An Educational Perspective at Digital Circuit Elements

Student ID: 1059430

Abstract

This paper considers application of Empirical Modelling principles to a system implementing logic gates in an interactive fashion. This implementation of logic gates is aimed for educational purposes. The user can interactively toggle the input values and the output is modelled as a dependency. Also, this model dynamically creates truth tables.

1 Introduction

Today's teaching has become more oriented with delivering information rather than understanding, interacting and experiencing the information being delivered. This paper discusses the merits of adopting an empirical based approach to overcome these shortcomings of teaching today.

The Empirical Modelling based model aims to fill these voids in teaching. Boyatt says that Learning is aided by exploration and an EM based model offers exactly that (Boyatt). It is believed by this a student can gain a better insight the subject of study using this approach.

This paper aims at using another approach to the work previously done in the ELS model (Warwick, 2007). The ELS model aims to create sophisticated circuits while this model aims to facilitate learning and understanding of the basic circuit elements by employing a truth-table that populates itself dynamically and also some text that would help the student gain a better understanding on the matter.

In EM, the end user can be the modeller. This adds additional flexibility and personalization power to the model. Assume, that a student is particularly interested in the circuits within a particular gadget, he/she has the freedom to model the dependencies and personalize the digital circuits model to their interest in these EM based models.

Chen 2007 suggests that "With conventional programming, all the possible states are considered to be preconceived at the start of the development, but this is not appropriate for reactive systems;" this in itself is a massive advantage EM has over the standard programming tools.

Another feature of EM that helps closely relate to the real world id that it is an observation-oriented state-based modelling framework (Roe, 2003).It helps include the 'what if' functionality in the model (Gooding, 1990).

2 Digital Circuit Elements and EM concepts

It is apparent that EM is closely associated with Digital circuit elements due to the various clearly notable observables and dependencies. This model can be used as a tool to help students think about challenges they might face with understanding the working of digital circuit elements.

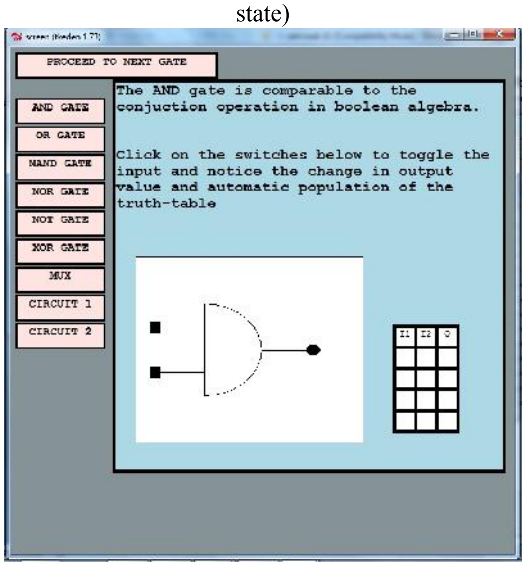
The key factor is to understand the working of the individual elements clearly and then understand how they work with one another. An understanding of these circuit elements are important for computer science, engineering and electrical students as these elements form the building blocks.

In the long run, this tool can be hoped to clearly model and troubleshoot problems in complex gadgets. Of course, in order to be able to do this, a number of complex dependencies will have to be created which would also include the basic circuit elements modelled here.

3 Digital Circuit Elements Model

This model was created using tkeden 1.73 in Windows. It aims to teach students the basic logic gates one by one and then proceeds to a complex circuit. Here, each of the gates is modelled as a separate layer. Once a student learns about it he/she can proceed to the next element using the 'Proceed to next gate' button on the top right of the screen as shown in the screenshot below. In the end, the model puts forth two complex circuits assuming that the students understand the working of the individual components. Apart from this, an extra button panel is also added towards the left for the ease of use of the student. On clicking the AND gate which is the first here the counter is reset to guide the student through all the steps again.

Figure 1: Screenshot of the model (start state)

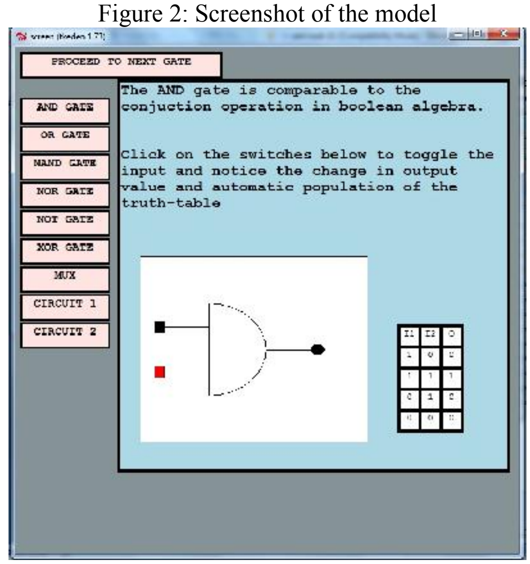


As shown in the above screenshot, this model gives the user an idea about the logic gate that is being shown. Next, by clicking on the switches of the gate as instructed in the text above, the user can understand and learn the functionality of the circuit element whilst witnessing the truth-table populate as shown in figure 2.

A widespread usage of macros is involved in this model to be able to create new scout windows in the EDEN perspective. Also, extensive use of DoNaLD type windows is employed. Majority of the objects here are of the above mentioned type.

There are many observables with respect to each circuit element namely inputs, outputs and a selector in the case of a multiplexer. These are shown as the colour of the LEDs and switches in the model. Red depicts on (or value 1) while black depicts 0. The inputs can be toggled by clicking on the switches in the DoNaLD window. Depending on the value of these, the output led's value changes. This change in value is represented as the colour of the led and the output value which will be displayed in the truth-table.

Figure 2: Screenshot of the model



Another feature of EM that is made use of in this model is the ability to track changes made to *andRes* variable which stores the result of the and gate. Here if the user enters any values other than 0 and 1, the model recalculates the output and returns to the stable state. While if the user modifies the output manually and the inputs are not in agreement with the entered output value then, a message is displayed which says the gate is dysfunctional and the led turns green to indicate the same. These are cases which cannot be dealt with in the structured programming approach. This clearly shows the consideration of the 'what if' conditions mentioned in Section 1. This feature has been implemented for all individual circuit elements.

A feature that could be added to EDEN is a method to track the medium by which a variable is changed. For instance, in the model implemented here, if the input values change through the use interface by interacting with the switches, then certain actions need to be taken while otherwise the action needs to be different. By having this distinction, some repetitiveness in the code can be reduced. This is the feature that provides ease of redefinition also so a balance must be struck between the two such that both exists in a complementary manner.

4 Learning

The model created here is aimed for educational purpose has advantages of flexibility and the capability of personalization or tailoring to one's own interest.

For instance, if a student would like to know how a gadget would behave if a certain circuit element was not functional this tool can be very helpful as it saves both time and hardware cost. Nevertheless, we will have to understand that the student may have to invest some time initially to understand and use EDEN definitions to be in a position to fully exploit the advantages of the model.

The model is not static that is the model can expand as the user's understanding, experiences and learning increases. This is one of the quintessential features of Empirical Modelling approach that make this approach desirable and a way of modelling for the future.

This feature of expanding with the growth of the user enhances the experience of the user and the ability to explore more topics in the desired subtopic. The teacher adapts herself according to the level of the student and in this case the model adapts and expands with the knowledge of the user. This is a very human-like approach to learning which models developed by traditional programming techniques cannot achieve.

5 Advantages and Disadvantages

5.1 Advantages

5.1.1 Flexibility

The world is a rapidly changing market with respect to hardware design. Let us assume, a company invests in creation of a model and it follows a standard programming approach. To keep up with the changing trends, the company will have to return to the developers for any amendment or addition that needs to be made. This is because the developer may use certain variables which may be very elusive to the end user. Hence, extra cost will be incurred and the software would be inflexible.

On the contrary, if the company were to adopt a empirical modelling based approach for the creation of the model then the users would be guaranteed flexibility to change behaviour and add features as and when the need arises for the same. Therefore, the model progresses with the user.

5.1.2 Ease of redefinition

An observer can be a student or the modeller. In a situation when the behaviour needs to be redefined, the program need not be stopped. This redefinition can happen at run-time itself. The new definition makes the old definition obsolete and hence deletion of the old code is not necessary.

5.2 Disadvantages

5.2.1 EDEN based issues

The disadvantages of using this approach as EDEN tools are still in their nascent stage when compared to its highly evolved structured programming languages like Java. The extensive use of Java is the reason for its highly sophisticated and efficient memory management system.

In EDEN, the processing speed and memory management are its limitations. These shortcomings can be attributed to the limited usage and testing of it.

Creation of truth-tables may seem to be a very trivial job from the perspective of the standard programming languages however, with EDEN this process was very time intensive as each cell needed to be modelled as a separate TEXT window each with its own declarations and attributes making the code repetitive and thereby increasing the number of lines of code.

Even though this model uses tables of limited size, it would be desirable to extend this work so as to create truth tables for complex circuits also. For this purpose, a suggestion would be introduction of a new MATRIX type window which would help to create these rows and columns with relative ease as tables are a part of everyday usage especially when one thinks of the EM model to be a learning tool.

Another disadvantage of using EDEN was that lack of stability of EDEN. When rendering an image type scout window, most times EDEN crashes. Therefore, to effectively render this type of scout window, EDEN needs to be restarted many times.

This is a cause of great inconvenience to the user and the modeller.

5.2.2 User based issues

Another limitation was the limited knowledge of the EDEN language syntax and working. This made the process of implementation challenging. Also, the various different components within EDEN each have its own unique syntax. This can be very confusing for a new user. A possible solution for this could be standardization of syntax.

Another roadblock was the limited examples in the help material. It may be said that there are innumerable models online but the effort in understanding a model to learn the syntax and working can be a very tedious process.

6 Future Work

Due to time constraints, all possible dependencies are not implemented here. For instance, in the complex circuits, the output of the first element could be taken as the input of the next as in the ELS model. This can be hoped to be implemented in future work.

7 Conclusion

There is a close association between the EM principles and Logic circuit elements. This makes EM based model a perfect match.Though to effectively use an EDEN model for teaching, the student's construal need to be considered while modelling.

This along with the flexibility of the model would be a great teaching tool. Despite its advantages, we need to understand that modelling using these tools may not be very easy for an end user as these tools are in their nascent stage. To employ a widespread use of EM tools, the first step is that these tools need to be made both easier to use and reliable.

Personally, I believe that the only hindrance to the widespread adoption of this technique is the quality of the tools. The improvement along the lines mentioned above would immensely help in increase the popularity of EM approaches.

Acknowledgements

To Meurig Beynon without whose help and support implementation and understanding of the Empirical Modelling concepts and EDEN would have been far from reality.

References

D. Gooding. Experiment and the Making of Meaning, Kluwer (1990)

Beynon, W. M., "Empirical Modelling for Educational Technology", ONLINE, <http://www2.warwick.ac.uk/fac/sci/dcs/research/em/publications/papers/downloads/047.pdf>

Chen, Yu-Yang, "An investigation of Empirical Modelling artefacts and its advantages over traditional modelling for real-world activities", ONLINE, <http://www2.warwick.ac.uk/fac/sci/dcs/research/em/publications/web-em/04/wii-pendulum.pdf>

Roe, C. (2003). Computers for Learning: An Empirical Modelling perspective. Doctoral dissertation, Department of Computer Science, University of Warwick

Warwick, S. a. (2007). ELS – An Eden Based Digital Logic Simulator. The Third Warwick Empirical Modelling Bulletin (WEB-EM-03)

Boyatt, Russel. Learning about and through Empirical Modelling, ONLINE http://www2.warwick.ac.uk/fac/sci/dcs/research/em/publications/papers/downloads/090.pdf
Access Date 30/01/2011