Web-Based Simulation of Electronic Circuit

Ugochukwu Kamalu, Esite Jeremiah Tamarankuroemi

***Abstract*— Over the years, advancements have been made in the world of internet technology. It has also gained significant application in various aspects of life. Its application in electronic circuit design and simulation is therefore inevitable.**

The paper explains the basis of web-based simulation (WBS) which has to do with the integration of web with the field of simulation.

It examines its importance, structure, implementation, advantages and disadvantages of the various implementation configurations of web-based simulation.

***Index Terms*— Web-based, Simulation, Web Service, electronic, internet.**

I. INTRODUCTION

The use of the internet has experienced a tremendous growth since its development. Several fields and disciplines have hence carried out a re-evaluation in order to integrate internet services into their various operations.

The use of the internet via web-applications for simulation purpose has had significant impact in areas of education and training, publications and simulation programs.

Simulation is an integral part of circuit realization and hence a vital tool used by both professionals and students. Various simulation tools allow for students to model a system, get understanding of it, and then explore alternate designs [10].

Typical traditional simulation tools lack portability and disseminating ability which can be solved by utilization of the internet via web platforms. This is achieved by a combination of the modelling power of traditional simulation with the communication ease of the internet.

Generally web-based simulation provides for wide accessibility, controlled access, efficient maintenance and increased integration.

In the area of education, repeated studies have been carried out which shows that often students in engineering core courses often encounter difficulty in understanding basic concepts because they lack a visual aid study approach [8].

For educational and training purposes, the web offers platform for the storage and retrieval of supporting material creating a highly interactive environment for learning [2].

A user can carry out a standardized simulation via a web platform and be able to publish it via the World Wide Web. On the other end, the recipient of such publication can then simulate the embedded models by simply clicking a button. This in essence provides a convenient system for submitting, refereeing and sharing of research results

A variety of systems based on web technologies currently exist. They are mostly characterized by a combination of client technology and circuit simulation environment with the provision of a graphic user interface for simulation capability. These simulation environments combine distance education, group training and real-time interaction [4].

II. GENERAL STRUCTURE OF WEB-BASED SIMULATORS

According to Sven and Benno [9], Web-based simulation service can be realized as a web service. Web services applications can be local, distributed, or web-based. Web services are built on top of open standards such as TCP/IP, HTTP, Java, HTML, and XML [7]

Every web service is made up of a service provider, service requestor and service registry. The service provider provides the web service, making the implemented service available on the internet. The service requestor is the web service customer who utilizes the web service by establishing a network connection. Finally, the service registry provides a central place where developers can publish new services or find existing ones.

A web-based simulation application is based on web interface, which utilizes web client-server conception. The computation and the interface programs run on the server and a user uses arbitrary graphic client, i.e., standard web browser [4].

There are several ways of realizing a web-based simulation system:

- Local simulation and visualization,
- Remote simulation and visualization, and
- Hybrid simulation and visualization.

A. Local simulation and visualization

In the case of local simulation and visualization (S&V), both the simulation engine and visualization components are downloaded to the user’s local computer seamlessly by the client, so that both the graphical interface and the simulation engine are made available in the same environment. Lorenz and Schriber [6], explains that this approach supports the incorporation of user interaction and animation into the simulation applet.

With this, there is a shift from the server side to the client side in terms of execution, allowing the server a central distribution point to the simulation but performing no real work [3].

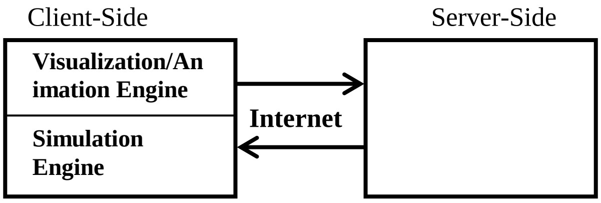


Fig. 1: Basic local simulation and visualization configuration post initial loading phase.

Fig. 1 shows the basic local simulation and visualization configuration after the initial loading stage.

When the user loads the web page (which usually contains an applet) of the web-based simulation system, the applet is downloaded to the user's computer seamlessly. This applet contains both the simulation engine and visualization/animation engine, which can then be executed on the client-side locally. The graphical interface and the simulation engine are incorporated into a single application which is then executed within the web browser on the user's computer. In this configuration, a database could also be available on the server-side.

The advantage of using the approach in developing a web-based simulation application is that there is usually no network delay between the user and the simulator since the simulation engine can be accessed locally. On the other hand, the disadvantage of using this architecture for developing a simulation application is that it relies heavily on the hardware of the user's computer in order to work efficiently.

B. Remote simulation and visualization

In remote simulation and visualization, both the simulation engine and any animation generation engine are located and execute remotely, on the server-side.

Parameters are sent to the web server where the simulation engine is located. At the end of the simulation, the result is sent to the user [1].

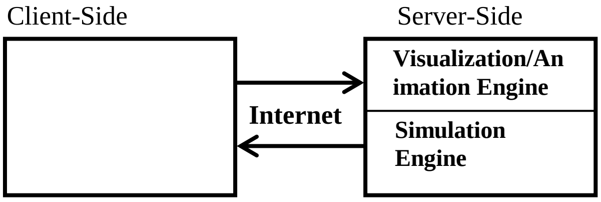


Fig. 2: Basic remote simulation and visualization configuration post initial loading phase

Fig. 2 depicts the basic remote simulation and visualization configuration post initial loading phase.

When the user accesses the web simulator using a browser, the basic interface is displayed. Both the visualisation and simulation engine are located at the server-side which is interfaced with through a browser. As with local simulation and visualisation, a database may also be present on the server-side.

An advantage of this approach is that larger simulations can run on powerful, high-end computers and the analysts can get access to the results from any low-end computer with a browser. Another note-worthy advantage is that with this approach, a centralized application exists, which makes maintenance by developers easy.

Some of the drawbacks of this configuration include:

1. It is not efficient for dynamic response observation.
2. Simulations cannot be interrupted midway as a result of the lag between the client and server.
3. Due to delays from the server, long running simulations may take extra time to generate output.
4. If multiple simulations take place at the same time, the server may get overloaded. The server load can also be excessive when complex visualizations are being generated even with a few concurrent users.

The problem of overloaded server can be solved by spreading the server load over a number of processors.

C. Hybrid simulation and visualization

A hybrid configuration is achieved by combining the approaches of remote simulation and local visualization configurations. The hybrid configuration combines the benefits of both local visualisation and remote simulation configurations.

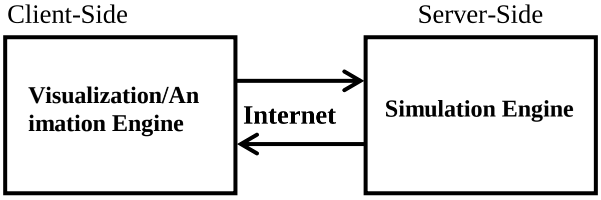


Fig.3: Basic hybrid simulation and visualization configuration

In this approach, the simulation runs remotely on a simulation server, and when the user connects to the server through a web browser, a visualization/animation engine is downloaded to the client-side. The web browser is used as a "thin client" with a limited calculation capacity, relying essentially on the browser integrated scripting languages (like JavaScript) and applets as client side java applications that are downloaded and embedded with a web page [1].

From fig. 3, a dedicated data connection is made back to the server after which the simulation results are then transferred to the client-side enabling the visualization engine to display results to the user in a dynamic nature. The data can continuously change, with delay being only due to the simulation model being executed and the latency present on the network connection [3].

The hybrid simulation architecture can also be implemented in such way that the user is able to build a simulation model at the client side using the visualisation/animation engine through an animated interface. The results are then displayed without tying the animation to the simulation during model execution.

The hybrid approach combines the best features of local and remote simulation and visualization. Majority of the simulation is performed on the server-side, the simulation can take advantage of more powerful hardware, and maintenance is eased with a centralized application. One of the advantages of this approach is that the server is not overloaded and there is more user interaction than traditional remote simulation and visualization approaches by sending the animation/visualization portion to the client-side.

A disadvantage of this approach is that there could be delay in communication between the client and the server. This can be eliminated by offering no interaction during an active simulation.

Another problem is that many of the systems using this approach make use of Java applets through a browser interface. In this case, the user must have the specific JRE installed in order to use the application. Though this is not a certainty on the internet today.

III. SIMULATION RESULT USING PARTSIM

In this section, we shall simulate a simple electronic circuit using a web-based simulator built using hybrid configuration called PartSim.

PartSim is a full-featured web-based simulator which can be accessed for free at www.partsim.com built on the basis of a full SPICE ("Simulation Program with Integrated Circuit Emphasis") simulation engine. The SPICE simulation engine is an open source electronic circuit simulator developed by the University of California.

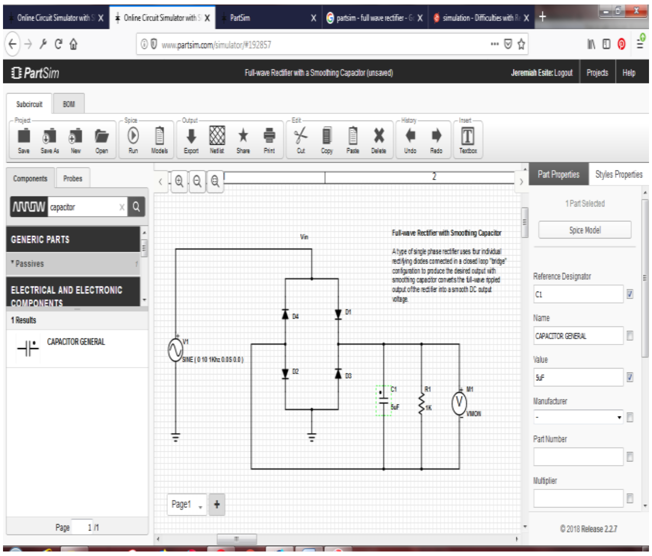
In this simulation, a full-wave rectifier circuit is implemented using the following components:

- One AC voltage source;
- four diodes (D1, D2, D3 and D4)
- a 5uF capacitor;
- a 1kΩ resistor;
- a voltmeter;
- and two ground symbols.

Transient analysis of a simple full-wave rectifier circuit

A full wave rectifier circuit is implemented as shown in fig. 4. Each circuit element is gotten from the sidebar. The value of each element can easily be edited by clicking on them.

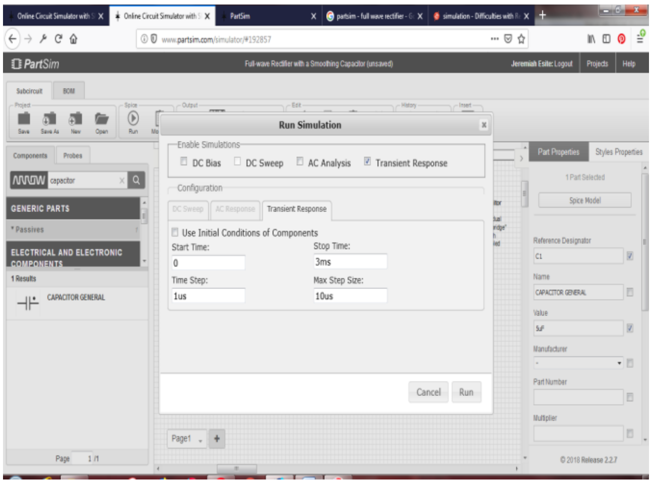
Once the circuit design is done, the "Run" button is clicked and then a dialog box pops as seen in fig. 5 where the "Transient Response" is selected and the "Run" button is clicked. The result of the transient response simulation can be seen as graphical response as shown in fig. 6.


Full-wave Rectifier with Smoothing Capacitor

A type of single phase rectifier uses four individual rectifying diodes connected in a closed loop "bridge" configuration to produce the desired output with smoothing capacitor converts the full-wave rippled output of the rectifier into a smooth DC output voltage.

Reference Designator: C1
Name: CAPACITOR GENERAL
Value: 5uF
Manufacturer: -

Fig. 4: Screenshot of a rectifier circuit design using PartSim


Run Simulation

Enable Simulations
☐ DC Bias ☐ DC Sweep ☐ AC Analysis ☑ Transient Response

Configuration
DC Sweep | AC Response | Transient Response

☐ Use Initial Conditions of Components
Start Time: 0
Stop Time: 3ms
Time Step: 1us
Max Step Size: 10us

Cancel Run

www.partsim.com/simulator/#192857
PartSim
Full-wave Rectifier with a Smoothing Capacitor (unsaved)
Page1
Part Properties
Styles Properties
C1
CAPACITOR GENERAL
5uF
© 2018 Release 2.2.7

Fig. 5: Screenshot of pop-up window to select a particular simulation type

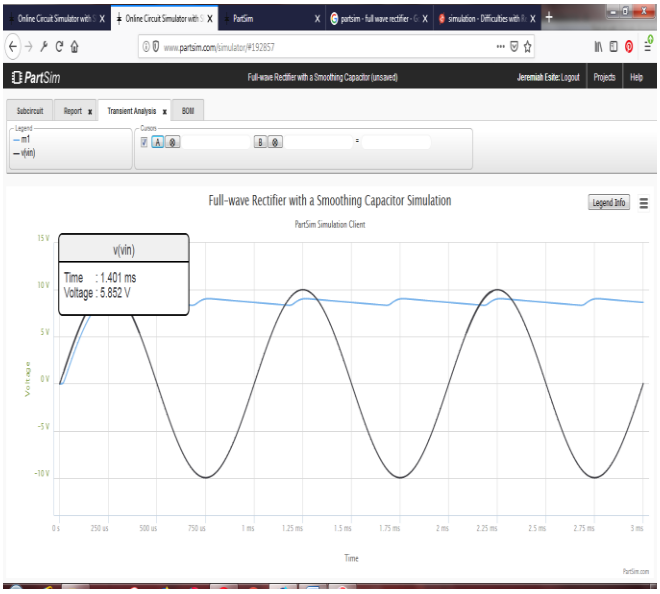

v(vin)
Time : 1.401 ms
Voltage : 5.852 V

Fig. 6: Screenshot of a graphical solution to transient analysis using PartSim

IV. BENEFITS OF WEB-BASED SIMULATION

Most simulators are in the form of executable files which have to be installed directly to a computer system.

While this simulation model is quite efficient, it has some limitations which have prevented its use by a larger numbers of people. First, the computer used may need to have the simulation language installed in order to run the model.

In most cases, this executable version has to be bought before installation. This executable file has to be compatible with all the computers it will be installed in. If the simulation is to be taken at a specific location, e.g., laboratory computers, there would be limitations to its accessibility and availability.

Interacting with the simulation may also require learning the simulation language, which would be time consuming.

In cases where the software is updated distribution of this update becomes difficult and new features cannot be made use of until the new update is implemented. Web-based simulation can help to address these underlying issues by providing wide accessibility, controlled access, efficient maintenance and increased integration [10].

A. Wide availability

A user can run a web-based model on numerous platforms irrespective of location without a need for additional hardware, software, file downloads or code recompilation. Furthermore, there is no time constraint since the Internet is available at any time of the day.

This provides economical widespread availability of a simulation tool. A user can easily access the simulation tool via any computer having a Java-capable web browser. The simulation interface requires no special training to use as it appears the same as a typical website.

B. Controlled access

Since Java does not allow users to save applets, a person must have access to the original model or package source code to permanently change a Java-based simulation model or duplicate the simulation package. This ensures users access the latest version of the model and reduces the potential for inadvertent or unauthorized alterations. This benefit can be particularly useful for teachers asking students to individually modify and analyse a particular model. If desired, access to the site can be controlled using passwords or network restrictions.

Such access restrictions are placed on the model or the website, not on the machine itself, ensuring uniform access privileges regardless of platform.

C. Efficient maintenance

The developer of the web-based simulator can easily carry out modification of the source code via the internet from the host server.

The modified model is then available instantly over the web and users can be notified through e-mail if necessary.

The updated model is available instantly, providing simplification of version control.

D. Increased integration

Generally, web browsers are available on the internet that supports web-based simulation packages. Most of these browsers are currently available for free download or are included in standard office software packages. On browser installation, no extra software or downtime is required in order to access a simulation model.

There is the availability of various web-based tools to help users while working with a model.

An example would be for each web-based simulation model to be linked to a hypertext markup language (HTML) page. This page can contain e-mail links, help forms and links to previous models. It can also provide extra information in the form of textual instructions, images, sound clips, and other multimedia tools.

The presence of technical assistance provides the user information on how to use the specific model, parameter ranges that are appropriate to employ, and a list-serve dedicated to usage questions and difficulties.

V. CONCLUSION

The development of electronic circuit simulators as an education tool is a necessity for enhancing proper understanding of students of electronic engineering. In the area of electronic manufacture, simulation of electronic circuits simplifies development of models.

Incorporating electronic simulation with the internet has helped in providing ease of accessibility by students as well as professionals to simulation engines. Making use of a simulator tool is now as easy as filling a registration form and logging in to your personal account. This account can contain multiple simulation projects which can easily be accessed by an internet-enabled computer system and mobile devices via web browsers.

By accessing simulation models through the
Web, a user can develop a model which can easily be shared to other users in cases of a team project or assessment of students by their instructors.

Simulation of electronic circuits can now be carried out irrespective of location by simple clicks of a button via any web browser of your choice.

REFRENCES

[1] Calciu M. (2013). Local, Remote and Hybrid Web Solutions to bridge the Academic-Practitioner Divide in Marketing Decision Models, 29-ème Congrès de l'Association Française de Marketing, La Rochelle, 15-16 ma

[2] Ernest H. Page, Robert L. Moose, Jr., & Sean P. Griffin. (1997). Web-based simulation in SimJava using remote method invocation. Proceedings of the 1997 Winter Simulation Conference ed. S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson pp. 468-474.

[3] James Byrne, Cathal Heavey, & P.J. Byrne. (2009). Simulation Modelling Practice and Theory 18 (2010) pp. 253–276.

[4] Jiri Hospodka, & Matej Koubik. (2009). Special Web-based Application for Electric Circuit Analysis. Pp. 140-145. ISBN: 978-1-61804-048-0

[5] PartSim SPICE circuit simulator, URL: http://www.partsim.com/examples/, 2018.

[6] Peter Lorenz & Thomas J. Schriber. (1997). Towards a web based simulation environment. Proceedings of the 1997 Winter Simulation Conference ed. S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson. IEEE Computer Society Washington, DC, USA, ISBN:0-7803-4278-X

[7] Tutorialspoint. (2018). What are Web Services?. Retrieved from: https://www.tutorialspoint.com/webservices/what_are_web_services.htm

[8] Shadi MS. Harb, Ahmad Harb, & Issa Batarseh. (2015). Development, Implementation, Assessment of a Web-based Circuit Solver for Teaching Basic Electrical Circuits Theory. Vol. 11, Issue 4, 2015.

[9] Sven Meyer zu Eissen & Benno Stein. (2008) Web-based simulation: Application scenarios and realization alternatives. Proceedings of the TMCE 2004, April 13–17, 2004, Lausanne, Switzerland, Horváth & Xirouchakis (eds.) 2004 Millpress, Rotterdam, ISBN 90 5966 0188

[10] Tamie L. Veith. (1998). World WideWeb-based Simulation. Int. J. Engng Ed. Vol. 14, No. 5, pp. 316-321.