# SimSmith: A New Program for the Venerable Smith Chart

Ward Harriman—AE6TY                                                                                    ae6ty@arrl.net

Whether I first got started in amateur radio, I was browsing the local bookstore and happened upon Wes Hayward's *Introduction to Radio Frequency Design*. After a quick examination I decided it would be a good primer to the topic, purchased it, and began to work my way through it. Chapter 4 of *IRFD* introduced the Smith chart and I read through the chapter several times. Unfortunately, understanding eluded me for quite some time. Still, I could glimpse its value and remained determined to understand the chart's utility. Finally it clicked and I started to use the Smith chart to solve a variety of problems around the shack. For those of you who remain mystified, I hope to help you along. For those of you who are already comfortable, I hope you'll find this a useful review.

Originally, the Smith chart provided two fundamental functions. The first and most important function was to simplify the calculations involved with the use of transmission lines and lumped, complex impedances. A secondary function was to provide a graphical aid in understanding the effects of combining the various circuit elements.

In the modern age, using the Smith chart to perform calculations has gone the way of the slide rule; such calculations are best done using a computer program. However, the utility of the Smith chart to enhance the understanding of complex impedances has not diminished. Indeed, use of the Smith chart has grown over the decades since its invention. The Smith chart remains one of the most widely used nomographs used in the practice of electrical engineering.

## Gettig Started

As mentioned above, the calculations involved with complex impedances are most easily done using a computer but visualization is still the job of graphs and pictures. A variety of computer programs are available which provide schematic capture, calculation and graphing all in a single package. Some of the more popular programs are Microsmith, LinSmith, QuickSmith and SuperSmith. Any of these programs provide the essential features necessary to learn about the use and utility of Smith charts. However, I found some shortcomings in each of these widely available packages. I had always wanted to learn the Java programming language and there is nothing like writing code to force you to understand "every little detail" and so, I wrote my own Smith chart software. The software, along with some instructional videos, is available on my website: www.ae6ty.com/software. I named it SimSmith.

SimSmith was developed with a few guiding principles. There are times when I will go months between uses of a particular tool and I often forget how to do things with it. I wanted to avoid the "relearning curve" and so made the following design choices.

First, I wanted all information about the circuit to be visible all the time. All circuit element values as well as their derivatives. For example, for a transmission line I wanted length and degrees and velocity factor all shown all the time. Second, I wanted there to be ONE window. Switching back and forth between schematic



**Figure 1—Ideal impedance transformation from 72 ohms to 50 ohms (not practical).**

capture and circuit analysis often slows the process. Third, the display should be real time and convey the maximum information. In following these principles, SimSmith requires the user to remember very little. Essentially, everything you might need to know about is always right in front of you.

It is probably best to use a specific and familiar example to get started. Suppose we wish to provide an impedance transform between a 50 ohm transceiver and a 72 ohm dipole. Figure 1 shows the Smith chart for this example. In it, the "dot" represents the starting impedance (the dipole) and the "X" represents the final impedance (the transmitter). NOTE: The Smith chart represents all impedances normalized and in most cases our target impedance will be the "normal" impedance. This means that usually, our goal will be to have the X at the center of the Chart.

Figure 1 shows the 'ideal' situation where the impedance transformation goes directly from 72 ohms to 50 ohms. The transition is shown as a straight line. Unfortunately, this transition requires an unrealizable, negative 22-ohm resistor. In the real world, we will need to use the familiar L network circuit shown in Figure 2.

Figure 2 shows a typical schematic with the *Load* on the left with 72 ohms of *real* and 0 ohms of *reactive* impedance. Moving to the right, Figure 2 shows the classic L network consisting of a shunt capacitor of 146 pF and a series inductor of 535 nH. On the far right is the *Generator* which is shown as operating at 10 MHz with a characteristic impedance of 50 ohms real and 0 ohms reactive. The Smith chart related to this schematic is shown in Figure 3.

One of the most valuable features of modern Smith chart software is to allow the user to vary the values of the various circuit elements and to see, immediately, the effect by viewing the Smith

**Figure 2—L network in SimSmith.**



**Figure 3—Smith Chart for the L network of Fig. 2.**

chart. For example, the user can select a parameter on the schematic and increase or decrease the value and watch the effect of changing that value. This is where the interactive nature of the tools comes into play. Unfortunately, it is very difficult to explain the experience in the written word. I highly recommend the reader go download one of the tools and PLAY! In the meantime, for the purposes of this paper, assume the user selected the inductor value and doubled the value. Figure 4 shows the resulting Smith chart. A significant feature of SimSmith is that it shows the path taken by the X as the various circuit elements are introduced.

Figure 4 shows an entire screen shot of SimSmith. Notice that the X has moved way from the center of the graph. On the lower right hand corner the impedance at X is shown. Specifically, the effect of doubling the L is to introduce an additional reactance of +34 ohms. In general, the goal of any impedance matching exercise is to start at the dot and add circuit components to get the X to move to the center of the Smith chart.

### SimSmith

As Figure 4 shows, SimSmith provides a comprehensive set of capabilities. The first step in any use of SimSmith is to specify the Load and the Generator. In the case of the Load, SimSmith allows the user to specify the load in two different ways. The first way is to simply enter the R and X values by clicking on the numbers and typing in a numeric value using a familiar notation. This technique works well when the Load does not vary with frequency. Unfortunately, this is not the case with antennas.

To deal with antennas, the user must be able to specify the load at the various frequencies of interest. SimSmith provides the ability to read a *load file* containing frequency/resistance/reactance triplets. Three common formats are supported, one compatible with EZNEC antenna modeling software, one compatible with the AIM4070 antenna analyzer and one compatible with the MiniVNA-pro. No practical limits are placed on the size of these load files.

The second step in using SimSmith is to specify the operating environment in the Generator box. Specifying the operating environment requires the user to set the operating frequency and default impedance. The vast majority of the time, the operating impedance will be 50 ohms real and 0 ohms reactive. However, the user is allowed to specify any value. The center of any SimSmith Smith chart always represents the R +jX of the Generator.

Schematics are created using an intuitive drag and drop interface. The user can add components by selecting from the menu of circuit elements on the left of the screen and simply moving them to the appropriate place in the circuit. Elements within the circuit can also be moved around using the same click/drag/drop interface. Unfortunately, it is difficult to demonstrate this simple interface in a written document. Consequently, the reader is encouraged to download SimSmith and give it a try. (In order to make SimSmith available to the widest possible user base, the author chose to write SimSmith in Java. In order to run a Java program the user must have the Java Runtime Environment or JRE installed. The JRE for windows and various unix derivatives is available, free of charge, from Sun Microsystems at http://www.java.com/en/download/index.jsp. Java for Apple computers is available directly from Apple. Java is often preinstalled on many computers. (As a final note… your browser might alter the download file extension from .jar to.zip. The file extension
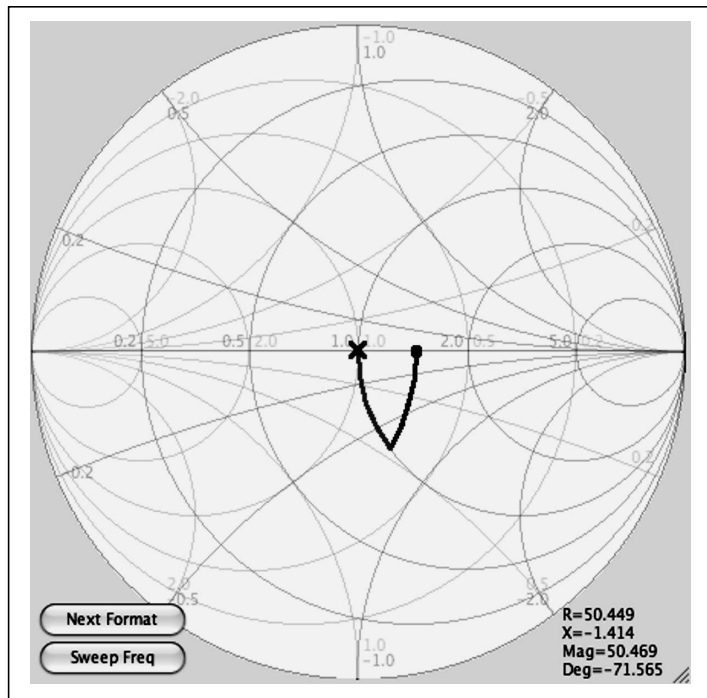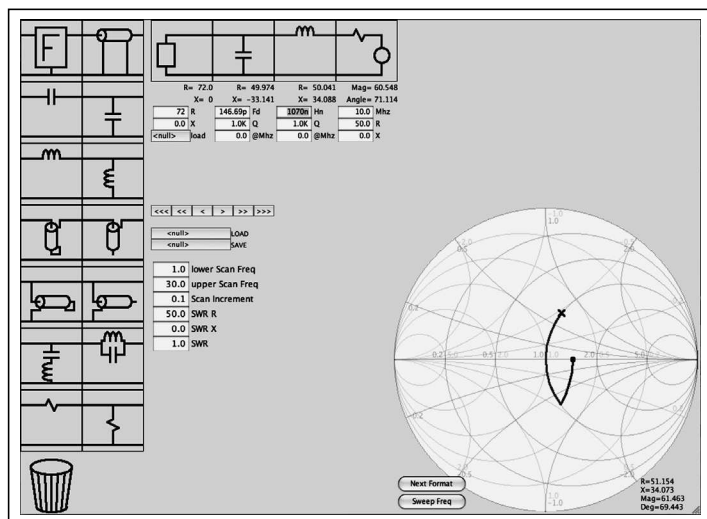


**Figure 4—Smith Chart with inductance doubled.**
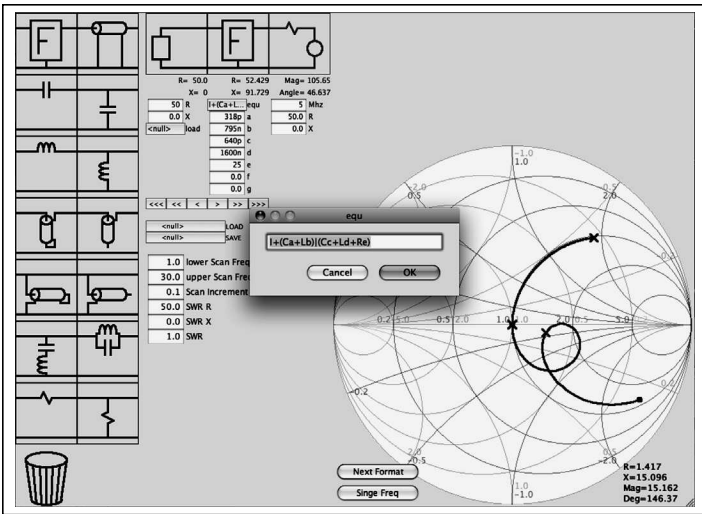
Figure 5—An example use of the F box.



Figure 6—SWR display for an antenna with two traps.

must be changed back to .jar for things to work properly. This problem is not unique to SimSmith.)

In developing SimSmith, I endeavored to provide some unique features sometimes lacking from other Smith chart software. For instance, the project was initially inspired by the need to faithfully model transmission lines; specifically to include the losses. These losses can prove quite significant when transmission lines are used as matching stubs. SimSmith allows the user to specify transmission line loss at a specific frequency. It computes the losses at other frequencies using traditional formulae.

SimSmith allows an operating impedance which can have a reactive component.

SimSmith allows both positive and negative component values. This can help the designer during the exploration phase.

SimSmith has no practical limits on the size of a circuit. As the circuit gets larger, SimSmith simply reduces the size of the components on the screen.

SimSmith scales its display to fit the window. Thus, the user can set the window to be any size and SimSmith can still be used; no information is obscured because the window size has changed. This is particularly valuable when using a large, high resolution monitor.

Finally, there are times when the user wants a circuit element not provided by SimSmith. For example, suppose the user wants to model an antenna that has traps with TWO resonant LC circuits. SimSmith allows for this by providing the "F" box. An example is shown in Figure 5. (Note that the equation for the F circuit element does not fit in the provided space. This is often the case. As a result, SimSmith uses a popup dialog when entering the equation. This capability is shown in Figure 5.) See Appendix A for a complete specification of the F box expression format.

Figure 5 also shows another extremely valuable feature always found in Smith chart software: frequency sweep. SimSmith can show how the circuit will operate over the range of frequencies specified. SimSmith can also plot the data in the more familiar SWR format shown in Figure 6. Notice that the SWR plot in Figure 6 shows two dips, one at 10 MHz (the resonance of Ca + Lb) and one at roughly 5 MHz (resonance of Cc + Ld). The valley at 5 MHz is slightly off because of the resistance Re.
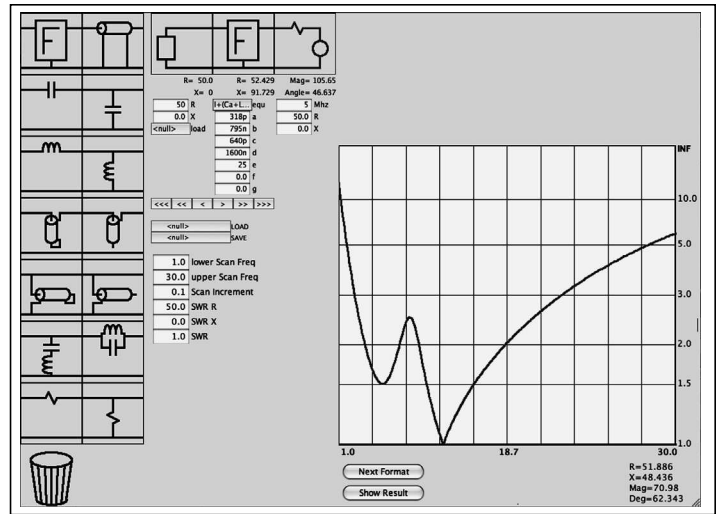
## Moving the X

As discussed above, using the Smith chart is essentially an exercise in moving the X to where you want it to be. In practice, there are three different ways to move the X. One can introduce a capacitance which generally moves the X downward. Conversely, one can introduce an inductance which generally moves the X upward. A final X movement is made by using a transmission line. All three movements are shown in Figure 7. Figure 7 starts with a load (the dot) at 25 ohms, adds a shunt capacitor, a series transmission line and a series inductor. The path taken by the X is represented by the three arcs. The X starts at the dot, proceeds along the downward arc, then goes clockwise around the center of the chart to about "3:30" and then traverses the final arc up to the center of the chart.

The example in Figure 7 demonstrates a few subtle features of complex impedances as represented in the Smith chart. First, all movement around the Smith chart takes place along arcs. When adding shunt reactance the arcs will be circular with an origin to the left of the center of the chart. Conversely, when adding series reactance the arcs will be along arcs with origins to the right of center. Series transmission lines result in arcs centered at the line's
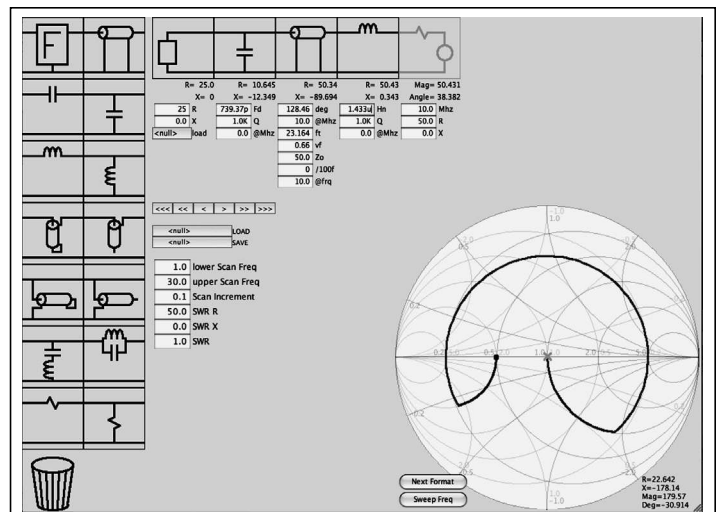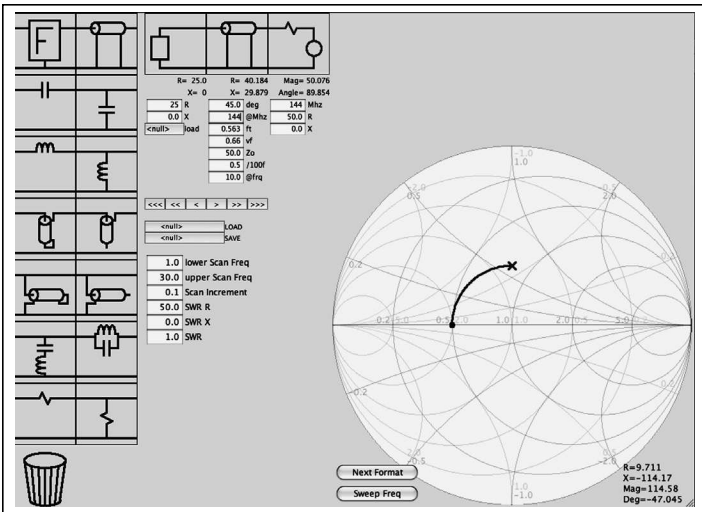


Figure 7—Smith chart for a three element circuit.

**Figure 8—Effect of a 1/8th wave line on 25 ohm load.**



**Figure 9—Changing the line to 1/4 wavelength.**

characteristic impedance. Typically, this impedance is that of the Generator.

Second, these arcs of movement are so important that they are always shown on the chart. Showing these arcs of movement helps guide the user while a components value is adjusted. Particularly—when attempting to reach the center of the chart, the user typically will move the X to the unit resistance or unit conductance curve, then simply move the X along that curve. This may sound complicated but is quite natural after a short time using the Smith chart.

Third, the Chart is not linear… for simple reactances, doubling the value of a component does not generally result in doubling the distance moved. For example, in Figure 7 the distance from the center of the chart to the starting dot is 25 ohms but is 1/3 of the way from the center out. The shunt reactance is over 5k and yet the distance from the center is only 3 times as far away.

Fourth, for standard Smith charts, the X cannot be moved out of the circle. Thus, the perimeter of the circle represents an infinite reactive impedance. Not clearly shown in Figure 7 is the fact that the extreme left of the Chart represents a real resistance of 0 ohms and the extreme right represents an infinite resistance.

Finally, although not shown, adding resistance moves the X along arcs that go through the leftmost and rightmost points of the Chart. Adding an infinite series resistance results in an arc ending at 3 o'clock. Adding a zero, shunt resistance results in an arc ending at 9 o'clock.

Having mastered these basic movements, a few more examples are in order.

**The Quarter Wave Matching Stub**

It is often the case that a Yagi-Uda antenna has an impedance significantly lower the typical 50 ohm feed line. A common remedy for this problem is to use a quarter wavelength matching stub. Figure 8 shows how one might get started solving this problem. Figure 8 shows a load of 25 ohms and a generator impedance of 50 ohms and an operating frequency of 144 MHz. Also shown is a 50 ohm, series transmission line which will be tuned to effect the impedance transformation.

The first step in this tuning process is to set the length of the
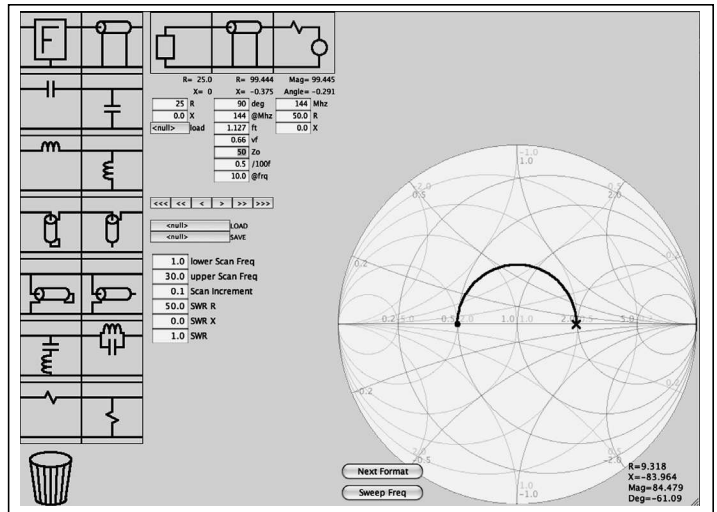
matching transmission line to 90 degrees. This is done by clicking on the *deg* box of the transmission line and setting the value to 90. Figure 9 shows the result of this step. The next step is to adjust the characteristic impedance of the quarter wave section. This can be done in SimSmith by selecting the characteristic impedance box labeled Zo and typing in a value. Alternatively, you can click on the Zo box and then use the tuning buttons "<<< | <<| < | > | >> |>>>". Figure 10 shows the result.

Note that, in the end, the X didn't end up at the exact center. This is because the user (I) chose to make the matching stub 37 ohms. A 37 ohm stub can be constructed by using two sections of 75 ohm transmission line in parallel.

**Modeling the Dipole Antenna**

Using the Smith chart to examine antenna parameters gives us a hint as to how to model a simple dipole antenna. Figure 11 shows a basic 72 ohm dipole swept across a fairly wide frequency (12 to 17 MHz). In this example, the antenna characteristics were generated by an antenna modeling program (EZNEC) and read into SimSmith. Note the resonant frequency is 14.167 MHz. The goal of this exercise is to see if we can develop a model of
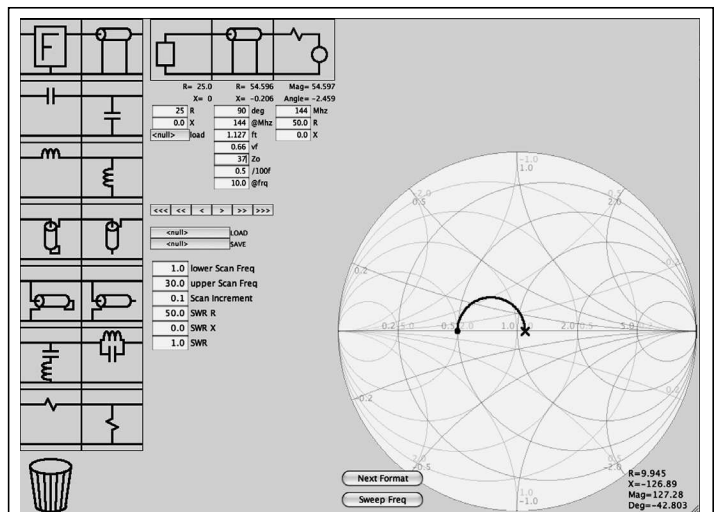


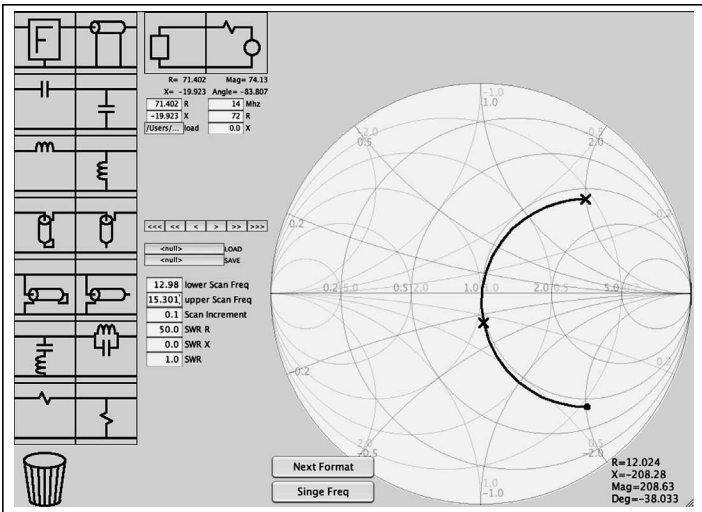**Figure 10—Changing line impedance to 37 ohms.**

**Figure 11—Impedance of an ideal dipole over a wide frequency range.**



**Figure 12—Using LCR elements to make a model equivalent to an ideal dipole.**

this simple antenna using the Smith chart. There are several reasons such a model might be useful. For example, one could build a reactive dummy load for amplifier and antenna tuner testing. Or, one could use the model in other simulation to ensure amplifier stability within the band of interest.

So, in examining Figure 11 we see that, to a first approximation, the antenna looks like a simple resistor in series with an LC network resonant at the center frequency. Lets start there. Figure 12 shows the result. We start by setting the Generator impedance to 72 ohms to simplify our work. We then set the load to 72 ohms and add a series capacitor and inductor. The capacitor/inductor circuit will be resonant at 14.167 MHz so that, at that frequency the antenna impedance will be purely resistive.

After adding the inductor and capacitor we begin the tuning process. By making the capacitor smaller and the inductor larger we can adjust the length of the arc passing through center. The effects of these adjustments are shown in Figure 12.

Actually, this is a fairly good model but we can do better. While the general arc is about right, we need to rotate the entire arc clockwise. This can be done by adding a transmission line in series

with the LC network. Figure 13 shows this transmission line and adjustments to all elements. The result is a surprisingly good model for the dipole antenna over a quite wide frequency range.

**Broadbanding**

One of my favorite ham radio examples is the broadband matching of a dipole antenna. This example draws on many disciplines and has several surprises along the way. This material is discussed in depth in the *ARRL Antenna Handbook*, chapter 9, "Broadband Antenna Matching". Typically, this example uses the 80m band where the dipole does not have enough bandwidth to cover the amateur privileges. Here I use the 20m band but the principles are the same.

This project starts by examining the properties of a simple dipole. Figure 14 shows the SWR of a dipole 34 feet long and up 34 feet above typical ground. The load file is created by EZNEC. Note that the 2:1 SWR bandwidth is about 13.9 to 14.5 MHz, about 600 kHz wide. The goal is to widen the 2:1 SWR bandwidth. The first step is to better match the dipole using the traditional L network. Figure 15 shows the result of this match. The 2:1
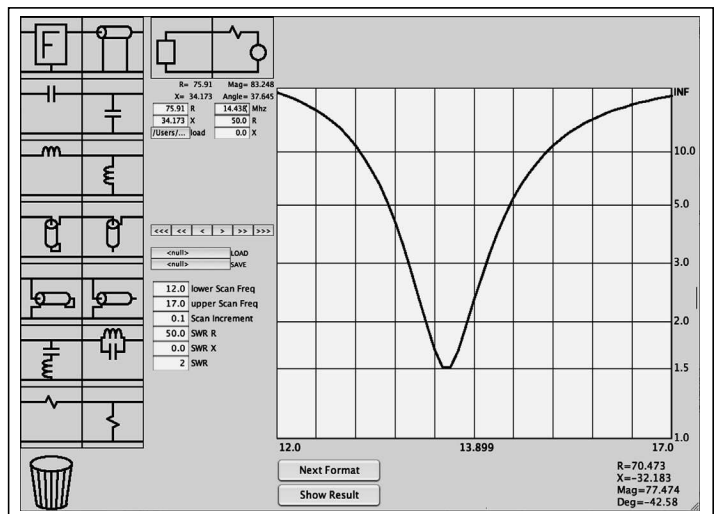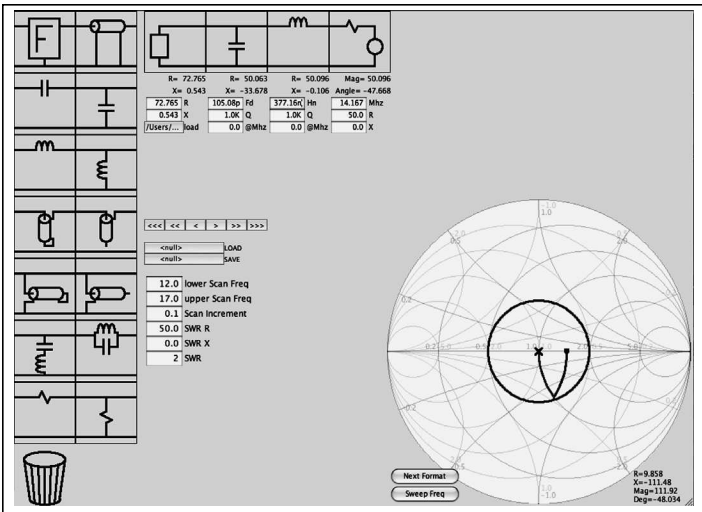


**Figure 13—Adding a transmission line to the model.**



**Figure 14—SWR plot for a practical dipole.**

**Figure 15—Extending frequency range with an L network.**



**Figure 16—Adding a transmission line.**

bandwidth has been extended to around 13.8 to 14.6, a modest improvement. However, as with any real antenna we need to introduce the feed line. Notice that Figure 15 also shows how SimSmith can draw the SWR circle. I simply set *SWR* to 2 and SimSmith drew the appropriate circle. The SWR can be around any impedance but it is typically drawn around the normalized impedance as specified by the Generator.

If I were to use best practices I would place the LC matching network directly at the feed point of the antenna. After all, I want the SWR on the feed line to be as close to 1 as possible. However, this is not what your average amateur does. Rather, the average amateur will use an antenna tuner to implement the match. So, I'll place the feed line between the match and the antenna. I'll make this feed line 1/2 wavelength long. The observant reader will note that I have retuned the LC network to account for the additional reactance introduced by the non-ideal transmission line. Figure 16 is the result. Note that the inner circle is the *path taken by the X* due to the 1/2 wave feed line. As one might expect, adding this feed line doesn't really change the SWR so I won't bother to show the plot.
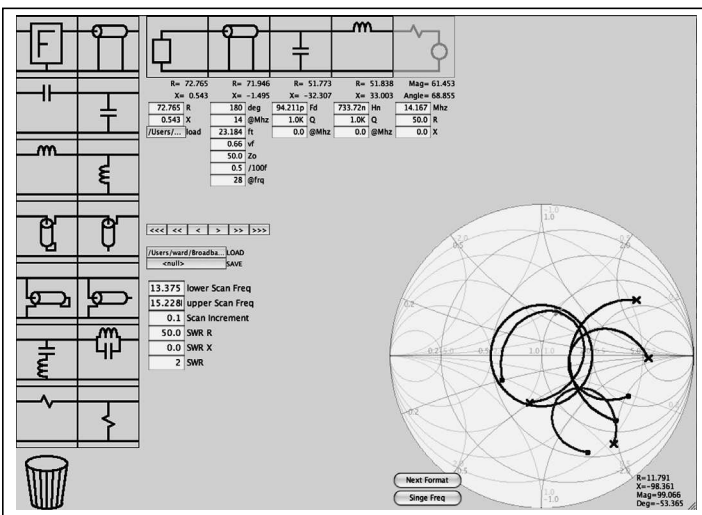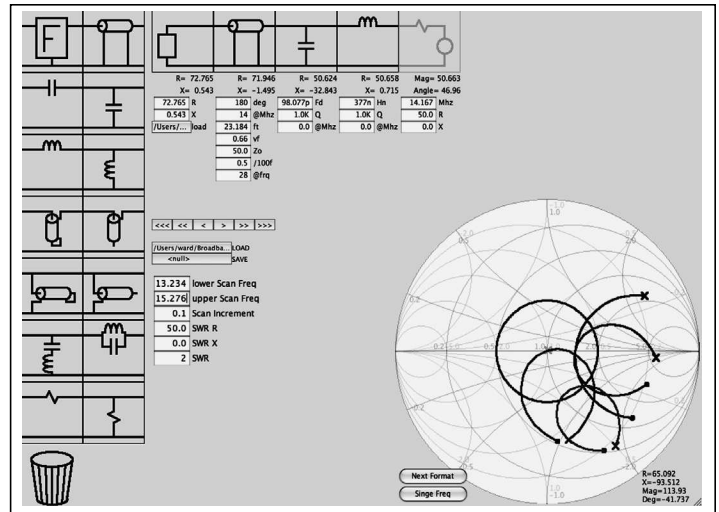
But, we aren't done. In Figure 16 I have SimSmith sweep the frequency. Note that SimSmith shows the path taken by X for all the components, not just the final result. The final trace is the one that goes through the center of the chart.

Figure 16 is very interesting because it shows that there is room for significant improvement in improving the SWR bandwidth. Specifically, if I could move the final curve I could significantly improve the 2:1 SWR bandwidth. As expected, I can move this curve simply by adjusting the LC network. I did so and Figure 17 shows the result. I have adjusted the Sweep lower and upper frequencies so that the entire curve fits within the 2:1 SWR circle. Figure 18 shows the SWR plot for this arrangement.

Notice that the 2:1 SWR bandwidth is now nearly 1.9 MHz. That's over 3 times the original.

As I said before, this is one of my favorite amateur radio examples. One reason I like it is because it shows how a simple LC match can improve the match between a simple dipole and the transmitter; nothing radical, but informative to the beginner. Another reason I like it is that it shows that the 1/2 wave feed line has little effect on the system even when the feed line is not
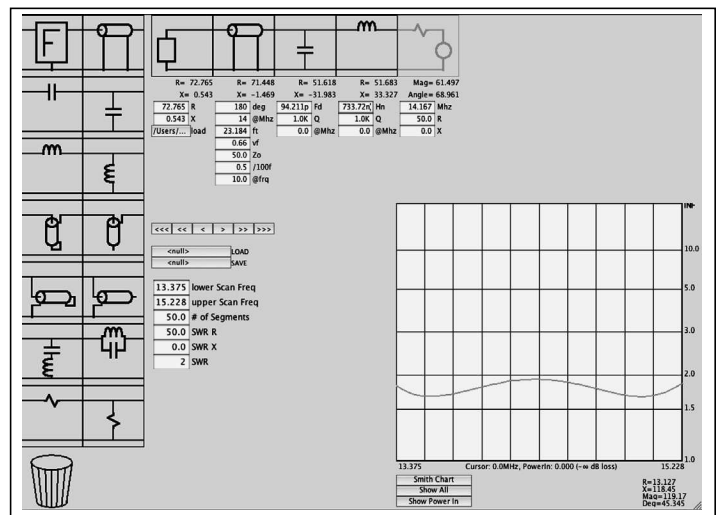


**Figure 17—Adjusting the L network to broaden frequency range.**



**Figure 18—Corresponding SWR plot for optimized dipole.**

matched. But the REAL reason I like it is because it shows that with a little exploration, one can do much better. The feed line can do more than be invisible. It can significantly improve our antenna/feed line system. And… the Smith chart is a terrific mechanism for doing this exploration!

**Wrapping Up**

Since its inception, the Smith chart has been used to perform calculations and gain insight into the intricacies of transmission lines and complex impedances. While the modern computer has supplanted the Smith chart as an aid to calculation, the Smith chart remains an important display technique. With the computer doing the heavy lifting and the Smith chart graphic providing the insight and guidance, the designer is freed to concentrate on the goals and the path toward achieving them. Modern Smith chart software represents a prime example of the melding of the traditional and the leading edge. I expect the Smith chart to hang around a long time to come. ●●

# Appendix A: Expression Syntax for F Element

(Note: Syntax for the F element changes occasionally. The reader is advised to visit www.ae6ty.com for the latest version.)

Expressions are always computed in complex arithmetic. Several unusual operators are available

    R means 'resistance' so you can write "Ra". An unadorned number works.

    X means 'multiply by j' so you can write (Ra + Xb)

    j means the same thing. You can write (5 + j10)

    w means omega. You can write "jw5u" for an inductor of 5u

    L means 'inductor with value' so you can say La or L5u

    C means 'capacitor'

    I means 'value on the left of the F block'

    T(Zl,length,Zo)  transmission line.

    "|" means 'in parallel' and is higher precedence than multiply.

Here's an example T matching equation: "(I + Ca) | Lb + Cc" and a Pi matching network might be "((I | Ca) + Lb) | Cc"

All spaces are ignored so "123   456" is the same as "123456".

A not so formal definition follows:

&lt;expression&gt;          := &lt;sum&gt;

&lt;sum&gt;              := &lt;product&gt;  or  &lt;product&gt; "+" &lt;sum&gt;  or  &lt;product &gt;"-" &lt;sum&gt;

&lt;product&gt;           := &lt;parallel&gt;  or  &lt;parallel&gt; "*" &lt;product&gt;
                       or  &lt;parallel&gt; "/" &lt;product&gt;

&lt;parallel&gt;          := &lt;unary&gt; or  &lt;unary&gt; "|" &lt;parallel&gt;
&lt;unary&gt;             := "+" &lt;unary&gt;
                       or "-" &lt;unary&gt;
                       or &lt;func&gt;

&lt;func&gt;              := "I" // I is the value to the left of the F block.
                       or "C" &lt;func&gt; or  "L" &lt;func&gt;
                       or "R"&lt;func&gt; or  "X"&lt;func&gt;
                       or "j" &lt;func&gt; or  "w"&lt;func&gt;
                       or &lt;TLine&gt;
                       or &lt;parens&gt;

&lt;TLine&gt;            := "T("&lt;sum&gt;,&lt;sum&gt;,&lt;sum&gt;")"  // T(Zl,length,Zo)

&lt;parens&gt;           := ( &lt;sum&gt; ) or &lt;number&gt;

&lt;number&gt;          := &lt;constant&gt;     or &lt;variable&gt;

&lt;constant:    :=  &lt;decimalNumber&gt; [ &lt;units&gt; ]

&lt;units&gt;      := "f" or "p" or "n" or "u" or "m" or "k" or "K" or "M" or "G" or "T" or "P"

&lt;variable&gt;   := "a" or "b" or "c" or "d" or "e" or "f" or "g" or "h"

&lt;decimalNumber&gt; := Just what you would expect