The Superchip: Innovative Teaching of IC Design and Manufacture

Peter R. Wilson, Reuben Wilcock, Iain McNally, Matthew Swabey and Bashir Al-Hashimi
{prw, rw3, bim, mas, bmah}@ecs.soton.ac.uk
University of Southampton

***Abstract*- In this paper we describe how through intelligent chip architecture, a large cohort (~100 students) of undergraduates can be given effective practical insight into IC design by designing and manufacturing their own individual ICs. To achieve this, the "Superchip" has been developed, which allows (without excessive cost in terms of time or resources) multiple student designs to be fabricated on a single IC, and encapsulated in a standard package. We demonstrate 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. The paper provides details of the chip architecture, test regime, test vectors, and an example design.**

I. INTRODUCTION

A. Background

Recent advances in IC CMOS process technology have forced Electronics departments world-wide to adapt their educational programs to equip students with the right skills and knowledge needed by industry. In addition, design cycle times (the time it takes to get from product specification to delivery to the market) are being driven ever shorter. The skills that are required to support this level of design are rapidly changing, as are the software and hardware tools required for engineers. In addition to the technical skills, we have taken an approach of team organization that is enabling for students and provides invaluable skills in terms of time management, team working, collaboration and interpersonal skills. The Electronics Engineering undergraduate program at the University of Southampton has run successfully for many years and provides a good grounding in hardware design. This paper demonstrates Southampton's adaptation to modern industry requirements, the educational rationale for this exercise and results are presented of its implementation and delivery.

B Learning Strategy

A key part of the strategy for learning has been to provide a solid experiential learning platform based on the Kolb learning cycle [1] and in particular by using small groups [2]. The strength of this approach is clearly the tutorial style with students able to progress at their own pace with a structured work plan to facilitate learning. While this is desirable in itself, there is the added benefit of empowering students by allowing them to organize their groups into whichever structure suits them best. This 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, with very clear operating instructions. Our experience in this design exercise has however been that the students welcome the responsibility and enjoy the fact that there is a "real" deadline for chip manufacture to meet, not just an artificial deadline, typical for most coursework at undergraduate level. Providing literature prior to the session [3] enables students to take a less linear approach to the design process and enable much more iteration and creativity to take place. This is intentionally in place to help develop students that can do design – not just rote type learning. Biggs [4] provides a useful framework to assist in the strategy of preparation and we apply different methods of delivery and assessment [6] to engage large classes more directly. A key aspect of this approach is to use student-oriented learning [5]. In our courses collaborative group work and peer review prove effective and useful in this context, and we incorporate this aspect into the design exercise.

C. 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 process of teaching, learning and assessment. This is essential to provide the student with a high quality of learning in a rapidly changing field. In this course we have taken a view of learning that considers the academic aspects of the work and links this to the industrially oriented aspects. Fourteen specific learning outcomes were devised for this course. The integration of key skills for industry is critical for engineering students in general as discussed by Woods et al [7], but in this particular field it is even more acute. In order to ensure that the learning outcomes in relation to the proposed course structure were appropriate, a matrix based approach was employed as described by Felder and Brent [8] to analyze the exercise structure in relation to learning outcomes. In this particular design exercise, we have identified relevant key skills and tied them into specific learning outcomes in a coherent manner that will provide the basic framework for the students to achieve a successful outcome and assessment has been considered in the context of the variety of skills, platforms and learning outcomes required, described by Felder and Brent in [5]. The approach we have taken to ensure this with a relatively inexperienced group of undergraduates is to provide design freedom, within a tightly constrained design tool framework.

D Design of the Exercise

The design exercise is divided into two main areas. In the first semester, the design and implementation stages take place. Teams will undertake the following activities, paper

design, schematic capture, design verification and simulation, layout, post layout verification and simulation, and finally design package production. These steps will be described in more detail later in this paper. 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 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 us to be a vital part of this exercise. 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 for fabrication. This takes around three months, and when the chips return, they can be tested during Semester two. In Semester two, the student teams re-convene to develop test vectors to enable automatic testing of their designs, carry out simulations to validate their test vectors using their original designs in simulation, and finally test their individual ICs.

II. The Superchip

A Introduction

A crucial aspect of the program is the ability to effectively support a large number of individual IC designs without excessive cost in terms of time or resources. In order to achieve this, the "Superchip" has been developed as shown in figure 1, which allows multiple student designs to be fabricated 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. There are 16 separate design slots within this single chip, and the cohort is divided into teams of around 6 students enabling a large number of students to develop separate designs as a group.

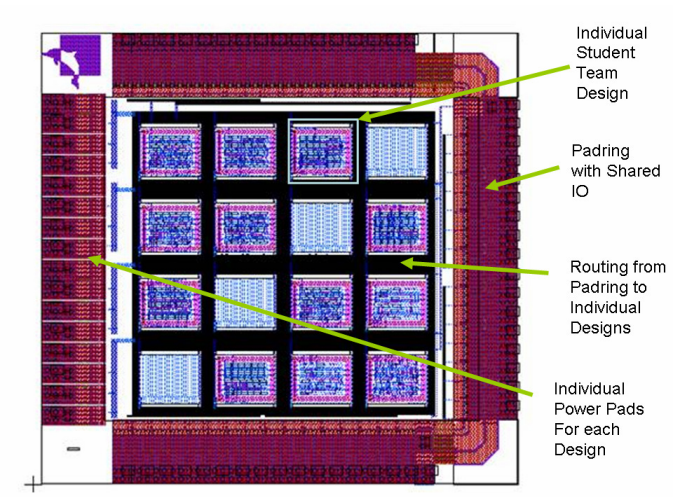

Figure 1: Typical Southampton Superchip IC layout

B Details of the Superchip Layout

The chip is designed using the Austria Microsystems C35B4 (0.35µm) CMOS process, with 4 metal layers available through multi project wafers (MPW). The Chip infrastructure consists of a padring which has 24 digital inputs, 24 digital outputs, 16 individual site power supplies (VDD) operating at 3.3V, a global VDD at 3.3V and a global Ground. This gives a total of 66 pins. The chip is packaged in a JLCC68 (68pin) package, with two spare pins, for use in general laboratory situations.

The individual student design sites are buffered and selected using separate power connections (VDD), so when a site is powered, then its inputs and outputs are also enabled. Within the ring of buffers and power for each site, a miniature pad ring has been created which is the interface to the Superchip that the students see.

III. The Design Process

A Introduction

In this section of the paper, we will introduce the key stages in the design process, particularly with reference to the students backgrounds from their first year in terms of knowledge, how this relates to the theoretical program of study and also the context of the skills looking forward to later on in their degree course. The design exercise is schedule as early as possible in the second year of the undergraduate program to provide as much time as possible to ensure that the ICs can be made in adequate time for testing the second semester. This has the obvious implication that we must assume the students only have the knowledge obtained in their first year by this stage. This has the effect of defining the type of designs that can be undertaken (simple digital synchronous or combinatorial logic design). Typical applications have therefore included sequence recognition, counter design, ALU design and oscillator design. In addition, the students have limited analogue electronics experience (basic CMOS transistor knowledge) and basic knowledge of electronic design tools.

B The Student Design Kit

As in any IC design, we provide the students with a complete design kit. This includes a library of schematic symbols, layout abstract cell views, simulation files, design rule check files and design extraction files. This is not however to be confused with the standard AMS design kit. In this case the design kit has a much reduced number of digital gates for the students to work with (shown in table 1). This greatly simplifies the scale of the design kit and it becomes markedly less intimidating psychologically. For example, a typical gate layout is shown in figure 2, where the inverter is simplified to 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 is hugely reduced, thus enabling student versions of software to be easily used, and also to minimise the design complexity. For more advanced students, the full layout cell views could certainly be used instead of the limited abstract views.

The routing is constrained, so that the students are not allowed to route over the cells, and are restricted to Metal 1 and Metal 2. They use a standard "routing channel" strategy

to manually connect up the cells. For each abstract cell, there is an equivalent Spice model for analog simulation, and a VHDL model for digital simulation. During the initial design phase, the students are restricted to Spice simulation, so it is important for them to ensure that not only are the cells connected correctly, but that the VDD and VSS are also connected.

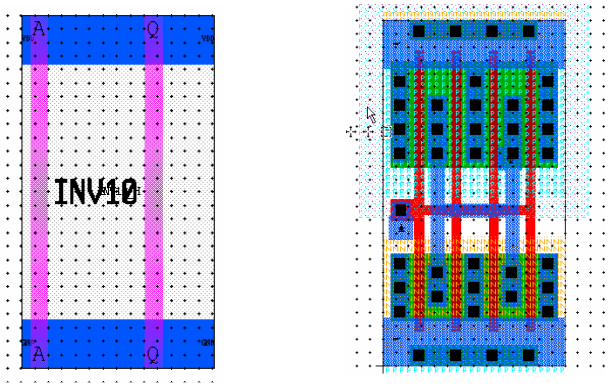

Figure 2: Typical Cell Abstract and Full Layout -Inverter

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 1: Design Kit Cells

C Design Tools and Methods

Given the limited knowledge of the students, we use the same PC based schematic design and simulation software they are familiar with in their first year studies. While these are necessarily optimum from an IC design perspective, as an introduction, they work well due to the existing familiarity of the students with the software. The overall process is shown in figure 2. Each stage of the process is discussed in the following sections of this paper.

D Design Specification

The design specification is published for all the teams and an introductory lecture is given to explain the detailed 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 similar level of functionality. In recent years, a ring oscillator has also been added as a specific item which can be used to test the process operation in a more "analog" function and enable the students to carry out some probing of high speed digital signals.

E Design process

A diagram of the standard design flow is shown in figure 3. The initial design phase is a typical "paper" design, where the team will discuss the options both for the functionality of the design, but also the implications for its fabrication.

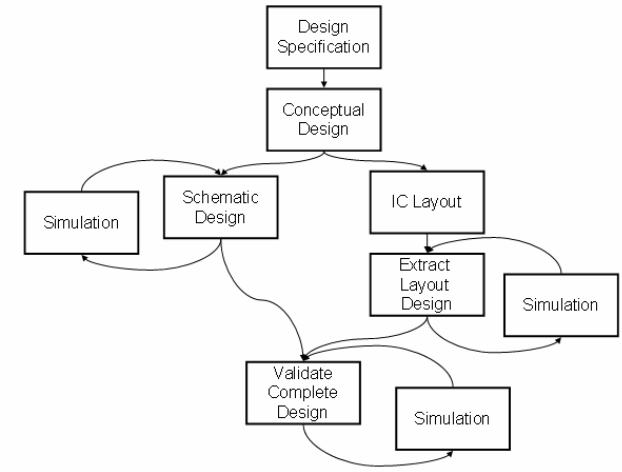

Figure 3: Design Process Flow Diagram

For example, in the ALU design one option is to design 8 bit functions in turn and link together, whereas an alternative approach would be to create a one bit "slice" and then simply replicate this 8 times. The student teams create a schematic of their design using the Orcad schematic capture software, from which they can simulate their design in Spice, or extract a VHDL model for digital simulation. We use an analog approach in the initial stages of the design to familiarize the students with the concepts of power consumption, realistic rise and fall times, overshoot, ground-bounce and device characteristics impacting on fan-out and loading.

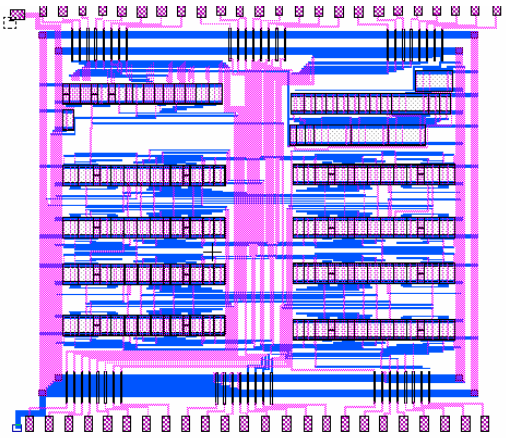

Figure 4: Student Design – 8-bit ALU

The IC Layout is carried out using the L-Edit software from Tanner EDA and 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 we introduce the students to the use of Design Rule Checking (DRC) at this stage also. 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 figure 4.

As can be seen from the layout, we use a standard cell based approach of manually laying out rows of abstract cells, with routing channels between the rows in two metal layers(M1 and M4). Although this is a standard 4 Metal Layer process, and it is possible to route over the top of the standard cells using M3 and M4, at this stage of the programme, it is useful to illustrate the concepts in channel based routing, and by manually routing, the students also have to think about the physical design, and implications of poor choice of cell placement.

V. IC Validation and Verification

A IC Test Board

When the ICs return from fabrication, the key task is to test them to ensure the basic functionality is correct, and also to carry out some basic performance measurements (timing, power consumption, oscillator frequency) to verify the design criteria have been satisfied. While it is straightforward in principle to carry out this type of testing by building a prototype test board, we take the view that it is more productive to provide a basic test infrastructure and to introduce the students to the concept of test vectors at this stage. To this end, we have developed a standard chip tester board, with a USB interface and chip socket, so semi automatic testing can be easily and quickly carried out on the individual designs. The credit card sized board is shown in figure 5, and uses a standard USB interface chip (CP2102) and a PIC to manage the interface between the PC and the test board, and the students also have full access to every pin via probe points directly next to the package.

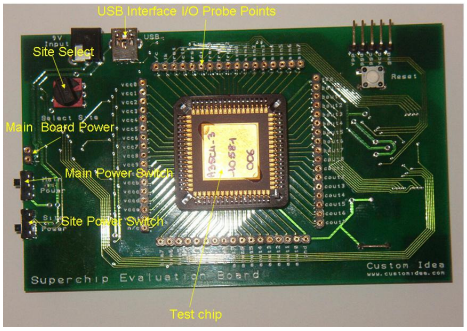

Figure 5: Superchip Test Board

B Test Vector Validation

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 test board can also export a VHDL test bench, so the test vectors can be tested using the model of the design, in this case in the Modelsim simulator, thus validating the test vectors prior to testing the IC itself. The test vectors are created in an XML style format, making editing and validation simple, with an example file shown below:

```
# Test Vector File
<PinDef>
clk,nrst,tclk,trst,tdi,freein0,freein1,freein2,ai
n0,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 either static 0, static 1 or a clock pulse denoted by C. The tester is not designed for high speed testing, but for the type of designs implemented (usually something like a frame decoder, ALU or sequence detector) these are perfectly adequate. (Lines beginning "#" are comments and ignored).

VI. Conclusions

This design exercise is unique in that the a cohort of second year undergraduates will have experienced a complete CMOS IC design process flow during their 4-year degree programme 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 and 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. The paper has described the architecture of the Superchip, the test board and the test vector approach used. We conclude that this demonstrates how large numbers of undergraduate or postgraduate students can be taught the essentials of IC design in a practical and cost-effective manner.

VII. References

[1] Kolb, D.A., "Experiental Learning: Experience at the source of learning and development", Prentice-Hall, Englewood Cliffs, NJ, 1984

[2] Brown S., "The art of teaching small groups", New Academic, Spring 1997, pp3-6

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

[4] Biggs J., "Teaching for Quality Learning at University", Open University Press, 2002.

[5] Felder, R.M., and R. Brent. 2001. "Effective strategies for cooperative learning," J. Cooperation & Collaboration in College Teaching, 10(2), 63–69.

[6] Qualters D., "Using classroom assessment data to improve student learning", Penn State University, USA.

[7] Woods, D.R., R.M. Felder, A. Rugarcia, and J.E. Stice. 2000. "The Future of Engineering Education. 3. Developing Critical Skills", Chem. Engr. Education, 34(2), 108–117.

[8] Felder, R.M., and R. Brent, "Designing and Teaching Courses to Satisfy the ABET Engineering Criteria", J. Engr. Education, 92(1), 7-25 (2003)

[12] W. Wolf, J. Madsen, "Embedded System Education for the Future", Proceedings of the IEEE, vol. 18(1), pp. 23-30, 2000