

Schematic Editor for Educational Purposes on the World Wide Web

Nikolay Georgiev Chillev, Tania Krumova Vassileva, Vassiliy Platonovitch Tchoumatchenko

Department of Electronics, Technical University of Sofia

nikic@lark.vmei.acad.bg

Abstract. The paper describes the architecture, functionality, implementation and user interface of Web based schematic editor, appropriate for educational purposes. It enables efficient design and manipulation of different circuits from industry-standard and user-defined components via standard Web browser. Incorporated online help and check facilities as well as feedback provided to the user, make the reported schematic editor a powerful tool in every e-learning environment in the field of electronics, hydraulics and telecommunication.

Introduction

The Internet has enormous power to improve the educational process. By using the Internet, education can be personalized to each user, so that each student is given a targeted set of materials based on his/her specific educational goals and previous achievements. The net effect is that education is becoming increasingly student centered and targeted to the individual.

Technology is a tool; by itself, it cannot teach anything [1]. The human element is a critical component of the educational process. The wide available on the web read-only teaching materials prevent users from getting response, based on their actions. The key is to create a set of interactive tools that guide the learner and can be used most effectively to leverage the teacher's time and energy, so that the teacher spends the most time doing those things that add the most value to the learning process.

Training in new design technology and obtaining skills for using sophisticated design automation tools are important topics in every engineering education. The most sophisticated technology and the most advanced design tool are useless if the student lack the training or motivation to take advantage of them. That is why our goal is to make learning interesting – to make it so relevant and exciting that student want to learn and to continue learning.

Schematic Editor is the front end for any design application. The function of this editor is to create a schematic diagram of the circuit. Our goal is to create interactive learning tool, which permits students not only to capture and edit circuit schematics, but also to learn given circuit configuration by incorporating back annotation functions for connectivity comparison. The main idea is to ensure platform independence, high level of interactivity and full control of student actions with appropriate feedback. The goal is to give students ability to

- Capture and modify simple schemes;
- Perform respective rules check;

- Obtain appropriate guiding and feedback;
- Learn how to manage with the most popular schematic editors.

To address above-mentioned goals a Web based schematic editor is developed. It is conceived as a Java student centered learning environment capable to manage circuit diagrams in different fields. Incorporating “learning by doing” approach it allows students to study, capture, modify and check schemes obtaining in the same time appropriate on-line guide functions and feedback.

Schematic Editor – Functionality

The schematic editor is intended not only to create a schematic diagram of the circuit, but also to check it and to prepare for further use. It applies intuitive design approach to make design methodology easy for the beginners. It implements all main drawing and editing functions, provided by professional editors. It allows:

- Flat schematic entry
- Use of standard symbols
- Creation of own symbols
- Definition of attributes for symbols, symbol instances and pins
- Flexible assignment of pin names and instance names
- Editing features
- Consistency design rule and connection checks

The created circuit is in the form of a logical diagram where components are placed using the library symbols. Multiple instances of the same symbol are supported. Each of the placed components gets a unique index and is labelled to display information about the symbol. User can rotate and flip the individual or grouped components during capturing to obtain the best symbol direction for the respective scheme.

The connections between component entries are established using wires linked in segments. Automatic orthogonal operation between segments is provided within the editor. Wires are connected to the components’ pins. Junction points between crossing wires are automatically placed and indicated.

For each component a set of attributes is provided which can be edited in a dialog box. Attributes include component name and value. Each attribute can be customized whether to be visible or not.

Schematic editor provides intuitive schematic editing facilities. The properties of the placed objects may be edited after selecting the objects. Wrong components or wires can be moved, rotated, mirrored or deleted as single objects or in groups. Components attributes are also accessible for editing.

Once a schematic diagram is complete it is checked for:

- Extra components. Each component, not required by the targeted scheme, is marked as wrong.
- Unconnected pins. Not connected pins are marked.
- Relevance with connectivity of the targeted scheme. A targeted and captured scheme is checked for equality in the terms of connectivity between their

components. The sequences in which components are placed in the circuit as well as components' direction are not taken into account.

On line help is also available.

- Help facility is context sensitive to the currently performed operation. Guided instructions differ depending on whether the component is selected for placing, wiring or editing.
- To help learner to manage with the targeted schematic a HTML page with the right schematic configuration is provided on request.

Schematic editor can be used in two modes. In the examine mode, schematic drawn by the learner is compared with the predefined targeted circuit. Appropriate guided messages are given in this mode. In the generation mode, the netlist of the scheme is generated in appropriate format for further use – simulation for example.

User Interface

Schematic editor provides a sophisticated, yet intuitive graphical user interface. It includes navigation buttons, drawing area, and a magazine with components. Components can be standard or user defined making the editor proper for building schemes in the field of electronics, telecommunications, hydraulics, etc.

The interface of the environment is built upon tree panels as shown in Figure 1.

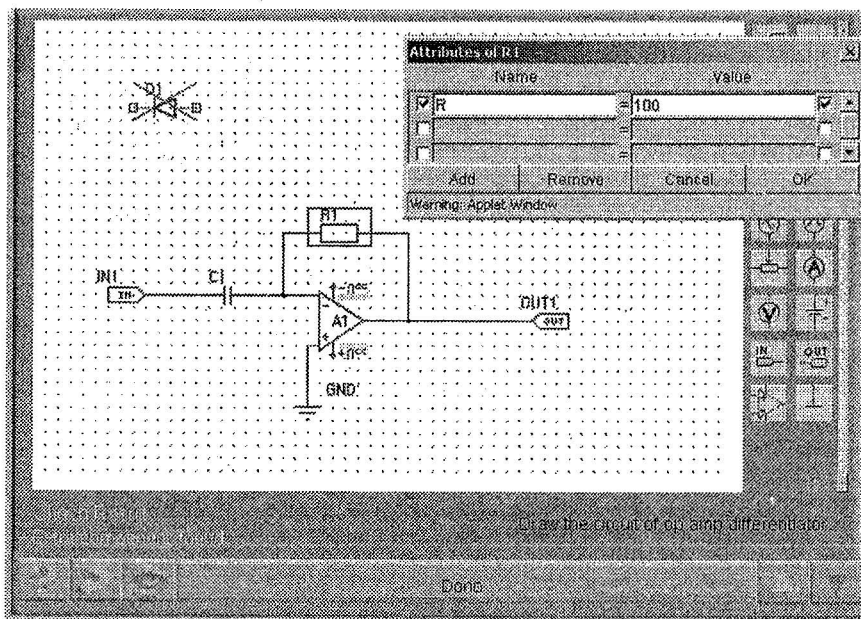


Figure 1 Graphic user interface of the schematic editor

The sheet panel contains the captured scheme. Attributes of the components are defined in a dialog form, invoked by double clicking on the component. Components have pins with tree states- connected, unconnected and unused. When the component is placed on the sheet all it's components are unconnected. Linking a pin with using wire with another pin makes it connected. Double clicking on it makes it unused. All of these states have their respective graphical symbols. The editor features stepping, multiple-object selection, moving, rotation and mirroring functions.

The magazine contains all elements, which can be placed in the scheme. The type of the component assigned to each item can be configured in an external file, which makes the magazine widely applicable. Each item is provided with a hint, giving information about the component. Custom defined components allow using in different areas as shown in Figure 2.

The navigation panel consists of buttons and labels, ensuring all drawing, editing and checking operations. The Reset button clears the working sheet and the internal data structures. Previous and next buttons clear the worksheet and initialize schematic editor for new task. The Hint button shows HTML page with the right solution for the current task. The Done button starts an algorithm, which compares the current scheme configuration with the targeted one. At first it checks whether all components are available in a circuit. After the algorithm verifies for unconnected pins. Finally the fundamental connectivity check is performed to make sure that the circuit is appropriately created. At each check step proper messages about current situation are given to the user as shown in figures.

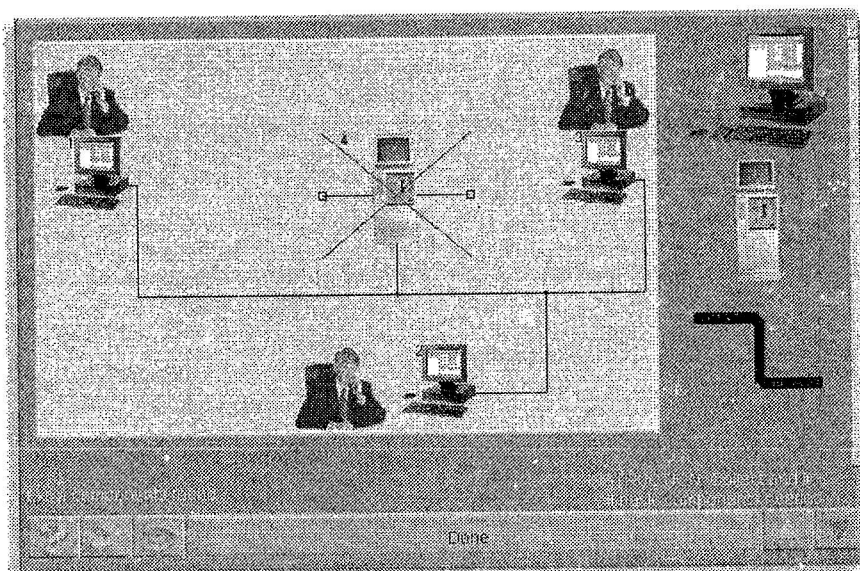


Figure 2. Custom components and background in telecommunication scheme

Implementation

The circuit editor implementation is based on the two-tier computing model. In order to make the editor portable it is realised as a Java applet [2]. It is built upon two separate layer – one for user interface and another for the schematic's logic.

The user interface uses custom AWT components to improve the visual appearance, for speed enhancement and double flickering reduction. The applet layout is based on relative positioning, which makes it proper for different resolutions and sizes. Applet has a user configurable background image, which improves its visual effect.

The applet's logical tier is concerned with graph comparison of the target and drawn circuits. The schemes are presented as oriented graphs. The algorithm for graphs polymorphism is used [3]. This makes the check for equality flexible since not consider the order of components entering. The algorithm at first determines the nodes and the edges of the graphs. After, the nodes of each compared graph are sorted in increasing number of neighbours. The algorithm attempts to find the respective components from the drawn and targeted schemes. At each step of the comparison an appropriate message, indicating type of error is displayed.

In terms of better performance, the algorithm is implemented at the lowest possible level – only generic data types are used. But inspite of all speed optimizations the algorithm remains with complexity $O(N)=N!$, where N is the number of vertices in the scheme. This classifies it as NP complete algorithm.

Schematic editor is configured with XML file, which is in text format, understandable for everyone and easily portable. This makes the schematic editor flexible and gives it capability for multilanguage support.

Conclusion

The developed Web based schematic editor allows students to study, capture, modify and check schemes obtaining in the same time appropriate on-line guide functions and feedback. It provides intuitive schematic drawing and editing facilities as well as powerful circuit check by incorporated graphs polymorphism algorithm.

The schematic editor is implemented as a Java applet which make it platform independent, flexible in terms of size, display resolution, custom components etc. It allows high level of interactivity with full control of student action.

Schematic editor is configured with external XML file. So it be easily adjusted to different circuit configurations and has a capability for multilanguage support.

The developed Web based schematic editor incorporates "learning by doing" approach. It helps the learners to gain a better sense of their knowledge gaps and increase learning with high moments of teaching value. It helps students acquiring:

- Ability to practice active learning
- Ability to cope with an unknown interactive environment, using online guiding and feedback

- Skills to deal with the most popular commercial schematic editors. The applet's user interface is designed to resemble Cadence OrCAD Capture® schematic editor.

The Web based schematic editor can be used in every e-learning environment. It is successfully applied in the Ingenatic Company Web Tr@iners [4] in the field of electronics, power electronics and telecommunication. As a further work it will be provided with simple simulator.

References

- [1] e-Learning and Educational Transformation: An Interview with Greg Priest by James L. Morrison, Vision May/June 2000
- [2] Eckel Bruce, Thinking in Java 1997.
- [3] Nakov Preslav, Fundamentals of the computer algorithms. Top Team books, 1998
- [4] http://www.ingenatic.com/html/_ingenatic_e-learning_solution2.html