easily adjust to it.

The experience in using Cadence IC tool suite within the microelectronics technologies and structures course, at the Electrical Engineering and Information Technologies Faculty in Skopje has shown that the process of studying this course would start by reading a professional literature, [10] - [13], which will help students learn the basics of microelectronic (for an average period of one month - 8 hours weekly). This would be followed by introducing students to the Cadence IC tool suite and NCSU design kit, as an integrated environment for integrated circuits design, supported by MOSIS (approximately one month - 8 hours weekly). Deeper knowledge would be acquired when the students will manage to design an integrated circuit with the Cadence IC tool. This can last for a month or two (around 40 hours per month) depending on the circuit complexity. However, the basic level of microelectronics, analog circuits design and other related electronics courses can be obtained through lab practice exercises and/or by completing a project work in Cadence IC.

The application of Cadence IC and NCSU CDK for educational purposes at FEEIT Skopje, expands the opportunities of the academic environment in our country, and increases the capabilities for collaboration with other companies, research and educational institutions from academia and industry, abroad.

References

[1] R. Brayton and J. Cong, "Electronic Design Automation: Past, Present, and Future", UC Berkeley, NSF Workshop Report, Virginia, July 2009.

[2] David W. Parent, E J. Basham, Y Dessouky, S Gleixner, G Young, and E Allen, "Improvements to a Microelectronic Design and Fabrication Course", *IEEE Transactions on Education*, 2005, pp. 497-502.

[3] Moreno F., Boemo E.I., Meneses J. and Lopez Barrio C., "A teaching experience on VLSI design at the technical university of Madrid", *In proc. of the third Eurochip Workshop*, September 1992, pp.60-66.

[4] Antonio J. Lopez Martin, "Cadence design environment", New Mexico State University, Tutorial paper, October 2002.

[5] MOSIS, "Design Rules: MOSIS Scalable CMOS (SCMOS)", Revision 8.00, May 2009.

[6] NCSU Electronic Design Automation (EDA) Wiki [Online]. Available: http://www.eda.ncsu.edu/wiki/NCSU_EDA_Wiki

[7] The MOSIS Service [Online]. Available: www.mosis.com

[8] Cadence Tutorial [Online]. Available: http://edp.wikispaces.asu.edu/Cadence+Tutorial

[9] Marin Razvan-Cristian and Marin Mihai-Eugen, "Simulation of a ring oscillator with CMOS Inverters", University Politehnica Bucharest, Tutorial paper, May 2011.

[10] Томислав Цеков, *Универзитетски учебник: Микроелектроника*. Скопје, Македонија, 1986.

[11] Richard C. Jaeger and Travis N. Blalock, *Microelectronic Circuit Design*. USA: Mc Graw Hill, 2011.

[12] Richard C. Jaeger, *Introduction to Microelectronic Fabrication, Modular Series on Solid State Devices*. USA: Mc Graw Hill, 2002.

[13] Behzad Razavi, *Design of CMOS Analog Integrated Circuits*. USA: Mc Graw Hill, 2001.