Low Cost, Compact Microwave Reflectometer for Non-Destructive Testing

Capstone Senior Project Report

completed by

Keith Bruno
Matt Rangen

Submitted to:
Dr. Huggins
May 9, 2005

Bradley University
Department of Electrical and Computer Engineering
EE451: Senior Design

TABLE OF CONTENTS

ABSTRACT
A low cost reflectometer for non-destructive is an instrument capable of measuring the phase and magnitude of the reflection coefficient of an unknown load. When used for non-destructive testing, an open-ended coaxial cable is inserted in a liquid or pressed against a material. Through additional analysis, the reflection coefficient from the material can be related to physical properties of the material such as conductivity and the dielectric constant. The reflectometer consists of a six-port passive microwave circuit integrated with a PC workstation. The PC workstation samples four output voltages for each load. The complex algorithm implemented on the PC requires the use of five known calibration loads which then allows the phase and magnitude of the reflection coefficient of the unknown load to then be determined.

1. PROJECT DESCRIPTION & OBJECTIVE

To analyze a material by measurement of its permittivity, an expensive and non-portable spectrum analyzer could be used as shown in Figure 1. However, such systems are too large for use in the field and too expensive (in excess of $100,000) for applications requiring low cost instruments. An alternative approach is to use a six-port reflectometer combined with a laptop computer. It is much cheaper since it based on a compact microstrip six-port microwave junction. The cost of such a hybrid circuit, integrated with detectors and voltage controlled oscillator (VCO), would be a few hundred dollars. In addition, the software only adds to development cost but not the Bill of Material. Finally, the system can be battery powered, compact, and lightweight, ideal for use in the field.

The objective of this project is to design a prototype of a low cost reflectometer for non-destructive testing. The reflectometer operates at 6 GHz and consists of a six-port network microwave junction with external VCO and microwave detectors. For a given load, the detectors generate voltage signals that are sampled by digital oscilloscopes. The reflection coefficient of an unknown load can be found through complex algorithms, implemented in MATLAB, by making multiple measurements of the detectors signals.

This paper discusses the design and implementation of the system. Section 2 gives a short background followed by a functional description of the system at the block diagram level in Section 3. Section 4 lists the division of tasks. Theory of operation is explained in Section 5 with Section 6 giving details of the radio frequency (RF) theory and design. Section 7 follows through simulation and testing of six-port and Section 8 recommends future work. Sections 9, 10, and 11 give specifications, standards, and patents, respectively.

Figure 1: Spectrum Analyzer

2. BACKGROUND/HISTORY

The use of a six-port junction as a network analyzer was considered by Glenn F. Engen in the late 1970's and described in a series of papers. He offered on optimized design consisting of four 90º hybrids and one Lange coupler which is described in his paper titled in "The six-port reflectometer: An alternative network analyzer," published in the IEEE Transaction of Microwave Theory and Techniques [ref 2].

An example of the use of a six-port reflectometer for nondestructive testing is the measurement of water content in cheese curd [ref 4]. In this application, the system allowed the real-time test of moisture content of cheese early in the manufacturing process to determine if water needs to be added or not to produce the best product.

3. FUNCTIONAL DESCRIPTION & SYSTEM BLOCK DIAGRAM

The system consists of an RF subsystem and computer software subsystem as shown in Figure 2. The RF subsystem is designated by a square dashed boundary in Figure 2. The RF component will be discussed followed by the software component.

Inputs/Outputs

The six-port junction is an optimum passive micro-strip design with quadrature hybrids. The VCO produces an RF reference signal and is controlled by the DC voltage, V_c. The signal from the VCO (a_{in}) enters the six-port junction at port-S and, in general, is split among all the other ports. At port-L, two signals exist simultaneously. One exits the port and is the incident wave (b_L) on the unknown load. The other enters the port and is the reflected wave (a_L) from the unknown load. The ratio of the reflected wave to the incident wave (relative to the load) is the reflection coefficient. The external detectors denoted: V_1, V_2, V_3 and V_4 on the ports convert the RF signal to low frequency voltages proportional to the RF power detected. The power detectors are connected to two oscilloscopes that read the power measurements. These signals are measured and stored in the PC where a complex algorithm computes the reflection coefficient of the unknown load. Table 1 summarizes the inputs and outputs to the RF block.

The system is controlled by software (MATLAB) installed on the PC shown outside of the junction in Figure 2. These voltages are sampled for five calibrating loads and then the unknown load. These measurements are used in a series of computations to find the reflection coefficient of the unknown load. Table 2 summarizes the inputs and outputs of the software.

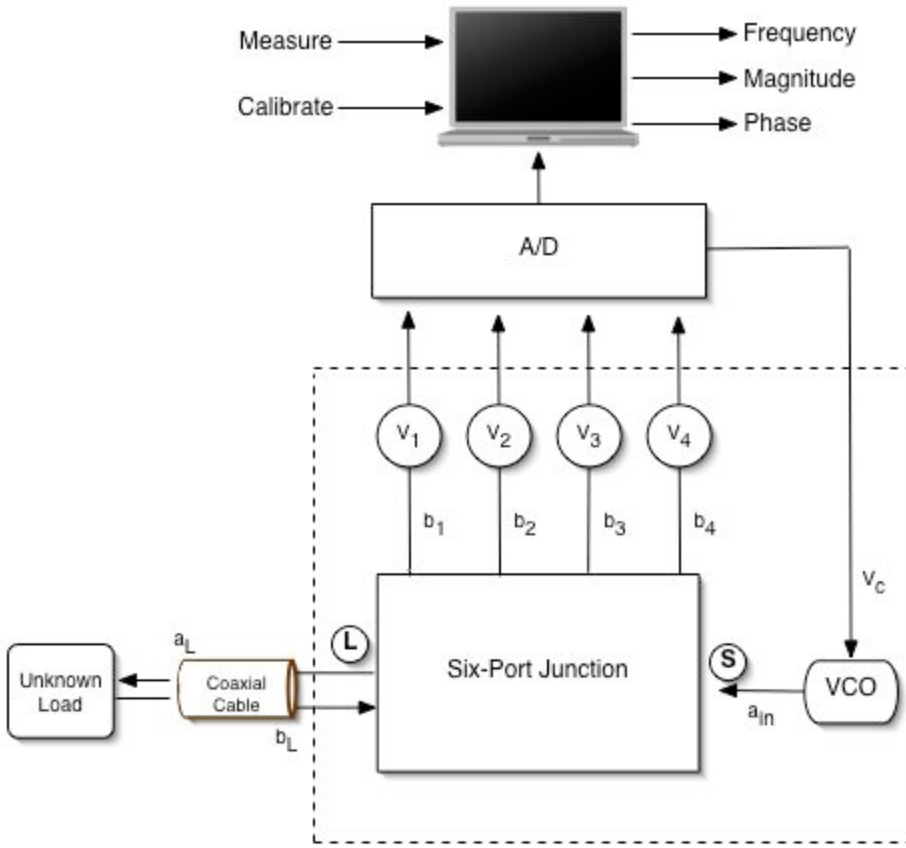


Figure 2: System Block Diagram

Table 1: RF I/O and Mode of Operations

Inputs	Outputs	Operations
Reflected Signal		Created from the incident signal which is used to calculate the Reflection Coefficient.
	V_1, V_2, V_3, and V_4	Sampled voltages to calc. Reflection Coefficient.
	Incident Signal	Sent to the load.

Table 2: Software I/O and Mode of Operations

Inputs	Outputs	Operations
V_1, V_2, V_3, and V_4		Sampled voltages to calc. Reflection Coefficient.
	Magnitude	Magnitude of the Reflection Coefficient.
	Phase	Phase of the Reflection Coefficient.

Modes of Operation

There are two modes of operation: calibration and measurement. Before a measurement of an unknown reflection coefficient can be made, the system must first be calibrated. The user must connect a specific calibration load. The program then samples the voltage levels (V_1, V_2, V_3 and V_4) and the four readings are recorded. This is repeated for each of the five specific calibration loads. An algorithm then uses the entered voltage readings to calculate the system parameters needed for measurement. This entire process is summarized in Figure 3.

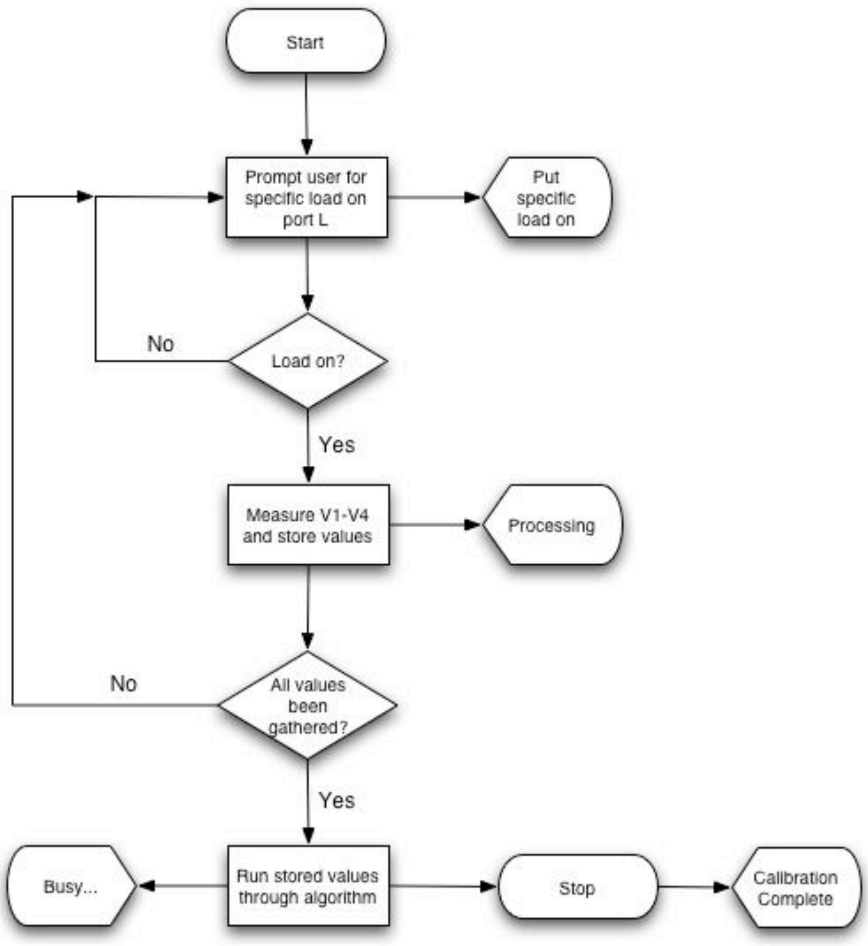


Figure 3: Calibration Flowchart

Once the calibration has been completed, the user will then proceed to the measurement mode of the software. The user connects the unknown load at port-L. The system will then proceed to take measurements of the voltage levels (V_1, V_2, V_3 and V_4) and the four readings are recorded. An algorithm then processes the entered values in correlation with the calibration results. The user is prompted with results showing the magnitude and phase of the load. The overall measurement process is shown in Figure 4.

Figure 4: Measurement Flow Chart

4. DIVISION OF TASKS

The design and implementation of the system required the design of the hardware, consisting mostly of RF components, and software. The hardware leader was Keith Bruno and he designed and implemented the optimal micro-strip six-port passive. His specific deliverables are listed in Table 3. The software leader was Matthew Rangen and he wrote the MATLAB code for the calibration and measurement calculations. His specific deliverables are also listed in Table 3.

Table 3: Division of Tasks

Matthew Rangen:	Keith Bruno:
-Research calibration & measuring equations	-Design & simulate 90º Hybrid
-Develop flow chart for calibration & measuring	-Design & simulate Lange Coupler
-Implement equations in MATLAB	-Design & simulate six-port
-Test programming code	-Design & simulate calibration loads
-Integrate MATLAB & oscilloscopes	-Design microstrip layout
	-Fabricate six-port components

5. THEORY OF OPERATION

At the start of this project, several different methods for calculating the reflection coefficient of an unknown load using a six-port were considered. J.D. Hunter and P.I. Somlo presented an exact approach in the paper titled "An Explicit Six-Port Calibration Method using Five Standards," [ref 3] and their process was used in this project. It designated the use of five different standards to calibrate the six-port network analyzer. As proposed in their paper, they suggested the use of one matched load and four different lengths of offset shorts. The five loads generate 20 power measurements, which are used to calibrate the six-port via a complex algorithm. Once calibrated, 4 additional power measurements allow the unknown reflection coefficient to be computed. These procedures are described in the next subsections and the exact code is given in Appendix 2.

Calibration Process

The calibration requires five known loads to be used. These five loads had a known reflection coefficient and were hard coded into the program. Once these power measurements are made, ratios are determined using port-4 as the reference as shown in Equation (1)

$$p_{i,k} = P_{i,k} / P_{4,k} \qquad \textbf{(1)}$$

where i (= 1, 2, 3) is the port number and k = 1, 2, 3, 4, 5 correspond to the calibration load with k = 5 denoting the matched load.

Once these ratios are computed, they are stored into the program. Next another ratio is computed using the results from above, with the fifth load (matched load) as the reference. These set of solutions are referred to as the T ratios. The T ratios are then computed and stored into memory, which is shown in Equation (2)

$$T_{S,R} = p_{i,k} / p_{i,5} \qquad \textbf{(2)}$$

where i (= 1, 2, 3) is the port number and k = 1, 2, 3, 4, 5 correspond to the calibration load with k = 5 denoting the matched load.

The next step in the process uses the known gammas of the calibration loads to compute a complex ratio shown in Equation (3)

$$\Gamma_k / |\Gamma_k|^2 = c_k + js_k \tag{3}$$

where k = 1, 2, 3, 4, 5 correspond to the calibration load with k = 5 denoting the matched load.

These ratios are used to compute the variable γ and as shown in Equation (4)

$$\begin{aligned}\gamma_i &= (c_j - c_k)[(s_i - s_j)(c_k - c_l) - (c_i - c_j)(s_k - s_l)] \\ &+ (c_k - c_l)[(s_l - s_i)(c_j - c_k) - (c_l - c_i)(s_j - s_k)]\end{aligned} \tag{4}$$

where i (= 1, 2, 3, 4), j (=i+1), k (=i+2), and l (=i+3). The reader will notice that j, k, and l refer to more ports than there are available. This is an error in the original paper so to fix this a wrap around is used, which creates i = i+4.

Once γ is computed, e, f, g. and h can then be calculated by taking the summations of the T ratios as shown in Equation (5)

$$\begin{aligned}f_n &= \sum_{k=1}^{4} T_{n,k} * \gamma_k \\ g_n &= \sum_{k=1}^{4} T_{n,k} * \gamma_k * c_k \\ h_n &= 2 * \sum_{k=1}^{4} T_{n,k} * \gamma * s_k \\ e_n &= \frac{\sum_{k=1}^{4} (T_{n,k} - 1) * \gamma_k}{|\Gamma_k|^2}\end{aligned} \tag{5}$$

where i (= 1, 2, 3) is the port number and k = 1, 2, 3, 4 which corresponds to the calibration load.

Once these summations and values have been tabulated, ξ can then be calculated. The computation of ξ requires values or T for 2 of the 4 ports. This allows for flexibility in the code and an error check. After this value has been calculated, it may be used to compute the variables M and N as shown in Equation (6).

$$\begin{aligned}\xi_1 &= (g_s * h_d) - (h_s * g_d) \\ \xi_2 &= (h_s * f_d) - (f_s * h_d) \\ \xi_3 &= (h_s * e_d) - (e_s * h_d) \\ \xi_4 &= (g_s * f_d) - (f_s * g_d) \\ \xi_5 &= (g_s * e_d) - (e_s * g_d)\end{aligned} \tag{6}$$

where s and d correspond to two of the four ports selected from e, f, g, and h.

M and N are used to calculate the magnitude of A_4 using the values of ξ as shown in Equation (7) and Equation (8)

$$M_{i,j} = \frac{(\xi_1^2/2) - \xi_2\xi_3 - \xi_4\xi_5}{\xi_2^2 + \xi_4^2}$$
$$N_{i,j} = \frac{\xi_3^2 + \xi_5^2}{\xi_2^2 + \xi_4^2} \quad \textbf{(7)}$$

$$|A_4|^2 = M_{i,j} - (M_{i,j}^2 - N_{i,j})^{(1/2)} \quad \textbf{(8)}$$

where i and j are reference to the port and the load and A_4 is in reference to port four. The magnitude of A_4 can then be split up into its real and imaginary numbers, which is shown in Equation (9).

$$a_4 = \frac{|A_4|^2 * \xi_{2+}\xi_3}{\xi_1}$$
$$b_4 = \frac{|A_4|^2 * \xi_{4+}\xi_5}{\xi_1} \quad \textbf{(9)}$$

The variable R is then computed and used for the last process of the calibration, which is shown in Equation (10)

$$R_{i,k} = \frac{T_{i,k} - 1}{|\Gamma_k|^2} + T_{i,k}[|A_4|^2 + 2 * c_k * a_4 - 2 * s_k * b_4] \quad \textbf{(10)}$$

where i (= 1, 2, 3) is the port number and k = 1, 2, 3, 4 which corresponds to the calibration load.

Finally, the values of a_1, a_2, a_3, b_1, b_2, and b_3 can be computed by using the computed results of R. This gives the final calibration values needed for the measurement process, which is shown in Equation (11)

$$a_i = \frac{R_{i,l}(s_m - s_n) + R_{i,m}(s_n - s_l) + R_{i,n}(s_l - s_m)}{2[c_l(s_m - s_n) + c_m(s_n - s_l) + c_n(s_l - s_m)}$$
$$b_i = \frac{R_{i,l}(c_m - c_n) + R_{i,m}(c_n - c_l) + R_{i,n}(c_l - c_m)}{2[c_l(s_m - s_n) + c_m(s_n - s_l) + c_n(s_l - s_m)} \quad \textbf{(11)}$$

where i (= 1, 2, 3, 4), m (= i+1), l (= i+2), and n (= i+3). The reader will notice that m, l, and n refer to more ports than there are available. This is an error in the original paper so to fix this a wrap around is used, which creates i = i+4.

Measurement Process

The measurement process is much less complicated than the calibration process. To obtain the unknown reflection coefficient, the algorithm requires the power measurements of the four ports along with the values of F, G, and H. The values F, G, and H are computed using the values computed in the calibration process. This equation for F, G, and H is shown in Equation (12)

$$
\begin{aligned}
F_i &= \frac{(-1)^i}{2q_i}[|A_j|^2(b_k - b_l) + |A_k|^2(b_l - b_j) + |A_l|^2(b_j - b_k)] \\
G_i &= \frac{(-1)^i}{2q_i}[|A_j|^2(a_k - a_l) + |A_k|^2(a_l - a_j) + |A_l|^2(a_j - a_k)] \\
H_i &= \frac{(-1)^i}{2q_i}[|A_j|^2(a_k b_l - a_l b_k) + |A_k|^2(a_l b_j - a_j b_l) \\
&+|A_l|^2(a_j b_k - a_k b_j)]
\end{aligned}
\qquad \textbf{(12)}
$$

where i (= 1, 2, 3, 4), j (= i+1), k (= i+2), and l (= i+3). The reader will notice that j, k, and l refer to more ports than there are available. This is an error in the original paper so to fix this a wrap around is used, which creates i = i+4.

Once these values have been processed, the unknown gamma can be calculated. This final equation is shown in Equation (13)

$$
\Gamma_t = \frac{\sum_{i=1}^{4} F_i * P_{i,t} + j\sum_{i=1}^{4} G_i * P_{i,t}}{\sum_{i=1}^{4} H_i * P_{i,t}} \qquad \textbf{(13)}
$$

where i (= 1, 2, 3, 4) is the port number and t = the unknown load.

6. RF THEORY & DESIGN

The RF subsystem consists of the six-port junction, VCO, four microwave detectors, and a coaxial cable. The junction is based on that proposed by Mr. Engen [ref 2] and consists of four 90˚ hybrids and a 6-dB directional coupler.

Advanced Design System (ADS) is the simulation package used to design, test, and simulate the components of the six-port network analyzer. ADS is equipped with a tool called Design Guide (DG) that helps design certain common components, including 90° hybrids and directional couplers, such as a Lange coupler. The designer enters the desired functional parameters into DG and allows the system to design the circuit internally. Simulations show the desired results of this design.

"MSUB"

The parameters entered into ADS for "MSUB" are characteristics of the board being used. This is a list specific to the microstrip board:

- h = substrate thickness
- ε_r = dielectric constant
- t = conductor thickness

Here are other parameters that can be entered:

- Mur = relative permeability
- Cond = conductor conductivity
- Hu = cover height
- TanD = dielectric loss tangent
- Rough = conductor surface roughness

Hybrid

A schematic of a narrow band 90° hybrid is shown in Figure 5 where port-1 is the input port; port-2 and 3 are the output ports; and port-4 is the isolated port. A 3-dB, 90° hybrid has the properties that the outputs from ports-2 and 3 differ by 90° and deliver half the input power to match loads. The input parameters of DG include center frequency (F), coupling (C) and microstrip transmission parameters.

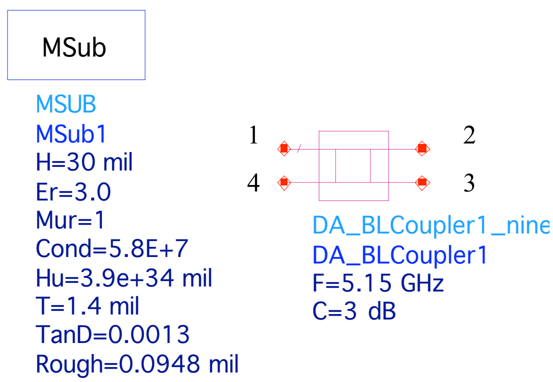


Figure 5: Design Guide's 90° Hybrid in ADS

Design Guide produces the equivalent microstrip circuit, shown in Figure 6, and gives the width and length parameters necessary for a 6 GHz center frequency and 3-dB output. The entered frequency is tuned to 5.15 GHz to obtain a 6 GHz center frequency to produce the desired results. The results for the 90° hybrid depicted in Figure 5 are shown in Figure 7. The simulation verifies correct operation.

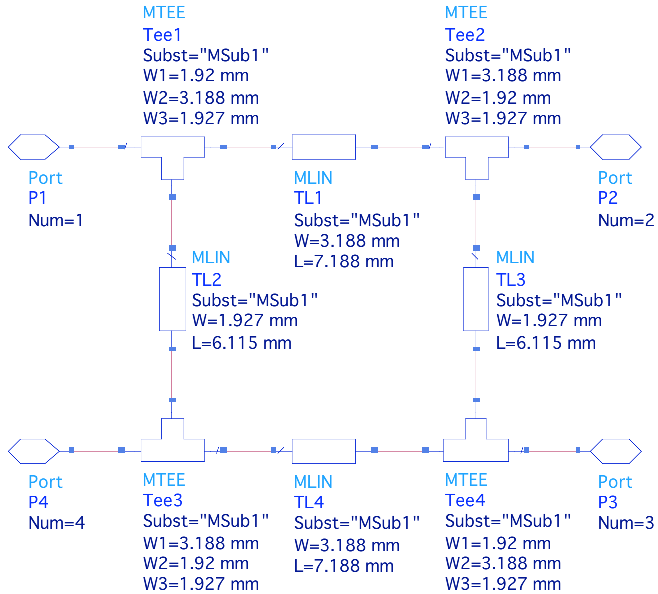


Figure 6: 90° Hybrid's Microstrip Parameters from Design Guide

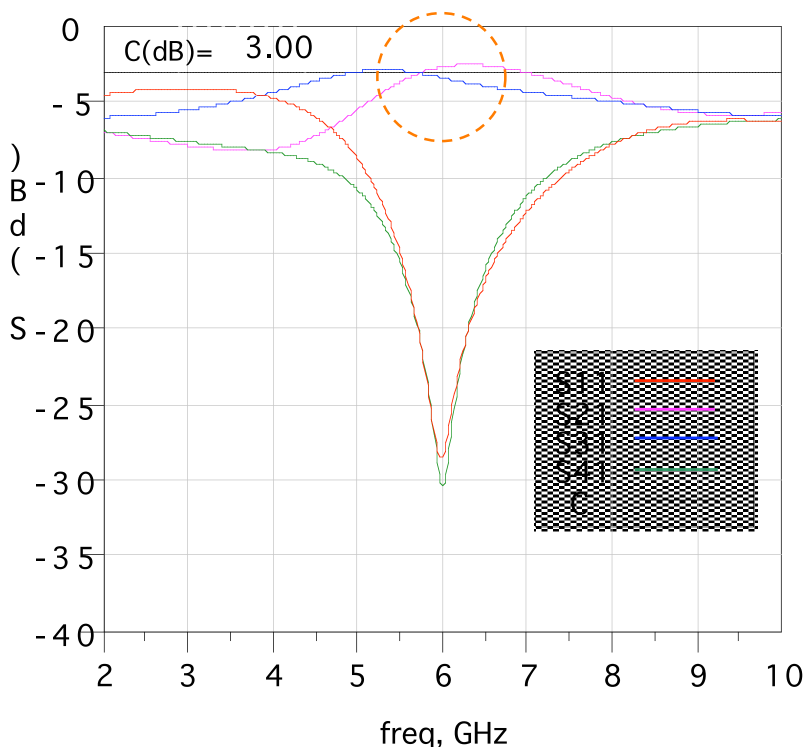


Figure 7: 90° Hybrid Simulation Results

	F	S11	S21	S31	S41	PhaseD
Desired Center Frequency	5.15	-10.24	-4.90	-2.94	-11.90	84.03
Actual Center Frequency	5.98	-28.55	-2.74	-3.42	-30.14	90.22
Change/Worst A->B						
Marker M1						
Marker M2						

F: Frequency (GHz) 1: Input Port 2: In-phase (I) Port 3: Quadrature (Q) Port 4: Isolated Port C: Desired Coupling PhaseD: I-Q Phase Difference	Note: Change/Worst A->B provides performance over the range from marker A to B. The change of F and PhaseD are given, and the worst case S-parameter values are given.

Figure 8: 90° Hybrid Simulation Results

Lange Coupler

A schematic of a Lange coupler is shown in Figure 9 where port-1 is the input port; port-2 is the through port; port-3 is the coupled port; and port-4 is the isolated port. With a 6-dB design, the coupled port one-fourth of the input power with the remainder at the through port. Design Guide includes a Lange coupler design. The input parameters are frequency (F = 6 GHz), coupling (C = 6-dB), and number of fingers (N = 4).

When designed by DB, three parameters of the Lange coupler are produced: width (W), spacing (S), and length (L), see Figure 10. Figure 11 shows simulation results of a negative 6-dB signal at the coupled output port at the center frequency of 6 GHz.

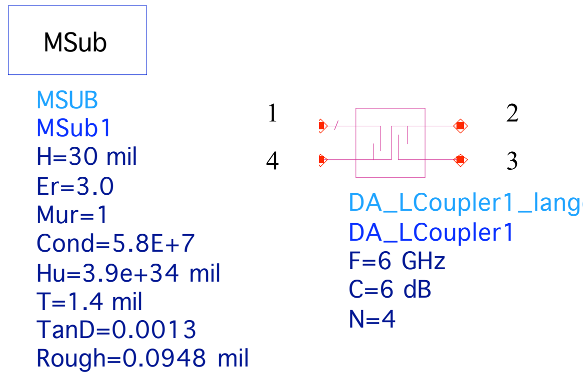


Figure 9: Design Guide's Lange Coupler in ADS

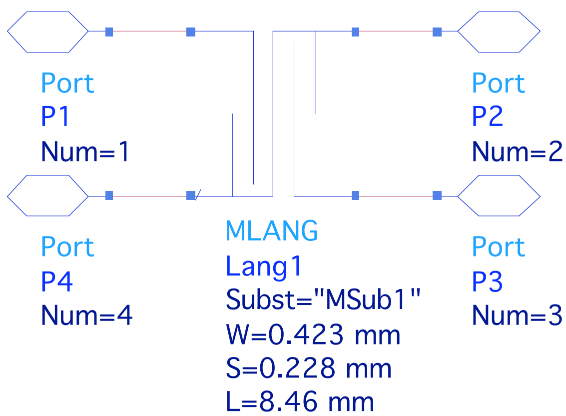


Figure 10: Lange Coupler's Microstrip Parameters from Design Guide

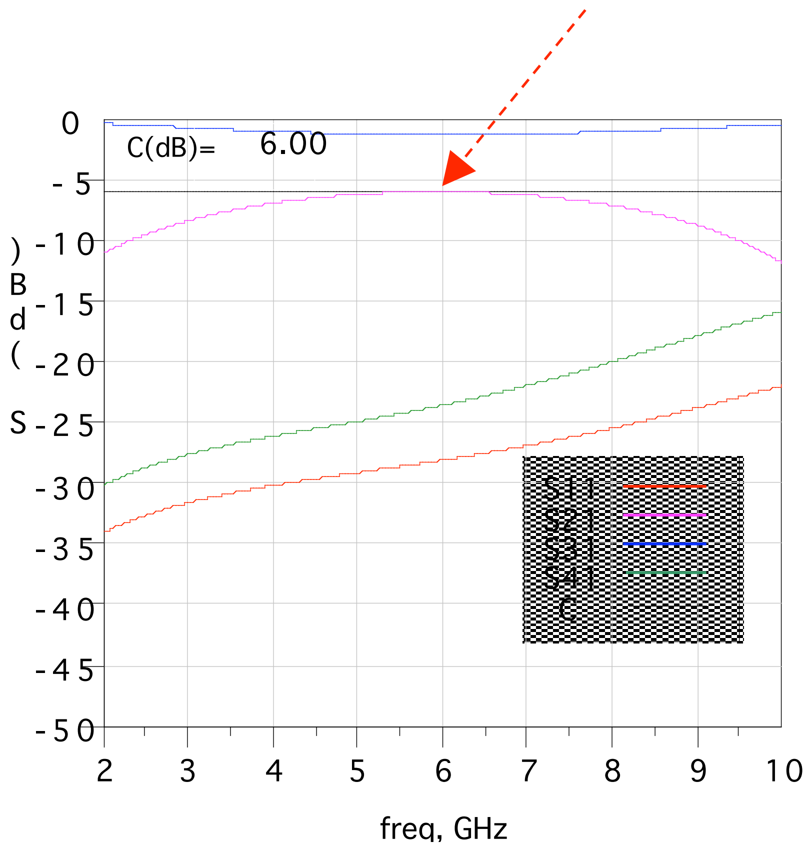


Figure 11: Lange Coupler Simulation Results

Six-Port Junction

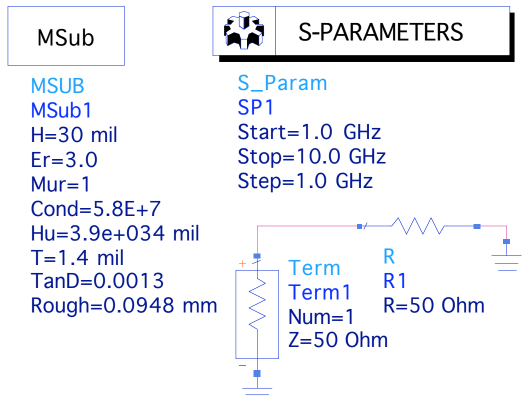
The six-port junction is implemented using four 90° hybrids and a Lange coupler in the configuration shown in Figure 16. The ports denoted by P_one, P_two, P_three, and P_four are where the detectors are attached, port-1 of the Lange coupler is where the VCO is attached, and port-4 of the Lange coupler is where the load is connected. The other ports are terminated with 50 Ω matched loads (Term).

However, since the VCO is external, this configuration leads to a cross over. Thus, for this project, the six-port was implemented using two printed circuit boards (PCB), one for the four 90° hybrids and the other for the Lange coupler. They will then be securely tightened to one platform and connected with semi-rigid coaxial cable. Simulation of the six-port is discussed in Section 7.

Calibration Loads

Five calibration loads are needed for the Hunter and Somlo calibration scheme. As suggested by the authors, one matched load and four offset-shorts of varying lengths were designed and tested using ADS.

The matched load is shown in Figure 12. A 50 Ω resistor is connected and terminates the signal, which results in a zero reflection coefficient since the microstrip line being used is also 50 Ω. The results of the matched load are shown in Figure 13 and shows that the impedance is 50 Ω.



Figures 12: Matched Load

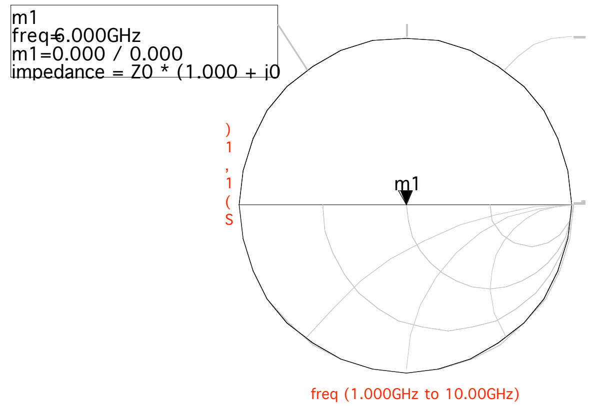


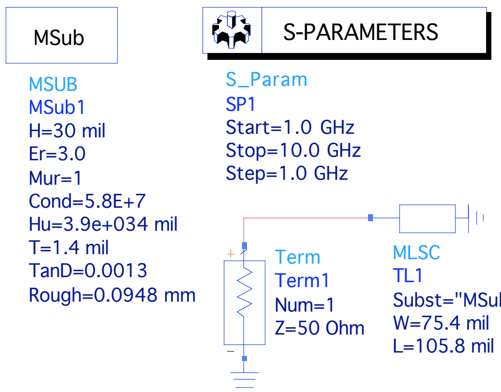
Figure 13: Matched Load Results

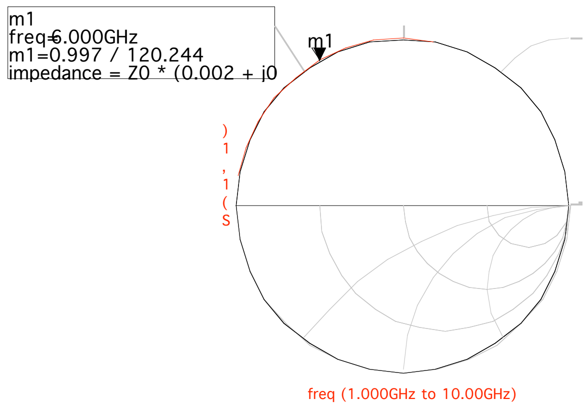
The four other calibration loads consist of short circuit microstrip lines of varying length. The characteristic impedance was chosen be 50 Ω for each and the lengths at 6 GHz were chosen as list. The lengths and widths were found by using the "mstrip" program on the G:\ee551 drive on the computers in the Electrical Engineering computer labs. There are two steps to use this program: Synthesize and Analyze. First, in synthesizing the designer enters the parameters of Zo, ε_r, and h. After doing this, width (W) is calculated and displayed. Then in analyzing, the designer enters h, t, frequency (GHz), and W (the width parameter from synthesizing). After doing this, λ_{eff} is calculated and displayed.

Table 4: Calibration Loads & Results

Load	Degrees	Length (L)	Gamma (Γ)
1. Matched	-	-	0
2. Offset-Short 1	120°	105.8 mil	(1.0 ∟-59.79°)
3. Offset-Short 2	60°	211.2 mil	(1.0 ∟-119.97°)
4. Offset-Short 3	-90°	471.5 mil	(1.0 ∟ 90°)
5. Offset-Short 4	-135°	550.5 mil	(1.0 ∟ 44.97°)

By entering the frequency and certain characteristics of the PCB, the width and effective wavelength (λ_{eff}) of the line is calculated. The length (L) is calculated by taking multiples of $\lambda_{eff}/4$ to obtain points around the edge of the Smith Chart. Half (180°) of the chart is $\lambda_{eff}/4$. A short is located on the left-hand edge of the Smith Chart. The degrees for the four shorts are shown in Table 4. The calibration shorts are of different lengths around the chart from the left-hand edge. Figure 14 shows the microstrip short component "MLSC" with the microstrip width and calculated length. The arrow in Figure 15 shows the location of the short at 6 GHz and the impedance of the short is given in the small box to the upper right. This impedance is normalized to 50 Ω and is used to calculate the reflection coefficient needed for the calibration process. The calculated gammas in Table 4 are used in the calibration code. The other three shorts are obtained the same way and their circuits and results are shown in Appendix B.





Figures 14: Offset-Short Load (1) **Figure 15: Offset-Short (1) Results**

7. ADS SIMULATION & TESTING OF SIX-PORT

The schematic in Figure 16 shows the complete six-port in ADS with the hybrids, Lange coupler, source and the load locations outlined. This is the circuit used to simulate the operation of the six-port and allow verification of the calibration and measurement software. For the simulation, a single frequency tone of 10-dBm with a frequency of 6 GHz was connected to port-1 of the Lange coupler and the rest of the ports are terminated as described in Section 6. This configuration was used to obtain power measurements to test the software through a multi-step calibration process and then measurement of the unknown load.

For the calibration process, one of the calibration loads is connected at the "Load" port. The schematic is simulated and four power measurements are obtained: P_one, P_two, P_three, and P_four. This is repeated for the rest of the calibration loads acquiring a total of 20 power measurements. These power measurements are in dBm, and are converted to milliwatts before being entered into the MATLAB code. As explained in Section 5, these power measurements are used in conjunction with the hard coded reflections coefficients of the calibration loads to compute the calibration parameters of the virtual six-port. For the measurement process, we connected a known load of 220 Ω so the reflection coefficient can easily be calculated by hand and then the code can be tested. The schematic was simulated with the known resistor as the load and four more power measurements were obtained. This yielded a total of 24 power measurement to be entered into the MATLAB code which first determines the calibration parameters of the six-port and then obtains the reflection coefficient of the unknown load.

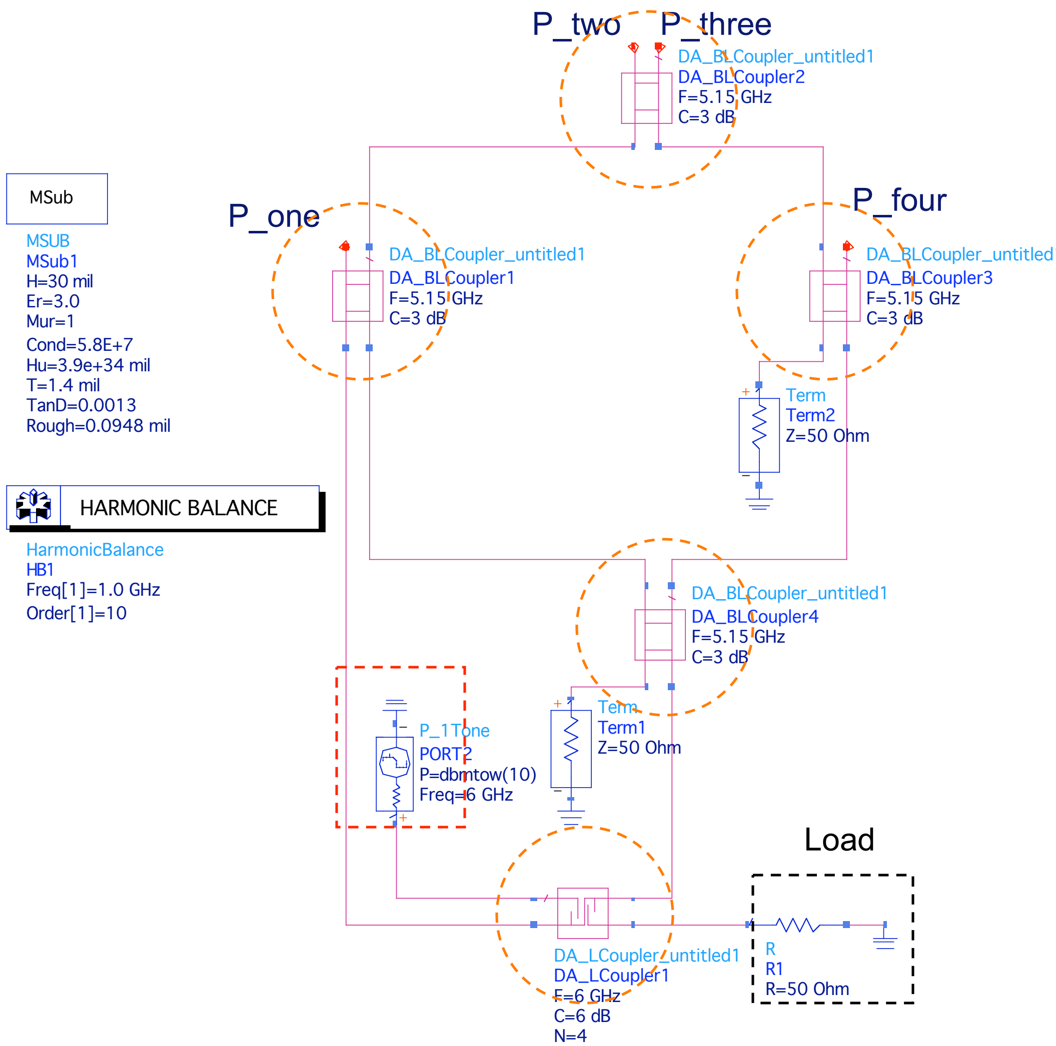


Figure 16: ADS schematic of Six-Port

The computed reflection coefficient was in error. This could be the due to either ADS simulations giving incorrect data results or an error in the MATLAB code. Since other projects using ADS show acceptable results, it is assumed that there are errors in the code.

8. FABRICATION

Fabrication is the process of making the microstrip design onto a PCB. First, ADS is used to design and simulate the RF circuit as well as to generate the microstrip layout as a .eps file. This file is sent to the graphics firm called Technicraft Display & Graphics where a mask is printed. This is an overhead sheet with a black print of the design. The masks of the four hybrids and the Lange coupler are shown in Figures 17 and 18, respectively. The results of the project include a completed six-port design, microstrip layout, and fabrication of the four hybrids and Lange coupler, shown if Figures 19 and 20, respectively.

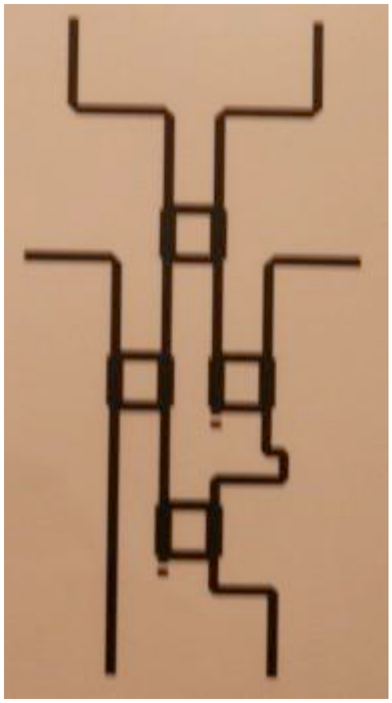


Figure 17: Mask of Hybrids

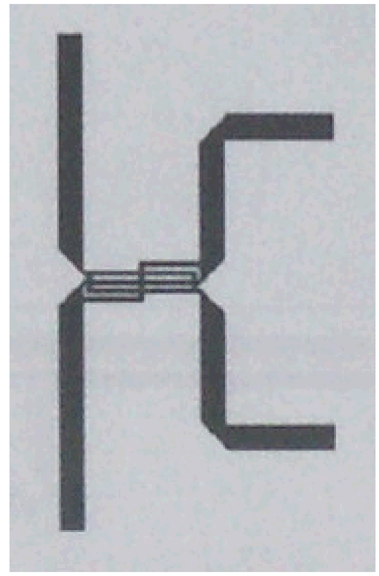


Figure 18: Mask of Lange Coupler

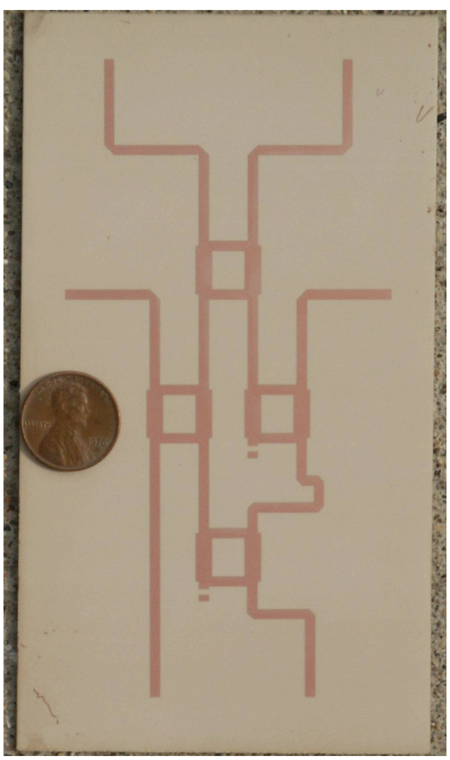


Figure 19: Microstrip of Hybrids

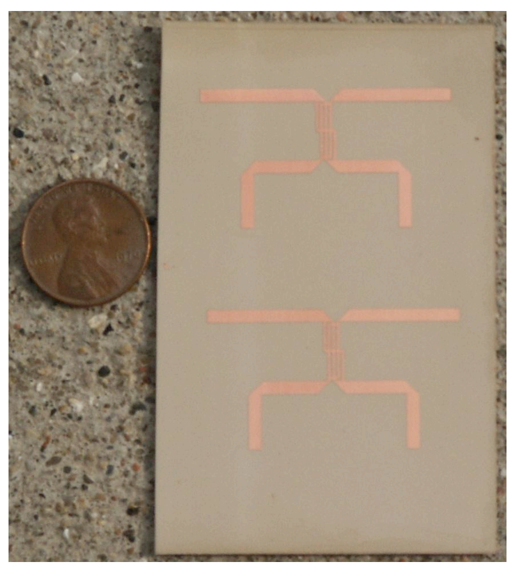


Figure 20: Microstrip of Lange Coupler

9. FUTURE WORK

The project can be expanded. The whole six-port microstrip component can be implemented onto a single PCB with the voltage detectors and VCO included on the board as well. Also, the calibration loads need to be fabricated to be able to test an unknown load.

The software needs to be debugged. Also it can be expanded by operating the oscilloscopes (A/D) through remote access, implement the oscilloscope readings into MATLAB, optimize the code, and create a Graphical User Interface (GUI).

10. DATA SHEET

Frequency of Operation	6 GHz
Accuracy of Γ Measurement	TBD
Power Requirement	5V
Dimensions of six-port junction	
Size	TBD
Weight	TBD
Operating System	Windows, Mac OS, Linux
Computer	USB capable, 256 Meg of Ram
Software	MATLAB

TBD = to be determined

11. STANDARDS

There are no current standards that exist for a six-port network analyzer.

12. PATENTS

There are a few patents that exist for a microwave reflectometer. Five patents were narrowed down that represented the project the most. These patents are shown in Table 3.

Table 5: Patent List

	Patent #	Title
1	5,274,333	Frequency Balanced six-port reflectometer with a variable test port
2	4,808,912	Six-port reflectometer test arrangement and method
3	4,680,538	Millimeter wave vector network analyzer
4	4,521,728	Method and a six port network for use in determining complex reflection
5	4,104,583	Six-port measuring circuit

13. AKNOWLEGMENTS

There are a few people that the designers would like to acknowledge for all their help, advice, and time they have contributed to the advancement of this project. They would like to thank Dr. B. D. Huggins, Dr. Prasad Shastry, and Balamurugan Sundaram.

14. Bibliography

[1] G F Engen. "The six-port reflectometer: An alternative network analyzer," IEEE Trans. Microwave Theory Tech., vol MTT-25, December, 1977, pp 1075-1080.

[2] G F Engen. "An Improved Circuit for Implementing the Six-Port Technique of Microwave Measurements," IEEE Trans. vol MTT-25, no. 12, December, 1977, pp 1080-1083.

[3] J D Hunter and P I Somlo, "An Explicit Six-Port Calibration Method using Five Standards," IEEE Trans., vol MTT-33, no. 1, January, 1985, pp 69-72.

[4] B Horsfield. (2004). "On-Line Moisture Measurement During Cheese Production," (orig. pub. August 1996). [on-line]. RMIT Symposium, Melbourne, Australia. http://www.is.irl.cri.nz/pubdoc/1996/RMIT96.pdf. pages: 5.

[5] Dr. Brian Huggins. Personal Interview. October 2004 till present.

APPENDIX A: MATLAB Code

```
%Calibration Process
%Matthew Rangen

%First of all take the impedance measurement from the ADS simulations

%USER ENTERS VALUES HERE
z1 = (j*0.5750);
z2 = (j*1.731);
z3 = (-j*0.9999);
z4 = (-j*0.414);
z0 = 50;

%Now we multiply these values by the impedance of 50 ohms
z1 = z1*50;
z2 = z2*50;
z3 = z3*50;
z4 = z4*50;

%Once this is calculated, the gammas can now be determined
gamma1 = (z1-50)/(z1+50);
gamma2 = (z2-50)/(z2+50);
gamma3 = (z3-50)/(z3+50);
gamma4 = (z4-50)/(z4+50);
gamma5 = 0;

%First enter the Gammas of the known loads
garray=[gamma1,gamma2,gamma3,gamma4,gamma5];
garray

%Now we must enter the power measurements in, but they are in dBm
%so they must be converted to mW

%USER ENTERS VALUES HERE
load1 = [-6.737,-30.75,-11.523,-12.616];    %port 1,2,3,4
load2 = [-11.819,-12.227,-8.25,-12.174];
load3 = [-8.462,-6.45,-12.871,-10.904];
load4 = [-6.288,-7.91,-20.72,-11.292];
matched = [-12.739,-11.575,-13.371,-11.663];

%Conversion from dBm to mW
for i=1:4,
   powerarry5(i) = 10^(matched(i)/10);   %load 5 matched
   powerarry1(i) = 10^(load1(i)/10);     %load 1   z1
   powerarry2(i) = 10^(load2(i)/10);     %load 2   z2
   powerarry3(i) = 10^(load3(i)/10);     %load 3   z3
   powerarry4(i) = 10^(load4(i)/10);     %load 4   z4
end

%Get the 4 power measurements from the 4 ports
%starting with load 1 through load 4
%This will be handled through the oscilloscope and
%worked on later.

%powerarry5=[.432414,.925763,.395094,.707457];    %load 5  matched
%powerarry1=[1.82222,2.24802,.186337,.837529];    %load 1  z1
%powerarry2=[.597998,2.74789,.818088,.816206];    %load 2  z2
%powerarry3=[1.52195,.18395,.983785,.576368];    %load 3  z3
%powerarry4=[2.43445,.145579,.406443,.632266];    %load 4  z4

%Now get the ratio of powers using the 4th port as
%reference

ratiopowers = [(powerarry1(1)/powerarry1(4)),   %Load1   1
   powerarry1(2)/powerarry1(4),  %Load1   2
```

```
   powerarry1(3)/powerarry1(4),  %Load1   3
   powerarry2(1)/powerarry2(4),  %Load2   4
   powerarry2(2)/powerarry2(4),  %Load2   5
   powerarry2(3)/powerarry2(4),  %Load2   6
   powerarry3(1)/powerarry3(4),  %Load3   7
   powerarry3(2)/powerarry3(4),  %Load3   8
   powerarry3(3)/powerarry3(4),  %Load3   9
   powerarry4(1)/powerarry4(4),  %Load4   10
   powerarry4(2)/powerarry4(4),  %Load4   11
   powerarry4(3)/powerarry4(4),  %Load4   12
   powerarry5(1)/powerarry5(4),  %Load5   13
   powerarry5(2)/powerarry5(4),  %Load5   14
   powerarry5(3)/powerarry5(4)];  %Load5   15

ratiopowers(1)

%Now we can get the T values using the ratios generated
%above, using load 5 as reference.

tratios = [(ratiopowers(1)/ratiopowers(13)),  %Load1  1
   (ratiopowers(2)/ratiopowers(14)),  %Load1  2
   (ratiopowers(3)/ratiopowers(15)),  %Load1  3
   (ratiopowers(4)/ratiopowers(13)),  %Load2  4
   (ratiopowers(5)/ratiopowers(14)),  %Load2  5
   (ratiopowers(6)/ratiopowers(15)),  %Load2  6
   (ratiopowers(7)/ratiopowers(13)),  %Load3  7
   (ratiopowers(8)/ratiopowers(14)),  %Load3  8
   (ratiopowers(9)/ratiopowers(15)),  %Load3  9
   (ratiopowers(10)/ratiopowers(13)),  %Load4  10
   (ratiopowers(11)/ratiopowers(14)),  %Load4  11
   (ratiopowers(12)/ratiopowers(15))];  %Load4  12

tratios(3)

%Now we can compute the complex number with the variables c and s
%using the gamma's given above for each load.

cands = [garray(1)/abs(garray(1))^2,
   (garray(2)/abs(garray(2))^2),
   (garray(3)/abs(garray(3))^2),
   (garray(4)/abs(garray(4))^2)];

cands(3)

c = real(cands);
s = imag(cands);

c(3)
s(3)

%Once we have the real (c) and imaginary (s) values, they can then
%be  used to calculate the eta values.  There are 4 different
%values of etas.

eta1 = (c(2)+c(3))*[((s(1)-s(2))*(c(3)-c(4)))-((c(1)-c(2))*(s(3)-s(4)))]+(c(3)-c(4))*[((s(4)-s(1))*(c(2)-c(3)))-((c(4)-c(1))*(s(2)-s(4)))];
eta2 = (c(3)+c(4))*[((s(2)-s(3))*(c(4)-c(1)))-((c(2)-c(3))*(s(4)-s(1)))]+(c(4)-c(1))*[((s(1)-s(2))*(c(3)-c(4)))-((c(1)-c(2))*(s(3)-s(1)))];
eta3 = (c(4)+c(1))*[((s(3)-s(4))*(c(1)-c(2)))-((c(3)-c(4))*(s(1)-s(2)))]+(c(1)-c(2))*[((s(2)-s(3))*(c(4)-c(1)))-((c(2)-c(3))*(s(4)-s(2)))];
eta4 = (c(1)+c(2))*[((s(4)-s(1))*(c(2)-c(3)))-((c(4)-c(1))*(s(2)-s(3)))]+(c(2)-c(3))*[((s(3)-s(4))*(c(1)-c(2)))-((c(3)-c(4))*(s(1)-s(3)))];

eta = [eta1,eta2,eta3,eta4];

eta

%Now that I have the different value of etas, I can now calculate
%the variables e, f, g, and h.

f1 = 0;
```

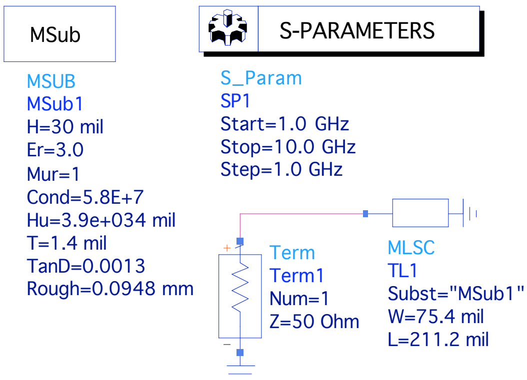
```
for i=1:4,
    ftemp = tratios(i)*eta(i);
    f1 = f1+ftemp;
end

f2 = 0;
for i=1:4,
    ftemp2 = tratios(i+4)*eta(i);
    f2 = f2+ftemp2;
end

f3 = 0;
for i=1:4,
    ftemp3 = tratios(i+8)*eta(i);
    f3 = f3+ftemp3;
end

f = [f1,f2,f3];
f

g1 = 0;
for i=1:4,
    gtemp = tratios(i)*eta(i)*c(i);
    g1 = g1+gtemp;
end
g1 = g1*2;

g2 = 0;
for i=1:4,
    gtemp2 = tratios(i+4)*eta(i)*c(i);
    g2 = g2+gtemp2;
end
g2 = g2*2;

g3 = 0;
for i=1:4,
    gtemp3 = tratios(i+8)*eta(i)*c(i);
    g3 = g3+gtemp3;
end
g3 = g3*2

g = [g1,g2,g3];

h1 = 0;
for i=1:4,
    htemp = tratios(i)*eta(i)*s(i);
    h1 = h1+htemp;
end
h1 = h1*2;

h2 = 0;
for i=1:4,
    htemp2 = tratios(i+4)*eta(i)*s(i);
    h2 = h2+htemp2;
end
h2 = h2*2;

h3 = 0;
for i=1:4,
    htemp3 = tratios(i+8)*eta(i)*s(i);
    h3 = h3+htemp3;
end
h3 = h3*2;

h = [h1,h2,h3];

e1 = 0;
```

```
for i=1:4,
   etemp = ((tratios(i)-1)*eta(i))/abs(garray(i))^2;
   e1 = e1+etemp;
end

e2 = 0;
for i=1:4,
   etemp2 = ((tratios(i+4)-1)*eta(i))/abs(garray(i))^2;
   e2 = e2+etemp2;
end

e3 = 0;
for i=1:4,
   etemp3 = ((tratios(i+8)-1)*eta(i))/abs(garray(i))^2;
   e3 = e3+etemp3;
end

e = [e1,e2,e3];
e

%Now zeta will be calculated using the values calculated
%above.  There will be 5 different zeta values, which will use
%randomly picked values from above.

zeta1 = g(1)*h(3)-h(1)*g(3);
zeta2 = h(1)*f(3)-f(1)*h(3);
zeta3 = h(1)*e(3)-e(1)*h(3);
zeta4 = g(1)*f(3)-f(1)*g(3);
zeta5 = g(1)*e(3)-e(1)*g(3);

%Since the zetas are calculated, M and N can be calculated.

M = (((zeta1^2)/2)-(zeta2*zeta3)-(zeta4*zeta5))/((zeta2^2)+(zeta4^2));
N = ((zeta3^2)+(zeta5^2))/((zeta2^2)+(zeta4^2));

M
N

%Now that M and N are calculated we can get lA4l^2

A4mag = M-((M^2)-N)^(1/2);

A4mag

A4temp = real(A4mag);

%Now I can compute a4 and b4 with the known value of A4.

a4 = ((A4temp*zeta2)+zeta3)/zeta1;
b4 = ((A4temp*zeta4)+zeta5)/zeta1;

%Now that a4 and b4 is computed, R can be calculated using
%the tratios, A4 magnitude, c and s values.

R = [((tratios(1)-1)/(abs(garray(1))^2))+tratios(1)*(A4temp+2*c(1)*a4-2*s(1)*b4),  %Load1  1
   ((tratios(2)-1)/(abs(garray(1))^2))+tratios(2)*(A4temp+2*c(1)*a4-2*s(1)*b4),  %Load1  2
   ((tratios(3)-1)/(abs(garray(1))^2))+tratios(3)*(A4temp+2*c(1)*a4-2*s(1)*b4),  %Load1  3
   ((tratios(4)-1)/(abs(garray(2))^2))+tratios(4)*(A4temp+2*c(2)*a4-2*s(2)*b4),  %Load2  4
   ((tratios(5)-1)/(abs(garray(2))^2))+tratios(5)*(A4temp+2*c(2)*a4-2*s(2)*b4),  %Load2  5
   ((tratios(6)-1)/(abs(garray(2))^2))+tratios(6)*(A4temp+2*c(2)*a4-2*s(2)*b4)   %Load2  6
   ((tratios(7)-1)/(abs(garray(3))^2))+tratios(7)*(A4temp+2*c(3)*a4-2*s(3)*b4)   %Load3  7
   ((tratios(8)-1)/(abs(garray(3))^2))+tratios(8)*(A4temp+2*c(3)*a4-2*s(3)*b4)   %Load3  8
   ((tratios(9)-1)/(abs(garray(3))^2))+tratios(9)*(A4temp+2*c(3)*a4-2*s(3)*b4)   %Load3  9
   ((tratios(10)-1)/(abs(garray(4))^2))+tratios(10)*(A4temp+2*c(4)*a4-2*s(4)*b4)   %Load4  10
   ((tratios(11)-1)/(abs(garray(4))^2))+tratios(11)*(A4temp+2*c(4)*a4-2*s(4)*b4)   %Load4  11
   ((tratios(12)-1)/(abs(garray(4))^2))+tratios(12)*(A4temp+2*c(4)*a4-2*s(4)*b4)];   %Load4  12
```

```
R(8)

%Now that the R values are calculated we can finally get the
%complex variables ai and bi.  These are the values we are
%achieving to get.

a1 = (R(1)*(s(2)-s(3))+R(4)*(s(3)-s(1))+R(7)*(s(1)-s(2)))/(2*(c(1)*(s(2)-s(3))+c(2)*(s(3)-s(1))+c(3)*(s(1)-s(2))));
a2 = (R(2)*(s(2)-s(3))+R(5)*(s(3)-s(1))+R(8)*(s(1)-s(2)))/(2*(c(1)*(s(2)-s(3))+c(2)*(s(3)-s(1))+c(3)*(s(1)-s(2))));
a3 = (R(3)*(s(2)-s(3))+R(6)*(s(3)-s(1))+R(9)*(s(1)-s(2)))/(2*(c(1)*(s(2)-s(3))+c(2)*(s(3)-s(1))+c(3)*(s(1)-s(2))));

ai = [a1,a2,a3,a4];

ai

b1 = (R(1)*(c(2)-c(3))+R(4)*(c(3)-c(1))+R(7)*(c(1)-c(2)))/(2*(c(1)*(s(2)-s(3))+c(2)*(s(3)-s(1))+c(3)*(s(1)-s(2))));
b2 = (R(2)*(c(2)-c(3))+R(5)*(c(3)-c(1))+R(8)*(c(1)-c(2)))/(2*(c(1)*(s(2)-s(3))+c(2)*(s(3)-s(1))+c(3)*(s(1)-s(2))));
b3 = (R(3)*(c(2)-c(3))+R(6)*(c(3)-c(1))+R(9)*(c(1)-c(2)))/(2*(c(1)*(s(2)-s(3))+c(2)*(s(3)-s(1))+c(3)*(s(1)-s(2))));

bi = [b1,b2,b3,b4];

bi

capAi = [complex(a1,b1),complex(a2,b2),complex(a3,b3),complex(a4,b4)];

capAi

%Measurement Mode

%Now that we have developed the complex values ai and bi, we can now
%calculate F, G, and H used to calculate the unknown reflection coefficent
%of an unknown load.

capF1 = ((-1)/2*ratiopowers(13))*((abs(capAi(2)))^2*(bi(3)-bi(1))+(abs(capAi(3)))^2*(bi(1)-bi(2))+(abs(capAi(1)))^2*(bi(2)-bi(3)));
capF2 = ((-1)^2/2*ratiopowers(14))*((abs(capAi(3)))^2*(bi(4)-bi(2))+(abs(capAi(4)))^2*(bi(2)-bi(3))+(abs(capAi(2)))^2*(bi(3)-bi(4)));
capF3 = ((-1)^3/2*ratiopowers(15))*((abs(capAi(4)))^2*(bi(1)-bi(3))+(abs(capAi(1)))^2*(bi(3)-bi(4))+(abs(capAi(3)))^2*(bi(4)-bi(1)));
capF4 = ((-1)^4/2)*((abs(capAi(1)))^2*(bi(2)-bi(4))+(abs(capAi(2)))^2*(bi(4)-bi(1))+(abs(capAi(4)))^2*(bi(1)-bi(2)));

capF = [capF1,capF2,capF3,capF4];

capG1 = ((-1)/2*ratiopowers(13))*((abs(capAi(2)))^2*(ai(3)-ai(1))+(abs(capAi(3)))^2*(ai(1)-ai(2))+(abs(capAi(1)))^2*(ai(2)-ai(3)));
capG2 = ((-1)^2/2*ratiopowers(14))*((abs(capAi(3)))^2*(ai(4)-ai(2))+(abs(capAi(4)))^2*(ai(2)-ai(3))+(abs(capAi(2)))^2*(ai(3)-ai(4)));
capG3 = ((-1)^3/2*ratiopowers(15))*((abs(capAi(4)))^2*(ai(1)-ai(3))+(abs(capAi(1)))^2*(ai(3)-ai(4))+(abs(capAi(3)))^2*(ai(4)-ai(1)));
capG4 = ((-1)^3/2)*((abs(capAi(1)))^2*(ai(2)-ai(4))+(abs(capAi(2)))^2*(ai(4)-ai(1))+(abs(capAi(4)))^2*(ai(1)-ai(2)));

capG = [capG1,capG2,capG3,capG4];

capH1 = ((-1)/2*ratiopowers(13))*((abs(capAi(2)))^2*(ai(3)*bi(1)-ai(1)*bi(3))+(abs(capAi(3)))^2*(ai(1)*bi(2)-
ai(2)*bi(1))+(abs(capAi(1)))^2*(ai(2)*bi(3)-ai(3)*bi(2)));
capH2 = ((-1)^2/2*ratiopowers(14))*((abs(capAi(3)))^2*(ai(4)*bi(2)-ai(2)*bi(4))+(abs(capAi(4)))^2*(ai(2)*bi(3)-
ai(3)*bi(2))+(abs(capAi(2)))^2*(ai(3)*bi(4)-ai(4)*bi(3)));
capH3 = ((-1)^3/2*ratiopowers(15))*((abs(capAi(4)))^2*(ai(1)*bi(3)-ai(3)*bi(1))+(abs(capAi(1)))^2*(ai(3)*bi(4)-
ai(4)*bi(3))+(abs(capAi(3)))^2*(ai(4)*bi(1)-ai(1)*bi(4)));
capH4 = ((-1)^3/2)*((abs(capAi(1)))^2*(ai(2)*bi(4)-ai(3)*bi(2))+(abs(capAi(2)))^2*(ai(4)*bi(1)-
ai(1)*bi(4))+(abs(capAi(4)))^2*(ai(1)*bi(2)-ai(2)*bi(1)));

capH = [capH1,capH2,capH3,capH4];

%Once we have F, G, and H calculated we can now get the reflection
%coefficent.

%USER ENTERS VALUES HERE
ukpow = [-26.568,-8.952,-9.679,-11.426];

%Unknown powers converted from dBm to mW
for i=1:4
   unknownpw(i) = 10^(ukpow(i)/10);
end
```

```
fp = 0;
for i=1:4,
   temp1 = capF(i)*unknownpw(i);
   fp = fp+temp1;
end

gp = 0;
for i=1:4,
   temp2 = capG(i)*unknownpw(i);
   gp = gp+temp2;
end

hp = 0;
for i=1:4,
   temp3 = capH(i)*unknownpw(i);
   hp = hp+temp3;
end

ukgamma = (fp+(j*gp))/hp;

ukgamma

((abs(ukgamma))^2)^(1/2)
```

APPENDIX B: Calibration Loads & Results



Figures 21: Offset-Short Load (2)

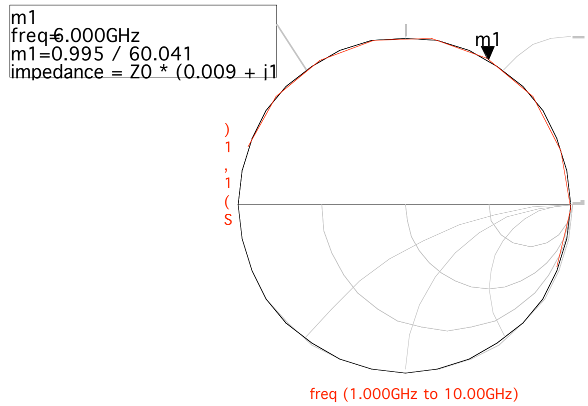
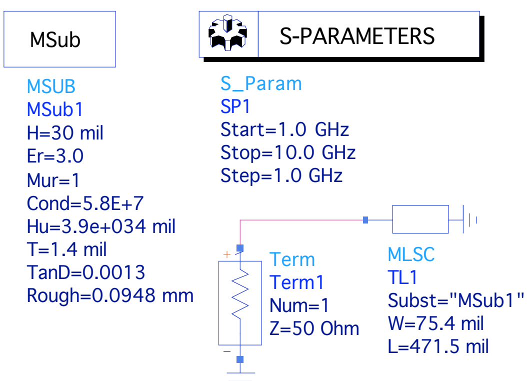


Figure 22: Offset-Short (2) Results



Figures 23: Offset-Short Load (3)

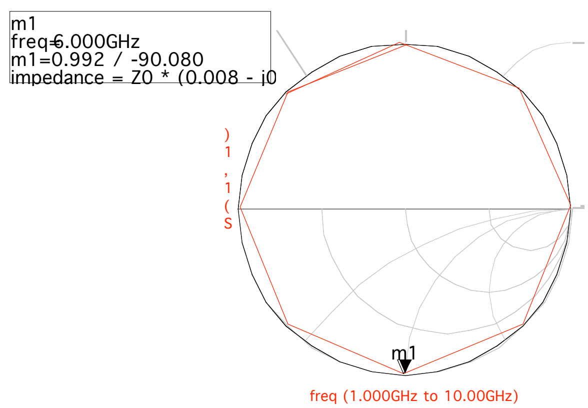
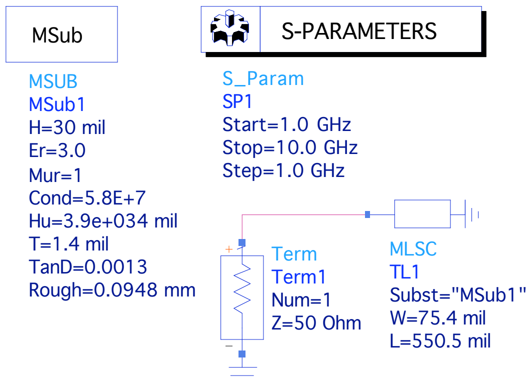
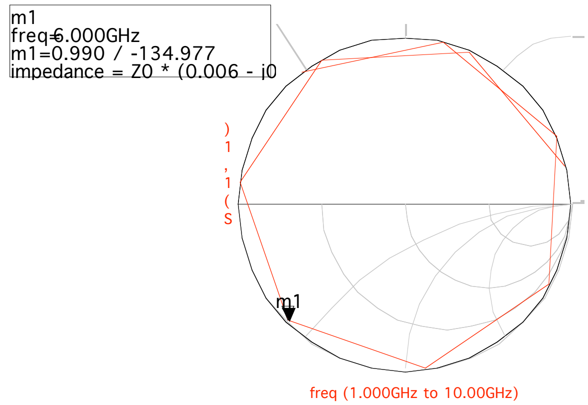


Figure 24: Offset-Short (3) Results





Figures 25: Offset-Short Load (4)

Figure 26: Offset-Short (4) Results