A NEW APPROACH TO EDUCATIONAL SOFTWARE FOR LOGIC ANALYSIS AND DESIGN

Javier García Zubía, Jesús Sanz Martínez and Borja Sotomayor Basilio
Faculty of Engineering of the University of Deusto, Apdo. 1, 48080
Bilbao, Vizcaya, Spain
zubia@eside.deusto.es

Abstract

BOOLE-DEUSTO is a software package capable of analysis and design of combinational circuits and finite state machines. Its ease of use, rich features, and didactic nature can make it a valuable teaching aid in Digital Electronics courses. The paper explains BOOLE-DEUSTO's pedagogical justification, its main features, and our teaching experience with it. The paper also matches the features and particular strengths of BOOLE-DEUSTO with LogicAid, a similar program developed by professor Roth from the University of Texas.

Keywords

logic design, pedagogical software, didactic software

1. PEDAGOGICAL JUSTIFICATION AND SCOPE

During an introductory course on digital electronics, software is one of the main teaching tools. It is not uncommon to use a professional package, such as OrCAD, ISE (Xilinx), ElectronicsWorkBench, MaxPlus II (Altera), etc. Table 1 shows that BOOLE is very different from professional software packages. Indeed, BOOLE isn't meant as a substitute for OrCAD, etc. BOOLE is meant as a complement, satisfying the needs of teachers and students during their first steps in a digital electronics course. Once this first phase is finished, a student will be more suitably prepared to use professional design software.

Since BOOLE-DEUSTO is intended for introductory courses, it can only carry out analysis and design of bit-level combinational and sequential circuits. Word-level and system-level circuits are currently out of its reach (see Figure 1).

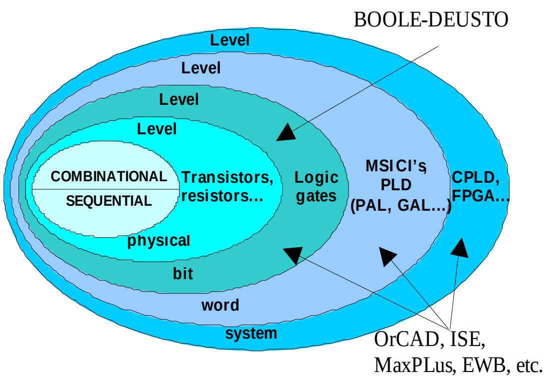

Figure 1 Abstraction levels of a circuit

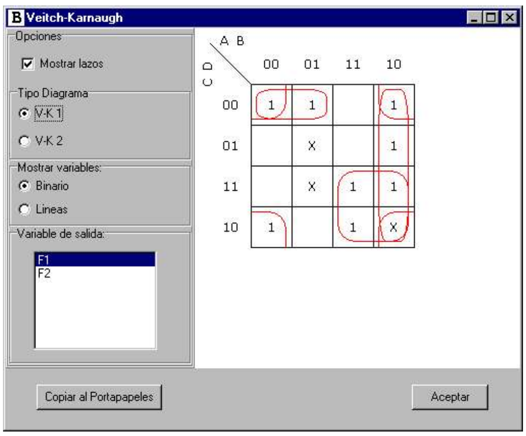

Figure 2 Veitch-Karnaugh diagram

BOOLE-DEUSTO	Professional Software
Educational	Professional
Easy to use	Difficult to use
Easy to install	Difficult to install
Free	Expensive
Simple projects	Complex projects
Focuses on methodology	Focuses on results

Table 1BOOLE-DEUSTO vs. Professional Software

2. THE BOOLE-DEUSTO ENVIRONMENT

The Boole-Deusto environment can be broadly divided into two parts: the analysis and design of combinational circuits, and the analysis and design of finite-state machines (sequential circuits). The program also has a set of features which are common to both the combinational and sequential module of BOOLE-DEUSTO. The most interesting features are described in this section.

2.1 Combinational Circuits

- ***Truth Table***
  The user can define a combinational circuit by filling in its truth table.
- ***Boolean Expression***
  The latest BOOLE-DEUSTO version includes a parser which adequately interprets any kind of boolean expression, including the XOR operator. The user can, therefore, define a circuit by inputting a boolean expression.
- **Canonical Forms**
  The user can also define a circuit by writing it in its canonical form (minterms or maxterms).
- ***Minimized Expressions***
  Once the user has defined the circuit, BOOLE-DEUSTO can obtain the minimized expression. The minimization method used is exact and non-heuristic, so it always yields the optimum minimized expression.
- ***NAND/NOR Expressions***
  BOOLE-DEUSTO can generate the equivalent NAND/NOR expressions from the minimized expression.
- ***Circuit Logic***
  The program is capable of generating the circuit diagram from any SOP or POS expression. The user can either write up his own SOP/POS expressions, or simply generate the circuit diagram from the minimized expression of the combinational circuit he/she has defined.
- ***Veitch-Karnaugh Diagrams***
  Veith-Karnaugh diagrams can be used as both an input and output format. The user can define a combinational circuit directly on a V-K map, or view the optimal V-K once the circuit has been minimized (see Figure 2).
- ***Veitch-Karnaugh Learning Mode***
  One of the most interesting features in the latest version of BOOLE-DEUSTO is the V-K Learning Mode. This new module tests the user's ability to find the minimal loops necessary to solve a V-K diagram. Once the user has defined the circuit (for example, by defining the truth table) he/she is shown a V-K diagram without loops. Using the mouse, the user can create and erase loops. Once all the loops have been defined, a “Test Solution” button is pressed, and BOOLE-DEUSTO will tell the user if the loops are (a) incorrect, (b) correct, but not minimal, or (c) correct and minimal.

2.2 Finite State Machines

- ***Moore-Mealy’s Diagrams***
  Both Moore and Mealy FSMs can be defined graphically using a simple point-and-click interface.
- ***FSM verification***
  Once the user has defined the system, BOOLE-DEUSTO can automatically verify if the user has made any mistakes (for example, defining duplicate transitions).

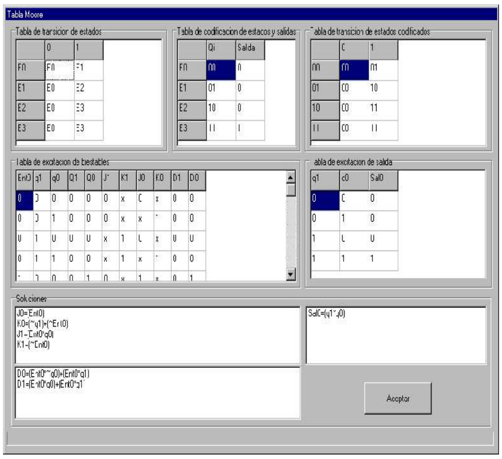

Figure 3 FSM design tables

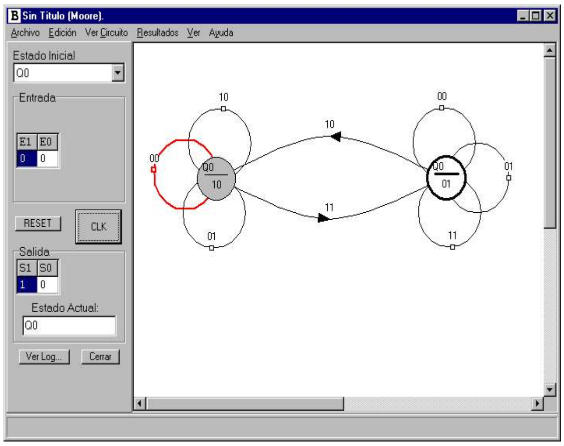

Figure 4 Interactive simulation of a FSM

- ***Tables and Minimized Expressions***
  One the system has been defined and verified, BOOLE-DEUSTO can obtain all the tables and minimized expressions necessary to implement J-K or D flip-flops. See figure 3.
- ***J-K and D Circuit Logic***
  The J-K and D circuit diagrams can also be obtained once the system has been defined and verified.
- ***State Minimization***
  BOOLE-DEUSTO includes a versatile FSM minimization algorithm capable of dealing with X conditions and incompletely-specified systems.
- ***Moore <-> Mealy Conversion***
  FSMs can be converted from Moore to Mealy, and viceversa.
- ***Interactive and batch simulations***
  Probably the most powerful and most popular feature in BOOLE-DEUSTO is the possibility of simulating finite state machines. Interactive simulation (see Figure 4) allows the user to interact with the system by modifying the input signal and sending a clock signal and inmediately seeing the result of his actions on the state-transition diagram (which highlights the current state, the transition which will be followed given the current input signals, and the current output). To review his/her actions, a detailed log of the simulation is kept which can be saved to disk, copied to the clipboard, or printed out. BOOLE-DEUSTO also supports batch simulation. In this case, the user specifies the complete sequence of input signals and clock signals, and can then follow the execution step by step, or run through the whole sequence and see the final result.

2.3 Program-wide features

- ***Code generation***
  BOOLE-DEUSTO can generate OrCAD-PLD, VHDL, and JEDEC code for both combinational and sequential systems. Currently, JEDEC code generation only supports the 22V10 device.
- ***Save and load systems to/from disk***
  Both combinational and sequential systems can be saved and loded to/from disk.
- ***Associate text with a system***
  The user can associate text with a combinational or sequential system. This text can contain a description of the system, or simply some personal notes. This text is saved to disk along with the system.
- ***Print systems in their various representations***
  The systems can be printed out in most of their representations (graphical, circuit, tables, etc.)

3. TEACHING EXPERIENCE

More than 700 students every year have used BOOLE at the University of Deusto since 1996. It is included in both Industrial and Computer Science Engineering curricula (in Digital Electronics and Computer Technology courses, respectively). This year a survey was carried out at the University of Deusto among students who have used BOOLE-DEUSTO this year in electronics courses. The questions where yes or no questions which could be rated on a scale of 1 (do not agree) to 5 (completely agree). The results of this survey are shown in table 2.

Question	Result
Relevance as a teaching aid	
Is BOOLE-DEUSTO didactic/pedagogical?	3,9
Is it useful to document exercises?	3,5
Has it fulfilled your needs as a student?	3,5
Should the use of BOOLE-DEUSTO and similar software packages be encouraged?	3,9
Features	
Is BOOLE-DEUSTO atractive?	3,2
Is it easy to use?	4,1
Is BOOLE-DEUSTO effective at showing the results of an operation?	3,5
Does BOOLE-DEUSTO generate correct results?	4,5
Is it easy to install?	4,8
Relevance in current course	
Has BOOLE-DEUSTO been useful to you in this course?	2,6
Has BOOLE-DEUSTO been useful to perform boolean minimization?	3,4
Has BOOLE-DEUSTO been useful to work with FSMs?	3
Have you used BOOLE-DEUSTO regularly during this course?	2,2

Table 2Results of BOOLE-DEUSTO survey

From this survey we can conclude that students regard it as a powerful and pedagogical tool, but that they do no yet use it regularly as a learning aid. The fact that most students are not accustomed to autonomous learning might be the reason for the limited use of BOOLE-DEUSTO, although we can probably expect this to change with the Bologna Process.

3.1 Use of BOOLE-DEUSTO outside the University of Deusto

BOOLE-DEUSTO is used not only in other Spanish universities, but also in several centers around the world (specially in Latin America). Although no formal studies have been conducted in those centers, the feedback we have received so far from instructors has been positive.

4. COMPARISON WITH A WELL-KNOWN DIDACTIC ENVIRONMENT: LOGIC-AID

LogicAid, developed by the University of Texas at Austin, is another well- known software package which is suited for its use in digital electronics courses. Both BOOLE-DEUSTO and LogicAid are high quality teaching aids, yet they do not provide the same functionality. The goal of this paper is to analyze their differences between them and see how each could benefit from the other's features.

4.1 Suited for digital electronics education

Both software packages have been designed from the start to be used mainly in a classroom setting, and therefore are well suited as an aid for digital electronics courses. Both programs feature a user-friendly interface which is easy to work with.

4.2 General strategy

One of the primary concerns when designing BOOLE-DEUSTO was that the user should be able to use different representation methods (see 3.3) interchangeably with ease, enhancing the notion that several representations (a truth table, a circuit diagram, a Veitch-Karnaugh map) can all represent the same system. In effect, the user can input a system using a boolean expression, and then instantly see how that system would be represented with a VK map, in canonical form, etc. As mentioned previously, this makes BOOLE-DEUSTO act as a 'boolean calculator', where the user can input the description of a system and then experiment with it, the same way a high school student can experiment with all the features of a scientific calculator.

On the other hand, although LogicAid does allow the use of different representation forms, passing from one form to another one is cumbersome (the user has to save the system to a file and reopen it from the other representation form). The user generally has to stick to one representation method at a time, which makes the learning process too guided and not very open to experimentation.

4.3 Forms of representation

BOOLE-DEUSTO and LogicAid both allow the user to work with the most common forms of representation used in digital electronics (truth tables, Veitch-Karnaugh maps, canonical forms, etc.) However, BOOLE-DEUSTO has a more complete and versatile set of representation forms. The following are some noteworthy features of BOOLE-DEUSTO which cannot be found in LogicAid:

- Input a digital system in the form of any boolean expression, using any combination of AND, OR, XOR, NOT and parenthesis operators.
- The Veitch-Karnaugh module can also represent any system (LogicAid can only show systems with up to 5 variables).
- Circuit diagrams can be generated from the system description (both combinational and sequential)
- Sequential systems can be simulated graphically (using the FSM diagram) once they have been specified.

4.4 Internal algorithms

LogicAid features much better internal algorithms than BOOLE-DEUSTO. While the latter only offers a simple (yet exact) single-output Q-M variant for boolean minimization, LogicAid offers several minimization methods, including the well-known Espresso method, which can perform multiple-output minimization.

4.5 Export formats

Both programs export combinational and sequential systems to well known formats. However, LogicAid currently only exports systems to JEDEC files, while BOOLE-DEUSTO can also generate OrCAD-PLD and VHDL code.

4.6 Bit-level and word-level

Neither BOOLE-DEUSTO nor LogicAid work at word-level, since they can only perform bit-level analysis and design. However, LogicAid includes a partner program called SimUaid that allows the user to perform word-level analysis and design by directly editing and simulating logic circuits.

Feature	BOOLE-DEUSTO	LogicAid
General learning strategy	Boolean calculator	Guided learning
Representation methods	Many	Few
Internal algorithms	Normal	Excellent
Export formats	JEDEC, PLD, VHDL	JEDEC
Level of operation	Bit-level	Bit-level and word-level (SimUaid)

Table 3Summary of comparison between BOOLE-DEUSTO and LogicAid

5. CONCLUSIONS

This paper has shown that the features found in BOOLE-DEUSTO make it a complete, didactic, and even professional environment for bit-level analysis and design of combinational and sequential digital systema. Its pedagogical focus, providing the students not only the result to a problem but also a clear idea of the process used to arrive at that result, make it an ideal teaching aid for Digital Electronics courses.

When compared to an existing didactic environment for Digital Electronics, professor Roth's LogicAid, we believe BOOLE-DEUSTO not only matches some of LogicAid's features, it also surpasses quite a few of them. BOOLE-DEUSTO could easily become the reference software teaching aid for basic Digital Electronics courses.

We realize that BOOLE-DEUSTO can still be improved in several ways, specially since users (both students and teachers) always point out new features they would like to see in BOOLE-DEUSTO. However, in our opinion, we think that BOOLE-DEUSTO has reached a point where it has all the basic elements required by most users, which is why we have decided to finally put the project to rest. We have so far been able to be true to the project's original spirit: simple, powerful, and easy to understand. However, we believe that adding more features would make BOOLE-DEUSTO stray from pedagogical to professional, which is not our goal specially considering that there are already several high-quality environments like Xilinx, Lattice, Altera, etc. which students can use to perform professional work.

References

[1] García Zubía, J. y Sanz Martínez, J., “Adecuación de los entornos computacionales a la clase de electrónica digital: BOOLE-DEUSTO”, *Actas del IV Congreso de Tecnologías Aplicadas a la Enseñanza de la Electrónica*, *TAEEE 2000*, pp: 573-576, Ed. UAB, ISBN: 84-600-9596-7, 2000.

[2] Angulo Usategui, J.M y García Zubía, J.,*Electrónica digital y tecnología de computadores*, 2º edición, Ed. Paraninfo, ISBN: 84-9732-042-5, 2003.

[3] García Zubía, J., Sanz Martínez, J y Sotomayor Basilio, B.,*BOOLE-DEUSTO v 2.0 Entorno de Diseño Lógico*, Ed. Universidad de Deusto, ISBN: 84-688-2621-9, 2003.

[4] García Zubía, J. “Educational software for digital electronics: BOOLE-DEUSTO”,*Proc. 2003 Int. Con. on Microelectronic Systems Education*, pp:20-22, Ed. IEEE Press, ISBN: 0-7695-1973-3, 2003

[5] Roth, C.H., *User´s guide and reference manual for LogicAid*, 2º edición, Ed. Thomson Learning, ISBN:0-534-38474-92002.