(

Integrating Nano-logic Knowledge Module into an Undergraduate Logic Design Course

Saket Srivastava, Student Member, IEEE *and* Sanjukta Bhanja, Member, IEEE

***Abstract*—This work discusses a knowledge module in the undergraduate logic design course introduced to electrical engineering and computer science students exposing them to nano-computing concepts. We have a two-fold objective from this knowledge module. First, we wish generate interest amongst our students regarding the fundamental logical behavior and functionality of future nano-devices. We hope that this motivates them to enroll in other elective courses related to Nanotechnology, being offered in most EE and CS departments. Further, we use this module to let students analyze, synthesize and apply their existing knowledge of Karnaugh-map based Boolean logic reduction scheme into revolutionary design context with majority logic. Whereas many efforts are focusing on developing new courses on nanofabrication and even nano-computing, we intend to augment the existing standard EE and CS courses by inserting knowledge modules on nano-logic structure for stimulating their interest without significant diversion from the course framework.**

***Index Terms*—logic design, K-maps, majority logic, QCA**

INTRODUCTION

There is a consensus among the computing researchers CMOS devices will stop scaling sometime around the year 2020 [7]. Some of the promising technologies such as Quantum-dot Cellular Automata (QCA) [6], Single Electron Transistors (SETs) [8], Carbon Nanotubes (CNTs) [9] and Tunneling Phase logic (TPL) [10], all have the potential to replace the current generation CMOS. Current generation engineering students will be the ones using these technologies in future and hence it is absolutely necessary for students of electrical engineering (EE) and computer science (CS) to have the knowledge of future nano-devices and the logic associated.

Logic design students in both EE and CS will have to design circuits in these technologies based on the logic associated with them. The logic primitive for one nano-device might not be same for another. For example, in QCA [6], the most optimum designs are those that make use of majority logic. Similarly, while using SET and TPL, minority gate logic design is used to implement circuits [2]. Apart from the logic associated with these new technologies, students also need to be aware of design methods that can make use of present Boolean logic designs and transform them into logic associated with the future technologies. Several research efforts are currently underway on logic level and circuit level designs of these nanotech devices. Our goal is to augment the existing courses with flavors of the nanotechnology in a friendly, abstracted manner. This generates interest amongst students and motivates them to enroll for other existing nanotechnology courses [3].

We wish to provide a seamless learning curve to the students to maximize the gains they obtain from this knowledge module without hindering their normal learning curve. In order to achieve this, we introduced the concept of majority network following the K-map based logic minimization techniques for AND-OR logic. Since K-maps are an integral part of any logic design curriculum, this was the right context to introduce the students to nano-computing.

The educational experiments are set to not only expose the students to newer technology but also to test the synthesis and comprehension skills on the learnt K-map in the previous lectures. The knowledge module consists of three parts. Initially, we introduce the students to some novel nano-devices and logic primitives for such technologies. Next, we help them analyze simple majority logic design examples that enable usage of familiar logical concepts to perform simple analysis. Finally, the students use the knowledge obtained in the first two parts to design a small circuit using a novel algorithm showing a higher level comprehension, synthesis and application. At each step our aim is to provide a smooth transition between consecutive stages of the module to maximize learning by applying familiar logic design concepts.

At each step we evaluated the learning progress by providing worksheets to the students. Finally, we showed the abstracted view of QCA logic and functionality to motivate students into nano-device and nano-computing research. It is clear that at every step, apart from exposing the students towards nanotechnology, our goal has been to enhance their critical thinking by reinforcement of the known idea and synthesis of novel logic structures applying the logic design concepts that they are familiar with.

We introduced this knowledge module in two semesters (Fall 2006 and Spring 2007). This helped us have a broader base of students to evaluate the outcome as well as the progress. In the next section, we provide an overview of the K-map based knowledge.

K-MAP BASED KNOWLEDGE MODULE

The knowledge module on nano-computing resides on the undergraduate logic design course. This is one of the core courses for electrical engineering students and a pre-requisite for a number of advanced courses such a CMOS VLSI design and microprocessor design. Students taking this course are usually in their sophomore or junior year. Most of the students are not yet familiar with Boolean logic and current technology being used in the industry to design circuits. *Since this is the first course of this type that they undertake, it lays the foundation of their future goals depending on the interest generated by this course.* Hence we thought it would be wise to introduce a flavor of nano-logic in this course. Our aim is to help students get some information and consequently to help generate interest in the field of nanotechnology.

1 Intellectual Merit

Intellectual merit of this work is two-fold. We not only wish to motivate students in electrical engineering towards future technologies, we also wish to reinforce their knowledge in Karnaugh maps. This module helps them apply their knowledge of K-maps into a novel majority logic synthesis application and helps to improve their analytical skills.

While developing this knowledge module, we first started with the study of various promising nanotechnologies that are being researched. Since this is a design oriented course, we did not wish to overwhelm the students with the fabrication details of the technology. We rather focused on the functionality and logic associated with the technology. We short-listed a few such technologies such as QCA, CNT and SET for this purpose. The primary reason to choose these three technologies is that all three of them use different logic schemes for implementation. While QCA uses majority logic, CNT implements traditional NAND logic and SET uses minority logic.

Since the knowledge module spanned over two lectures, it was not possible to educate the students about the quantum mechanical properties of these devices. We rather focused on abstracting the logical behavior of these devices and present it to the students in a way that would be easy to comprehend. The algorithm we use in this module is complex to understand for students of this class. The algorithm uses a multi-level iterative design procedure which is beyond the scope of undergraduate logic design course. Hence to simplify the complexity of this algorithm and use it in the class, we first solved a number of common logic expressions with it such as an even parity generator and an odd parity generator. These expressions are regularly used in the undergraduate logic design class and students are quiet familiar with them. Next, we demonstrated the algorithm to the students with these simple examples and then students were asked to design a new circuit based on that. Our objective was to let the students understand the algorithm without getting overwhelmed with the iterative procedure used to optimize the logical function. In order to enable our students to use the algorithm we only provided them with a library of five such K-map patterns (overall *38* such patterns could be there) out of which they needed to choose three patterns to effectively represent a Boolean logic function into a majority logic function.

We also intend this module to work as a reinforcement of their digital concepts namely algebraic manipulation, logic minimization, the true meaning of min-terms. The students were able to synthesize and apply the knowledge of achieving irredundant prime-implicants into achieving minimum majority network.

2 Teaching Methodology

Our teaching methodology is three-fold. First, we demonstrate to our students how they can relate the already familiarized Boolean logic concepts to logic associated with other nano-devices. This step helps the students grasp the underlying idea behind the new information and relate it to the information they have learnt in the previous lectures of this class. It will also help them reinforce the existing knowledge resulting in increased information acquisition and retention.

Secondly, students are shown how they can use this acquired knowledge to analyze the new logic style by making use of a simple AND/OR mapping scheme. Since students are already familiar with AND/OR gates, they find it really interesting how the same type of analysis can be performed using majority logic. This step helps the students visualize and analyze the new logic style with a minimum diversion and we observe that this step has great impact on our students (discussed in section III). It is important for students to realize that while logic styles might change when technology changes, the inherent concepts regarding design styles remain the same.

Students are taught how a 3-input majority gate can be represented as an AND or OR gate by fixing one of its inputs as a 0 or 1 respectively as shown in Fig 1(a). In this way students are able to visualize and synthesize small Boolean circuits using majority gate logic just by representing each AND and OR gate by its equivalent majority gate. Fig. 1(b) shows QCA implementation [1] for the Boolean expression mapped by AND/OR logic in Fig 1(a). Also we emphasize that any Boolean function can be implemented by using majority gates and inverters. Fig 2(a) shows the AND/OR mapping of a reduced Boolean logic function (shown in Fig. 1) and Fig 2(b) shows its QCA implementation.

As a final part of this logic module, we ask the students to perform a relatively complex design algorithm by applying the already-known K-map to the new logic style that they have been recently exposed in the earlier part of this module.

3 K-maps in Logic Design Course

K-maps are the graphical representation used for reduction of complex logical expressions to a reduced form. K-maps are one of the most essential elements of any Digital logic design curriculum. They facilitate learning by representing information in a graphical format that is easier to comprehend, analyze and evaluate. The patterns that the students are looking at traditionally in a logic design course are connected to AND-OR logic and the patterns resemble squares/rectangles of 2^m by 2^n cells due to the algebraic adjacency. However for majority logic the patterns are irregular shape and students need to understand and comprehend the differences and similarities between the logic styles.

A Boolean logic function can be minimized by mapping all the zeros and ones in the truth table to a K-map and then combining adjacent ones in the K-map in groups, each representing a min-term. We try to group ones in a K-map by first trying the largest 2^p group and then group them in decreasing order of 2^{p-1}, 2^{p-2} and so on. This grouping is done only I a rectangular or square matrix of 2^m by 2^n cells. In this way we have to cover each one mapped on the K-map and each group represents a min-term of the reduced Boolean logic function.

In order to change a Boolean logic function to a majority logic function, we do not group the ones in the way described above, rather we group them according to one of the shapes in the pre-built library of 38 such shapes, each of which denotes majority logic function of at most three variables.

Fig. 3(a) shows how a logic expression mapped on a K-map will appear in the reduced form in Boolean logic. We can see from the figure that the reduced Boolean expression contains three min-terms, where each min-term is a group of two adjacent ones. As we can see that we cannot group all four ones in a matrix of 2x2 hence we have to group them in three matrices of 2x1. Each grouping represents a min-term in the reduced expression. The schematic of reduced expression is shown in Fig. 3(b). In Fig. 3(c) we show that all the four ones can be grouped together to represent a majority gate of three variables. The schematic of a three input majority gate (derived from the K-map) is shown in Fig. 3(d).

4 Algorithm

The algorithm demonstrated for majority logic in [2] is complex to understand and had to be simplified significantly to enable undergraduate logic design students to understand and implement it in their worksheets. The algorithm makes use of a library of 38 predetermined K-map patterns. An iterative algorithm then chooses three most optimum K-map patterns each of which represents a majority logic function (M1, M2 and M3). Using [4] it as been proven that any three input logic function *f* can be represented using a maximum of four majority gates.

$$f\ (a,\ b,\ c) = M(M1,M2,M3)$$

Fig. 4(a) shows the schematic of a majority logic function obtained after using the algorithm that reduces a complex three input Boolean logic function to a majority logic function. Fig. 4(b) shows the QCA implementation of a four majority gate design derived using the K-map based design algorithm discussed above.

In the next section we will evaluate the impact of this knowledge module based on the performance of students in each of the worksheets handed out to them.

EVALUATION

In order to evaluate the level of understanding that students gained from this course module, they were given worksheets at the end of each step of this knowledge module. We followed the Bloom's taxonomy [5] in the cognitive domain while formulating these worksheets in order to maximize the gains to the students. The cognitive domain deals with the development of knowledge and development of intellectual attitudes and skills.

1 Worksheet Details

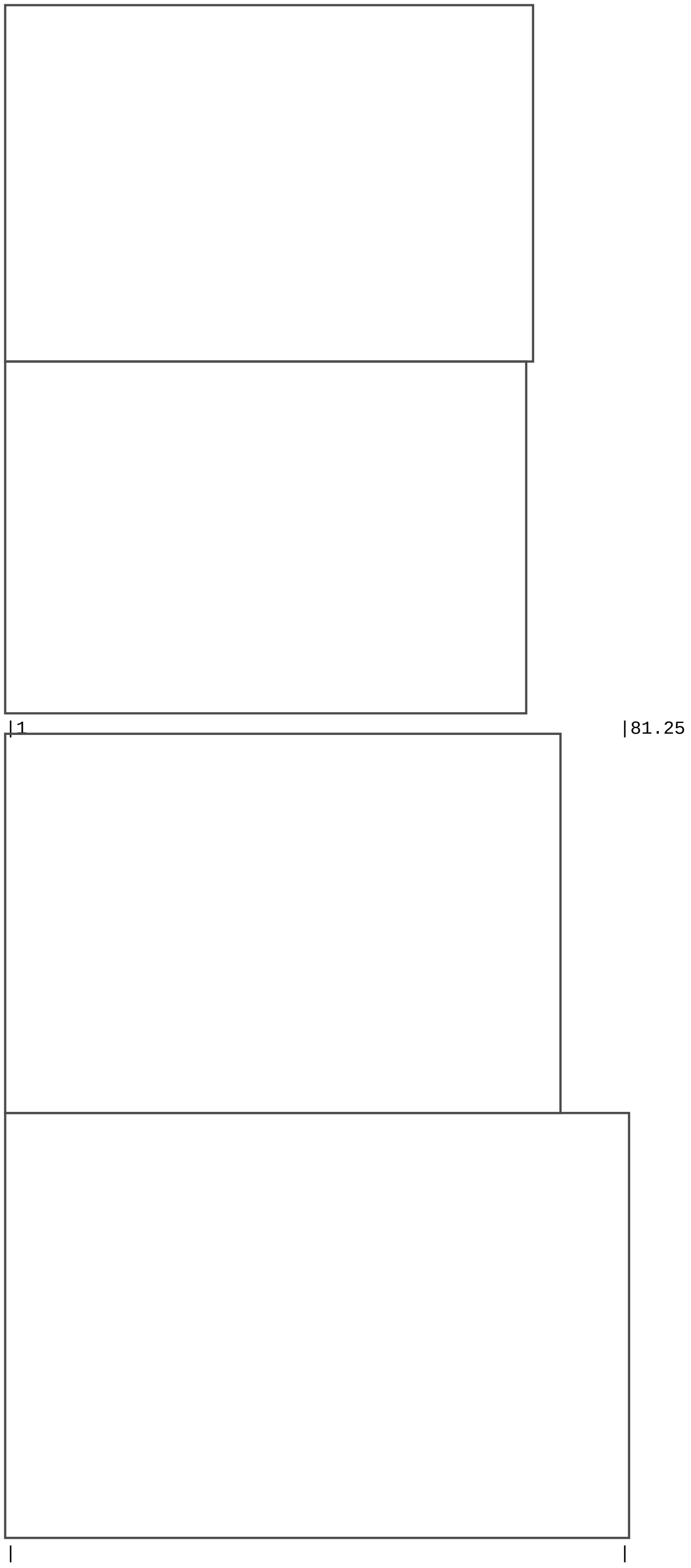
Worksheet-1 evaluated their understanding of QCA majority logic. This worksheet gives us idea about the interest and the level of understanding shown by the class to the new concepts. For the students, this worksheet evaluates information acquisition of a novel concept. This worksheet deals with the knowledge and comprehension skills.

Worksheet-2 evaluated their level of understanding to perform an AND/OR mapping of simple Boolean expressions. This worksheet helps the students not only perform the simple design, but also evaluates the seamless transition between known concepts using old logic style to a novel logic style. This worksheet deals with the application and analytical skills of the students.

Finally, in **worksheet-3,** students were asked to perform a K-map based synthesis method to reduce a complex three input logic function. Students were also asked to perform an AND/OR mapping for the same function to understand the advantage of K-map based method over AND/OR mapping method. Not only does this worksheet challenge the analytical skills of the students to apply the knowledge attained in this course to a totally new logic style, it also enhances their critical thinking by reinforcement of known ideas. They see how K-maps while being an important part of logic design curriculum can also be used as a design tool for other logic styles. This worksheet falls under the category of synthesis skills (different from majority logic synthesis discussed above) in the cognitive domain.

TABLE I
RESULTS OBTAINED FOR EACH WORKSHEET IN FALL 2006 AND SPRING 2007

Worksheet	Percentage of students receiving grade					
	Grade A		Grade B		Grade C	
	F06	S07	F06	S07	F06	S07



|1
|81.25
|
|

CONCLUSION

We believe that this module will serve as a good resource for faculties teaching logic design class. Our results from the worksheets and feedback survey suggests that we were able to achieve our twin objectives We motivated our students towards learning more about Nanotechnology and nano-computing in a nice and easy way. We were also able to strengthen their knowledge in K-maps and help them develop cognitive skills to apply this knowledge in a very novel way. The deliverables of this work are the lecture notes, sample student worksheets and feedbacks. In future semesters we also intend to introduce students to other promising nano-logic devices such as SET and TPL and logic associated with them.

TABLE III

PIE-CHARTS OF STUDENTS RESPONSE TO THE FEEDBACK SURVEY. SAMPLE SIZE INCLUDES STUDENTS FROM BOTH THE SEMESTERS.

QUESTION 5 TO QUESTION 7

APPENDIX

We have added scanned copies of the graded worksheets and feedback survey from the students. We are also attaching power point presentation that was used as a study material for this class.

References

1] QCADESIGNER,www.qcadesigner.ca

2] Zhang R; Gupta, P; Jha, N K, "Synthesis of majority and minority networks and its applications to QCA, TPL and SET based nanotechnologies," *18th International Conference on VLSI Design*., 3-7 Jan. 2005, pp. 229- 234

3] Minsu Choi; Park, N., "Teaching nanotechnology by introducing crossbar-based architecture and quantum-dot cellular automata," *Microelectronic Systems Education, 2005. (MSE '05). Proceedings. 2005 IEEE International Conference on* , vol., no.pp. 29- 30, 12-14 June 2005

4] S. B. Akers, "Synthesis of combinational logic using three-input majority gates," in *Proc. Third Annual Symp. Switching Circuit Theory & Logical Design*, Oct. 1962, pp. 149–157.

5] Bloom B. S. (1956). *Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain.* New York: David McKay Co Inc.

6] P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," J. Applied Physics, vol. 75, no. 3, pp. 1811--1817, Feb. 1994.

7] International Technology Roadmap for Semiconductors, 2006.

8] M. A. Kastner, "The single-electron transistor", Rev.Mod. Phys. 64, 849 (1992).

9] Adrian Bachtold, Peter Hadley, Takeshi Nakanishi, and Cees Dekker "Logic Circuits with Carbon Nanotube Transistors", *Science* **294** (5545), 1317, 9 November 2001

10] H. A. Fahmy and R. A. Kiehl, "Complete logic family using tunneling-phase-logic devices," in Proc. Int. Conf. Microelectronics, Nov. 1999, pp. 22--24.

Fig. 4. Schematic diagram and QCA implementation of a Boolean expression represented in majority logic.

Fig. 1. AND/OR mapping of a Boolean Logic function n = x1’.x3’+x2.x3’

Fig. 2. AND/OR mapping of reduned Boolean logic function n = x1’.x3’+x2.x3’ = (x1’+ x2) x3’

Fig. 3. (a) Grouping of 1’s in a K-map for Boolean reduction (b) AND/OR schematic of the reduced Boolean expression (c) grouping of 1’s in a K-map for a majority logic design and (d) a majority gate equivalent of the reduced K-map expression.

P = 0 (for AND Gate)