SCRIPTING COMPLEX ARITHMETIC IN A WEB APPLICATION: MICROWAVE ENGINEERING SOLUTIONS

ANM Shariful Islam* ***Robert M. Nelson*****

ABSTRACT

Many aspects of microwave engineering involve detailed calculations. To allow students to focus on fundamental concepts rather than mathematical operations a web-based design tool named "Microwave Engineering Solutions" (MES) was developed as a graduate student project in the Electrical and Computer Engineering Department at North Dakota State University (NDSU), USA. Most of the programming code used in MES was written in JavaScript. Although JavaScript has many positive attributes, it is not capable of performing many of the complex mathematical operations often needed in microwave engineering. To overcome the limitations of JavaScript, server-side programming languages like PHP and Perl were used. In addition, some of the complicated mathematical equations were written in a form that could be handled by JavaScript. In this paper, a brief description of MES and its component tools is provided. This is followed by a discussion of many of the programming aspects involved in MES. MES is used by ECE students studying various aspects of electromagnetic and is freely available on the web http://venus.ece.ndsu.nodak.edu/~ronelson/mes/index.html.

INTRODUCTION

Aspects of high-frequency circuit design typically addressed in microwave engineering courses include analysis and design of analog filters and couplers, impedance matching, network analysis (scattering parameters, chain matrices, etc.) and the design of microwave amplifiers [1-11]. To obtain a thorough grasp of new concepts students are frequently asked to carry out calculations needed for these tasks. Although hand calculation significantly helps students understand fundamental concepts, students can get so focused on carrying out the calculations that they lose sight of what they are really learning. Computer-based tools can be very helpful at this stage of the learning process. They can greatly minimize calculation time, and allow the student to ask "what if" questions.

Various computer-based tools have been developed to assist students by performing many of the required calculations. Many such tools require the use of licensed software [12-15] (e.g., MATLAB® and MathCAD). Such tools are typically restricted to use on dedicated computers equipped with the licensed software. On the other hand, web-based tools are readily available to all students. Such tools have been developed for filter design [16-17], digital signal processing (DSP) [18-21] and transmission line calculations (available at sites such as http://www.circuitsage.com/ http://www.rfcafe.com/site_map.htm, and http://www sci.lib.uci.edu/HSG/RefCalculators.html). Although helpful, these tools are typically single-purpose tools (i.e., strictly filter design, etc.).

A new web-based tool was designed at NDSU that allows students to quickly carry out many of the operations involved in microwave engineering. "Microwave Engineering Solutions" (*MES*) [22,23] is available to students at http://venus.ece.ndsu.nodak.edu/~ronelson/mes/index.html and consists of three stand-alone tools: (1) *FilTech* - a design tool for analog filters, (2) *LMatch* - a design tool for impedance matching, and (3) *MicroLines* – a design/analysis tool for striplines and microwave network analysis. The first page of *MES* is shown in Figure 1.

Selection of the programming language used to write *MES* was a key element in its development. Since standard web browsers such as Internet Explorer and Netscape are equipped with JavaScript support, the decision was made to use JavaScript to provide client-side

* *Department of EEE, Islamic University of Technology, Gazipur-1704, Bangladesh.*

** *Department of ECE, North Dakota State University, Fargo, ND 58105, USA.*

programming. Although JavaScript has the advantage of providing client-side programming, it does have some drawbacks – including limited internal support for some mathematical functions and operations. To overcome these limitations, server-side programming languages PHP and Perl were used in conjunction with separate JavaScript subroutines. Complicated mathematical equations involved in different designs also had to be simplified to be used with JavaScript.

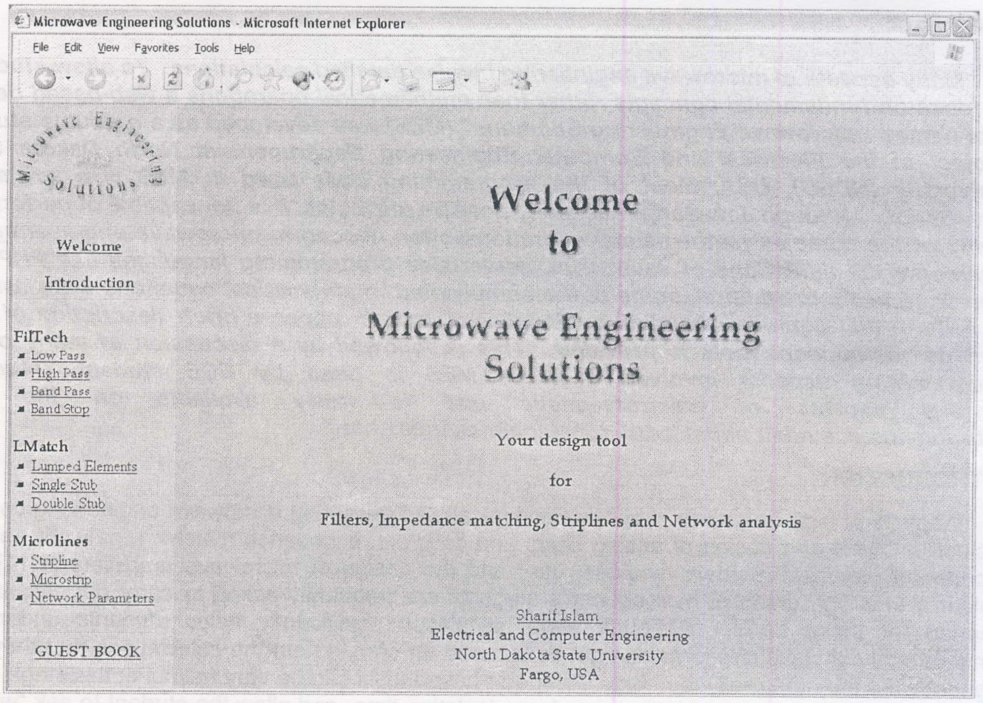

Figure 1. Web page of Microwave Engineering Solutions.

The purpose of this paper is to highlight and elucidate several of the programming aspects involved in MES. A complete description of each of the three tools used in *MES* is given in [22] and [23]. As such, only a brief overview of these parts will be provided here. This will be followed by a description of the unique scripting aspects applied in *MES*.

SOFTWARE DESCRIPTIONS

A. *FilTech*: A Design Tool for Analog Filters

Since analog filters are present in just about every piece of electronic equipment the accurate and efficient design of analog filters is an important task. A user-friendly, web-based tool called *FilTech* can be used to design passive low-pass, high-pass, band-pass, and band-stop filters with either Butterworth or Chebyshev responses. *FilTech* uses the insertion loss method [10] and performs the synthesis by first transforming the filter specifications to a low-pass prototype, synthesizes the low-pass prototype, and then does an element transformation to get the desired filter. *FilTech* calculates the exact values of the elements of the filter and then draws the corresponding circuit diagram and plots the frequency response of the designed filter in a separate window.

B. *LMatch: A Design Tool for Impedance Matching*

A common assignment for microwave engineering students is to eliminate standing waves on a transmission line via impedance matching – i.e., matching the load impedance (Z_L) with the characteristic impedance of the transmission line (Z_0). Three commonly used methods of impedance matching / tuning include lumped element tuning, single stub tuning, and double stub tuning. *LMatch* can be used to design an impedance match using any of these methods. For lumped element tuning, two reactive elements (i.e., an inductor and capacitor) are used. Single-stub tuning (both series and parallel) and double stub tuning are also accommodated with *LMatch*. Proper selection of the position and length of open- or short-circuited transmission lines are determined. *LMatch* provides all the solutions available for any particular impedance matching problem (e.g., values for the reactive elements used for lumped element tuning, etc.). In addition, the frequency response of each possible solution is provided, which the student can examine to help determine the optimal solution. It is noted that not all loads can be matched with double-stub tuning. *LMatch* will alert the user if a load is given that cannot be matched. The solutions are provided in table form, and appropriate circuit diagrams of the impedance match are displayed. The frequency response is also generated. Figure 2 shows the frequency response of a single stub-series-tuning network generated by *LMatch*.

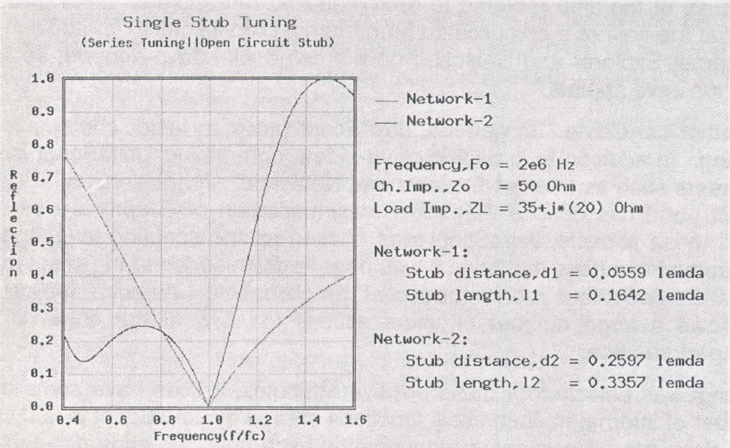

Figure 2. Frequency response of a single stub series-tuning network

C. *MicroLines*: *A Design Tool for Striplines and Microwave Network Analysis*

When designing microstrip filters and couplers proper selection of the width and length of the conducting traces is critical for a successful design. *MicroLines* calculates these values given inputs for the characteristic impedance of the transmission line, substrate thickness, conductor thickness, conductivity of the strip, operating frequency, and dielectric constant and loss tangent of the dielectric material. If given the strip length, *MicroLines* also provides equivalent circuit parameters (R, L, C) of a stripline [24], the maximum frequency of operation, signal attenuation, phase shift and the effective characteristic impedance. *MicroLines* also tries to help the student select reasonable values for some of the input quantities (strip conductivity, dielectric constant, and loss tangent) by providing values of the respective parameter for commonly used materials in a separate help window linked beside the respective input text box.

Microwave engineers often find it helpful to use various network parameters, such as the chain (or ABCD) matrix, the impedance (Z) matrix, admittance (Y) matrix, and the scattering (S) matrix. Depending on the problem to be solved, it may be advantageous to use a specific networkdescription (*Z, Y,* etc.). In such cases, it may be necessary to convert one network description into another. When carried out by hand, such conversion can be very tedious. *MicroLines* is designed to help with such tasks and can be used to relate all four types of network matrices for any two-port network.

PROGRAMING ASPECTS

As indicated above, *MES* can be used for filter design, impedance matching and a number of network functions. Each of these tasks requires repetitive calculations that can be carried out very quickly with *MES. MES* is characterized by easy-to-use "push buttons" and requires little instruction. *MES* generates frequency response graphs dynamically, and also electrical circuit diagrams as needed. In addition, help pages are provided throughout *MES.*

To carry out all of these functions using a web-based program requires careful consideration of many factors. Selection of the computer language to be used with *MES* was one of the first decisions to be made. Java is a possibility for web-based applications like *MES.* However, Java is losing popularity because of the time required to download and run applets. Java gives program and data security, but at the cost of a complicated programming architecture. In addition, standard web browsers like Internet Explorer and Netscape do not come with Java support, so a separate program is needed to run Java applets.

JavaScript is another candidate. In general, JavaScript provides faster computation through client side programming. In addition to providing client-side programming, JavaScript is supported by standard web browsers such as Internet Explorer and Netscape. In addition, JavaScript is easy to interface with HTML, and has options that allow implementation of Graphical User Interfaces (GUI). Considering all these aspects, JavaScript was chosen as the scripting language for *MES.* Using client-side programming means that once a web page is downloaded to the client machine, all the required mathematical operations are performed at the client side. As such, the use of client-side programming allows a large number of users access to *MES* at the same time without degradation of server performance.

Although JavaScript was selected for these positive attributes, it does have some drawbacks. In particular, the number of internal mathematical functions offered by JavaScript is fairly limited. In particular, JavaScript does not support some trigonometric hyperbolic functions nor can it perform rounding or exponentiation of calculated results. JavaScript does not support options used to draw lines or plot equations, and does not support complex arithmetic. Since *MES* involves many complex calculations (e.g., calculating and plotting the frequency response of filters) several major hurdles needed to be overcome. To overcome these limitations, server-side programming languages PHP and Perl were used in conjunction with separate JavaScript subroutines. This combination provided the tools necessary to carry out the required functions. As an example of the interaction, consider the problem of plotting the frequency response of a filter or impedance matching network. To carry this out, JavaScript receives user inputs from the web page which are passed to Perl via PHP. Complex calculations are carried out via routines written in Perl. The results are then passed back to PHP and PHP draws the actual graph in a separate window. Separate scripts are written for communicating among JavaScript, PHP, and Perl. Figure 3 shows the command line to exchange parameters among JavaScript, PHP and Perl. Interested readers are referred to [22] for the specific programming techniques used in the development of *MES.*

As mentioned, JavaScript does not support functions to carry out mathematical operations involving complex numbers. Since many of the parameters used in microwave engineering involve complex numbers, specific functions were written that dealt with the real and imaginary parts of the functions. As an example of such scripting, consider the sections of JavaScript shown in Figure 4, which are snippets from the parts of *MicroLines* that convert ABCD parameters to S parameters (S11 in particular).

```
// JavaScript for communicating with PHP file: filter.php

        window.open ('../graph/filter.php?N='+N+'&f0='+f0+'&BW='+BW+'&R='+R+'&
        rs='+rs+'&rl='+rl+'&response='+response+'&k='+k,'bodu','height=450,width=6
                    00');

// PHP script for communicating with Perl file: filter.pl

        $cmd = "perl /home/ekhan/public_html/mes/graph/filter.pl ".$R." ".$ep."
                ".$N." ".$k." ".$response." ".$Xmax." ".$Xmin." ".$increment;
        $result=`$cmd`;
        $data=explode("\n",$result);
```

Figure 3. Script for communicating among JavaScript, PHP, and Perl.

MES also checks the validity of each input value before carrying out any computations. For example, working with *FilTech*, the order of the filter must be a nonnegative integer value between 1 and 10, and the design frequency must be a nonnegative numeric value. As such, *FilTech* will not accept any negative number or any character input for the order of the filter or operating frequency.

```
// Functions for calculating other parameters from ABCD parameter input

function from ABCD()

//A=Ax+i*Ay;
//B=Bx+i*By;
//C=Cx+i*Cy;
//D=Dx+i*Dy;

        var Ax=parseFloat(document.form1.W12.value);
        var Ay=parseFloat(document.form1.W13.value);
   ⋮
// Calculating S from ABCD

//numS11=A+B/Z0-C*Z0-D;
//den=A+B/Z0+C*Z0+D;
   ⋮
        var im_numS11=(Ay+By/Z0-Cy*Z0-Dy);
        var real_numS11=Ax+Bx/Z0-Cx*Z0-Dx;
        var im_denS=Ay+1/Z0*By+Z0*Cy+Dy;
        var real_denS=Ax+1/Z0*Bx+Z0*Cx+Dx;
        var abs_denS=real_denS*real_denS+im_denS*im_denS;
         ⋮
        var real_S11=(real_numS11*real_denS+im_numS11*im_denS)/abs_denS:
        var im_S11=(im_numS11*real_denS+real_numS11*im_denS)/abs_denS:
         ⋮
```

Figure 4. Sections of JavaScript in Microlines for converting ABCD parameters to S parameters.

If an input error is detected, a message is generated by *MES*, which alerts the user of the error and provides suggestions for how to correct the error. For example, Figure 5 shows the message generated by *FilTech* for the case when the input frequency is invalid. As an example of helpful suggestions, Figure 6 shows the recommendations generated by *LMatch* for an error in a double stub tuning network.

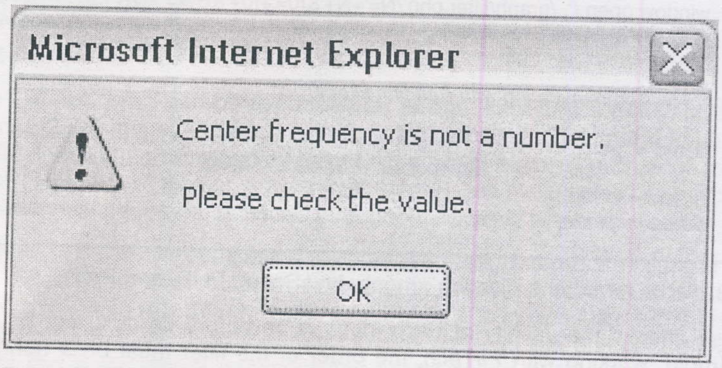

Figure 5. Message window from *FilTech.*

Although *MES* is an excellent design tool for filters, striplines and impedance matching, it has some limitations at the present time. In particular, *FilTech* uses the filter pass-band characteristics provided by the user. It does not consider stop-band and transition-band characteristics (which some designers may prefer). Double stub tuning is presently available for parallel stubs only. Quarter-wave transformers are commonly used for impedance matching, and will be included in future versions.

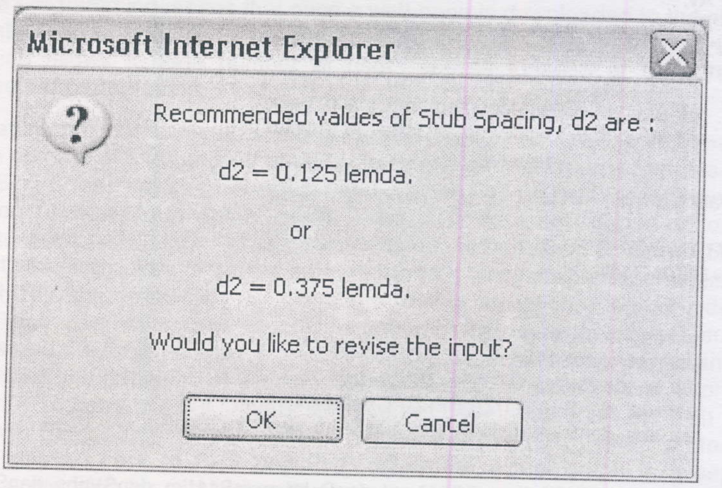

Figure 6. Message window from *LMatch.*

It should be pointed out that *MES* underwent several phases of testing. The first phase involved evaluating each of the component tools (e.g., *FilTech*, etc.) by solving different types of design problems using MATLAB®, by hand calculation, and with *MES*. After successfully passing that test, *MES* was evaluated by a group of NDSU graduate students who had experience with microwave engineering. These students filled out an evaluation form which included suggested modifications for *MES*. The feedbacks from all tests were incorporated into *MES*, resulting in improved robustness, accuracy and reliability. To maintain on-going improvement a guest book and comment option is included in *MES*. Comments suggested by users will be sent to the author, thus providing further testing opportunities and evaluation options.

CONCLUSION

"Microwave Engineering Solutions (*MES*)" is a web-based design tool developed at North Dakota State University that can be used for analog filter design, impedance matching and various microwave network functions. Most of the source code for *MES* was written in JavaScript, although portions in PHP and Perl were needed to overcome limitations of JavaScript. Examples of specific programming scripts are provided. *MES* is web-based - thus available to any student connected to the Internet via http://venus.ece.ndsu.nodak.edu/~ronelson/mes/index.html

ACKNOWLEDGMENT

The authors are grateful to the NDSU students (current and former) Ezaz Khan, Ahmad Fallah, Mike Reich, Peng Jin, Yuliang Xu, and Debashis Banarjee for their help in debugging and evaluating *MES*. The constructive comments and suggestions provided by these students were very significant in the development of *MES*.

REFERENCES

[1] I. Bahl and P. Bhartia, *Microwave Solid State Circuit Design*, Hoboken, NJ: Wiley, 2003.s

[2] K. Chang, I. Bahl, V. Nair, *RF and Microwave Circuit and Component Design for Wireless Systems*, New York: Wiley, 2002.

[3] R. Collin, *Foundations for Microwave Engineering*, 2^{nd} ed., New York: McGraw-Hill, 1992.

[4] R. Elliott, *An Introduction to Guided Waves and Microwave Circuits*, Englewood Cliffs, NJ: Prentice-Hall, 1993.

[5] G. Gonzalez, *Microwave Transistor Amplifiers*, 2^{nd} ed., Upper Saddle River, NJ: Prentice-Hall, 1997.

[6] K. Gupta, R. Garg, R. Chadha, *Computer-aided Design of Microwave Circuits*, Norwood, MA: Artech House, 1981.

[7] J. Hagen, *Radio-Frequency Electronics*, Cambridge, U.K.: Cambridge University Press, 1996.

[8] R. Ludwig, P. Bretchko, *RF Circuit Design*, Upper Saddle River, NJ: Prentice-Hall, 2000.

[9] D. Misra, *Radio Frequency and Microwave Communications Circuits*, New York: Wiley, 2001.

[10] D. Pozar, *Microwave Engineering*, 2^{nd} ed., New York: Wiley, 1998.

[11] R. Weber, *Introduction to Microwave Circuits*, New York: IEEE Press, 2001.

[12] D. Baez-Lopez and T. Escalante, "MATLAB based analog filter design," *29^{th} Annual Frontiers in Education Conference (FIE '99)*, vol. 3, 1999, p. 13B4/1.

[13] L. Huelsman, "FILTERD-MathCAD worksheets for analog filter design," *IEEE Circuits and Devices Magazine*, vol. 17, no. 3, pp. 4–5, May 2001.

[14] R. Koller, B. Wilamowski, "LADDER-A microcomputer tool for passive filter design and simulation," *IEEE Transactions on Education*, vol. 39, no. 4, pp. 478–487, November 1996.

[15] M. Bialko, P. Dobrychlop, M. Klosowski, D. Mlynski, "Prototype hybrid expert system for analog filter design," *Systems-Science*, vol. 22, no. 2, pp. 81-91, 1996.

[16] J. Bicak, J. Hospodka, J. Vrbata, P. Martinek, "Design of electric filters in Maple and through WWW interface," *IEEE International Conference on Electronics, Circuits, and Systems, (ICECS 2001),* Malta, Sept. 2001, pp. 1619-1622.

[17] J. Teng, T. Luor, W. Hsieh, C. Chen, "Design harmonic filter for industrial distribution system on WWW," *Proceedings Of Ninth Internaional Conference on Harmonics and Quality of Power,* Orlando, FL, 2000, pp. 142-147.

[18] T. Welch, M. Morrow, C. Wright, "Poles and zeroes, and MATLAB®, Oh My!," *Computers in Education Journal,* vol. 10, no. 2, pp. 70-72, April 2000.

[19] A. Spanias, et. al, "Development of a web-based signal and speech processing laboratory for distance learning," *Computers in Education Journal,* vol. 10, no. 2, pp. 21-26, April 2000.

[20] A. Spanias, et. al, "Development and evaluation of a web-based signal and speech processing laboratory for distance learning," *Proceedings of 2000 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASP 2000),* Istanbul, Turkey, June 2000, pp. 3534-3537.

[21] A. Spanias and F. Bizuneh, "Development of new functions and scripting capabilities in Java-DSP for easy creation and seamless integration of animated DSP simulations in web courses," *Proceedings of 2001 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASP 2001),* Salt Lake City, Utah, May 2001, pp. 2717-2720.

[22] A. Islam, "*MES*-Web based design tool for Microwave Engineering." Master's Thesis, Department of Electrical and Computer Engineering, North Dakota State University, Fargo, 2002.

[23] A. Islam and R. Nelson, "MES – A Web-based Design Tool for Microwave Engineering," submitted for publication in *IEEE Transactions on Education.*

[24] K. Gupta, R. Garg, I. Bahl, P. Bhartia, *Microstrip Lines and Slotlines,* 2nd ed., Norwood, MA: Artech House, 1996.