

MICROSTRIP TRANSMISSION LINES AND MODELING

Lab #5

This lab, which will lay the groundwork for the next several labs on design and construction of a microstrip-based microwave circuits, will serve as an introduction to microstrip transmission lines and the concept and effects of transmission line discontinuities. Over the course of the next several lab sessions, you will design, build, and test an amplifier, mixer, or oscillator for signals above 1 GHz. The basic structure of the next several lab sessions includes a study of the microstrip transmission lines to be used and the active device to supply the amplification (Agilent AT-41485, a microwave bipolar junction transistor), the design of appropriate bias and matching networks, and finally full characterization of the circuit you have designed, including its gain, frequency response, input and output reflection coefficients or impedances, noise performance, and large-signal (intermodulation) performance. In this lab session, we'll focus on the measurement and assessment of the performance of microstrip lines for your circuit.

Please watch the demonstration videos online before lab; they will help you understand the procedures more clearly.

Microstrip Line Characterization

In this subsection of the lab, measured data from a test microstrip line will be taken using the network analyzer. This data will be used in subsequent sections to obtain the behavioral parameters of the line (e.g. characteristic impedance, loss) and from this the parameters of the substrate material (e.g. permittivity, conductivity, loss tangent, etc.) can be determined. These substrate parameters will be used in your circuit designs in later labs, as the same substrate material will be used. **Note: make sure you take notes as you go along, so that you can do these types of measurements later without instructions!**

- a. To automate the transfer of measurement data from the analyzer to the CAD software, we'll use the IC-CAP software package. While we have used Matlab scripts to collect data in previous labs, IC-CAP provides a more convenient framework for making measurements, particularly as the measurements become more complicated (e.g. when coordination between multiple instruments is required). Since this is (likely) the first time you've run IC-CAP, you will need to set up the licensing information. Select a "specify a network license server" and enter "1712@linux3-lic.crc.nd.edu" for the server.
- b. For IC-CAP to be able to make measurements, you need to set up the connection between the software and the measurement hardware. Click on the "GPIB" icon in the toolbar (looks kind of like an oscilloscope above an arrow). In the window that appears, click on the "pencil" in the "Available connections" box (on the left), and select (with a checkmark) the "VISA IO Libraries Interfaces" in the pop-up window and then close the window. To make the instruments available for IC-CAP to control them, we need to establish a set of interface connections. To do this, click on the "circle arrow" button in the "Active instrument connections" box. This will cause the interface bus to be scanned for active instruments; if things go right, you will see some instruments be identified and added to the list. When this completes, close the hardware setup window.
- c. To read the s-parameters from the analyzer, a pre-configured measurement setup is available on the course web page (lab5.mdl). Load this into IC-CAP (under the "File" menu), and click on the "microstrip" device (the icon looks like a FET). Select

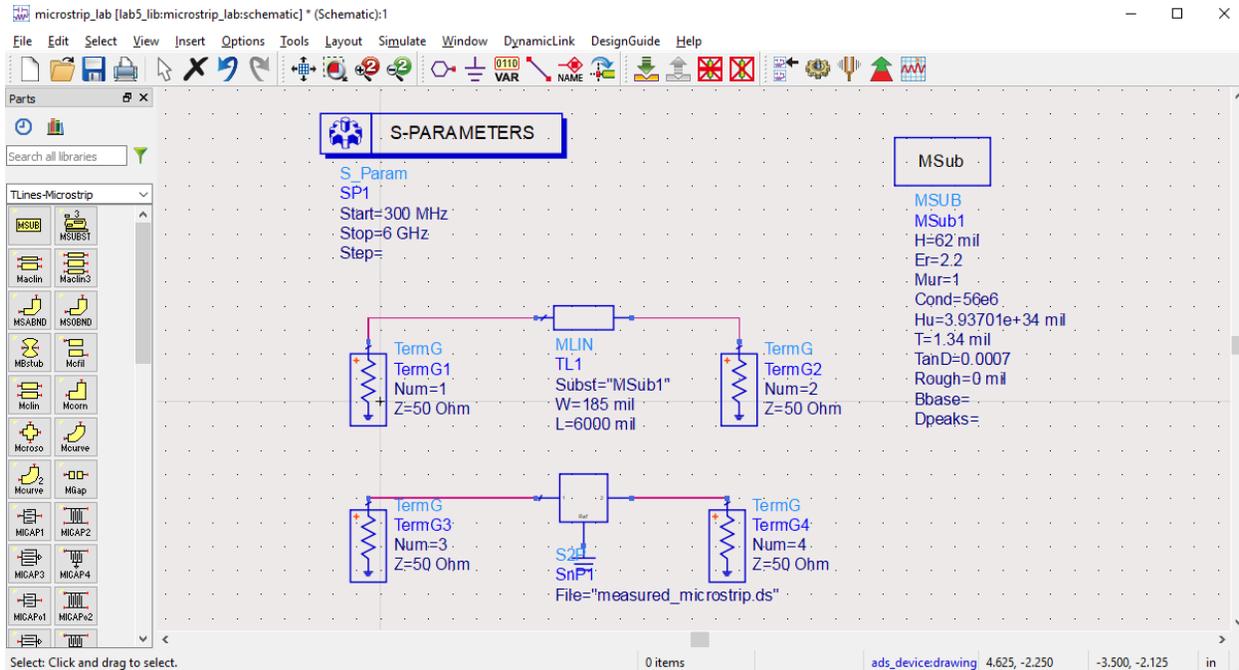
- “microstrip_s” in the left-hand window pane to show the pre-configured measurement setup.
- d. To set up measurements using IC-CAP, you enter the desired settings in IC-CAP (not on the front panel of the instruments). For a typical network analyzer measurement, this means we set the start and stop frequencies and number of points in the “freq” block on the “Measure/Simulate” tab. In our case, confirm that the start frequency is set to 300 MHz and the stop frequency is 6 GHz, with 201 points. Also check that the IF bandwidth (under the “Instrument Options” tab) is set to 1 kHz for noise suppression, and that “Cal Type [Hardware/None]” is set to “H”. When this is set correctly, on the “Measure/Simulate” tab, click “Calibrate.” This will download the measurement setup into the analyzer and prepare it for calibration (but paradoxically will not actually do a calibration).
 - e. From the front panel of the network analyzer, perform a 2-port calibration, using the precision 3.5 mm calibration kit and male connectors on both network analyzer ports. Note: you will need to push the “local” button on the analyzer to enable the front panel (when the instrument receives commands from the computer, it “locks out” the front panel). For the calibration, press “Cal” then select the “Basic cal” softkey. Since we are using SMA connectors, choose “APC 3.5” as the connector type (this is the formal name for what are often called SMA connectors), choose “85033D/E” as the calibration kit, and indicate that both ports 1 and 2 are male (this is the end of the cables we have). The calibration type should be set to “SOLR” (short, open, load, reciprocal) to match the standards we have available. Pressing next will move to the calibration measurement screen; using proper connection techniques, connect and measure each of the required standards to each port. Note that the short, open, and load must be measured on each port; the “unknown thru” connects the two ports together for the measurement. After all the measurements are complete, press “Finish.” The analyzer will ask you for the “adapter delay” associated with the through and present you with an estimate of the adapter delay (it may ask the same question multiple times—this is annoying, but ok). For the through we are using, make sure the value that the analyzer suggests is 0.052 ± 0.04 ns. If it is, click “OK”; if not, something has gone wrong (perhaps a loose connector or the standards got mixed up during measurement) and the calibration needs to be re-done.
 - f. After calibration, you *must* save (or re-save) the calibration, and *make a note of the state filename used*. Do this by pressing the “Save/Recall” button, followed by the “Save state” softkey. Check to be sure that “Save type” is set to “State + Cal Data.” Pressing “Save State” will save the calibration with the name indicated in the box on the button—this usually over-writes the previous calibration. If you want to preserve the previous calibration, using “Auto Save” saves the file in an automatically-selected name (usually something like atXXX.csa). Either way, make a note of the name used for saving the calibration; it must be entered (exactly as it appears, including case sensitivity and extension) in IC-CAP in the “Instrument options” tab, in the “Cal/State File Name” field. Note 1: the analyzer will “increment” the filename when you push the “auto save” button; the value shown is the *next* filename it will use. You need to record the value of the filename it was showing before you pushed the button. Note 2: IC-CAP says that it accepts “.sta” files only, but it is kidding. A .csa file is required.
 - g. With this calibration established, measure the s-parameters of the microstrip test line (this board consists of a single microstrip line with 3.5 mm (sma) connectors on both

- ends; this is not the microstrip stub board we've used in previous labs). To measure and collect the data, connect the board to the network analyzer, and click “Measure” on the “Measure/Simulate” tab of IC-CAP. IC-CAP will take over control of the VNA, and collect the data. To check that the measurement was successful and that the data is reasonable, click the display plots icon in the toolbar (looks like 3 transistor IV curves). This will bring up plots of the measured S_{11} , S_{12} , S_{21} , and S_{22} (magnitude and phase) for inspection. If satisfied with the data, save the file (to a .mdl file) for importing into ADS for analysis and modeling, being sure to note the filename used. If everything has worked OK, close IC-CAP.
- h. Measure and record the physical length and width of the top conductor of the microstrip line as accurately as possible (use the dial calipers for this measurement); these dimensions are needed for computer simulations of the microstrip line. When making this measurement, be careful not to scratch the surface of the microstrip line or to gouge the edge of the line with the calipers.

Microstrip Line Modeling

In this portion of the lab, a model of the microstrip test line will be built and simulated to compare with the actual measured results. The real-world effects of discontinuities will be examined, and a simple approach for modeling these effects will be implemented.

- a. Start ADS and make a new workspace (give it a descriptive name so that you can remember it). Add a schematic page to the workspace (click on the icon that looks like a resistor and capacitor in the toolbar), and give it a descriptive name (anything is better than “cell_1”). To import the measured data, choose “Tools/Data File Utilities/Data File Tool”. In the dftool window, choose the input file format to be ICCAP, and browse for the .mdl file you saved the measurement data in. Enter a dataset name to hold this data within ADS in the lower part of the window, and click “Read File.” If all goes well, the status box will indicate the data has been read successfully. Close the data file tool window.
- b. To develop a model for the microstrip line (so that we can design our own circuits using this board substrate material), we will start by examining the expected s-parameter results for a microstrip line, and comparing those to the measurements. This is similar to what we did with lumped-element components previously, but based on 2-port measured data and transmission line components instead of ideal lumped elements. To do this, we will build the schematic page shown in the screenshot below. In the “Parts” window, select the “TLines – Microstrip” palette. This will change the palette on the left-hand side of the schematic window to include a number of microstrip components.
- c. Since the microstrip test line consists of a single section of microstrip line, select the MLIN element from the palette. Place one of these elements in the schematic window by clicking; the component will be placed where the outline is shown in the schematic window. Change the L and W parameters to match the measured values for the width and length of the physical board tested. Note: 1 mil = 0.001 inch.
- d. To provide the simulator with information about the microstrip substrate, add an “MSUB” element to the schematic page. Choose “Standard ADS Layers, 0.0001 mil layout resolution” in the pop-up box. To set up the physical parameters of the microstrip substrate, double-click on the MSUB element to bring up a parameter editing box. Change the parameters to the nominal values from the manufacturer’s data sheet in Table 1 below.



Screenshot: Initial simulation of microstrip transmission line

Table 1: Datasheet values for substrate material

| | |
|-----------------|---|
| H = 62 mil | thickness of substrate |
| Er = 2.20 | relative permittivity |
| Mur = 1 | relative permeability |
| Cond = 56.2e6 | Conductivity |
| Hu = 3.9e34 mil | cover height – use large value to neglect cover effects |
| T = 1.34 mil | conductor thickness |
| TanD = 0.0007 | dielectric loss tangent |

- e. Since we're interested in the s-parameters of this line, we need to add the appropriate simulation elements to the circuit. From the “Simulation – S_Param” palette in the “Parts” window, add two “TermG” elements to the circuit; these are equivalent to the test ports on a network analyzer.
- f. To control the simulation, add a "SP" element to the schematic page as well; this sets the frequency range and other simulation options. Double-click on the “S-Parameters” item. Set the start frequency to 300 MHz, the stop frequency to 6 GHz, and the number of points to 201 (to match the measurements that were taken).
- g. All of the required elements are now added to the schematic; what remains is to wire them together and set the physical dimensions and parameters for the circuit. To wire the circuit, click on the "wire" icon (or press control-w) and click on successive pairs of nodes to connect them.
- h. To facilitate comparison of the measured data and the simulation results, it is helpful to “import” the measured data into the simulation. While we could use the “DAC” element we used in lab 1, for s-parameter data there is an easier way. Add two more “TermG” elements to the schematic page (from the “Simulation-S_Param” palette), and place an SnP element from the “Data items” palette between them (icon looks like a box with an N on it). The SnP is a funny “placeholder” component; it can have

any number of ports, but defaults to 1. To change it to a 2-port element (to match our board, which has 2 ports) change the “n” in SnP to a 2, turning it into an S2P element. Be sure to do this for the light blue component identifier, not the dark blue instance name. Wire port 1 of the S2P to the TermG with Num=3, and port 2 of the S2P to the TermG with Num=4. Ground the “ref” terminal of the S2P element. To select the data that the S2P element should import, double-click on the S2P element and browse for the dataset name you used in the data import step (step a). Also, set the file type to “Dataset” (the default is Touchstone). In simulation, the s-parameters associated with ports 3 and 4 will be the measured data obtained previously from the test line measurements.

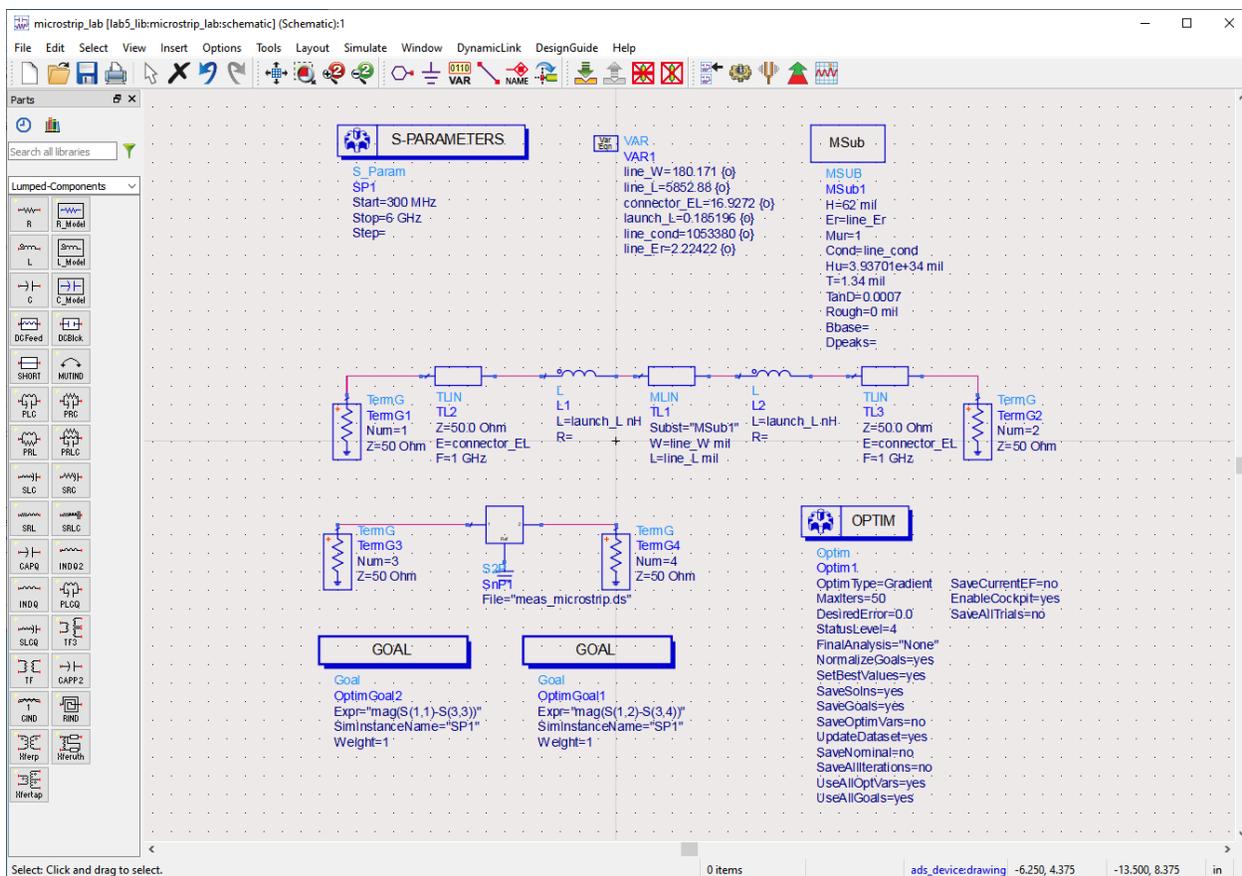
- i. To run the simulation, click on the "gear" icon (or press F7); a simulation status window should appear and indicate when the simulation has completed. A blank graph window will also (probably) pop up when the simulation is complete.
- j. Plot the results of this simulation. Note that the schematic page has four test ports, so the simulator calculated a 16-element matrix of s-parameters at each simulation frequency. Of these, the only physically meaningful elements are S_{11} , S_{12} , S_{21} , S_{22} , S_{33} , S_{34} , S_{43} , and S_{44} . The others are all cross-terms between the two separate circuits, and will all be zero. Note that S_{33} is the measured data for the line that is simulated as S_{11} , that S_{43} is the measured data for the simulation of S_{21} , and so on. How well does the simulation and the measurement match? Are there any clear similarities or clear discrepancies? Can you detect any patterns in the results? **Be sure to document your work as you do it in your lab notebook.**

Except under unusual circumstances, the measured data and the simulation data generally won't match well on the first try. This can be due to several physical effects: i) not including all of the circuit elements in the simulation, ii) not modeling, or modeling incorrectly, the effects of discontinuities, and iii) inaccurate substrate parameter values. Each of these potential problems can be fixed by increasing the sophistication of the model. For example, in the simple model used so far, the connectors on the board and the discontinuity between the coaxial geometry of the cable and connector and the planar geometry of the microstrip on the board have not been accounted for. Also, the nominal dimensions and substrate parameters have been assumed to be exactly correct; in reality, these are only approximately known.

- k. To include the effects of the finite length of the connectors, insert a length of ideal transmission line (TLIN) between the test ports and the microstrip line. This line can be found in the “TLines – Ideal” component palette. Set the reference frequency to 1 GHz; any frequency could be chosen, but this is a convenient choice.
- l. To account for the discontinuity between the coax connector and the microstrip board, insert an inductor (found in the “Lumped – Components” palette) between each of the TLIN elements and the microstrip line. At modest frequencies, this is a reasonably accurate model to account for the transition of the electric and magnetic fields from the cylindrical field distribution in the cable and connector to the quasi-planar distribution in the microstrip line.
- m. Since it is difficult to determine with sufficient accuracy the electrical length of the connector or the inductance of the discontinuity from first principles, the optimization capabilities of ADS can be used to find these values. To do this, insert a VAR element (by choosing the VAR icon next to the ground icon on the toolbar) in the schematic. Insert two variables in the VAR element—one for the electrical length

- of the connector and one for the inductance of the discontinuity. Choose descriptive names for these variables; generic names like “L” are used internally by the simulator and will generate confusing error messages. To start with, select “reasonable” estimates of the electrical length and inductance; 20 degrees of electrical length (at 1 GHz) and 0.5 nH of inductance is a decent starting point. Insert the variable names you chose for the electrical length of the connectors and inductance of the discontinuities in the schematic elements. Be careful about the units; if your VAR element contains units then the schematic element should not, and visa-versa.
- n. To account for possible measurement uncertainty in the line dimensions and parameter variation in the substrate, add additional variables to the VAR element. Entries for the permittivity (ϵ_r in the MSUB) of the line, the conductivity of the line (Cond in the MSUB), microstrip line length, and microstrip width should be added. Set these initial values to those used in the table previously, and modify the schematic elements to use the VAR element variables instead of hard-coded constants. As with the electrical length and discontinuity inductance, be sure to choose descriptive names to avoid name conflicts.
 - o. To set up the optimization, limits on the variation of all of the parameters must be established. To set the limits for the variation, double-click the VAR element. For each variable, select “Tune/Opt/Stat/DOE Setup...” and set the “Optimization Status” to “enabled” in the “Optimization” tab. Enter appropriate values for the minimum and maximum values. Be sure that the nominal value lies between these two limits, and be careful to be consistent with the units. A reasonable range for the electrical length is something like 12 to 35 degrees, for the discontinuity inductance is from 0.1 to 0.7 nH, for the permittivity is 2.1 to 2.3, for the conductivity is from $1e5$ to $56e6$, for the length is approximately ± 0.1 inch (100 mils), and for the width is approximately ± 5 mils.
 - p. The final step in setting up the optimization process is to give the simulator the criteria for a good solution – it needs to be told what the best solution would look like so that it can determine if a variation in the parameters has improved the situation or made it worse. To do this, add two “Goal” elements from the “Optim/Stat/DOE” palette. In each goal element, a mathematical expression that indicates the “quality” of the solution must be provided; the simulator will attempt to make this expression evaluate to a value within the “acceptable” range. Since the goal of this optimization is to make the simulation model match the measured s-parameters, two expressions that could be used are $\text{mag}(S_{11}-S_{33})$ and $\text{mag}(S_{21}-S_{43})$. In both cases, a perfect match between the model and the measurement would be signaled by these expressions evaluating to zero; any imperfections in the match will cause these expressions to evaluate to a positive value. Double-clicking on the Goal element will allow you to set it up. Enter one of these expressions in each of the goal elements, select the Analysis from the drop-down menu (usually SP1), and select an optimization type of “<” so that the optimizer will try to minimize the goal expression. Set the “Max” field to 0.001 so that the optimizer will aim for a very good fit.
 - q. Add an “Optim” element to the schematic to control the optimization process. Double-click on the “Optim” element, make sure that “use all goals in design” is checked, set the optimization type to gradient, and set the number of iterations to 50. The screenshot below depicts a schematic configured for optimization. To perform the optimization click the “Christmas tree” icon. The optimization cockpit window

- will open, and you can watch the progress as it tries to identify the best set of parameters. When it stops, check these values over to make sure they appear to be reasonable values. If they are, close the optimization cockpit and choose “Update design.” This will put the values found in the optimization into the schematic automatically. To verify that the optimization worked, you can run a simulation (without optimizing) by clicking the “gear” icon.
- Using the same data display window format as previously, how do the simulation and measurement data compare now? Do they match better? Are there any discrepancies or points of significant disagreement between the two? If so, it may be necessary to optimize again, either with different starting values or with different bounds in order to find a better solution. Be sure to evaluate your work as you do it, and track your progress in your lab notebook.
 - Once acceptable agreement between the measurements and model have been achieved, examine the parameters and dimensions obtained by the optimizer. Do these seem reasonable? Make note of the parameters and dimensions obtained; these will be used for the design of the project design. **Be sure to document a comparison of the measured and modeled microstrip test line in your lab notebook.**



Screenshot: Schematic with optimization for microstrip line, including discontinuities and connector electrical length.

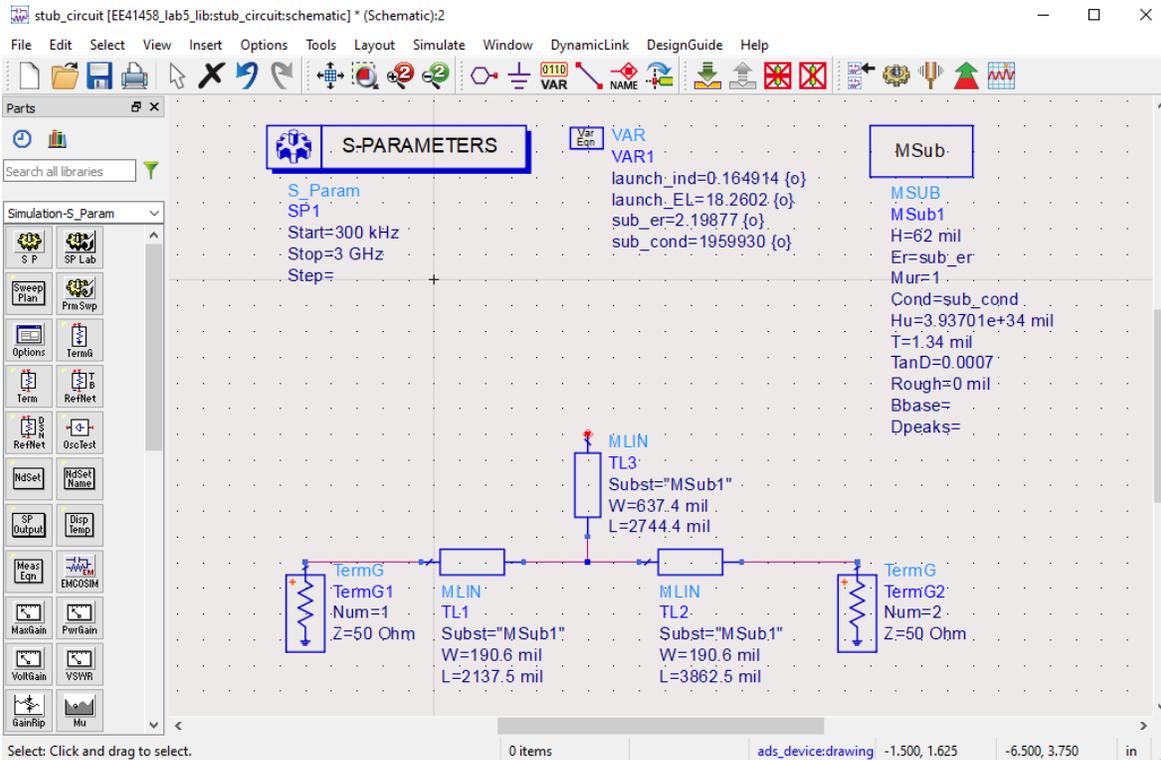
Microstrip Stub Filter Circuit Simulation

In this part of the lab, the behavior of the microstrip stub filter circuit that was measured in labs 2 through 4 will be simulated, using the substrate parameters obtained in the last section.

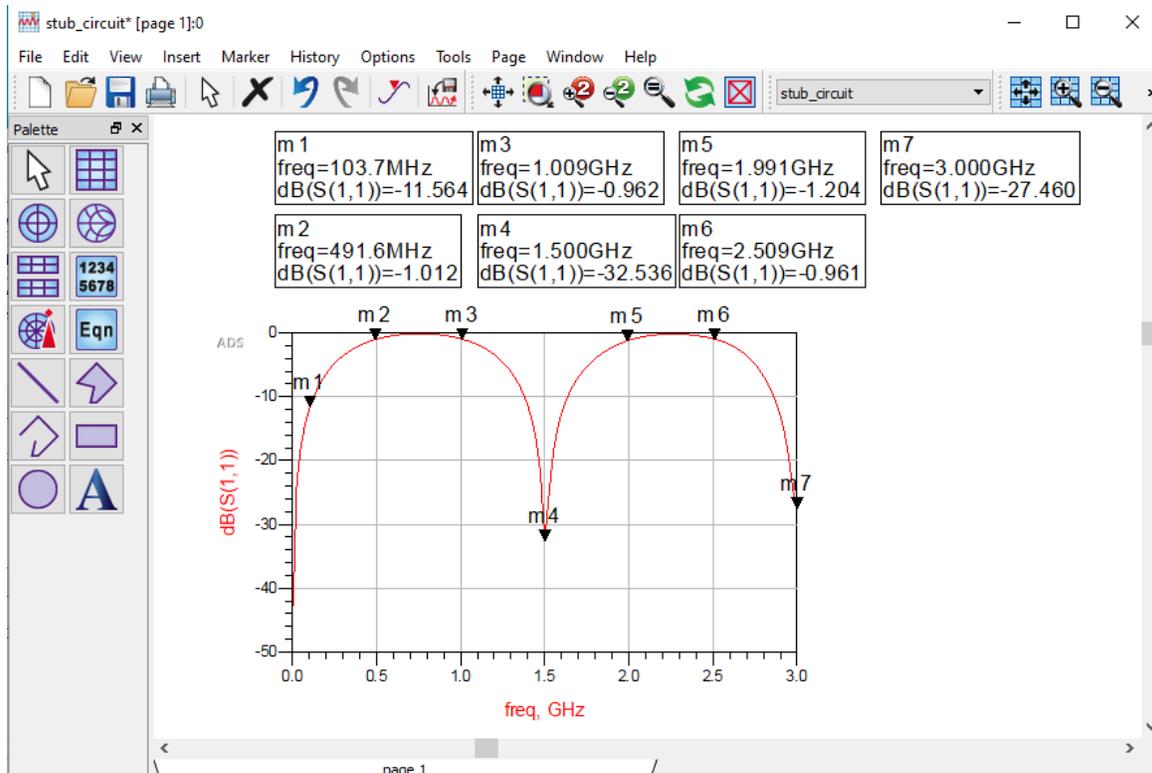
- a. In the ADS schematic window, select “File/New” to create a new schematic page for simulating this circuit. Name the new schematic something meaningful and click OK. A blank schematic window should open. Construct a CAD model of the microstrip stub filter circuit in the schematic window. This can be done using the following steps:
- b. Since the microstrip stub filter circuit consists of three sections of microstrip line, select the MLIN element from the “TLines – Microstrip” palette. Place three of these in the schematic window.
- c. In the actual circuit, the fat stub is oriented at 90° with respect to the main line; to make the schematic reflect this, select the element you'll use for the stub portion of the circuit by clicking on the element (this will outline the selected element in a black box) and click on the "rotate" icon in the tool bar, or press ctrl-r. When satisfied with the orientation, press escape to exit rotation mode.
- d. Copy the MSUB and VAR elements from the previous design to the schematic page. This is easiest by first opening the previous design, and copying/pasting them into the current design.
- e. Add s-parameter terminations (TermG) and an s-parameter simulation control (SP), and wire the circuit together. Set the simulation to run from 300 kHz to 3 GHz with 201 points to match the measurements made previously.
- f. To set the dimensions of the three transmission line segments, double click on each and set the width and length as appropriate. For the filter circuit, these values are:
 - port 1: $W = 190.6$ mil, $L = 2137.5$ mil
 - stub: $W = 637.4$ mil, $L = 2744.4$ mil
 - port 2: $W = 190.6$ mil, $L = 3862.5$ mil

The circuit should look similar to the screenshot below. Note that the stub has one end that is not connected to anything. This is correct; the stub is open-circuited. We are also neglecting the impact of the connectors in this simulation.

- g. Run the simulation, and graph the results. To read numeric values from the plot, use the marker function. Select Marker/New, and click on the appropriate plot trace to add a marker. Add a marker to the s_{11} trace, and record the simulation's prediction of $|s_{11}|$ for the frequencies measured in previous labs (approximately 100 MHz, 500 MHz, 1 GHz, 1.5 GHz, 2 GHz, 2.5 GHz, and 3 GHz; the frequencies will not match exactly due to the frequency sweeps being slightly different). Save the plot under the file menu. How do these values compare to those obtained in previous labs? Discuss any discrepancies and their possible origins. As always, be sure to document your work in your lab notebook.



Screenshot: Microstrip stub filter schematic example.



Screenshot: typical microstrip stub filter circuit simulation response.