KARNUMA AN EDUCATIONAL TOOL FOR DIGITAL SYSTEMS

Valentim Freitas and Fernando Morgado-Dias

Madeira Interactive Technologies Institute and Centro de Competências de Ciências Exactas e da Engenharia, Universidade da Madeira
Campus da Penteada, 9000-039 Funchal, Madeira, Portugal.
Tel:+351 291-705150/1, Fax: +351 291-705199
morgado@uma.pt

Abstract: This paper presents an application called *KarnUMa* which aims to demonstrate the operation of the Karnaugh Maps which are used to simplify Boolean algebra expressions. This application is available in two variants with different target platforms, the first, *KarnUMa* available for computer and the second *Pocket KarnUMa* available for mobile terminals in either Midlet or Android Package. This paper addresses the existing applications in this area that served as inspiration, the technologies chosen for the development of this software, and finally it presents the *KarnUMa* applications identifying the new features introduced by them as well as new approaches that were made to already existing features in the reference applications. . Copyright CONTROLO 2012.

Keywords: digital systems, computer software, educational aids, e-learning.

1. INTRODUCTION

In the last decade there has been a constant technological evolution in what regards computers and mobile terminals. As such it became advantageous to use these new technologies, which have become part of our everyday life, as sources of information for education. These technologies have shown great potential when harnessed for pedagogical proposes hence the emergence of concepts such as E-Learning and M-Learning. E-Learning is a term used to refer to "electronic learning" (Bowles, 2004), i.e. when there is the use of electronic technologies for educational purposes, on the other hand when talking about M-Learning (Mobile -Learning) we refer to the use of mobile terminals to this same end, therefore M-Learning can be defined as "educational source, where a single technology or dominant technologies are handheld devices" (Traxler, Defining Mobile Learning, 2005).

The work presented in this paper is a part of a master's thesis in Computer Science in which the purpose was to create an application to use as a component of E-Learning, since this application would aim to aid learning the Karnaugh Maps tool used in the subject of Digital Systems.

Aiming to help teach the use of Karnaugh maps tool, the application should not limit itself to presenting results, but be able to demonstrate how to achieve these results. With this in mind the application should be able to display the whole process of solving a Karnaugh Map as well as providing detailed information for each step of this process. Another important aspect for teaching is the practice, as such the application should enable the user himself to perform the Karnaugh Map resolution process. Hence the motivation to develop a new application of Karnaugh maps lays in providing software from which the user could learn to use this tool and also put his knowledge to the test.

2. STATE OF THE ART

In a first stage of this work the aim was to check all the existing applications within the same area in order to determine what has already been achieved. This process consisted in testing the multiple applications found to identify features that made them stand out from each other giving primary importance to the ones that had some educational component. After this stage the following applications have been identified as reference cases to the development of the *KarnUMa* applications.

2.1. Karnaugh Minimizer

The application shown in Fig. 1 is available in both free and commercial releases whereas the last one offers more features. Since the applications presented in this article are intended to the academic community, only the free version of the application developed by (ShurikSoft, 2008) was taken into consideration. The Karnaugh Minimizer was regarded as a reference application for its wide range of features among which the following ones should be highlighted:

- Allows the introduction of the Karnaugh Map's content by the map itself and also by a truth table;
- Indifferent terms can be used;
- Solves maps up to four variables;
- Provides solutions using the Sum of Products method and the Product of Sums method;
- It has options to assist the map's filling such as inverting its content or deleting more than one cell at a time;
- It generates a report with the relevant information from the Karnaugh map such as the initial function, the Karnaugh Map itself, the Truth table and the map's Solution with the representation of each minimized function's term.
- It generates the circuit schematics corresponding to the solution of the map.

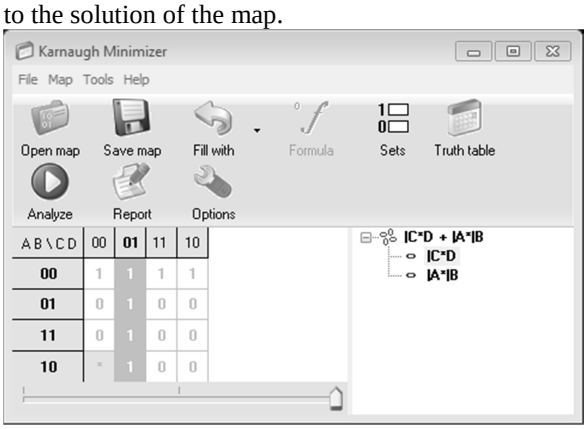

Fig. 1. Karnaugh Minimizer (ShurikSoft, 2008).

At the time of this paper's writing this software was in the 2.0 release.

2.2. EasyKarnaugh

The approach taken to Karnaugh maps by (Sena *et* Torres, 2009) is represented in Fig. 2. This approach stands out from the previous application for being targeted for education. This application aims to teach its users how to solve a Karnaugh map. Taking into consideration the purpose of this application it can be said that its design was made with the objective of E-Learning given that the description made by the authors of the application is as follows "A computational tool to aid in the teaching of Karnaugh maps in Computational Logic" (Sena *et* Torres, 2009). This application has four main components:

- Presentation of Karnaugh Map's Theory;
- Solution's visualization in which the solution's expression is shown as well as each Implicant is properly marked in the Karnaugh map;
- A component that allows the user to try to deduce the Boolean expression that corresponds to the Karnaugh map' solution;
- A component in which the user tries to form the solution's Implicants of the Karnaugh map. A fault was detected in this section since it only accepts Implicants that are part of the solution set by the computer, so when there's more than one solution to a Karnaugh Map the program doesn't accept Implicants that only belong to an alternative solution.

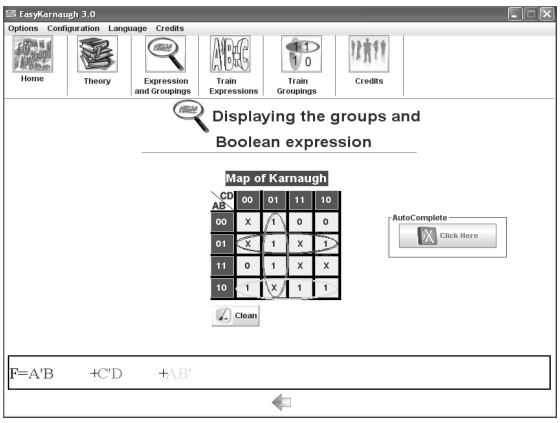

Fig. 2. EasyKarnaugh (Sena *et* Torres, 2009).

This application solves maps from 3 to 4 variables, works with both methods Sum of Products and Product of Sums, supports terms with indifferent value and also has the functionality of filling in the map's content automatically, an appreciated feature when you just wish to test your knowledge of solving Karnaugh maps.

At the time of this paper's writing this application is at version 3.0.

2.3. Karma

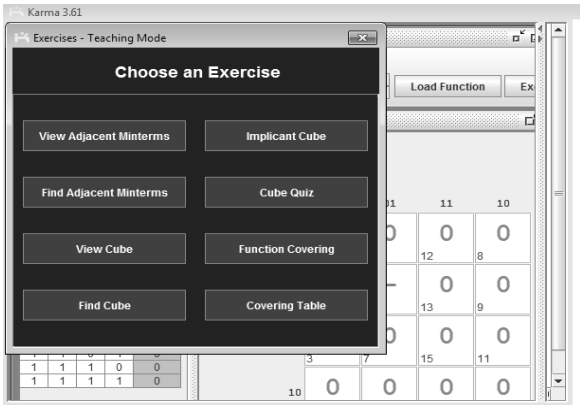

Fig. 3. Karma (Reis *et* Ribas, 2011).

The Karma application developed by (Reis *et* Ribas, 2011), is a powerful application that includes among its features a component devoted to Karnaugh maps. This component supplies the user with the Karnaugh Map's solution with the usual data like the solution's boolean expression, the properly filled map, but it also provides other detailed data as the Quine

McCluskey process steps and its Prime Implicants Table.
Within all its features the most relevant to this study consists of a series of exercises that this application provides to its user. This feature offers a considerable variety of good exercises for the user to test his knowledge. This feature provides eight different types of exercises as seen in Fig. 3
At the time of writing this article this application was in its 3.61 version.

3. CHOSING TECHNOLOGIES

Considering that the outcome of this work is intended to be used as an auxiliary tool for learning the choice of technologies to use focused on the following criteria:

- The technologies used shall be part of the intended community's everyday life and the choice should focus on the one that is available to the highest possible proportion of individuals in this community;
- Acknowledging that this is an academic community, the chosen technologies should be based on the principle of Free Software. Thus the developed applications should not require pre-installation of any commercial software.

In order to meet the above criteria the chosen technologies were as follows:

- **Computer Application –** For this platform, the application was developed in Java. Currently, this technology is found in Release 6. The choice fell on Java because it offers versatility, efficiency and portability. In order to run *KarnUMa* on a computer it only requires an installed Java virtual machine, known as Java Runtime Environment (JRE). Note that this technology is free (ORACLE, 2012).
- **Mobile Application –** For this platform the choice fell on the Java MicroEdition (ME). This (ORACLE, 2012) technology is currently on its 3.0.5 version. The main factor for choosing this technology lies in the fact that currently, "around 80% of all mobile handsets worldwide support the mobile Java standard (Java ME formerly known as J2ME)" (Balaz *et* Kromp, 2011). Yet in order to ensure greater accessibility of the application, it was adapted to Android due to the high growth this technology has recently sustained. Current Android version is 4.0 (Google, 2012).

The development of the *KarnUMa* applications in both PC and handset version required the usage of source code obtained from (LiteratePrograms, 2010) to implement the method of Quine Mckluskey, method which is used as background for the Karnaugh map's resolution.

4. KARNUMA APPLICATION (PC VERSION)

This section contains a description of how *KarnUMa* application works.

4.1. New Map

When starting the application, a form is available to the user where he can choose the Karnaugh map's specifications. Fig. 4 shows the available options on this form:

- **Number of variables –** The user can choose from 3 to 6 variables;
- **Method of resolution –** The user can choose between Sum of Products and Product of Sums;
- **Order of input variables –** Lets the user choose the map's orientation;
- **Filling method –** To choose between introducing the Karnaugh map's content by the map itself or by a Truth table.

Once the Karnaugh map's specifications have been chosen, the user can proceed to the next screen by pressing "Begin" at the bottom of the screen. Note that the user could simply leave the default specifications.

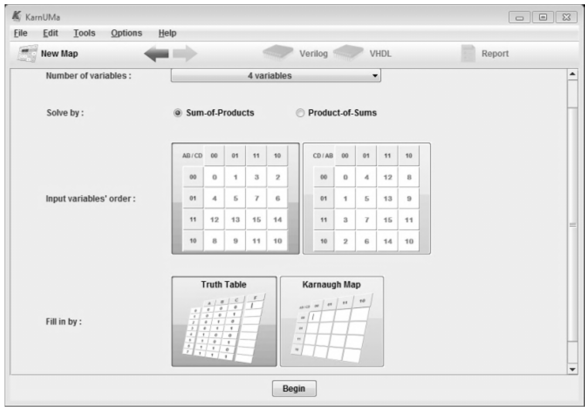

Fig. 4. *KarnUMa*'s new map screen.

4.2. Filling the map's content

After choosing the specifications in the previous screen (Fig. 4) the user will insert the Karnaugh map's content. This filling can be done in the map itself or through a Truth table as shown in Fig. 5.

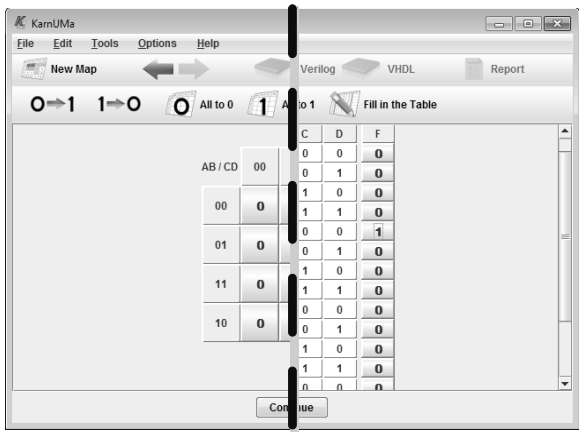

Fig. 5. Filling the Karnaugh map either by the map itself (right) or by Truth table (left).

Regardless of the chosen input method the procedure is the same, the user just needs to press the map or

table cells he wishes to change the value, or choose any of the available options to quickly fill the map or table:

- **0 → 1 –** This option allows the user to modify all terms of value 0 to 1;
- **1 → 0 –** Makes exactly the opposite of the previous option, i.e. modifies all terms of value 1 to 0;
- **All the 0 –** Sets all terms to 0;
- **Everything for 1 –** Sets all terms to one;
- **Automatically Fill –** Fill the entire map or table randomly.

4.3. Choose between learning and testing knowledge

In this section the screen is divided, on the left there's the Truth table, and on the right side the matching Karnaugh map. Here the user can choose from four available options described in the following paragraphs.

4.3.1. View Solution

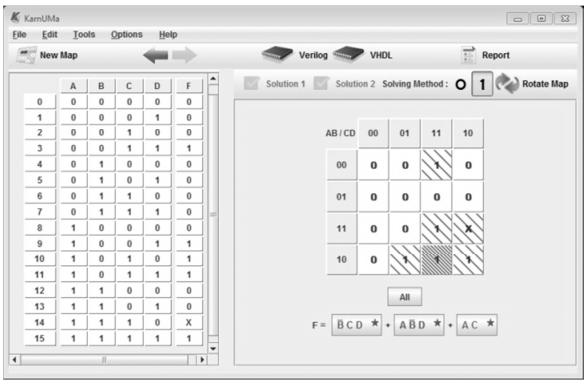

Fig. 6. View solution.

Once the user chooses this option the application presents him with the Karnaugh map's solution has seen in Fig. 6.

When viewing the solution a set of new features is available. On the map's toolbar the user can choose the following features:

- When a map has more than one solution the user may watch one of these solutions at a time simply by pressing the desired solution ("Solution 1" or "Solution 2");
- The solution can be viewed by the Sum of Products or by the Product of Sums by pressing buttons "1" and "0";
- The user can also change the map orientation (this option is always available for as long as a Karnaugh map is shown).

In this application's section there are yet another set of important features available on the top toolbar:

- From the buttons "Verilog" and "VHDL" the user can get the solution's source code in the respective Hardware Description Language;
- By pressing "Report" the user may also get a full report with all Karnaugh map's relevant information.

Finally it remains to be said that below the Karnaugh map is the solution's boolean expression in which there is a correspondence between the Implicant's color in the expression and its representation on the map. The user can click on each Implicant to see it exclusively represented on the map. To redisplay the representation of all Implicants simply re-press the Implicant that is selected or press "All".

4.3.2. View Solution (in detail)

This option allows a detailed view of how to get the map's solution. It consists of several steps where each step corresponds to one of the needed stages to reach the Karnaugh map's solution. As can be seen in Fig. 7, this option displays the screen organized as follows, on the right side is the description of the current stage of the resolution process. At the bottom of the description is usually a button that allows for the definition of the main element of this stage of resolution. On the left side is the map depicting the current step of the resolution.

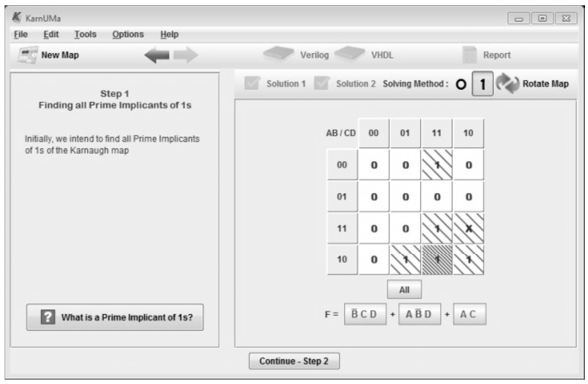

Fig. 7. First stage from "View Solution (in detail)".

This option typically comprises the following steps:

1st – Presentation of all Prime Implicants;

2nd – Identification of which Prime Implicants are Essential. When the Karnaugh map's solution lacks of Essential prime Implicants the user is properly informed and the application proceeds to the next step;

3rd – Of the remaining Prime Implicants (those that aren't Essential), the ones necessary to obtain the final solution are selected. This step may also not be performed, in this case when the solution consists solely of Essential Prime Implicants. The user is informed and the final solution presented.

Once this step by step solution process is finished, the application ends on the same screen as described in section *4.3.1*.

4.3.3. Solve Map

This option presented in Fig. 8, allows the user to solve the Karnaugh map quickly, i.e. in a single step. For this the user must form Implicants until he achieves the solution established by application or an equivalent one. To form an Implicant the user must select the map cells he wants to constitute the Implicant and then press "Validate Implicant" where

the application will validate it. If it isn't well-formed the application informs the user of the errors found.
In this mode you should get exactly the Implicants needed to achieve a minimal solution, to help in this matter there's a feedback bar at the bottom of the screen that indicates how many Prime and Essential Prime Implicants are needed to achieve the minimal solution. This bar is filled in accordance with the user's progress, and once it is completely filled the "Check Solution" option is unlocked.
The most important aspects in this section consist of the feedback bar mentioned above and in the fact that the application accepts any minimal solution, not being restricted to the solution previously established by it.

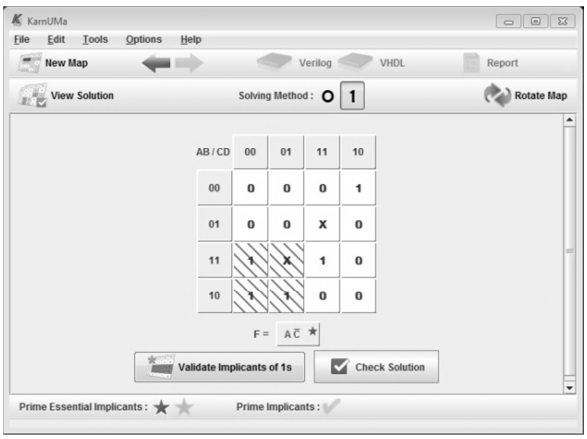

Fig. 8. Solve map.

4.3.4. Solve Map in detail

This user resolution mode follows exactly the same steps described in section *4.3.2*, but this time it is the user that must complete each step. This allows him to test the knowledge acquired in *4.3.2*, knowledge which is relevant for obtaining a correct Karnaugh map's Solution.
This mode also counts with the same feedback described at point *4.3.3Solve Map*, but this time with the proper elements of each resolution's step. Once finished every step, it becomes possible to validate the final solution.

4.4. Other important details

Other relevant features which the application provides are the availability of two languages (Portuguese and English) that can be accessed from the Options menu, and the ability to Redo and Undo, two features increasingly common these days but always important.

5. *POCKET KARNUMA* APPLICATION (HANDSET)

This version of the application is an adaptation of the PC application. As such it consists of the counterpart features that could be implemented and that are relevant in this platform. Since these are the same features their description in this paper's section will be more succinct.

5.1. New Map

As can be seen in Fig. 9, this screen provides exactly the same Karnaugh map's specifications as the previously displayed version, with the only exception to the number of variables which in this version can only be 3 or 4. This difference is due to the fact that the screen dimensions don't allow a good representation of more than four variables in the majority of handset terminals.

Fig. 9. (Left) New Map (Pocker *KarnUMa*)
Fig. 10. (Right) Solve map.

5.2. Filling the map's content

The screen shown in Fig. 11 covers exactly the same options described in *4.2Filling the map's content* of the PC application.

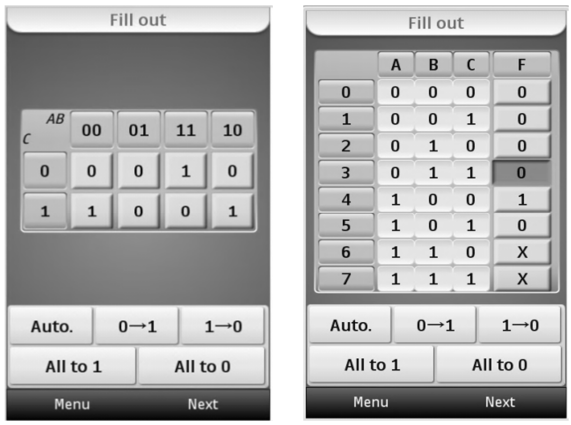

Fig. 11. Fill out the Karnaugh map.

5.3. Solve or get the solution

Unlike the PC version this one doesn't have a major educational component since this section only provides the option to view the solution and solve the map. Due to their complexity the options to see the detailed solution and solve in detail aren't included in this version.

5.3.1. View Solution

Again, this option has the same features described in section *4.3.1*, except for the possibility of obtaining the HDL source code and get a Karnaugh map's report. These features were not included since they were not of importance in this version of the application.

5.3.2. Solve Map

This option operates in the same way as reported in *4.3.3Solve Map*. As can be seen in Fig. 10 there is a different organization of the display since the feedback bar now placed above the Karnaugh map and due to its small size each bar element's description can only be seen in the Help menu.

5.4. Other important details

Similarly to what happens with the PC version, this one has the same two languages and allows Undo and Redo actions.
In addition to the features not included in this release, the most important difference lies in the different organization of the screen due to its small size. The differences in the organization focused mainly on diverting some options to the menu and also removing some descriptions that now can only be viewed in the Help menu.

6. CONCLUSION

KarnUMa applications unlike another existing applications emphasizes the Essential Prime Implicants identifying them unequivocally. These Implicants are essential to the resolution of a Karnaugh map, because without them it is not possible to obtain the minimal solution of a Karnaugh map .
Another important feature of this application is that it provides all the steps constituting the Karnaugh Map solving process, adding even the opportunity for the user to solve it the same way, which reinforces the educational component.
The application *KarnUMa* is capable of delivering up to two map's solutions and check for any alternative solution, since it checks if the alternative solution is whether identical to the two solutions found by the application or equivalent.
Considering the other features that were already present in existing applications, the application tries to make these features more complete and intuitive, such as the resolution by the user in which was introduced a feedback bar, which allows the user to become aware of his performance in this process. Still in user's resolution mode, this application is able to identify the mistakes made by the user and provide tips in order to avoid making the same mistakes in the future attempts.

ACKNOWLEDGMENTS

The authors would like to acknowledge the Portuguese Foundation for Science and Technology for their support for this work through project PEst-OE/EEI/LA0009/2011.

REFERENCES

Balaz, A., *et* Kromp, J. (2011). *Don't Panic Mobile Developer's Guide to the Galaxy* (9th ed.). Sogestrass, Breman, Germany: Enough Software.

Bowles, M. (2004). *Relearning to E-Learn - Stratagies for electronic learning and knowladge* (Vol. Cap. 1). Merbourne University Publishing Ltd, Polar Design Pty Ltd.

Google. (2012). *Android 4.0 Platform Highlights*. Retrieved February 16, 2012, from http://developer.android.com/sdk/android-4.0-highlights.html

LiteratePrograms. (2010). *Quine-McCluskey algorithm (Java)*. Retrieved February 22, 2012, from http://en.literateprograms.org/Quine-McCluskey_algorithm_(Java)

ORACLE. (2012). *Java*. Retrieved February 16, 2012, from http://www.java.com/pt_BR/about/

ORACLE. (2012). *JAVA FOR MOBILE DEVICES*. Retrieved February 16, 2012, from http://www.oracle.com/technetwork/java/javame/javamobile/overview/getstarted/index.html

Reis, A. I., *et* Ribas, R. P. (2011). *Karma*, 3.61. (Logic Circuit Synthesis Labs, Instituto de Informática, Universidade Federal do Rio Grande do Sul, Brazil) Retrieved October 2011, from http://www.inf.ufrgs.br/logics/index.php?option=com_docman&task=doc_download&gid=202&Itemid=58

Sena, A., *et* Torres, M. (2009). *EasyKarnaugh 3.0 - Uma ferramenta computacional para o auxílio no ensino de Mapas de Karnaugh em Lógica Digital*. Universidade Estadual de Santa Cruz, Departamento de Ciências Exatas e Tecnologias, Brazil.

ShurikSoft. (2008). *Karnaugh Minimizer*, 2.0. (Shuriksoft) Retrieved November 2010, from http://karnaugh.shuriksoft.com/

Traxler, J. (2005). Defining Mobile Learning. *International Conference Mobile Learning*. Wolverhampton, United Kingdom: IADIS.