Innovative Teaching of IC Design and Manufacture Using the Superchip Platform

Peter R. Wilson, *Senior Member, IEEE*, Reuben Wilcock, *Member, IEEE*, Iain McNally and Matthew Swabey, *Member, IEEE* [1]

***Abstract*—This paper describes how an intelligent chip architecture has allowed a large cohort of undergraduate students to be given effective practical insight into IC design, by designing and manufacturing their own ICs. To achieve this, an efficient chip architecture, the "Superchip", was developed, which allows multiple student designs to be fabricated on a single IC, and encapsulated in a standard package without excessive cost in terms of time or resources. This paper demonstrates how the practical process has been tightly coupled with theoretical aspects of the degree course and how transferable skills are incorporated into the design exercise. Furthermore, the students are introduced at an early stage to the key concepts of team working, exposure to real deadlines and collaborative report writing. This paper provides details of the teaching rationale, design exercise overview, design process, chip architecture and test regime.**

I. INTRODUCTION

A. Background

Recent advances in CMOS IC process technology have forced Electronics departments worldwide to adapt their educational programs to equip students with the Integrated Circuit (IC) design skills and knowledge required by research and industry. In addition, the time taken from specification to market, often referred to as the design cycle time, is being driven ever shorter, emphasizing the necessity of teaching up to date and advanced design skills in a structured environment. The skills required to support this level of design are rapidly evolving, as are the software and hardware tools that support the design process. In addition to

[1]Manuscript received 3rd July 2008, revised 6th December 2008

Peter R. Wilson, Reuben Wilcock, Iain McNally and Matthew Swabey (prw,rw3,bim,mas}@ecs.soton.ac.uk are with the School of Electronics and Computer Science, University of Southampton, Highfield, Southampton, UK, SO17 1BJ.

teaching the required technical content, it is also vitally important that a team approach is employed, enabling students to gain experience in time management, group collaborations and interpersonal skills. The Electronics Engineering (EE) undergraduate program at the University of Southampton, England, has run successfully for many years and provides a good grounding in hardware design. One key element is advanced microelectronics in particular the design of integrated circuits [1]. The view has been taken that there is no substitute to learning through real experience, gained from a practical design context with real examples; this work summarizes the approach taken for IC design. A key element of the approach has been the development of a new chip infrastructure, called the "Superchip", which allows up to 16 separate circuit designs to be fabricated on a single cost effective IC.

The rest of this section describes previous work and the learning strategy and outcomes for this exercise. Section II details the Superchip platform itself. Section III outlines the student design process and the steps taken to maintain industrial relevance whilst reducing complexity, to ensure that the scale of the exercise is clearly within the scope of the overall taught program. Finally, results are presented of the project's implementation and delivery, from a technical and student perspective.

B. Previous Work in This Area

In the last twenty years, significant work has been presented in the development of advanced curricula and the teaching of Very Large Scale Integration (VLSI) systems. O'Keefe *et al* [2] was an early example of bringing practical IC design from a "systems" perspective into the classroom, and a good case of delivering a realistic IC design flow to a student environment. Although this work was significant at the time, it did not look at the impact and skills required for modern EDA (Electronic Design Automation) tools –primarily due to their limited availability at the time. While it is common practice to teach the theory of VLSI architectures, perhaps the most effective mechanism to improve student engagement and understanding of such topics has been the development of design projects and laboratory courses with a strong practical content, such as that described by Brown *et al* [3]. In particular, it is desirable to provide students with an

opportunity to have their own circuit designs fabricated as part of teaching the modern design flow. However, a very common issue for educators in the VLSI domain is that a single semester course is simply not long enough to design and fabricate integrated circuits in time for testing in that semester. One route is to use an FPGA (Field Programmable Gate Array) based approach [4], and this is now pervasive in most, if not all, EE curricula at undergraduate level. Another popular approach is to develop a set of individual modules that cover the whole range of VLSI activities including IC layout, schematic design, simulation, synthesis and fabrication, for example in the Canadian Model [5] by Serra *et al*. This approach has been adopted by numerous universities worldwide in some form or other [6], [7] with this university being no exception. Harris in [6] has also undertaken a number of exercises using microprocessors in particular, as a design basis for this general type of teaching.

At the University of Southampton (UOS) the wider context of the student program concentrates on fundamentals in Year 1, with generally formulaic laboratory exercises throughout. In the third year, the focus is on a year-long individual project, and so the Year 2 laboratories have been designed with two key aspects in mind. The first is to continue the fundamental laboratory work that directly maps onto individual theoretical taught modules of the program; the second is to deliver a series of design exercises which will encourage team working, creative thinking and the development of a wider range of skills. The Integrated Circuit (IC) design exercise was designed within a two semester design program to allow more flexibility in the timing of individual design elements. This structure allows the design and post-fabrication test to take place in Semesters one and two respectively. The previous incarnation of the course run at Southampton employed a pre-manufactured gate array that was post-processed by an in-house CMOS facility to add high-level connections. However, this approach is rarely used in industry today and one of the key reasons to alter the course was the need to provide students with more realistic projects, from both an application and technology perspective as discussed in [8] and [9]. In searching for an alternative it became apparent that there was just enough time between the two semesters for a chip to be fabricated on a mainstream CMOS

process. However, the financial cost of fabricating numerous designs for different student teams was prohibitive, and getting to a stage where multiple full custom designs were ready for manufacture would require a lot of time for the supervisors involved. The solution to this problem was the development of a multi-design infrastructure, called the Superchip, which allows up to 16 separate and individually-selectable designs to be placed on a single chip, thus greatly reducing both the financial cost of fabrication and also the time spent readying designs for manufacture. The main advantages of this approach are:

- The experience gained by students of having their designs made on a mainstream CMOS process
- Low financial cost to the university, since one IC supports 16 group designs
- A saving in supervisor time, as placing all the designs onto the Superchip can be highly automated.
- Chips are back by the second semester ready for testing, due to the short manufacturing turnaround.

More details of the Superchip infrastructure are provided later in the paper.

C. Learning Strategy

A key part of the strategy for learning in this design exercise has been to provide a solid experiential learning platform based on the Kolb learning cycle [10], and in particular the use of small groups [11]. The strength of this approach is its tutorial style, with students able to progress at their own pace with a structured work plan to facilitate learning. A tight feedback loop that enables students to experiment, but also gain insight with the help of a quick response from staff, is also beneficial. Students are allowed to organize their groups into whatever structure suits them best, which was an interesting step to take, as the intuitive assumption is often made that students must be given a tight framework within which to work. However, the experience in this design exercise has been that the students welcome the responsibility, and enjoy the fact that there is a real deadline set by the chip manufacture, not just an “artificial” deadline, typical for most coursework at the undergraduate level. Providing literature prior to the session [12] enables students to take a less linear approach to the design process, and enable more iteration and creativity to take

place. In addition, briefing seminars are run prior to the laboratory sessions, and informal support and question-and-answer (QA) sessions are run throughout the period of the design exercise. While the design exercise is aligned with taught elements of the course, it is not proscriptive, and creativity and fresh thinking are encouraged. This is an intentional move to help develop students who can design solutions to real and unfamiliar problems – an ability not encouraged by rote-type learning. In fact, some aspects of the technical background are deliberately given as taught elements after the design exercise, to provide further insight later in the year. Biggs [13] was found to provide a useful framework to assist in the strategy of preparation, and different methods of delivery and assessment [14] were applied to engage large classes more directly, such as seminars and QA sessions rather than standard lecture formats. A key aspect of this approach is to use a student-oriented learning technique, with self-assessment and reflective review being a key part of the final deliverable report [15]. Experience has shown that a combination of collaborative group work and peer review proves effective and useful in this context, and this has been incorporated into the design exercise as a result. While no system is perfect, the feedback from students and their resulting enthusiasm and commitment was a strong indicator of the success of this strategy.

D. Learning Outcomes, Key Skills and Assessment

In order to ensure that the individual student's experience is satisfactory, learning outcomes have been designed in the context of an integrated overall process of teaching, learning and assessment. This is, of course, essential to provide the student with a high quality of learning in a rapidly changing field. In this course a view of learning was taken that considers the academic aspects of the work, and links this to the more industrially oriented "real world" aspects. Fourteen specific learning outcomes were devised for this design module – ranging from purely technical to more general transferable skills, with an assessment of these to ensure that when the task was framed it addressed these learning outcomes. Within this range of fourteen learning outcomes, six specific learning outcomes were identified as being relevant to this particular design exercise; they are discussed in more detail later in the paper. The integration of key skills

for industry is critical for engineering students, as discussed in some detail by Woods *et al* [16], and in this particular field the need is even more acute. In order to ensure that the learning outcomes were appropriate in relation to the proposed course structure, a matrix-based approach was employed as described by Felder and Brent [17] to analyze the exercise structure in relation to these outcomes. In this particular design exercise, key relevant skills were identified and tied into specific learning outcomes in a coherent manner. The assessment has also been considered in the context of the variety of skills, platforms and learning outcomes required [17]. The approach taken to ensure these requirements were met, with a relatively inexperienced group of undergraduates, is to provide design freedom, within a tightly constrained design tool framework. In addition, the scope of the design kit is artificially limited, to ensure that the students are not intimidated by the scale of the possibilities. The learning outcomes of this exercise, driven by the national strategy as defined in [18], are summarized as:

1. To carry out a complete ASIC design flow
2. To implement a specification
3. To carry out self-study of the design tools
4. To manage a workload effectively as a team
5. To experience industrial conditions and practice
6. To test the fabricated design and compare it to simulations

As part of the assessment of this exercise, a breakdown of marks is given for individual or groups of learning outcomes, and this breakdown is also provided to the students. The next section of this paper will discuss the implementation of the “Superchip” architecture itself.

II. THE SUPERCHIP

A. *Introduction*

A crucial aspect of the program is the ability to support a large number of individual IC designs effectively, without excessive cost in terms of time or resources. The Superchip has been developed to achieve this goal and allows up to 16 design sites to be placed on a single IC and encapsulated in a standard package. This has been achieved through innovative design techniques, some of which are discussed in the following sections of this paper. The cohort is divided into teams of around six students, enabling them to develop separate designs as a group.

B. *Details of the Superchip Layout*

The Superchip [19], was designed to be fabricated on the Austria Microsystems C35B4 (0.35µm) CMOS process, which supports four metal layers and is available through one of the multi project wafer organization (MPW) world-wide [20],[21],[22]. It is important to stress, also, that the infrastructure has been designed to use only two of the available metal layers, and therefore can be implemented on a wide variety of similar processes. In fact the authors believe that this technique can be implemented very straightforwardly on any standard 0.35µm CMOS process with a minimum of two metal layers. This is a very cost-effective way of fabricating a small number of ICs, since manufacturing costs are shared with a large number of partners. The Superchip takes the multi-project concept to the next level, by sharing a single IC between the 16 different designs. The novel framework on the Superchip has been implemented to provide and service a 4 x 4 array of separate design sites with shared I/O, and separate supplies. Sharing I/O pins is essential to avoid the cost of fabricating an IC with many hundreds of pads, which would be financially unfeasible. The chip infrastructure is shown in Fig. 1 and consists of the following features:

- 16 separate design sites each 260µm by 300µm in size
- 24 digital input pins, shared between all design sites
- 24 digital outputs pins, shared between all design sites

- 16 separate Vdd pins, one dedicated to each design site
- 1 global Gnd pin for all design sites and infrastructure circuitry
- 1 global Vdd pin to power the site buffers and I/O pad ring
- 68 pin JLCC package (2 pins unused)

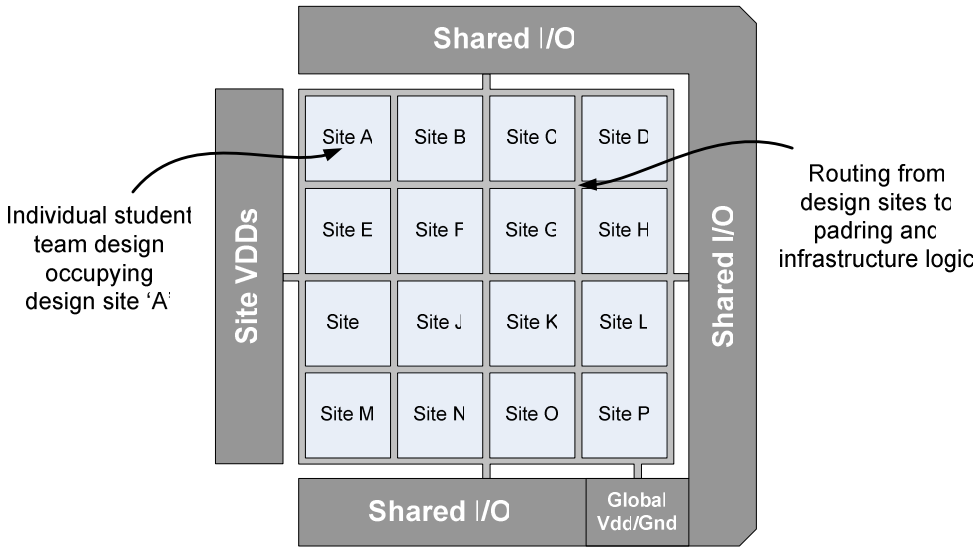

Fig. 1: Southampton Superchip IC layout

When a design site is powered, the shared inputs and outputs are connected to this design site and disconnected from all others. Within each design site, a pseudo pad ring has been created which the students see as the interface to the Superchip, abstracting the complexities of the overall layout. An example of a fully populated Superchip is shown in Fig. 2. The JLCC package choice allows practical general use in electronics laboratories.

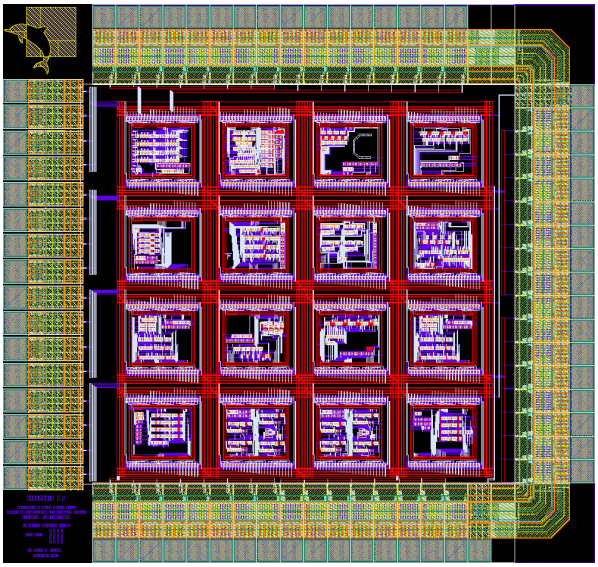

Fig. 2: Layout view of an example of a fully populated Superchip

III. The Design Process

A. *Process Overview*

In this section the key stages in the design process are introduced, with particular reference to the skills developed in the student's first year. The relationship of the project to the theoretical program of study and the rest of the student's degree course is also considered. During the first year, the students are exposed to the basics of gate level logic and processor design, fundamentals of analog electronics, and the use of schematic, PCB (printed circuit board) layout and basic simulation software. The design exercise is scheduled as early as possible in the first semester of the second-year undergraduate (UG) program in order to ensure that there is enough time for the ICs to be made ready for testing during the second semester. Since it must be assumed that the students only have their first-year knowledge at this stage, the type of designs undertaken must be relatively simple. To date, typical designs have included a sequence recognition block, counter design, ALU (arithmetic logic unit) design and oscillator design.

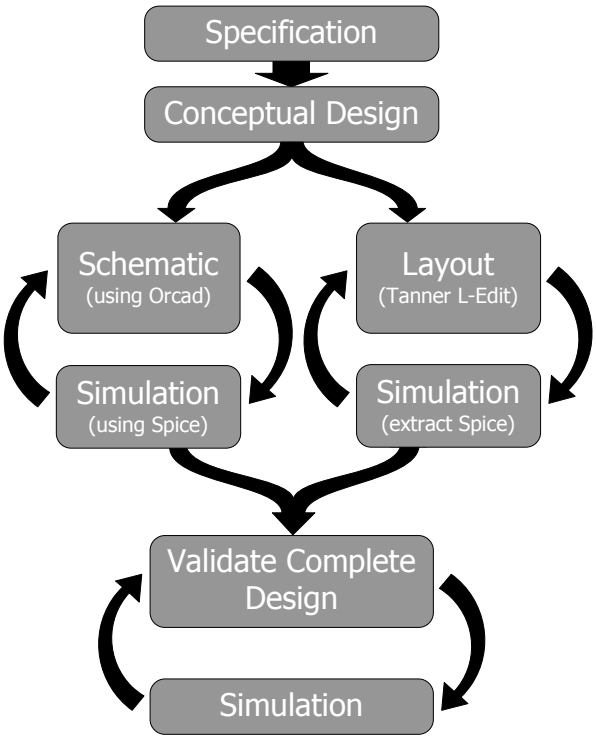

Fig. 3: Design Process Flow Diagram

The design exercise is divided into two main sections. In the first semester, the design and implementation stages take place. In Semester two, the student teams reconvene to develop test vectors to enable automatic testing of their designs. The main activities undertaken in the first semester are shown in Fig. 3 and the different steps in this design process are now considered in more detail.

B. Design Specification and Student Design Kit

The design specification is published for all the teams and an introductory briefing is given to explain the concepts, deadlines, tools and methods in detail. This is also an opportunity for the students to meet up with their other team members. As discussed previously, a typical design specification may be an 8-bit ALU or a circuit with a similar level of functionality. In recent years, a ring oscillator has also been added as a specific item that can be used to test the process operation in a more "analog" function and enable the students to carry out probing of high-speed digital signals. The exercise is designed to give a range of possible goals in

order to enable the maximum number of students to achieve some level of functionality whilst allowing the outstanding students to demonstrate their abilities. This addresses learning outcome 2.

Cell Name	Cell Decription
inv10	Inverter
nand2	Two Input Nand gate
nand3	Three Input nand Gate
nand4	Four Input Nand Gate
nor2	Two Input Nor gate
nor3	Three Input Nor Gate
nor4	Four Input Nor Gate
xor2	Two Input XOR gate
xnor2	Two Input XNOR gate
dff	D-Type Flip Flop with reset
Tie1	Tie to VDD
Tie0	Tie to GND
MUX21	Two Input Multiplexer

Table I: Design Kit Cells

The students are provided with a complete design kit, including a library of schematic symbols, layout abstract views, simulation files, design rule files and design extraction files. The design kit provided is not to be confused with the standard AMS design kit since it has a much-reduced number of digital gates for the students to work with (shown in Table I).

A typical gate abstract layout is shown in Fig. 4, where the inverter is simplified to show only the power rails (VDD and VSS) in Metal 1 as horizontal tracks, and the vertical signal tracks (A, Y) in Metal 2. In comparison with the full layout cell, the complexity of the abstract view is greatly reduced, which has the interesting advantage of enabling student versions of software to be easily used. For more advanced students, the full layout cell views could be used instead, to give greater insight into the internal cell structure.

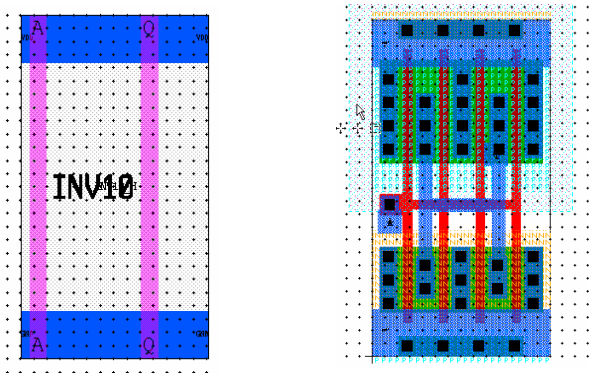

Fig. 4: Typical Cell Abstract and Full Layout -Inverter

At this stage in their course program, the students have only basic knowledge of electronic design tools having covered gate level schematic design and PCB layout in their previous year, with more intensive IC design courses only coming in their third year of study. This limit to their knowledge means that it is essential to target the level of abstraction of the design kit, so that the students are not swamped with too much information. Given the limited knowledge of the students, the same PC-based software for schematic design and simulation that they used during their first-year studies is used for this course. While this software is not necessarily optimal from an IC design perspective, it works well as an introduction. The software tools are introduced in a structured manner through a series of lab sessions after the introductory briefing. These lab sessions allow the basics of the software to be learnt in a closely supervised environment, whilst student versions of the software are provided for self study in students' own time, satisfying learning outcome 3. These initial labs are followed up by a more design-oriented lab, allowing the students to transition from learning basic skills to those with a greater design perspective. One option that was considered was to introduce the students to Cadence Design Systems software, however, but this was considered to be a "step change" in both complexity and learning. This problem has been discussed by Bouldin in [23]. The authors feel that teaching design software with such greater complexity is more appropriate for later in the EE program, and in fact a dedicated option in UOS's System on Chip Masters level course does exactly this. In UOS's broad-based undergraduate curriculum, however, it was not

considered appropriate for second-year undergraduates undertaking a wide-ranging set of modules to learn a completely new set of software *and* layout a chip for fabrication in a matter of two weeks.

C. Conceptual and Schematic Design Stage

The initial design phase is a typical "paper" conceptual design, where the team will discuss both the functionality of the design and also the implications for its fabrication. The teams must weigh up the tradeoffs in using different approaches for the design of each block. For example, when designing an ALU, one option is to design 8-bit functions in turn and link these together, whereas an alternative approach would be to create a one bit "slice" and then simply replicate this eight times. During the design stage there is an emphasis on best design practice, design for testability and fault tolerance. Since integrated circuit designs must, in principle, be right first time, CAD (Computer Aided Design) tools are of course used extensively in this exercise for design entry, verification and simulation. Exposure to such CAD tools and techniques is considered by UOS to be a vital part of this exercise. The student teams create a schematic of their design using the schematic capture software, from which they can simulate their design in SPICE, or extract a VHDL model for digital simulation. An analog approach is used in the initial stages of the design to familiarize the students with the concepts of power consumption, rise and fall time, overshoot, and the device characteristics that impact on fan-out.

D. Layout

The IC Layout is carried out manually using the L-Edit software from Tanner EDA. While it is arguable that automatic place and route (APR) could be used, it has been useful to allow the students to investigate the principles of channel routing, and experiment with different layout strategies, which would not be possible with synthesis and automatic layout. In addition, the synthesis and APR is so mechanized that it is entirely conceivable that 16 identical designs could result. By making mistakes, the students learn a great deal about using the Design Rule Checker, and gain understanding of the possible errors, how to correct them and also how to be efficient in their design. The authors have noticed a marked interest in IC design as a result of

taking this approach. This coverage of the main aspects in a design flow addresses learning outcome 1 directly.

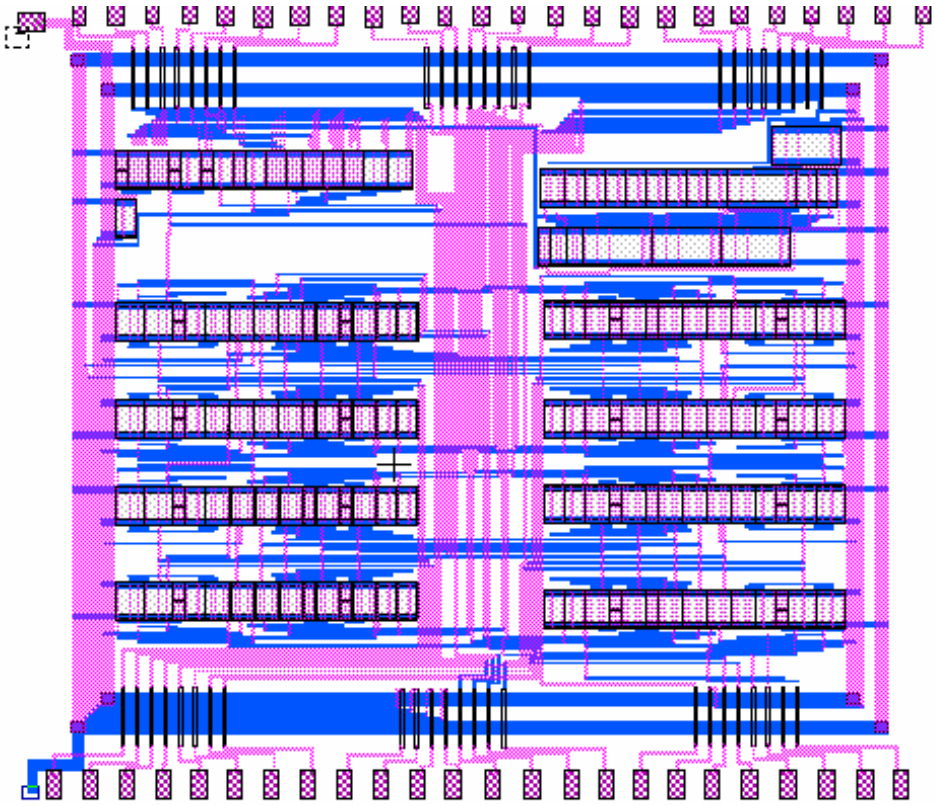

Fig. 5: Typical Student Design – 8-bit ALU

For each abstract cell, an equivalent SPICE and VHDL model exists for analog and digital simulations respectively. The same SPICE test benches are used to validate the extracted SPICE model from the layout to ensure the designs are consistent. LVS (Layout Versus Schematic) is also possible within L-Edit, and the students are also introduced to the use of Design Rule Checking (DRC) at this stage. The routing of the design sites is restricted to the Metal 1 and Metal 2 layers, and is constrained to prevent the students from routing over cells. This approach encourages students to consider the physical design and the implications of cell placement more carefully. A standard “channel routing” strategy is encouraged to connect the cells manually. This approach is simple to manage, and a number of students can create individual blocks of

layout that can be connected together later in the process. The student designs must pass DRC prior to completion of the lab, and the functionality of the design (schematic and layout) has to be fully demonstrated to the lab supervisor prior to "sign-off". A typical example design is an 8-bit ALU, which has a completed layout as shown in Fig. 5 where a cell-based approach has been used with rows of abstract cells and routing channels between these.

E. Validation of the Design

It is important to ensure that the design is correct in the schematic, but even more crucially, in the layout, since the designs are going to be fabricated on a real process. A small ring oscillator is always placed in the design as a test circuit, which allows simple validation of basic operation and power supply connectivity. At this stage transistor-level simulation is used to illustrate the analog concepts of finite rise and fall times and over-shoot. A typical waveform is shown in Fig. 6:

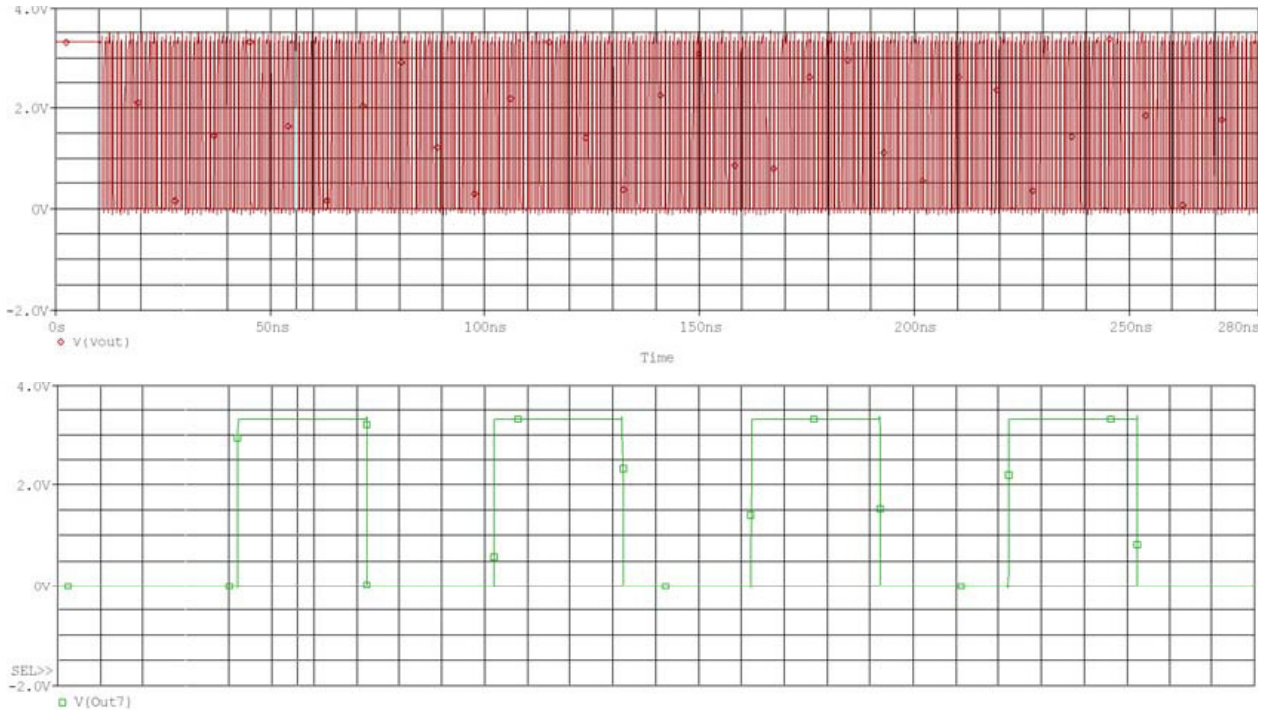

Fig. 6: Typical Analog Simulation (Oscillator and Divider outputs)

F. Student Deliverables and Assessment

The deliverable from the "Design Phase" is the "Design Package". This package mimics the output produced from an industrial ASIC design project and consists of:

1. Design schematics
2. Design simulations (SPICE)
3. Design simulation test circuits (SPICE)
4. Layouts (L-Edit)
5. Layout simulations (SPICE)
6. Layout simulation test circuits (SPICE)
7. The "Design Phase" report

There are two deadlines for the deliverable. The first is for the silicon layout and is one week after the second lab session. The second is for the complete "Design Package", and is two weeks after the second lab session. An integral part of the design package is a report that not only details all design decisions, but also a comprehensive testing regime and data validation set. The first section of the lab is assessed on the contents of the "Design Package" with the majority of the emphasis on the content of the report. This report must contain three sections:

Section 1: The design – a description of the design approach

Section 2: The schematic designs – simulation results confirming the design's functionality.

Section 3: The layout –simulations demonstrating the design's functionality.

The report must also contain a DRC report showing no errors. The active involvement of all team members is also assessed, not only by staff, but also by the team members themselves, to recognize outstanding contributions and also perhaps identify weaker members. The aspect of team working and project

management is vital to this design task, especially since a commercial fabrication "tape out" deadline has to be met. This deadline forces teams to collaborate effectively, and in what is the first occasion in this program that they will have experienced working in a team of more than two. It is not mandated *how* the teams are structured, or managed, simply that they *are* managed. This management is an integral part of the task, and is to be documented in the report. This vital aspect of the design exercise addresses learning outcomes 4 and 5.

After the design package has been completed by each team (A to P), the individual designs are incorporated onto the single Superchip layout and final checks undertaken. The complete layout package is delivered to AMS (Austria Micro Systems) for fabrication, which takes around three months; when the chips return, they can be tested during Semester two.

IV. IC Validation and Verification

A. *Superchip USB Test Board*

In Semester two, after the ICs have returned from fabrication, the key task is to ensure the basic functionality is correct, and to carry out performance measurements (timing, power consumption, oscillator frequency) to verify that the design criteria have been satisfied. A standard chip tester board was developed, with a USB interface and chip socket, so semi automatic testing can be easily and quickly carried out on the individual designs. The compact board is shown in Fig. 7, and uses a USB interface chip and a microcontroller to manage an interface to a PC. The students have full access to every pin via probe points directly next to the package, and test vectors can be uploaded and analyzed automatically. This testing is a fascinating and essential final stage of the design process, and in fact confirming the functionality of the integrated circuits (ICs) is a validation of all the hard work put in by the teams in Semester one. The significance of the student's achievement in such a short time is brought home in an extremely positive way. This stage of the task satisfies the final learning outcome, 6. Similar low-cost testers exist for separate integrated circuits designs, with [20] being a good example of this. With the multiple core architecture,

however, the authors have extended this to allow one tester to validate all the designs using the site "selector" switch. This tester applies power to each individual design core uniquely, at the same time as enabling the shared I/O to that same core.

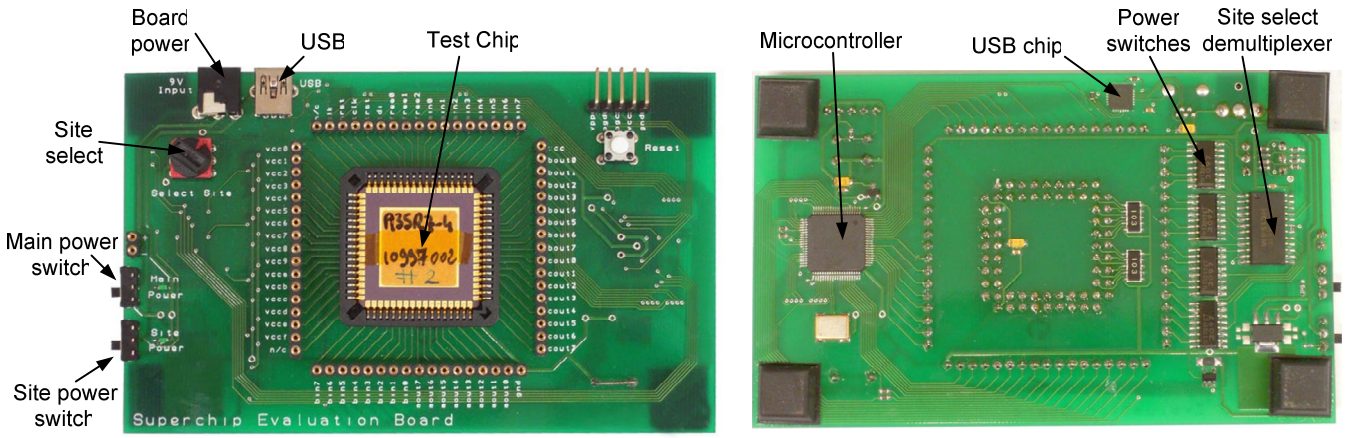

Fig. 7: Superchip Test Board

B. Test Vector Validation

The USB tester board allows test vectors to be automatically uploaded from the PC to the Superchip, and the responses returned. In order to develop the test vectors efficiently, the same schematic used to design the layout can also be used to extract a VHDL model of the design. Using this digital model, the test software used to connect to the USB tester board can also export to a VHDL test bench. This feature means that the test vectors can be tested using the model of the design, in this case in the Modelsim simulator, allowing their validation prior to testing on the Superchip itself. This approach is in contrast to the analog simulation carried out in Semester one. The same circuit schematics are used as in the original design, so even though a different model is extracted, the source is identical. This way, potential issues can be identified in the design at a functional level. At this stage the topics of test benches and assertions, fundamental to digital circuit test, are introduced and this is an excellent practical vehicle to demonstrate these concepts. A typical output waveform of such a test is shown in Fig. 8.

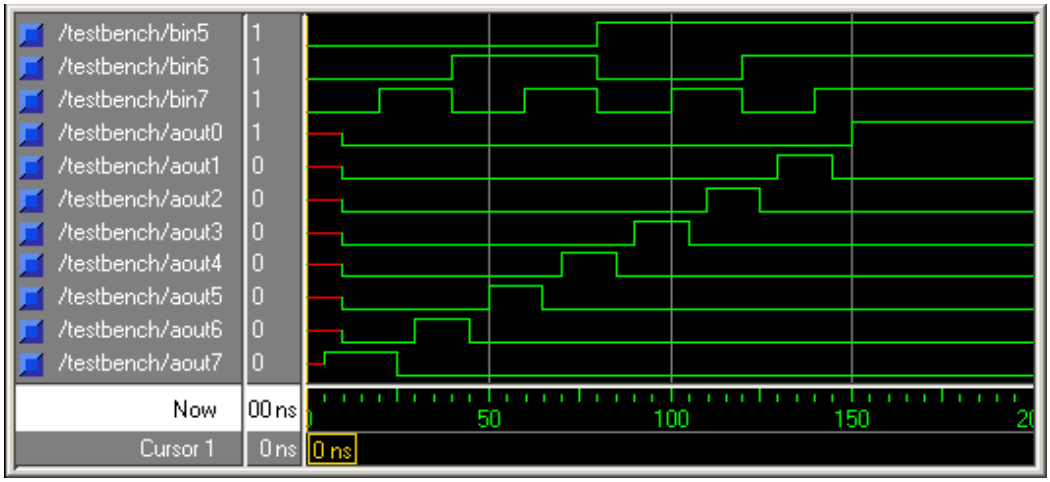

Fig. 8: Digital Simulation test results

This stage is vital and ensures that the test vectors themselves are correct. The authors take the view that if a group's design is flawed, then students can redeem themselves in the testing stage by identifying the location of the flaws in their original design through intelligent selection and application of test vectors. The test vectors are created in a simple XML style format, making editing and validation simple, with an example shown below:

```
# Test Vector File

<PinDef>

clk,nrst,tclk,trst,tdi,freein0,freein1,freein2,ain0,ain1,ain2,ain3,ain4,ain5,ain6,ain7,aout0,
aout1,aout2,aout3,aout4,aout5,aout6,aout7,bout0,bout1,bout2,bout3,bout4,bout5,bout6,bout7

</PinDef>

<TestVector>

#  cin      ain       aout     bout

10000000 00000000 10000000 00000000

01000000 00000000 01000000 00000000

00100000 00000000 00100000 00000000

</TestVector>
```

The basic format is divided into two sections: **PinDef**, which is the pin definitions, and **TestVector**, which gives all the individual test stages. The test vectors can be static 0, static 1 or a clock pulse denoted by 'C'. Lines beginning "#" are comments and ignored by the compiler.

V. Assessment

Since the introduction of the Superchip exercise, student marks in this area have improved, although it is difficult to establish a direct link due to the improvement of other design exercises and student intake qualifications. Nevertheless, a steady improvement in IC design course grades can be shown after the introduction of the new IC design exercise over a number of years, as shown in Fig. 9.

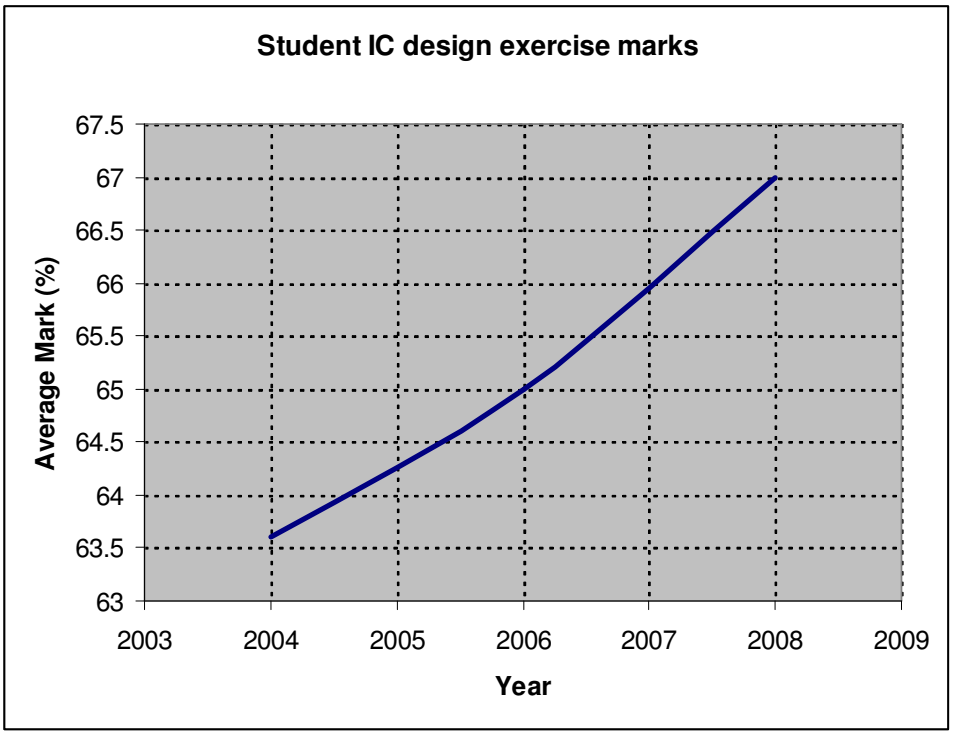

Fig. 9: Student Mark Trend from 2004 to 2008

Evaluation of annual student feedback forms shows that each year the IC design exercise is one of the highlights, with repeating comments such as "Interesting" and "Challenging". The students award marks out of 5 for aspects of the module including the technical content and the teaching quality, and provide an overall rating of the module. Comparing the student feedback for courses every year is dependent on

individual cohorts, but looking at the student responses to the overall unit quality and teaching, it can be seen that compared to the school average of 3.5, the IC design exercise scores highly. Fig. 10 shows the feedback results for 2007 and 2008.

In addition to the overall course quality evaluation, the cohort of students is asked to provide comments. Recurring highlights for the Superchip course were “Learning to use the EDA tools to design”, “practical”, “very important”. The student results and also student feedback responses are felt to have been very positive and the authors will continue to strive to maintain and improve further on these.

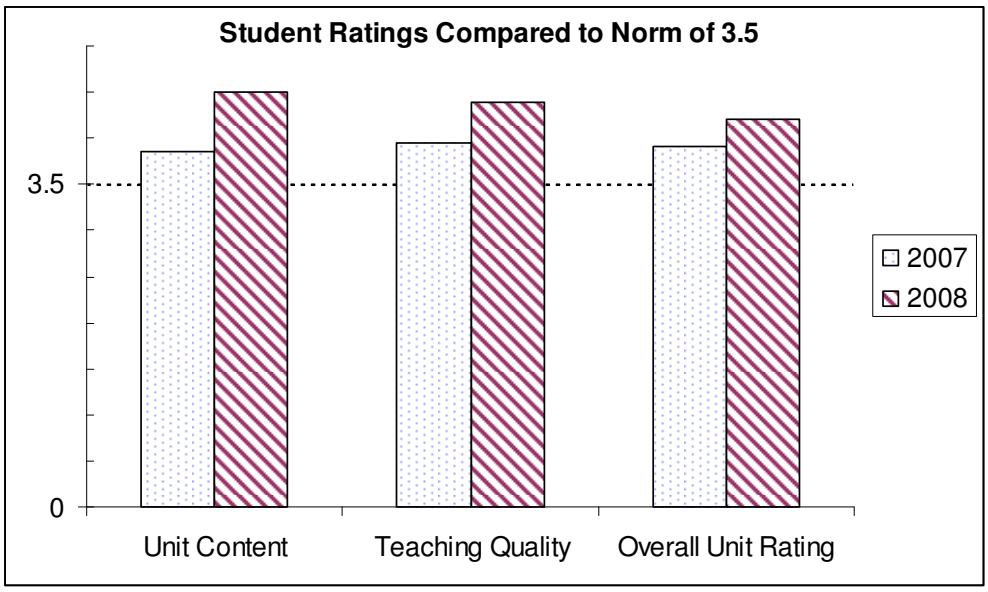

Fig. 10: Student feedback for 2007 and 2008

As far as the authors are aware, this is the only example of a course at this stage in any undergraduate curriculum that completes the full IC design, fabrication and test cycle, with delivered and packaged CMOS circuits. Significantly, student numbers are maintained at a high level, despite the national trends of reduction in electronic engineering courses. Furthermore, feedback from prospective students applying to these courses indicates that this is one of the differentiating factors that influenced their decision to select this particular degree course.

VI. Conclusions

This design exercise is unique in that a cohort of second-year undergraduates will have experienced a complete CMOS IC design process flow during their three or four-year degree program, including making their own ICs. This is the most recent innovation in a long history of CMOS design and fabrication undertaken by undergraduates at Southampton; since 2004 over 400 students have produced their own designs on silicon using this approach. The benefits to industry are clear, as the students leave the university with not only the theoretical and design skills, but also a practical knowledge of real design deadlines, team-working and the achievement of designing, making and testing their own ICs. An increase has also been noticed in awareness and enthusiasm for the general area of IC design (for which there is a particular shortage of engineers in certain areas), and this has resulted in a greater number of students wishing to carry out projects and research. This paper has described the architecture of the Superchip, the test board and the test vector approach used. The authors conclude that this demonstrates how large numbers of undergraduate or postgraduate students can be successfully taught the essentials of IC design in a practical and cost-effective manner. The key to this is the innovative chip architecture inherent in the Superchip which allows up to sixteen separate designs to be fabricated on a single IC.

REFERENCES

[1] P.R. Wilson, I. McNally, M. Swabey and B. Al-Hashimi, "IC Design and Manufacture for Undergraduates: Theory, Design and Practice," *European Workshop on Microelectronics Education*, 2008.

[2] M.T. O'Keefe, J.C. Lindenlaub, S.C. Bass and T.P. Wahlen, "Introducing VLSI computer-aided design into the EE curriculum: a case study," *IEEE Trans. Edu.*, vol. 32, pp. 226-236, Aug. 1989.

[3] R.B. Brown, R.J. Lomax, G. Carichner and A.J. Drake, "A microprocessor design project in an introductory VLSI course," *IEEE Trans. Edu.*, vol. 43, pp. 353-361, Aug. 2000

[4] P. Murphy, P, J.P. Frantz, E. Welsh, R. Hardy, R., T.Mohsenin, J. Cavallaro, "VALID: custom ASIC verification and FPGA education platform," *Microelectronic Systems Education*, pp. 64-65, June 2003.

[5] M. Serra, B. McLeod, B. Sorbin, L. Turner and Y. Savaria, "New advances in microelectronics education: a Canadian model," *IEEE Trans. Edu.*, vol. 36, pp. 141-147, Feb. 1993.

[6] D. Harris, "The microprocessor as a microcosm: a hands-on approach to VLSI design education," *Frontiers in Education*, vol. 3, pp. S2D1-S2D5, Nov. 2002.

[7] D. Harris and D. Diaz, "TestosterICs: a low-cost functional chip tester", *Microelectronic Systems Education*, pp. 74-75, Jun. 2003.

[8] R. J. Niederjohn and R. J. Schmitz, "The case for project-oriented courses with educationally useful student design projects," *IEEE Trans. Edu.*, vol. 25, pp. 65-70, May 1982.

[9] J. M. Feldman and M. B. Silevitch, "Design as a regular component in regular lecture courses-incentives and disincentives," *IEEE Trans. Edu.*, vol. 28. pp. 85-91, May 1985.

[10] D.A. Kolb, *Experiental Learning: Experience at the source of learning and development*, (1st Ed.), Englewood Cliffs:Prentice-Hall 1984.

[11] S. Brown, "The art of teaching small groups", *The New Academic*, vol. 6, pp. 3-6, 1997.

[12] C. Atman and K. Bursic, "How Effective are Textbooks in Teaching the Engineering Design Process," *Frontiers in Education Conference*, Nov. 1995.

[13] J. Biggs, *Teaching for Quality Learning at University*, (2nd Ed.), Berkshire:Open University Press, 2002.

[14] D. Qualters, "Using classroom assessment data to improve student learning," *Classroom Assessment Guidebook*, Northeastern University, USA.

[15] R.M. Felder and R. Brent, "Effective strategies for cooperative learning," *J. Cooperation & Collaboration in College Teaching*, vol. 10, pp. 63–69, 2000.

[16] D.R. Woods, R.M. Felder, A. Rugarcia, and J.E. Stice, "The Future of Engineering Education. 3. Developing Critical Skills," *Chem. Engr. Education*, vol. 34, pp. 108–117, 2000.

[17] R.M. Felder and R. Brent, "Designing and Teaching Courses to Satisfy the ABET Engineering Criteria", J. *Engr. Education*, vol. 92, pp. 7-25, 2003.

[18] R. Dearing, National Committee of Inquiry into Higher Education (Dearing Report), Higher Education in the Learning Society, *Report of the National Committee*, HMSO, 1997

[19] P.R. Wilson, R. Wilcock, B.I. McNally, M.A. Swabey and B.M. Al-Hashimi, "The Superchip: Innovative Teaching of IC Design and Manufacture, *Custom Integrated Circuits Conference*, Sep. 2008, pp. 535–538.

[20] CMP Foundry Services, http://cmp.imag.fr

[21] Europractice IC Manufacture, http://www.europractice.com/

[22] MOSIS IC Fabrication Services, http://www.mosis.com/

[23] D. Bouldin, A. Miller and C. Tan, “Teaching Custom Integrated Circuit Design and Verification”, *IEEE MSE*, 2003, pp. 139-140.

Dr Peter R. Wilson (M’99, SM’06) was born in Edinburgh, Scotland, and received the B.Eng. (Hons.) in Electrical and Electronic Engineering from Heriot-Watt University, Edinburgh, Scotland, in 1988; an M.B.A from the Edinburgh Business School, Soctland in 1999, and Ph.D. from the University of Southampton, England in 2002.

He worked in the Navigation Systems Division of Ferranti in Edinburgh, Scotland from 1988-1990 on Fire Control Computer systems, before moving in 1990 to the Radar Systems Division of GEC-Marconi Avionics, also in Edinburgh, Scotland. During the period 1990-1994 he worked on modeling and simulation of Power Supplies, Signal Processing Systems, Servo and Mixed technology systems. From 1994-1999 he worked as European Product Specialist with Analogy Inc., based in Swindon, UK and Beaverton, Oregon. During this time he developed a number of models, libraries and modeling tools for the Saber simulator, especially in the areas of Power Systems, Magnetic Components and Telecommunications.

Dr Wilson is currently a Senior Lecturer in Electronics at the School of Electronics and Computer Science, University of Southampton, England. His current research interests include modeling of magnetic components in electric circuits, power electronics, renewable energy systems, integrated circuit design, VHDL-AMS modeling and simulation, and the development of electronic design tools. He has published more than 70 articles in these areas, and one book. He is a Member of the IET and a Chartered Engineer, both in the UK. He is Vice Chair of IEEE Std. 1076.1.1, for which work he received an IEEE award in 2006, Secretary of the IEEE Power Electronics Society Standards Committee, General Chair of Behavioral Modeling and Simulation 2009, and has served as Technical Program Chair of Behavioral Modeling and Simulation 2008 and Finance Chair of the European Test Symposium 2006.

Dr Reuben Wilcock (M'01) received the B.Eng. degree in Electronic Engineering from the University of Southampton, UK, in 2001 and the Ph.D. degree in Electronic Engineering from the University of Southampton in 2004.

He was appointed as a research fellow in the Electronic Systems and Devices research group in the University of Southampton from 2004 to 2005 and was later appointed as a senior design engineer in a high tech start-up company from 2005 to 2006.

In 2006, Dr Wilcock rejoined the Electronic Systems and Devices research group at the University of Southampton as a senior research fellow. His current research interests include high frequency, low power, analog circuit design in CMOS, and analog design for yield techniques for deep submicron process technologies. He is a member of the IET.

Iain McNally received the B.Sc. degree in Electronics from the University of Southampton, UK, in 1986.

He has worked in research for British Telecoms, Marconi Maritime Labs and for the University of Southampton.

Mr McNally is currently a senior teaching fellow at the University of Southampton where he has been teaching integrated circuit design since 1991.

Dr Matthew Swabey (M'08) received the B.Eng. (Hons.) degree in Electronic Engineering from the University of Southampton, UK in 2001 and the Ph.D. degree in Electronic Engineering from the University of Southampton in 2006.

He has worked as a research fellow in the fields of otoacoustics and renewable energy at the University of Southampton.

Dr Swabey is currently a teaching fellow at the University of Southampton where he teaches analog electronics, digital electronics and computer engineering.