

Computational Learning Objects Using Interactive Spreadsheet-Based e-Forms

Abdullah A. S. Basuhail*

Computer Science Department, Faculty of C&IT, King Abdulaziz University, Saudi Arabia

Abstract— This paper presents a methodology that can be used to help instructors in developing educational e-learning objects by the aid of interactive e-forms on their owns. In turn, these objects can be exploited to assist the learners for better understanding and exploration of scientific and technological concepts. Spreadsheet-based e-forms for learning some subjects that are taught in introductory college courses were designed and implemented using this technique. Moreover, advanced spreadsheet features and capabilities were exploited to simulate several scientific notions which are taught to undergraduate students. The paper demonstrates few examples that illustrate implementation using the suggested methodology. The designing method demonstrated in this paper is characterized by simplicity, flexibility, affordability and speed of delivery compared with other implementation techniques. This implementation approach can be used as a framework for structuring similar electronic models related to many disciplines in education, business, science, and technology. Generally, the developed computational spreadsheet-based e-forms can be used by instructors to illustrate topics in attractive and professional fashions in e-teaching sessions. In addition, students and learners can use the developed educational models without instructor supervision in self-education or e-learning environment.

Keywords— Computational learning objects, e-learning, e-teaching, e-content, spreadsheets, interactive learning

I. INTRODUCTION

Scientific problems of computational nature can be made more grasp to students by using modern technologies and tools. One of the common tools for simulation and realization of such problems is computer programs written in high-level programming languages, such as Java, C#, C++, etc. However, this paradigm requires knowledge and experience with the programming techniques, and familiarity with the visualization tools, as well as, it requires dedicating resources, such as time and budget, for the design and implementation.

Spreadsheets are powerful computing tools that are offered for flexible and rapid use. They are used to store data in tables, perform calculations of mathematical and logical nature, and demonstrate data and computation results in assorted formats and styles. Mostly, spreadsheets are used as a computational tool for a subject or a problem to be studied. Such use can offer, to the learners, the chance to examine several cases related to the problem under investigation. Spreadsheets also facilitate exploration of what-if scenarios and build hands-on experience for scientific concepts and facts.

Modern electronic spreadsheets are used in various educational and scientific fields. Their features and capabilities make them attractive tools to solve complex problems in an attainable fashion. They have been used in a wide range of engineering and technology applications [1]–[4]. They are used to introduce practical applications to undergraduate college students [5]. Some researchers reported the use of spreadsheets for simulation in diverse scientific fields [6]–[8]. Other researchers used them for implementing, studying, and analysing systems and models [9]. Still some researchers used electronic spreadsheets for applying mathematics for computer processing purposes [10]. One researcher illustrated the use of spreadsheets in implementing models for simulating some types of computer networks [11]. The motivations of using spreadsheets are: popularity, simplicity, portability, and operability.

The technique of modelling and implementing systems and problems using spreadsheets can be practiced by instructors who have deficiency or who are inexperienced with the implementations using computer programming languages and techniques or using more sophisticated software packages.

Generally, the use of available tools and software packages, such as spreadsheets, can facilitate the design and implementation stages for non-expert developers of e-content [12]. Although there are some features in other electronic developing and implementation packages and tools which are superior to spreadsheets based designs, or they are preferred by some users, the popularity of the later one can make the development and use of spreadsheet-based designs as a building process over previous expertise [12].

II. INTERACTIVE SPREADSHEETS

Spreadsheets are a highly flexible programming environment which are widely used in industry and adopted for decision making [13].

Spreadsheet can contain data of assorted types. A spreadsheet cell can be formulated to compute simple or complex formulas and mathematical functions.

Spreadsheet packages are enriched with many built-in functions related to various scientific disciplines and applications. The use of the pre-programmed functions reduces the programming burden needed by the designed task.

This feature encourages the developers to integrate designated functions in their applications without the effort of entering, verifying, and validating those formulas and functions.

Visuals, such as pictures and computer graphics, can also be integrated in spreadsheet-based designs. The use of these objects enhances the look-and-feel user interface for the topic or the problem under design. It can make the designed product more informative and attractive to the learners.

Another powerful spreadsheet feature is representing data using charts with variety of styles. This capability enhances displaying and examining relations among data. They also enable representing output results with the cost of almost few fingertips during the implementation of a computational e-form.

Other spreadsheets features, such as decision-making, sorting, filtering, hyperlinks, conditional formatting, and many other characteristics, facilitate implementing numerous powerful and flexible electronic computational spreadsheet-based educational objects that are suited for the use in various applications and fields.

III. DESIGN METHODOLOGY

The ultimate goal of designing learning object using the technique that will be explained in this paper is to transform a spreadsheet table into an interactive computational e-form for educational and technological purposes.

The technique to implement a computational e-form passes by several stages in order to have control over the design phase. Specifically, these phases can be taken as a guideline for developing e-form based learning objects that range from simple to complex designs using spreadsheet tables. Figure 1 illustrates the stages for the methodology that is applied during the design and implementation of computational e-forms for a learning object.

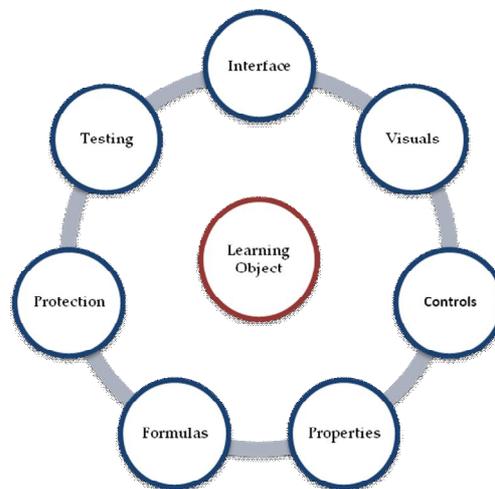


Fig. 1 Stages for developing spreadsheet-based computational learning object

A. Interface Design

This stage deals with the design of a user interface for the educational e-form using spreadsheet table. First, the borders of the spreadsheet cells are made invisible by repainting the table. The locations of the cells which are to be used are refilled to become distinguished and resemble input textboxes (look-and-feel). To facilitate the use of an e-form, it is resized to almost one screen size. Next, proper labels are added next to each used input cell in order to aid the users to enter the precise input values. The overall view of the e-form can be improved by adding proper explanatory titles and exterior borders. Setting borders and shadows to the input-boxes can also enhance the visualization of the designed e-form.

B. Adding Visuals

Integrating graphic and image objects enhances the look-and-feel user interface for the computational designed e-form. It makes the learning object more attractive and informative. Picture and graphical clips are added to the e-form according to the need by the designed topic. Furthermore, drawings are integrated in the e-form such that it imitates the real problem or topic. This stage facilitates the self-use of the learning object, and without an external facilitator.

C. Adding Controls

Controls, such as radio buttons, spin buttons, scroll bars, ... etc, are incorporated in the design of a learning object. They offer flexibility and easiness when using the developed spreadsheet-based learning objects. The controls require programming and linking to the appropriate e-Form components for accurate operation and proper functioning.

D. Setting Properties

Usually, input textboxes represent the locations for entering input data by the users. These fields are considered as the user interaction to a computational e-form. On the other hand, the resulted numerical or textual computations which are to be displayed to the users are the output fields. Setting appropriate properties to both types of fields is crucial. Such properties include field size, data type, number format, etc. Text formats, such as colour, font and size, assist the user in locating and grasping the intended information.

E. Formulas and Built-in Functions

Typically, a computational learning object includes one or more mathematical formulas and/or equations. The use and integration of the built-in spreadsheet functions in a computational e-form helps in designing rapid and accurate implementations with high reliability. Mostly, formulas are entered directly into spreadsheet cells. In some cases, VBA is used to embed sophisticated subroutines in the e-Form, and will be invoked according to the need. This choice is governed by the existence of suited built-in formulas and by the level of complexity of the problem under consideration.

F. Setting Protection

After the e-form design is completed, the entire e-form is secured by protecting all cells except the input fields. The users should interact through the input-boxes and only viewing the other parts of the e-form. Securing e-form protects it from any unintentional alterations or misuse and keeps it intact. This leads to more reliability in using spreadsheet-based computational e-forms designed and implemented for user's interaction systems.

G. Testing

The last stage in the design process that precedes distributing and delivering of a developed computational e-form is to run a thoroughly test on the entire learning object in order to validate the proper operation and functioning of the entire e-form as an integrated whole object. In case of a learning object that have malfunction operation, or being not in accordance with the design specification, the corresponding design stage is revisited and validated to meet the requirements.

The designing methodology explained in the previous sub-sections has been used to implement spreadsheet-based computational e-forms in diverse learning fields related to computer science courses. For the purpose of this paper, the following section will shed the light on two interactive spreadsheet-based models that have been designed and implemented using the pre-mentioned technique. The first implemented model is for one electricity principle. The second model is for a digital circuit that is demonstrated in digital logic design course which is taught by the paper's author.

IV. APPLICATIONS

In this section, the use and application of the previous explained methodology for designing and implementing of spreadsheet-based computational e-form is demonstrated with two illustrative examples. The first example is a learning object for Ohm's law, and the second example is a learning object for a digital logic combinational circuit. Learning objects for other fields of science and technology can be designed and implemented in a similar fashion.

A. Ohm's Law Computational Learning Object

Ohm's law is a popular electric principle introduced to students in introductory physics and electrical engineering courses. Basically, the law illustrates the relation between applied voltages in the presence of an electric resistance which resulted in a current that flows in an electric circuit. Table I list the various components used to model and implement this law using spreadsheet.

TABLE I
 COMPONENTS USED IN OHM'S LAW SPREADSHEET-BASED E-FORM

Object component	e-form component	Use
Battery	graphic-cell	indicate battery life
Battery scroll-bar	control	adjust voltage
Voltage	input-box	voltage value
Resistance	input-box	resistance value
Resistance Spin button	control	adjust resistance
Ammeter	output-box	current value
Text	textbox	topic theory
Graph	chart	graph relations

The next step is to design an e-form that can fit the entire learning object with the various components in accordance with Table I. Figure 2 depicts the computational e-form that is designed and implemented for this purpose. As can be noticed, three clipart drawings were added to the exhibition; specifically, battery image that visually represent the voltage source, resistance image that controls the amount of the flowing current in the circuit, and ammeter image that visually represent the measuring device for the flowing current. Several input boxes, output boxes, textboxes, and controls are integrated with the e-form.

The graphic-cell indicated by the green color inside the image of the battery is changed using the scroll-bar next to the battery. As a result, the voltage level is changed interactively to reflect the battery voltage level as set by the user. The resistance can be controlled by the spin button underneath it.

Once any of the two parameter values, voltage or resistance, is changed by the user, the resulted electric current value is computed and promptly displayed in the ammeter computational cell which is located at the bottom of the ammeter image.

The e-form is supplemented with chart that graphically depicts the relation between the three Ohm's law parameters. This component of the e-form enables the user to interactively observe the relation between the object parameters. Colors

are selected properly in accordance with each component to facilitate the follow-up for the learner and to enhance learning, understanding, and investigating of the subject.

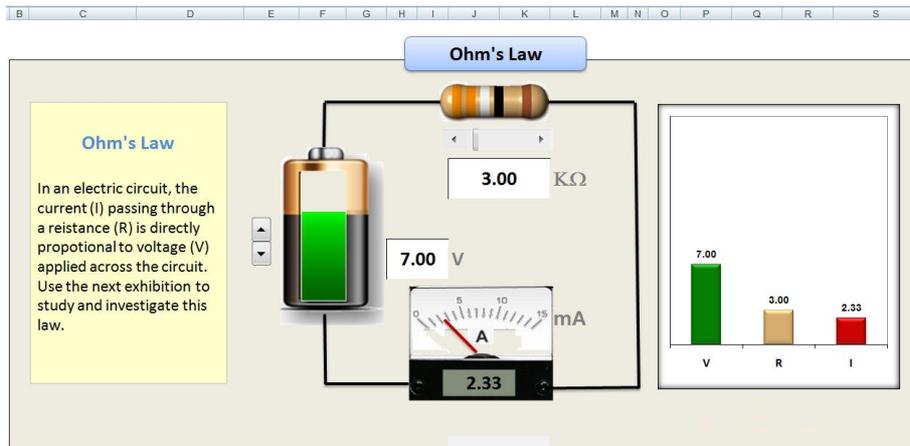


Fig. 2 Ohm's law computational e-form

Moreover, a related text is added to the object as a theoretical background about the topic. Learner who use and manipulate with this e-form can understand and deduce Ohm's law relation in an illustrative and interactive manner.

B. Combinational Circuit Computational Learning Object

Combinational circuits are used in digital logic courses to learn the design and operation of digital systems. The system is composed of several logic gates that are all together perform certain function. The operation requires triggering the system with binary input values until the final binary output is yielded. Figure 2 shows an example of a combinational circuit system.

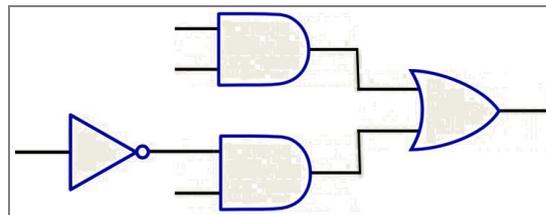


Fig. 3 Combinational circuit for digital system

Usually, the system operation is examined by the means of truth table that traces the circuit from its inputs to the final outputs. The learner needs to trace and perform exhaustive testing in order to find the output values for all possible input combinations. To facilitate the testing stage of this system, a spreadsheet-based e-form is designed and implemented. Table II list the components integrated and added to the circuit to manipulate and test this system.

TABLE II
 COMPONENTS USED IN COMBINATIONAL CIRCUIT SPREADSHEET-BASED E-FORM

Object component	e-form component	Use
4 Gate inputs spin buttons	control	set the inputs
4 Gate outputs	output textboxes	output indicators
Output LED	graphic-cell	LED state indicator
Text	textbox	topic theory

The next step is to design an e-form that can fit the entire learning object with the various components in accordance with Table II. Figure 3 depicts the computational e-form after adding the digital circuit as a graphic and the other components in Table II. As can be noticed, the inputs to the system are applied using the spin buttons located at the inputs of the circuit. The learner can examine any combination and observe the resulted output. The figure shows one case in which the output is logic value 0. The entire circuit output is also indicated by the non illuminated LED.

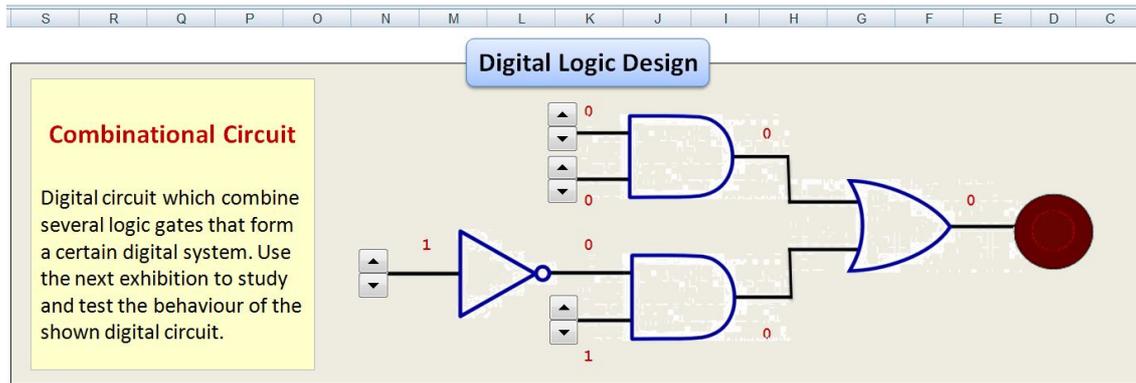


Fig. 4 Combinational circuit spreadsheet e-form

Still another test that is performed to the same design, Figure 4 shows a case with another input combinations which resulted in a logical output of 1, consequently, the LED is turned on and illuminated. Other tests can be investigated by simply changing the input triggers of the system.

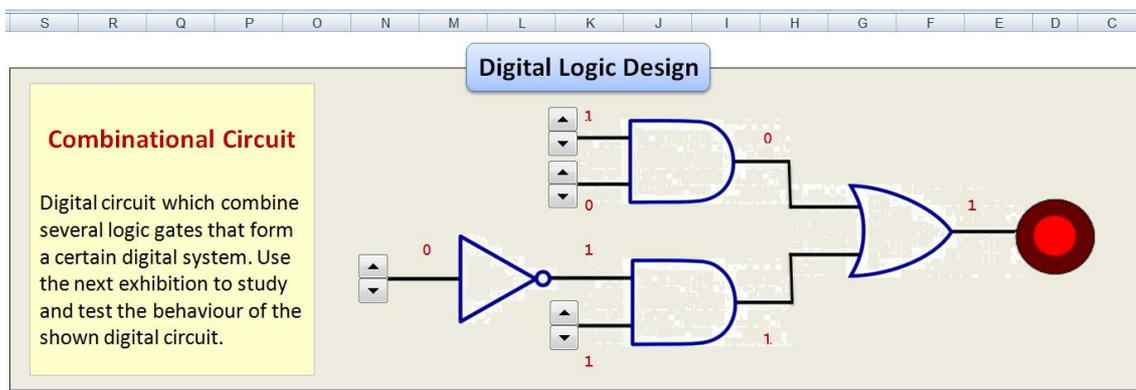


Fig. 5 Illuminated LED due to certain input triggers

This model simplifies the task of learning the system and generally gives the learners hands-on experience for better understanding of the digital combinational circuits. In a similar fashion, more sophisticated and complicated systems can be designed and implemented.

V. CONCLUSIONS

This paper demonstrated a methodology that has been applied to design and implement spreadsheet-based computational e-forms for learning objects that are used to demonstrate educational topics and problems. Two applications have been demonstrated in the paper using the mentioned technique. The implemented examples are both related to the engineering and computer science fields. The same technique could be applied in other disciplines. The designing method can relieve the developers from many of tedious tasks required by other implementation techniques, such as programming and use of sophisticated software packages. The features and capabilities of spreadsheets to construct interactive learning objects considered as powerful characteristics of this technique. Some limitations of using spreadsheets in designing learning objects include the limited animated interaction and less flexibility when integrating images within the computational e-forms.

For future work, other fields of technology can be investigated to develop more general platform for the design of spreadsheet-based computational e-forms for learning objects. Instructors and students will be encouraged to use the same design and implementation technique and adopt it in their implementations in order to perform thoroughly experimental tests for scientific problems that requires exploration of what-if scenarios and build hands-on experience for scientific concepts and facts.

Furthermore, spreadsheet sophisticated features, such as exploiting other controls in the design and utilizing advanced VBA capabilities will also be incorporated for more flexibility and enhancements of using the final developed educational spreadsheet-based e-forms for learning objects.

REFERENCES

- [1] A. Banta, "Using Computer Spreadsheets in Engineering and Education," *Proc. Frontiers in Education Conference Proceedings*, 1998.
- [2] W. Xu, Y. Liu, D. Koval, and M. Lipsett, "Using spreadsheet software as a platform for power system analysis," *IEEE Computer Applications in Power Journal*, vol. 12, pp: 41-45. 1999.



- [3] R. Pearson, and G. Atkinson, "Teaching vacuum technology using spreadsheet calculations," *Proceedings of the 15th Biennial University/ Government/ Industry Microelectronics Symposium*, pp. 332-335, Jul. 2003.
- [4] S. Alsagoff, "Microsoft Excel as a tool for digital forensic accounting," *IEEE International Conference on Information Retrieval & Knowledge Management, (CAMP)*. Pp. 97-101, Mar. 2010.
- [5] J. Svoboda, "Using spreadsheets in introductory EE courses," *IEEE Transactions on Education*," vol. 35, no. 3, pp. 226-229, Aug. 1992.
- [6] A. El-Hajj, and M. Hazim, "On using spreadsheets for logic networks simulation," *IEEE Transactions on Education*, vol. 41, no. 4, pp. 311-319, Nov. 1998.
- [7] S. Baicher, and J. Sherrington, "Learning about digital signal processing using spreadsheets and simulation software," *Journal of Engineering Science and Education*, pp. 41-48, Feb. 1996.
- [8] A. El-Hajj, K. Kabalan, and S. Mahmoud, "An improved spreadsheet toolbox for simulating digital networks," *IEEE Circuits and Devices Magazine*, pp. 3-5, Jan. 2003.
- [9] D. Walter, and M. McMillan, "A spreadsheet method for studying neural networks," *IEEE Proceedings of the 1990 Symposium on Applied Computing*, pp. 42-44, Apr. 1990.
- [10] J. Waldock, "Applying mathematics to digital image processing using a spreadsheet," *MSOR Connections*, vol. 10, no. 3. Pp. 11-16, 2010
- [11] A. Basuhail, "Modeling slotted-ALOHA simulator using computational spreadsheet based e-forms," *Journal of Spreadsheets in Education (eJSiE)*: Vol. 7: Iss. 3, Article 1, 2014.
- [12] A. Basuhail, "Design and implementation of e-learning content using simple and obtainable tools," *First International Conference e-Learning and Distance Education*, 2009.
- [13] R. Abreu, J.Cunha, J. Fernandes, P. Martins, A. Perez, J. Saraiva "Smelling faults in spreadsheets," *Proceedings of the 30th IEEE International Conference on Software Maintenance and Evolution: IEEE Computer Society*, pp 111–120, Washington, DC, USA, 2014.