12-1-2012

De-Embedding in High Speed Design

Don DeGroot
Andrews University, degroot@andrews.edu

David Dunham
Molex Inc.

DesignCon 2012

De-embedding in High Speed Design

Dr. Don DeGroot, CCN & Andrews University
Don@ccnlabs.com, 303-872-7700

Dr. Kaviyesh Doshi, LeCroy Corporation
Kaviyesh.Doshi@lecroy.com, 845-425-2000

David Dunham, Molex Inc
DDunham@molex.com, 630-718-5817

Peter J. Pupalaikis, LeCroy Corporation
Peter.Pupalaikis@lecroy.com, 845-425-2000

Abstract

A common problem encountered while measuring S-parameters of a device under test (DUT) is that it is relatively easy to calibrate a network analyzer to the end of coaxial test cables, but difficult to connect coaxial test ports directly to board-level DUTs. One needs a probe or fixture to connect the coaxial test port to the board-level DUT. Since the transfer function of the fixture/probe significantly changes the high-frequency S-parameter measurements, one needs to de-embed the DUT response from a broadband measurement of DUT *plus* fixture/probe. For example – A high-speed backplane connector test vehicle, consisting of SMAs, transmission lines, and connector vias, introduces unwanted high-frequency effects to the response of mated connector pair under test, limiting the ability to characterize and model the connector pair alone. This tutorial reviews and demonstrates three de-embedding methods used to extract DUT S-parameters from total response measurements made at the boundary of a test fixture. The paper provides an overview of the math, inherent assumptions, and strengths of each method, while the presentation focuses more on practical application.

Author(s) Biography

Don DeGroot operates CCN (www.ccnlabs.com), a test and measurement business he co-founded in 2005 to support high-speed electronic design. Don has over 25 years experience in high-frequency electrical measurements and design, including his PhD degree from Northwestern University and 12 year of research at NIST. Don currently focuses on interconnection and PCB material characterization for serial data applications.

Kaviyesh Doshi received his Ph.D. in Electrical Engineering from University of California, Santa Barbara in June 2008. He joined LeCroy Corporation in August 2008, as a Research & Development Engineer. At LeCroy, he has been involved in design and development of SPARQ – a TDR based instrument for measuring S-parameters. He has filed patents in the area of de-embedding and signal processing for time domain network analyzer.

Dave Dunham is an Electrical Engineer in Connector Product Division at Molex. His primary focus is in design and characterization of next generation backplane and I/O systems and components. He has been also involved in the test and measurement methodologies, such as TRL calibration, 3.5mm and 2.4mm field replaceable connector optimization. Dave has a BSEE degree from University of New Mexico.

Peter Pupalaikis was born in Boston, Massachusetts in 1964 and received the B.S. degree in electrical engineering from Rutgers University, New Brunswick, New Jersey in 1988. He has worked at LeCroy Corporation for 16 years and currently manages integrated-circuit development and intellectual property as Vice President of Technology Development. Prior to LeCroy he served in the United States Army and as a consultant in embedded systems design. Mr. Pupalaikis holds numerous patents in the area of measurement instrument design and is a member of Tau Beta Pi, Eta Kappa Nu and the IEEE signal processing, communications, and microwave societies.

Introduction

Signal integrity engineers must know the electrical response of devices and interconnects that are often buried in packaging or placed on a circuit board at a point that is distant from any practical test point. At high frequencies, the electrical response of the pieces that lie between the instrument and the device under test (DUT) must be carefully removed from the composite measurement in order to characterize the device response alone. This de-embedding process gives engineers the tools they need to separate the electrical response of each individual piece of an electrical network, so the pieces can be used in design or to verify individual circuit element performance.

This tutorial describes the general framework for the classical de-embedding problem – given the measured S-parameters of the composite system (DUT plus fixture or DUT plus probes), recover the S-parameters of the DUT alone.

When test fixtures have "nice" places to make coax and probe contacts to the fixture interconnections, it is possible to acquire the S-parameters of your fixture from the manufacturer or a test lab. With this data and some assumptions about circuit symmetry, de-embedding is a set of matrix operations applied to the total response measurements.

What happens when we have to build or acquire a custom test fixture where the connections to the DUT are in buried layers and not accessible with probes?

One approach is to build an approximate equivalent circuit model using the best available information on the test fixture interconnections. This de-embedding model is often limited since it is difficult to access physical details of the fabricated fixture.

With the general framework for de-embedding covered in this tutorial, in conjunction with time domain gating and impedance peeling, engineers can address the more complicated and customized cases wherein no fixture/probe information is available beforehand.

The sections that follow describe three de-embedding methods that only need a limited knowledge of the test fixture interconnections. In particular, this paper and accompanying presentation demonstrate the application of:

- TRL Calibration Method;
- Second-Tier TRL Method; and
- Time-Domain Gating & Peeling

Each method is applied to an example backplane connector test vehicle to show their strengths and key assumptions, but the framework is general and applicable to other methods that may be pertinent to a given de-embedding problem.

Patent Disclosure

Portions of the information provided in this paper are the subject of patents applied for.

Need for calibration

An overview of de-embedding and DUT reference planes

Inherent to de-embedding is the concept of at least two measurement reference planes: one where the instrument can readily measure; and the other where we want to measure our device.

For instruments used to measure network parameters (e.g. resistance, reactance, impedance, and scattering parameters), calibration is the process that corrects observed values, and at the same time defines the measurement reference plane. For all network-measuring instruments, correction of the measured parameter only makes sense if we know the precise location (reference plane) of valid calibration (like the ends of the test cables vs. the front panel connectors). The point where we attach calibration standards defines this location, and it is known as Calibration Reference Plane.

The second reference plane is known as the DUT Reference Plane. It is the point where our board-level DUT starts. This may not coincide with the point where a manufacture or test engineer connected their calibration standards, so we de-embed to correct the measurements in order to get closer to the DUT we defined. De-embedding may be thought of as the process that extends the Calibration Reference Plane to the DUT Reference Plane.

If the factory provides a calibration for an ohmmeter, the Calibration Reference Plane is most often specified as the connection points at the face of the instrument (banana jacks). If the user wants to remove the resistance of their leads and probes from their measured values, there is often a button that modifies the calibration by the simple linear operation of subtraction: the resistance measured with the probe tips in contact with each other is recorded and subtracted from all subsequent DUT measurements. The manufacturer defined the Calibration Reference Plane to be the front-panel connectors since they connected their standards at that point; the user extended the reference plane to their probe tips using a de-embedding operation.

That is the same general process used for extending vector network analyzer (VNA) and time-domain refelction (TDR) calibrations to the DUT reference plane, though the transformation is a bit more complicated than subtraction.

Calibrations and corrections in general

Few electrical instruments actually measure the physical parameter of interest. They may measure a voltage amplitude that is related to the measurement parameter, so that the measurement M is actually a function of the real parameter x: $M = f(x)$. If the function is linear and we can find an approximate form of the function (like $M = ax + b$), then we can estimate the model parameters a and b with measurements from a sufficient number of known standards. This is calibration.

With a measurement M and calibration coefficients a and b, we can then estimate the actual physical parameter: $x = (M - b)/a$. This is correction.

If the function *f* is the same for all time, we can determine the calibration of an instrument at the factory and ship the instrument with a single set of calibration coefficients and software that performs the correction.

Network analyzer calibrations
Why would a signal integrity engineer even need to worry about vector network calibrations and reference planes since the factory should handle this, right?

The answer is that the calibration function *f* of high-frequency instruments varies significantly with time. Signal sources, receivers, and interconnections drift with vibrations, cable flexing, temperature, humidity, etc. The network analyzer is only approximated as time-invariant over short time periods (like 2-8 hours), so the VNA has to get calibrated often (and we include both frequency- and time-domain network analyzers in "VNA").

In traditional VNA's, this means the user has to attach a series of standards to all of the ports and execute the internal calibration routines. There is still the need for factory service and calibration on quantities such as amplitude and frequency, but the calibration coefficients for the S-parameters, must be performed often.

Since the VNA calibration defines a reference plane, Signal integrity engineers, then, establish the Calibration Reference Plane for their network-measuring instrument every day they use (or calibrate) it.

The network analyzer calibration also establishes the function and coefficients that link measured values to actual values. This is further complicated in that VNAs measure ratios (like voltage to current, or reflection amplitude to incident amplitude). Network analyzers cannot measure voltage and current directly at the frequencies of multi-gigabit electronics, so it measures voltages that correspond to the amplitudes of transmitted and reflected traveling waves. The linear combination of the incident and reflected traveling wave amplitudes can give us the ratio of the voltage to current at some measurement reference plane. All this means is that VNA calibration is a calibration of network parameters, like impedance or scattering parameters, both of which are ratios.

The standards for such a calibration much be known ratios: known impedances and known S-parameters. Over the past 3-4 decades, a large number of VNA standard sets have been developed, and with them, the software to determine the calibration coefficients of a VNA.

There is an alphabet soup of VNA calibration standards: SOLT, LRM, LRRM, LRL, TRL, etc. The designations like TRL refer to the standards used (Thru, Reflection, Line) *and* to a class of algorithm used to estimate the calibration coefficients from the measurements of the known standards. Not all TRL calibration algorithms are the same, so the user of an instrument must be aware of the specific limitations and assumptions of the software or instrument they are using (which is well beyond the scope of this tutorial).

Since the place we attach the VNA standards locates a reference plane, calibrations can be viewed as part of de-embedding (the movement of the Calibration Reference Plane close to the DUT Reference Plane) when we can place known standards at the DUT Reference Plane, just like the ohmmeter can remove the resistance of the probes down to their tips. However, the location of the Calibration Reference Plane is only part of the process, calibration, not de-embedding, determines the correction equation for the instrument.

General de-embedding framework

In this section, the general de-embedding problem is defined. *Figure **1*** indicates a basic setup of the de-embedding problem.

It is desired to measure the s-parameters of a P-port device under test (DUT) using a P-port network analyzer. The network analyzer is connected to the DUT via a 2P-port fixture. Either the S-parameters of the fixture are known or enough information about the fixture is available so that a model of the fixture can be created. Using the network analyzer, the S-parameters of the system comprising of the cascade connection of the fixture and the DUT is measured and now it is desired to obtain the S-parameters of the DUT by removing the effects of the fixture from the measured s-parameters. This is the classical de-embedding problem.

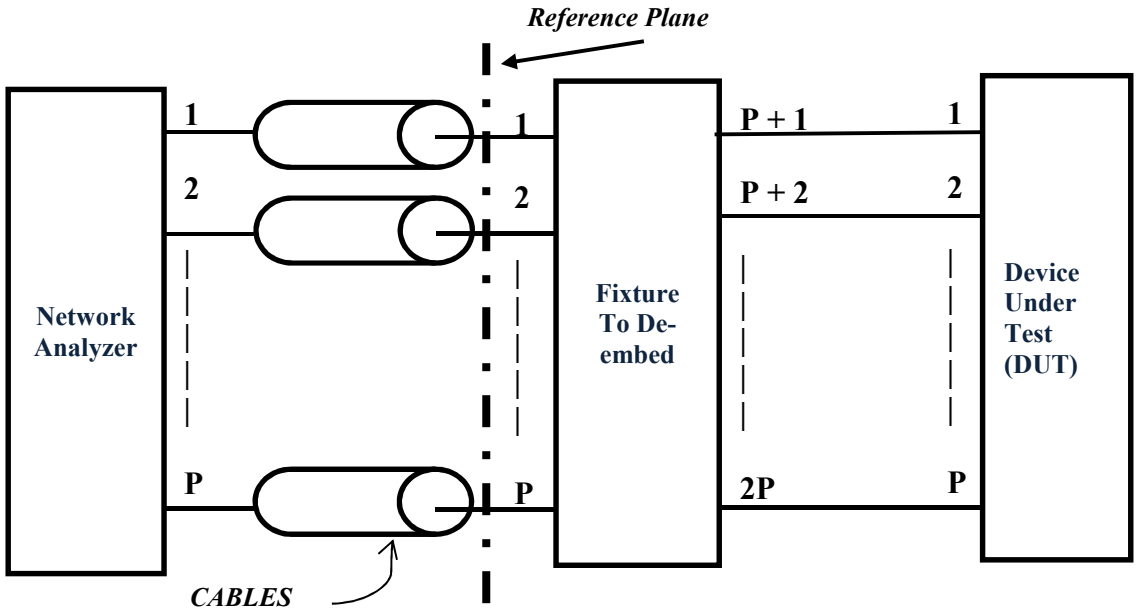


Figure 1: Block diagram depicting a de-embedding problem

The above problem is general enough to include different cases. For example, *Figure **2*** shows the setup for adapter de-embedding. When the connector type of the end of the cable attached to the network analyzer is such that it cannot be connected directly to the

DUT, one needs to use adapters. To characterize the DUT, these adapters need to be de-embedded from the measurement.

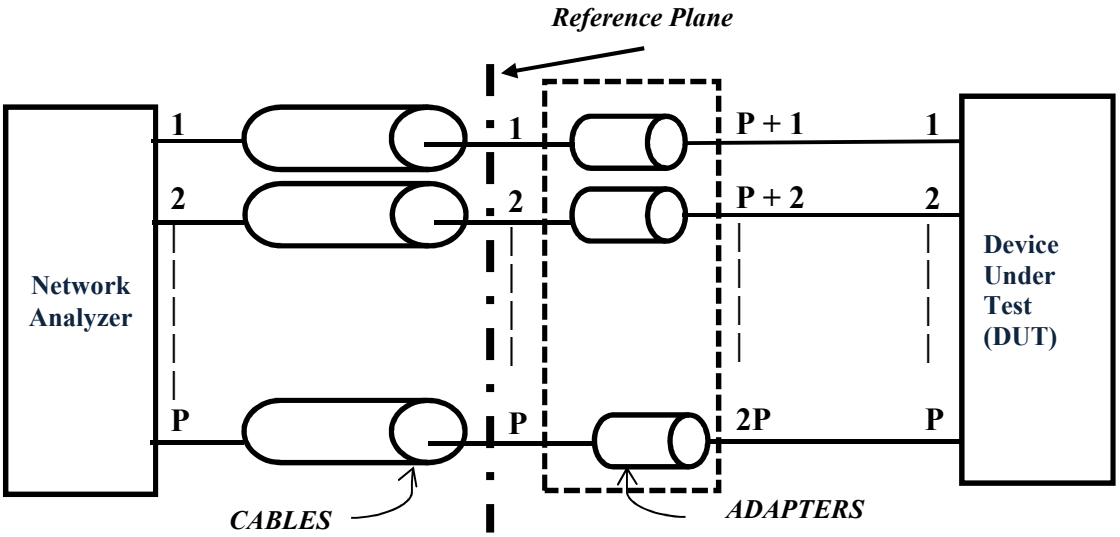


Figure 2: Adapter De-embedding problem

Yet another example is for measuring S-parameters of a USB cable. One must use a fixture so that the network analyzer can be connected to one end of the fixture, while the other end is connected to the USB cable.
Figure 3 shows one such possible arrangement.

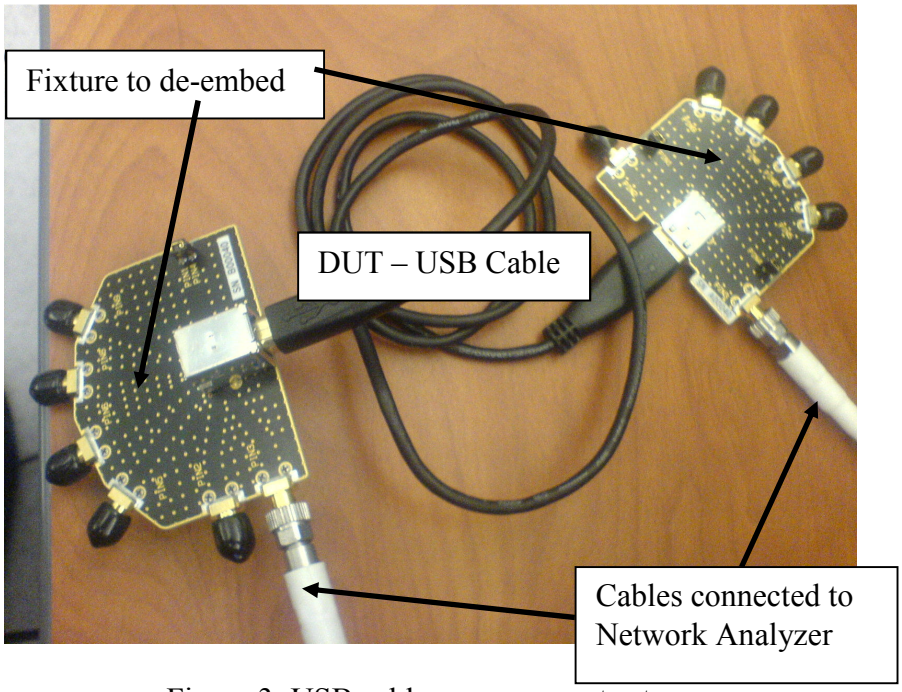


Figure 3: USB cable measurement set-up

In order to address the de-embedding problem, the first step is to write the equations that describe the measurements for the circuit shown in *Figure 1*. In the next section a simple case of a one port DUT is considered to describe the concept of signal flow diagram which will be used to write the system of equations that describe the measurements. The de-embedding problem will be then solved for the one port DUT case.

De-embedding for one port DUT

Consider the block diagram shown in *Figure 4*. Here we have a one-port DUT labeled Γ_{dut} connected to a two-port fixture labeled D. There is a dotted-line box surrounding all of these elements with a single, exposed port on the left labeled 1. This port where the measurements are made is termed as the system port. As a one port de-embedding problem, it is presumed that the S-parameters of the device labeled D are known, as well as the overall s-parameter measurements of the entire system within the dotted line. The de-embedding problem is to determine the unknown S-parameters Γ_{dut} from the known S-parameters of the system, the known S-parameters of D, and the known interconnectedness of the system.

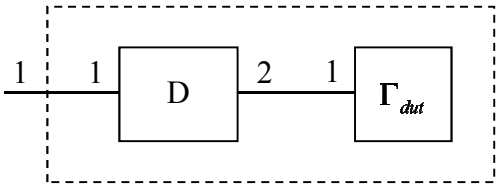


Figure 4: One Port De-embedding Example - system diagram

To write the equations for the measured wave, first the block diagram in *Figure 4* needs to be represented in an equivalent signal flow diagram. *Figure 5* is a signal flow diagram representation of the two-port fixture shown as D. The S-parameters of this network are:

$$S = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix},$$

and the signal flow diagram defines the equations:

$$\begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}. \qquad [\,1\,]$$

Note that [1] is a set of equations that only define the port-port relationships. In other words, given a pair of incident waveform values, it computes the values of the reflected waves.

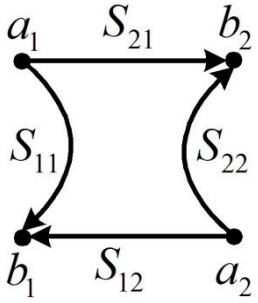


Figure 5: Signal Flow Diagram Representation of a Two-Port Network

Figure ***6*** is a signal flow diagram representation of the system shown in *Figure* ***4***, where the fixture D is cascaded with the one port DUT. This system is driven by the stimulus shown as e and the reflected wave under this driving condition is labeled as Γ_{msd}. The dots in the figure correspond to nodes. The value of a node contains the value of a power wave present at that node. There are always two nodes per port of a device: one for the incident wave and another for the reflected wave. Arrows in a device correspond to paths for waves to follow. The arrows always point from incident nodes within a device to reflect nodes in that device. The weight of the arrow is given by an s-parameter value which is written with a subscript notation such as S_{xy}, where the x refers to the reflect port and the y corresponds to the incident port. Again, an s-parameter value S_{xy} always refers to the weight of an arrow from the incident node of port y to the reflect node of port x. Note that the a and b naming conventions on the nodes is dropped when moving from *Figure* ***5*** to *Figure* ***6*** because they can no longer be referred to as incident or reflect nodes. For example, the node containing the incident wave on the DUT is n_1, but this is the reflected wave from port 2 of the two-port fixture D.

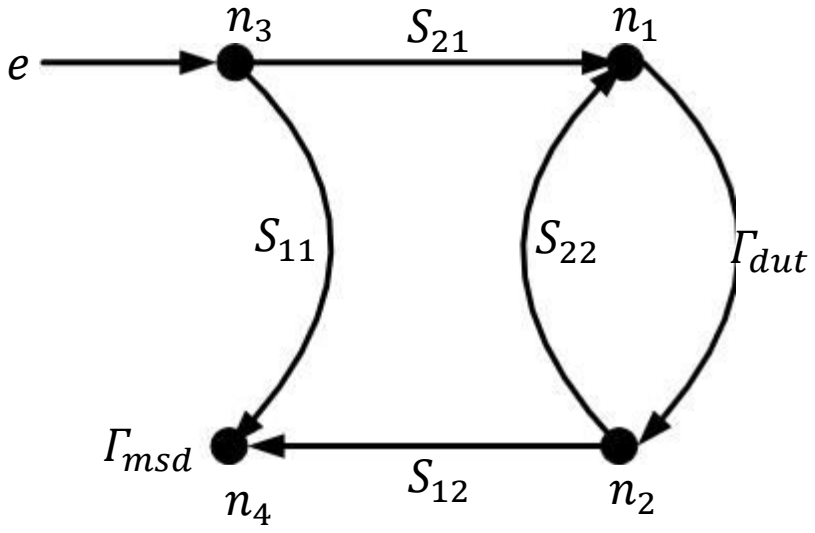


Figure 6: One Port De-embedding example – signal flow diagram

This is a very specific occurrence in s-parameter system signal flow diagrams, specifically, all device ports are connected by connecting the reflect node of the port of one device to the incident node of the other device and vice-versa. There is a duality between the signal flow diagram and the system of equations describing the system. This is seen by writing the equations for each node. For the purposes of maintaining this duality easily, we write the equations as follows:

$$\begin{aligned} n_1 - S_{22}n_2 - S_{21}n_3 &= 0 \\ n_2 - \Gamma_{dut}n_1 &= 0 \\ n_3 - e &= 0 \\ n_4 - S_{12}n_2 - S_{11}n_3 &= 0 \end{aligned}$$

Here, the equation for each node is written by first writing the node name, subtracting the weights on the arrows entering the node multiplied by the node from which the arrow originates and setting this equal to the sum of the weights of arrows that enter the node but do not originate from another node. These equations can therefore be written in matrix form as:

$$\begin{pmatrix} 1 & -S_{22} & -S_{21} & 0 \\ -\Gamma_{dut} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -S_{12} & -S_{11} & 1 \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ e \\ 0 \end{pmatrix}.$$

[2]

If we solve this system for node n_4, substituting 1 for e, and replacing n_4 with Γ_{msd}, which is now the measured system S-parameters, we obtain:

$$n_4 = \Gamma_{msd} = \frac{S_{11} - \left(S_{11}S_{22} - S_{12}S_{21}\right)\Gamma_{dut}}{1 - \Gamma_{dut}S_{22}}$$

[3]

Solving for Γ_{dut}, we obtain:

$$\Gamma_{dut} = \frac{\Gamma_{msd} - S_{11}}{\Gamma_{msd}S_{22} - |S|}.$$

[4]

As we can see, the de-embedding in this case is straightforward mathematically. Given a one-port measurement Γ_{msd} of a system and the known S-parameters of device D, we plug these into [4] to obtain Γ_{dut}.

One should note that if the S-parameters of the fixture and the DUT are known, then [3] is the equation that can be used to obtain the S-parameters of the cascaded system.

Some notations are defined from the above example before describing a general algorithm. In equation [2], the large, left-hand matrix is referred to as the *system characteristics matrix* and it is denoted as E. It is named this because all of the characteristics of the system, namely the S-parameters of the devices and the interconnectedness of the devices are contained in this matrix. The vector that it's multiplied by is called the *node vector* because it holds the values of all of the nodes in the system. It is usually denoted by v. The vector to the right is called the *stimulus vector* because it contains the values of all of the stimuli entering from sources of power waves. It is denoted as e. Thus, equation [2] can be written as:

$$Ev = e.$$

Note that because of the way the equations are written, a one appears on the diagonal of the system characteristics matrix. This is because each set of equations was written starting with the node value corresponding to the nodal equation and these equations were listed in the same order as the node numbering. This is called a *canonical* form of the equation, meaning that the nodes are ordered in the same order as the nodal equations[1]. Writing the equations in canonical form maintains the ease of transitioning back and forth between equation and signal flow diagram. To understand what is meant by this, consider that in [2] any given element at row x and column y of the system characteristics matrix E_{xy} is the weight of an arrow from the node named v_y to the node named v_x. This is understood clearly by understanding that row x of the equation corresponds to the equation for node v_x and therefore all of the elements in the row x of E are weights of arrows that enter node v_x. Similarly, understand that column y of E all multiply the element v_y and therefore are weights of arrows leaving node v_y. Once the system of equations corresponding to the signal flow diagrams written in the form shown in [2], the solution for the node values as a function of the stimulus is easy to determine as $v = E^{-1}e$.

In the next section, using a two-port DUT example, a few concepts not found in the one-port example are extended. The procedure for converting the signal flow diagram into the system equation for the two-port DUT and fixtures as shown in *Figure 7* is described. The de-embedding algorithm described for this two port case will be generalized to the P-port DUT.

De-embedding for two-port DUT

In *Figure 7* we have a two-port DUT labeled U, with unknown s-parameters and bracketed by left and right devices labeled L and R. We know the S-parameters for L and

[1] note that for a canonical form, we do not require that the equations and nodes are ordered exactly as numbered, only that the node ordering of the node vector (i.e. the columns of system characteristic matrix) correspond to the equation ordering (i.e. the rows of the system characteristics matrix).

R and that we wish to determine the S-parameters of the device labeled U from the two port S-parameters for the entire block surrounded by the dotted lines.

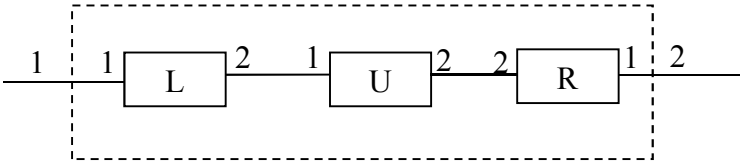


Figure 7: Two Port De-embedding Example - system diagram

A signal flow diagram that represents this block diagram is shown in *Figure **8***, where we have arbitrary labeled the nodes, inserted the known S-parameters SL and SR and the unknown S-parameters Su and driven the inputs with arbitrary signals e_1 and e_2.

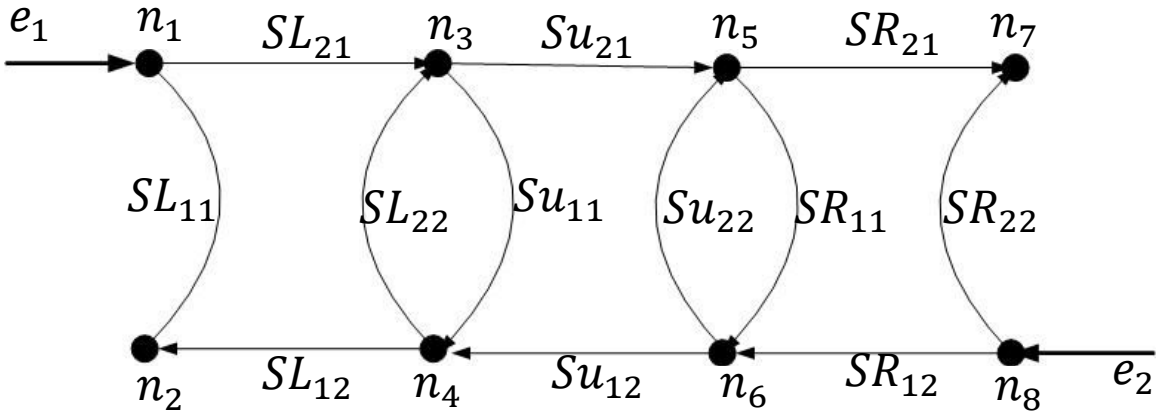


Figure 8: Two Port De-embedding Example - signal flow diagram

The process of converting a signal flow diagram into the system equation starts with a preparation step. We will use the example signal flow diagram shown in *Figure **8***:

1. For an N node system, label the nodes in the system from 1…N.
2. List the node names that have been numbered in order of numbering in the nodes vector
3. Identify the sources of stimuli. These are arrows that enter a node, but do not come from another node (i.e. are set to a value externally from the system). The nodes are the ones directly connected to the network analyzer.
4. Write an empty $N \times N$ weights matrix subtracted from an $N \times N$ identity matrix multiplied by the nodes vector. (The identity matrix minus the weights matrix is the system characteristics matrix).
5. Set this equal to another vector with an empty list of stimuli names

When we've finished these steps, we have the signal flow diagram shown in *Figure **8*** and the partially set up equation in [5].

$$\left[I - \begin{pmatrix} \square & \square & \square & \square & \square & \square & \square & \square \\ \square & \square & \square & \square & \square & \square & \square & \square \\ \square & \square & \square & \square & \square & \square & \square & \square \\ \square & \square & \square & \square & \square & \square & \square & \square \\ \square & \square & \square & \square & \square & \square & \square & \square \\ \square & \square & \square & \square & \square & \square & \square & \square \\ \square & \square & \square & \square & \square & \square & \square & \square \\ \square & \square & \square & \square & \square & \square & \square & \square \end{pmatrix} \right] \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \\ n_6 \\ n_7 \\ n_8 \end{pmatrix} = \begin{pmatrix} \square \\ \square \\ \square \\ \square \\ \square \\ \square \\ \square \\ \square \end{pmatrix}$$

[5]

Now, all that needs to be done is to fill in the weights matrix and the stimulus matrix. To do this, find each arrow in the diagram. If an arrow points into a node and does not come from another node, it is a stimulus. Put the weight of the arrow in the row corresponding to the node it points at. In other words, if stimulus value e points at node x, put e at row x of the stimulus vector. If the location is already occupied (more than one stimulus points into a node), add the value to value currently at the location (in other words, the element in the stimulus vector contains the sum of all stimuli pointing into the node). If an arrow comes from another node, put the weight of the arrow in the weights matrix at the row corresponding to the node that the arrow points at and the column corresponding to where the arrow originated. In other words, if an arrow with weight A points from node y to node x, and the weights matrix is designated as **W**, then put the value A at location W_{xy}. If the element in the weights matrix is already occupied (more than one arrow points from one node to another), then add the weight to the value already occupying the location. When done, set all the unused locations in the weights matrix and the stimulus vector to zero. When the above mentioned steps are completed for *Figure **8***, we get equation [6]:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -SL_{11} & 1 & 0 & -SL_{12} & 0 & 0 & 0 & 0 \\ -SL_{21} & 0 & 1 & -SL_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & -Su_{11} & 1 & 0 & -Su_{12} & 0 & 0 \\ 0 & 0 & -Su_{21} & 0 & 1 & -Su_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 & -SR_{22} & 1 & 0 & -SR_{21} \\ 0 & 0 & 0 & 0 & -SR_{12} & 0 & 1 & -SR_{11} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \\ n_6 \\ n_7 \\ n_8 \end{pmatrix} = \begin{pmatrix} e_1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ e_2 \end{pmatrix}$$

[6]

Equation [6] is in a form with little hope for solving the unknown S-parameters, unless something different is done. Without any rational as yet, observe that it is possible to group the nodes according to certain criteria:

1. Nodes 1 & 8 are incident waves on the entire system and are therefore the measured incident waves and are labeled a_{msd}.
2. Nodes 2 & 7 are reflected waves from the entire system are therefore the measured reflected waves and are labeled b_{msd}.
3. Nodes 3 & 6 are the incident waves on the DUT and are labeled a.
4. Nodes 4 & 5 are the reflected waves from the DUT and are labeled b.

Equation [6] needs to be re-written in a form that re-orders things according to a special grouping of nodes. Note that the default way of writing the equations for systems is always the canonical form, meaning that the node ordering is always the same as the equation ordering. By using *permutation matrices* the nodes and equations need to be ordered in the following order:

$$\begin{pmatrix} b_{msd} & a & a_{msd} & b \end{pmatrix}. \qquad [7]$$

A permutation matrix is a matrix that when placed to the left of a matrix reorders the rows and when placed to the right of a matrix reorders the columns. When placed to the left, it is said to perform row operations and when placed to the right it is said to perform column operations. A row permutation matrix is formed by reordering the rows of the identity matrix corresponding to the desired row ordering. In this example, we want to group the nodes and equations for nodes n_1 through n_8 in the order $\begin{pmatrix} b_{msd} & a & a_{msd} & b \end{pmatrix}$. To do this, we say that the current system of equations is in the form $\hat{E}\hat{v} = \hat{e}$ where $\hat{E}$ represents the system characteristics matrix, $\hat{v}$ represents an arbitrary node ordering (and equation ordering), and $\hat{e}$ represents the corresponding arbitrary stimulus ordering. We develop a row permutation matrix P such that $P\hat{v} = v$ has the new desired ordering. We know that we must multiply P from the left of $\hat{E}$ to get the desired equation ordering. Less obviously, if we multiply $P^{-1} = P^T$ from the right of $\hat{E}$ we reorder the nodes. Thus, we have:

$$P\hat{E}P^T P\hat{v} = P\hat{e} \qquad [8]$$

where

$$E = P\hat{E}P^T,\ e = P\hat{e} \text{ and } Ev = e.$$

To solve the current de-embedding problem, we first group the nodes in order $\begin{pmatrix} b_{msd} & a & a_{msd} & b \end{pmatrix}$. In order to do so we write node order as $\begin{pmatrix} n_2 & n_7 & n_3 & n_6 & n_1 & n_8 & n_4 & n_5 \end{pmatrix}^T$. Observe that [9] achieves such a reordering:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \\ n_6 \\ n_7 \\ n_8 \end{pmatrix} = \begin{pmatrix} n_2 \\ n_7 \\ n_3 \\ n_6 \\ n_1 \\ n_8 \\ n_4 \\ n_5 \end{pmatrix}$$

[9]

So, we define P as the left side of [9] and substitute equation [6] into [8] and we obtain:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & -SL_{11} & 0 & -SL_{12} & 0 \\ 0 & 1 & 0 & 0 & 0 & -SR_{11} & 0 & -SR_{12} \\ 0 & 0 & 1 & 0 & -SL_{21} & 0 & -SL_{22} & 0 \\ 0 & 0 & 0 & 1 & 0 & -SR_{21} & 0 & -SR_{22} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -Su_{11} & -Su_{21} & 0 & 0 & 1 & 0 \\ 0 & 0 & -Su_{12} & -Su_{22} & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} n_2 \\ n_7 \\ n_3 \\ n_6 \\ n_1 \\ n_8 \\ n_4 \\ n_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ e_1 \\ e_2 \\ 0 \\ 0 \end{pmatrix}$$

[10]

Some observations are in order regarding [10]. First, we can drive the system any way we want, so we choose to alternately drive the system first with $e_1 = 1$, then with $e_2 = 1$. Under these conditions, $\mathbf{e} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & I & \mathbf{0} \end{pmatrix}^T$. When we do this, the node vector, as grouped becomes $\mathbf{v} = \begin{pmatrix} \mathbf{A} & \mathbf{B_{msd}} & \mathbf{A_{msd}} & \mathbf{B} \end{pmatrix}^T$ where the elements of $\mathbf{v}$ are two-by-two block matrices.

We can also group E into two-by-two element block matrices. When we do this, we obtain [11] :

$$\begin{pmatrix} 1 & 0 & -F_{11} & -F_{12} \\ 0 & 1 & -F_{21} & -F_{22} \\ 0 & 0 & 1 & 0 \\ 0 & -Su & 0 & 0 \end{pmatrix} \begin{pmatrix} B_{msd} \\ A \\ A_{msd} \\ B \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ I \\ 0 \end{pmatrix}$$

[11]

In [11], all block matrices are here two-by-two. Expanding these equations, we have:

$$B_{msd} - F_{11} \cdot A_{msd} - F_{12} \cdot B = 0$$

[12]

$$A - F_{21} \cdot A_{msd} - F_{22} \cdot B = 0$$

[13]

$$A_{msd} = I$$

[14]

$$B - Su \cdot A = 0 \,.$$

[15]

Equation [14] says that the measured incident waves are identity as we expect since we have alternately driven the ports with 1. Under these conditions, we know that the measured reflected waves are equal to the measured overall system S-parameters, so we substitute $B_{msd} = Sk$ where Sk is the known system S-parameters. We substitute [14] into [12] and solve for B :

$$B = F_{12}^{-1} \cdot (Sk - F_{11})$$

[16]

We substitute [16] into [13] and solve for A :

$$A = F_{21} - F_{22} \cdot B$$

[17]

Finally, we substitute [17] and [16] into [15] and solve for the unknown S-parameters Su :

$$Su = BA^{-1} \,.$$

[18]

Equations [16], [17] and [18] provide the recipe for de-embedding. To perform the full calculation for the two-port case, note that

$$F_{11} = \begin{pmatrix} SL_{11} & 0 \\ 0 & SR_{11} \end{pmatrix}$$

$$F_{12} = \begin{pmatrix} SL_{12} & 0 \\ 0 & SR_{12} \end{pmatrix}$$

$$F_{21} = \begin{pmatrix} SL_{21} & 0 \\ 0 & SR_{21} \end{pmatrix}$$

$$F_{22} = \begin{pmatrix} SL_{22} & 0 \\ 0 & SR_{22} \end{pmatrix}$$

$$B = \begin{pmatrix} SL_{12} & 0 \\ 0 & SR_{12} \end{pmatrix}^{-1} \left[\begin{pmatrix} Sk_{11} & Sk_{12} \\ Sk_{21} & Sk_{22} \end{pmatrix} - \begin{pmatrix} SL_{11} & 0 \\ 0 & SR_{11} \end{pmatrix} \right]$$

$$B = \begin{pmatrix} \frac{Sk_{11} - SL_{11}}{SL_{22}} & \frac{Sk_{12}}{SL_{12}} \\ \frac{Sk_{21}}{SR_{12}} & \frac{Sk_{22} - SR_{11}}{SR_{12}} \end{pmatrix}$$

$$A = \begin{pmatrix} SL_{21} & 0 \\ 0 & SR_{21} \end{pmatrix} + \begin{pmatrix} SL_{22} & 0 \\ 0 & SR_{22} \end{pmatrix} \begin{pmatrix} \frac{Sk_{11} - SL_{11}}{SL_{22}} & \frac{Sk_{12}}{SL_{12}} \\ \frac{Sk_{21}}{SR_{12}} & \frac{Sk_{22} - SR_{11}}{SR_{12}} \end{pmatrix}$$

$$A = \begin{pmatrix} \frac{Sk_{11}SL_{22} - |SL|}{SL_{12}} & \frac{Sk_{12}SL_{22}}{SL_{12}} \\ \frac{Sk_{21}SR_{22}}{SR_{12}} & \frac{Sk_{22}SR_{22} - |SR|}{SR_{12}} \end{pmatrix}$$

$$Su = \begin{pmatrix} \frac{Sk_{11} - SL_{11}}{SL_{22}} & \frac{Sk_{12}}{SL_{12}} \\ \frac{Sk_{21}}{SR_{12}} & \frac{Sk_{22} - SR_{11}}{SR_{12}} \end{pmatrix} \begin{pmatrix} \frac{Sk_{11}SL_{22} - |SL|}{SL_{12}} & \frac{Sk_{12}SL_{22}}{SL_{12}} \\ \frac{Sk_{21}SR_{22}}{SR_{12}} & \frac{Sk_{22}SR_{22} - |SR|}{SR_{12}} \end{pmatrix}^{-1},$$

[19]

where $|SL| = SL_{11}SL_{22} - SL_{12}SL_{21}$ is the determinant of the matrix SL and similarly $|SR|$ is the determinant of the matrix SR.

Again note that if the three S-parameter matrices - SL, SR and Su are known then once can calculate the S-parameters of the cascaded system (B_{msd}) by substituting equations [14] and [15] in equation [13] to obtain:

$$A = (I - F_{22}Su)^{-1}F_{21},$$

which can then be in equation [12] to obtain

$$S_{cascade} = B_{msd} = F_{11} + F_{12} \cdot Su \cdot A.$$

The next section describes how equations [16], [17] and [18] are used to perform de-embedding for an P-port DUT shown in *Figure **1***.

De-embedding for P-port DUT

To see how the two-port DUT de-embedding algorithm described above can be used in a general case, note that the system shown in *Figure 7* can be redrawn to resemble the general problem shown in *Figure **1***.

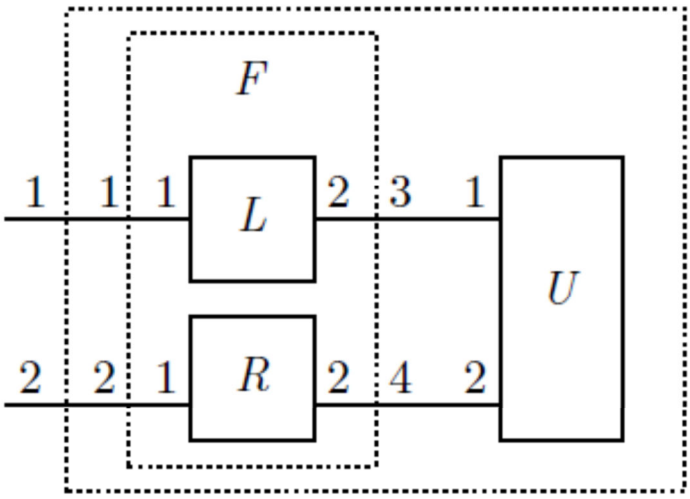


Figure 9: System diagram for de-embedding in two-port DUT

In *Figure **9***, we have exactly the same connection of the two-port elements L and R between the DUT and the system ports and the exact same de-embedding problem. Here, however, we have chosen to place all of the system ports and the DUT ports to the left. Furthermore, we have chosen to place a box around elements L and R, treating them in aggregate as a four-port fixture device F. We have numbered the ports on F exactly as specified for the fixture de-embedding arrangement as shown in *Figure **1***.

It is a trivial matter to establish the s-parameters of the fixture as drawn. They are:

$$\mathbf{F} = \begin{pmatrix} SL_{11} & 0 & SL_{12} & 0 \\ 0 & SR_{11} & 0 & SR_{12} \\ SL_{21} & 0 & SL_{22} & 0 \\ 0 & SR_{21} & 0 & SR_{22} \end{pmatrix}$$

Note that these s-parameters (or more precisely, the negative of these s-parameters) fit neatly into the upper right quadrant of the system characteristics matrix shown in [10] and we can solve the general fixture de-embedding problem as a more general case of the two-port de-embedding case.

To de-embed a $2P$-port fixture with S-parameters F from a P-port unknown DUT with s-parameters Su given known system s-parameters Sk for a fixture de-embedding arrangement as shown *Figure **1***, partition the fixture S-parameters in the block matrices:

$$\mathbf{F_{11}} = \begin{pmatrix} F_{11} & F_{12} & \cdots & F_{1P} \\ F_{21} & F_{22} & \cdots & F_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ F_{P1} & F_{P2} & \cdots & F_{PP} \end{pmatrix}$$

$$\mathbf{F_{12}} = \begin{pmatrix} F_{1(P+1)} & F_{1(P+2)} & \cdots & F_{1(2P)} \\ F_{2(P+1)} & F_{2(P+2)} & \cdots & F_{2(2P)} \\ \vdots & \vdots & \ddots & \vdots \\ F_{P(P+1)} & F_{P(P+2)} & \cdots & F_{P(2P)} \end{pmatrix}$$

$$\mathbf{F_{21}} = \begin{pmatrix} F_{(P+1)1} & F_{(P+1)2} & \cdots & F_{(P+1)P} \\ F_{(P+2)1} & F_{(P+2)2} & \cdots & F_{(P+2)P} \\ \vdots & \vdots & \ddots & \vdots \\ F_{(2P)1} & F_{(2P)2} & \cdots & F_{(2P)P} \end{pmatrix}$$

$$\mathbf{F_{22}} = \begin{pmatrix} F_{(P+1)(P+1)} & F_{(P+1)(P+2)} & \cdots & F_{(P+1)(2P)} \\ F_{(P+2)(P+1)} & F_{(P+2)(P+2)} & \cdots & F_{(P+2)(2P)} \\ \vdots & \vdots & \ddots & \vdots \\ F_{(2P)(P+1)} & F_{(2P)(P+2)} & \cdots & F_{(2P)(2P)} \end{pmatrix}$$

Then, solve for the unknown DUT s-parameters using the solution already established in the previous section:

$$\mathbf{B} = \mathbf{F_{12}}^{-1}\left(\mathbf{Sk} - \mathbf{F_{11}}\right)$$

$$\mathbf{A} = \mathbf{F_{21}} + \mathbf{F_{22}}\mathbf{B}$$

$$\mathbf{Su} = \mathbf{B}\mathbf{A}^{-1}$$

The de-embedding methodology described so far provides a way to obtain DUT S-parameters by de-embedding known S-parameters from the measured system S-

parameters. It also provides a way to analyze such interconnected system and can be used not only for de-embedding but also to obtain S-parameters of cascade combinations of different blocks. Next sections describe de-embedding methods when it is not easy to obtain the S-parameters of the fixture or when only partial information about the fixture is known.

TRL Calibration

TRL (for Thru Reflect Line) calibrations can be used for partial de-embedding of test fixtures, including transmission lines and launch effects, including probes SMA's, 3.5 mm's and 2.4mm connectors. There are limitations in de-embedding small structures, such as vias to connectors, due to their non transmission line like structure characteristics. This part of the paper will address SE TRL structures, used as a first tier or second tier tool.
The acronym TRL identifies both a method for correcting two-port network analyzer S-parameters and the set of calibration standards used in the network analyzer calibration. The method involves measuring the standards at the ends of the test connections, such as the coax test cables on the VNA or the tips of circuit probes. The TRL standards consist of two or more lengths of uniform transmission line that differ only in length (including connector response), and a reflection standard that can be either an open- or short-circuit reflection (or any arbitrary, non-zero reflection coefficient). The shorter transmission line is taken as the THRU standard, and in the case of mating test ports (called insertable connections), the Thru standard may be zero length. Figure 10 depicts the signal flow diagram of the calibration standards.

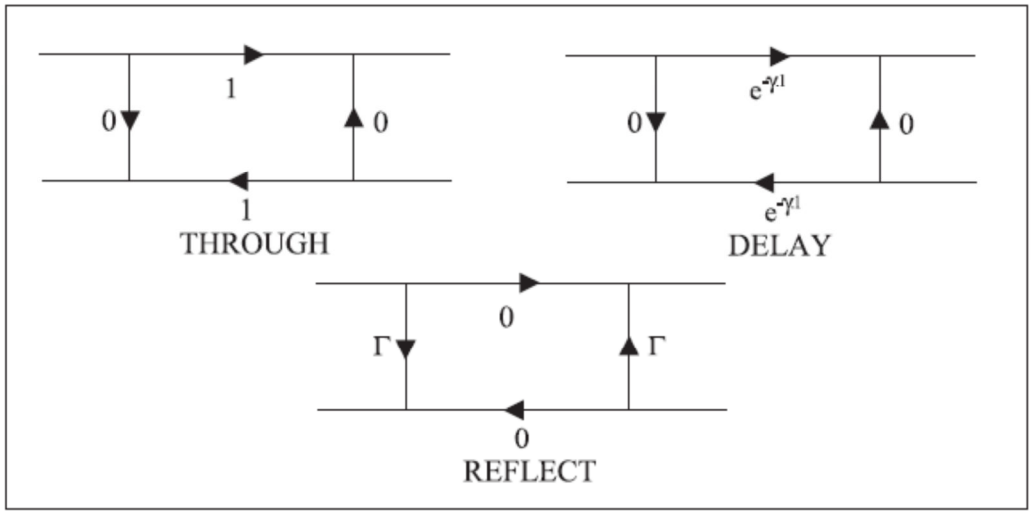


Figure 10: Flow Graph TRL Standard

Today, TRL is used as a routine method to correcting commercial vector network analyzer instruments, providing an alternative to the more common SOLT (Short-Open-Load-Thru) method, however, this method goes back to the early days of VNA research and may be attributed to (Engen & Hoer, 1979), and the method encompasses more

recent work of the NIST Multiline Method (Marks, 1991), (DeGroot, Jargon, & Marks, Multiline TRL revealed, 2002), and (DeGroot, Rolain, Pintelon, & Schoukens, 2004).

Engen and Hoer were looking for a traceability path for the impedance normalization necessary to define scattering parameters. That is, in order of two labs to exchange S-parameters, they both need to agree on the impedance environment in which the S-parameters were obtained, like the common 50 Ω reference. Engen and Hoer chose the coaxial line filled with air (or vacuum) as their impedance standard since the RLGC (resistance, inductance, conductance and capacitance) parameters of the lines could be calculated from first principles given the transmission line geometry and the uncertainty of the geometrical measurements. Otherwise, they were working with a pure ratio that would vary based on the internal source and termination impedance of their network analyzer, and could not link scattering parameters to any fundamental standard.

They defined the S-parameters of their standards to be ideally matched to reference impedance (Zref). For the Thru and Line Standards, the defined S-parameters were given as:

$$\mathbf{S}_L^D = \begin{pmatrix} 0 & e^{-\gamma\ell} \\ e^{-\gamma\ell} & 0 \end{pmatrix},$$

where l is the length of the line and γ is the propagation factor (loss and phase per unit length) of the line,

$$\gamma = \alpha + j\beta = \sqrt{(R + j\omega L)(G + j\omega C)}.$$

Like wise, they defined the S-parameters of the reflection standards, allowing for a short length of offset line going between the coaxial connector and the actual standard:

$$\mathbf{S}_R^D = \begin{pmatrix} \Gamma_R e^{-2\gamma\ell} & 0 \\ 0 & \Gamma_R e^{-2\gamma\ell} \end{pmatrix},$$

where Γ_R is the non-zero reflection coefficient of the termination.

As a consequence of their TRL calibration algorithm, the calibration reference plane would be located in the exact middle of their Thru standard (the shorter line). If they used a finite length Thru standard, the reference plane would be located at a nice region of a uniform line that would be free from all connector disturbances and discontinuities.

Researchers working on wafer-level S-parameter testing (Jargon & Marks, 1995), quickly adopted TRL because of this ability to locate the measurement reference plane a distance away from contact pads. While contact pads and probe connections could be highly made repeatable, the wafer-level contacts created all sorts of electromagnetic field

discontinuities and mode-conversion problems when the measurement reference plane coincided with the contact pads.

Further, on-wafer measurements completely prohibited the location of any other standards right at the port connections of their transistors or other device under test (DUT). Since the TRL algorithm computes the propagation factor of the transmission line during the calibration, that "ideal" reference plane is mathematically moved from its default location (half the length of the Thru standard). It can be mathematically translated closer to the launch connector, or it can be translated forward to the point where we want the DUT to start.

Figure 11 shows a set of on-waver coplanar waveguide (CPW) standards and the location of the TRL reference planes for two Lines and the Reflect

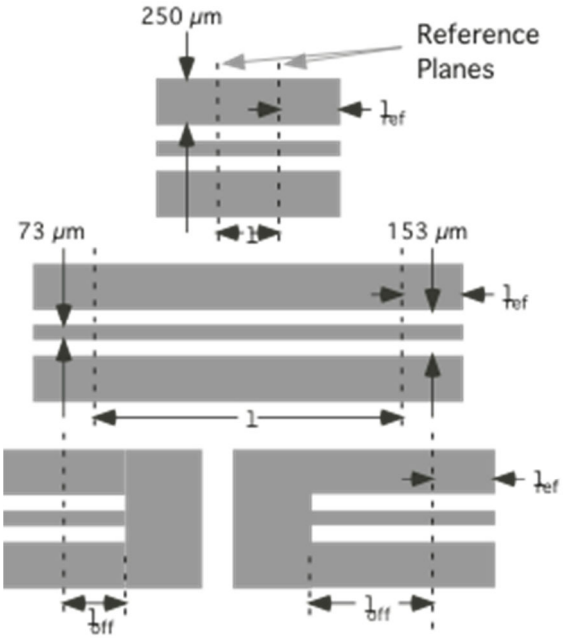


Figure 11: Example TRL standards for on-wafer CPW applications

In this way, TRL was used as an on-wafer de-embedding method, in addition to a vector network analyzer correction method. It could naturally be both at the same time, but if the DUT were not located at nice distance away from the contact pads (exactly half of the Thru length), then the engineer could de-embed a section of connecting transmission line by mathematically translating the reference plane to the port of the DUT.

The trick was to make the TRL lines match the geometry of the device connections so that the propagation factor γ would be the same for the standards and the DUT.

With the increase in operating frequencies in signal integrity design, circuit board designers are now facing the same problems of the on-wafer world—the signal launches are terrible places to take measurements and the DUTs are not located at a connector that accepts VNA calibration standards.

In the Signal Integrity world, TRL can again be used as both VNA correction method and de-embedding method.

This comes with a problem to solve. We need a standard transmission line in order to define the reference impedance of our S-parameters. For near-ideal coaxial measurements, the standards are easily created for standard connection types, like 3.5mm or K-type connectors. The coaxial TRL standards clearly define the impedance and allow the reference plane to coincide with the physical connection points. However, circuit boards have transmission lines that are not that uniform over long lengths and they have significant signal losses. Both unwanted features of circuit boards present frequency-dependent line impedance that must be accounted properly.

The line impedance,

$$Z_{line} = \sqrt{\frac{R + j\omega L}{G + j\omega C}},$$

depends on frequency, whenever the line resistance (R) or the dielectric losses (G term) are significant.

Later in the section, we discuss a calculator for choosing line geometries to select the lengths necessary for a chosen frequency band and to reduce losses to keep the reference impedance nearly constant with frequency.

There are a few practical considerations to remember before applying TRL as shown later in this paper. First, we must realize that not all vector network analyzers measure S-parameters the same way. Figure 12 shows an ideal vector network analyzer that measures both waves traveling incident to the DUT and waves reflecting back from the DUT. The correction method compensates all the non-ideal connection paths at every frequency point so that the instrument reports the actual S-parameters at the DUT plane, not the uncorrected ratio of forward and reverse waves measured internally.

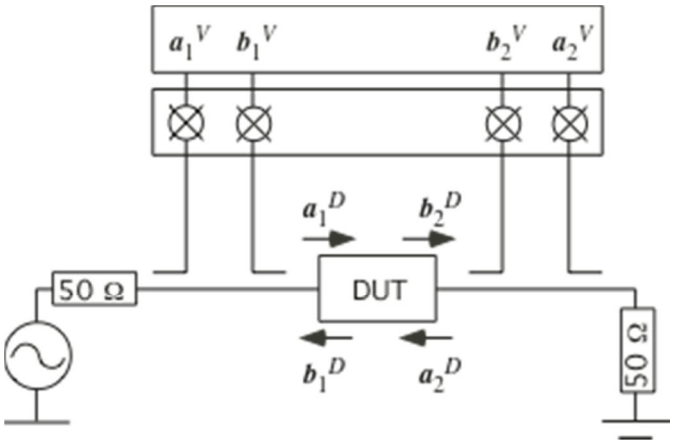


Figure 12: Ideal two-port VNA configuration with source at port 1.

If we consider the actual S-parameters for a VNA measurement (not the simplistic definition in an engineering text book), we would see that measured S-parameter value includes all incident and reflecting waves.

$$S_{i1} = \frac{\left(\frac{b_{i1}}{a_1}\right)^F - \left(\frac{a_2}{a_1}\right)^F \left(\frac{b_i}{a_2}\right)^R}{1 - \left(\frac{a_1}{a_2}\right)^R \left(\frac{a_2}{a_1}\right)^F}$$

$$S_{i2} = \frac{\left(\frac{b_i}{a_2}\right)^F - \left(\frac{a_1}{a_2}\right)^R \left(\frac{b_i}{a_1}\right)^F}{1 - \left(\frac{a_1}{a_2}\right)^R \left(\frac{a_2}{a_1}\right)^F}$$

The unwanted terms only go away with perfect impedance match between DUT, cables, and instruments, and that never happens in practical terms.

As a result, the four wave amplitudes must be measured simultaneously to provide sufficient information to map uncorrected VNA waves to actual DUT waves using the early version of the TRL algorithm [6]. Since only the upper-end VNA's measure sufficient information for this model to hold, we often have to perform two calibrations to accommodate lower-end VNAs. The first-tier calibration can be SOLT right at the ends of the VNA's coaxial test cables. The second-tier calibration can be TRL using a custom calibration standard like that computed below, or provided by a third-party service. The SOLT cal takes care of the VNA issues, and the TRL cal gets the second reference plane located on the circuit board so it can be mathematically translated towards the DUT port. The second tier calibration method is described in detail in the later part of the paper.

Another practical consideration is that not all VNAs provide full TRL calibration capabilities, and when they do, the translation of the reference plane may not be obvious. NIST and others provide stand-alone software [7,8] to perform Multiline TRL corrections and de-embedding. We show both the practical steps of performing TRL de-embedding with a commercial VNA capable of the full TRL algorithm, and we show the practical steps of using third-party software as the second-tier de-embedding step for a TDR-based vector network analyzer.

Practical Considerations for TRL

As mentioned earlier, care must be taken in designing the TRL calibration standards. Errors appear when the standard use for the calibration is different from the device under test fixture. For example if the fixture does not have the same board thickness as the thru standard, its frequency response will not be the same as the standard. A good TRL

calibration standard can provide poor results if it is not well implemented, by design. Factors such as board layout, PCB thickness and PCB routing play a role in the accuracy of the TRL standard.

Since the measurement of delay lines is a substantial part of the calibration definition, the difference between an open, a through, and a line can be measured and defined on a smith chart, by looking at phase shift. To avoid having several standards to close in phase, the delay lines must comply with several rules. The S12 differential phase of the delay line has to be within a 20 to 160° limit when compared to the through line.

The fixture used for this paper, shown below, consists of 2.5" SE routed transmission lines on either side of a standard Molex Impact RA backplane connector.

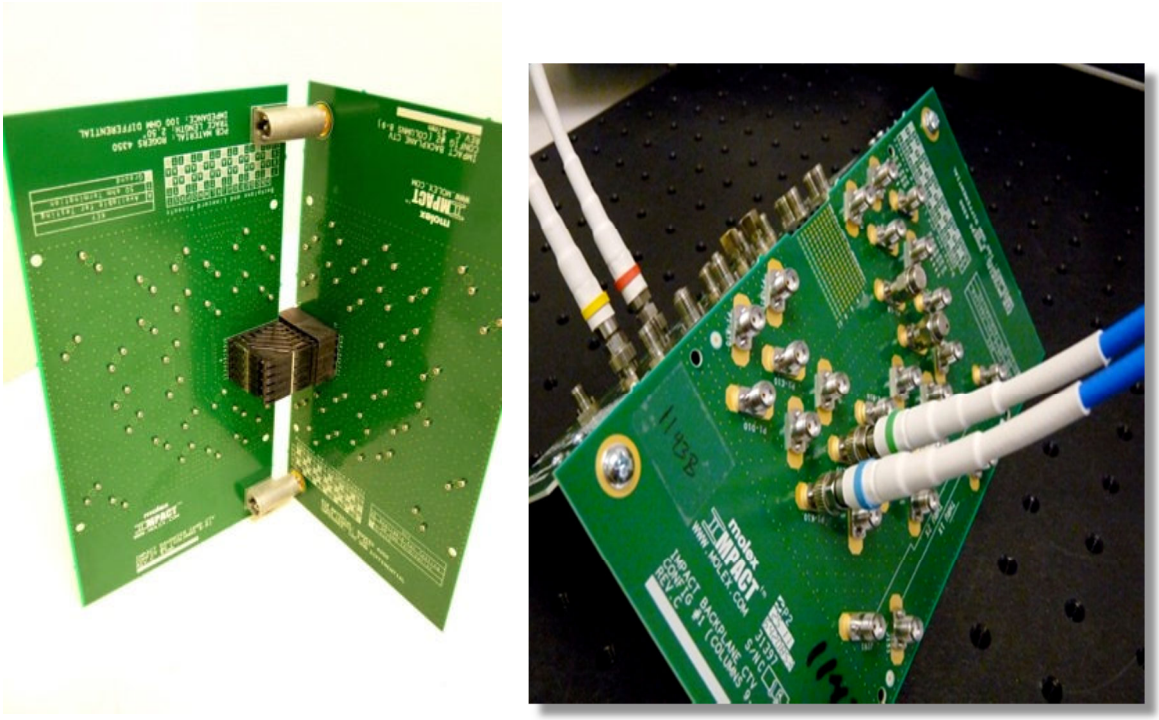


Figure 13: DUT & fixture: Backplane connector on the test board

In order to de-embed the 2.5 in traces a 3 line TRL calibration structure was also created on the same ctv panel, shown in Figure 14.

Some of the variables in this TRL structure are: TRL traces only reside on 1 layer and in one x/y orientation. Actual test traces have various orientations and fan_outs and are on 3 different signal layers. The design of cal kit was created using the TRL calculator shown in Figure 15.

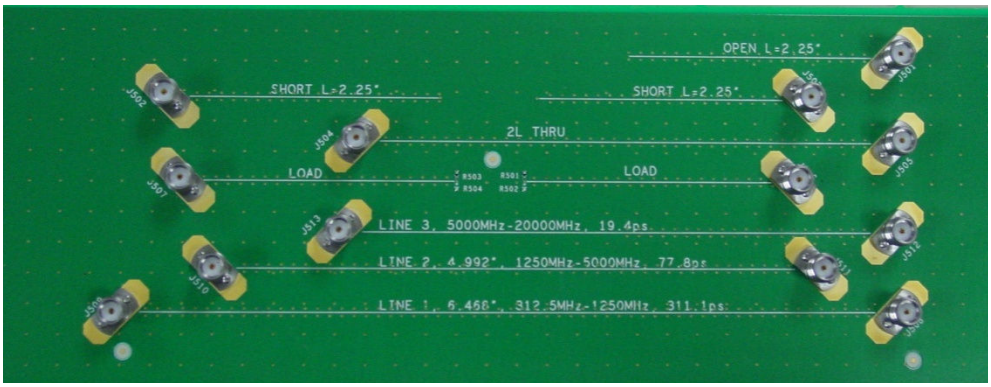


Figure 14: TRL calibration structure

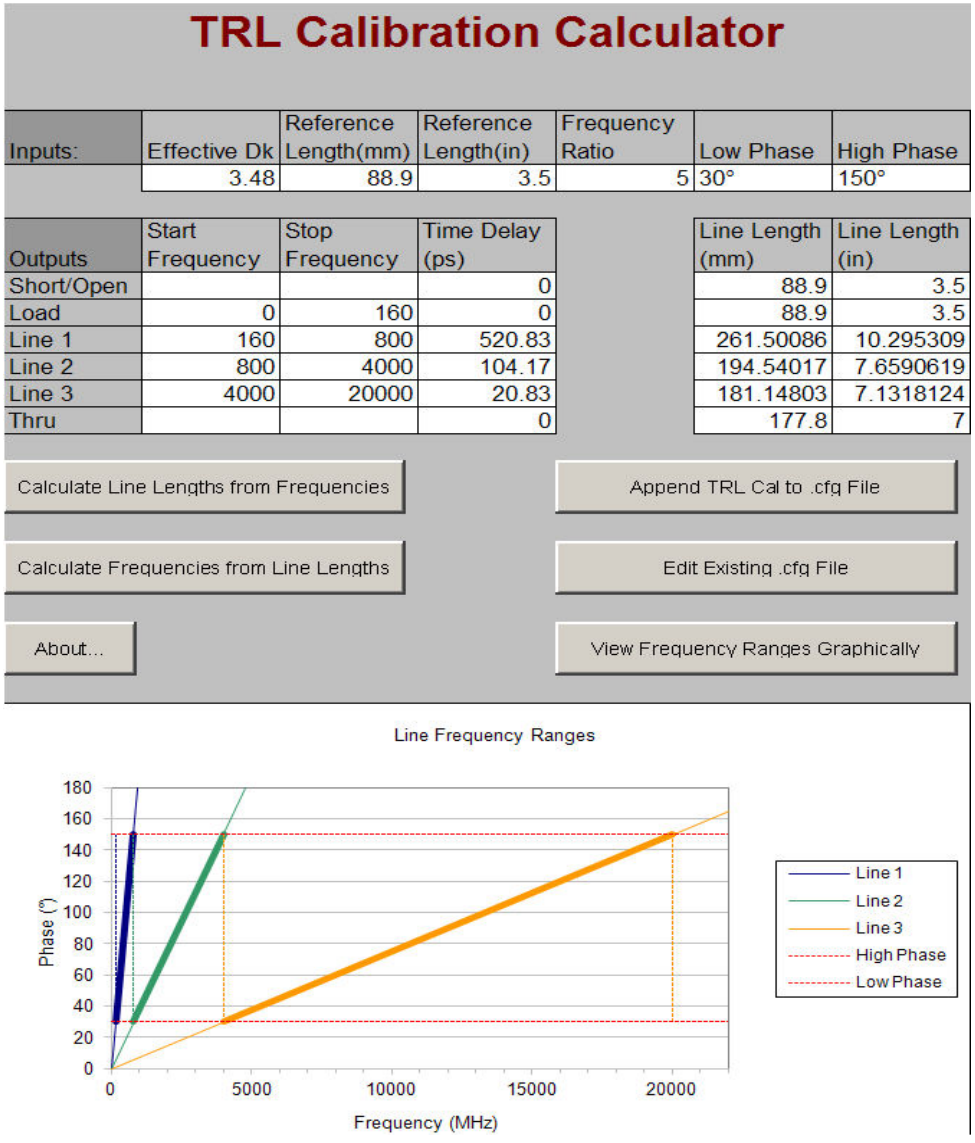


Figure 15: TRL calculator

De-embedding of the SMA's and pcb loss gives a more accurate representations of the DUT performance, as shown in the plots below:

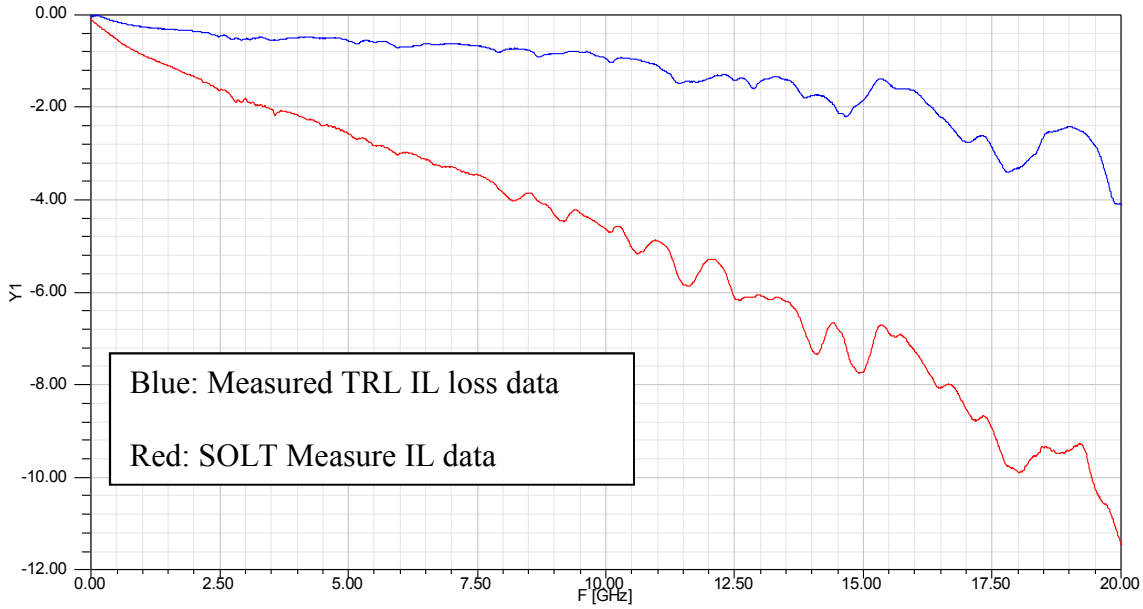


Figure 16: SOLT Data vs TRL Data

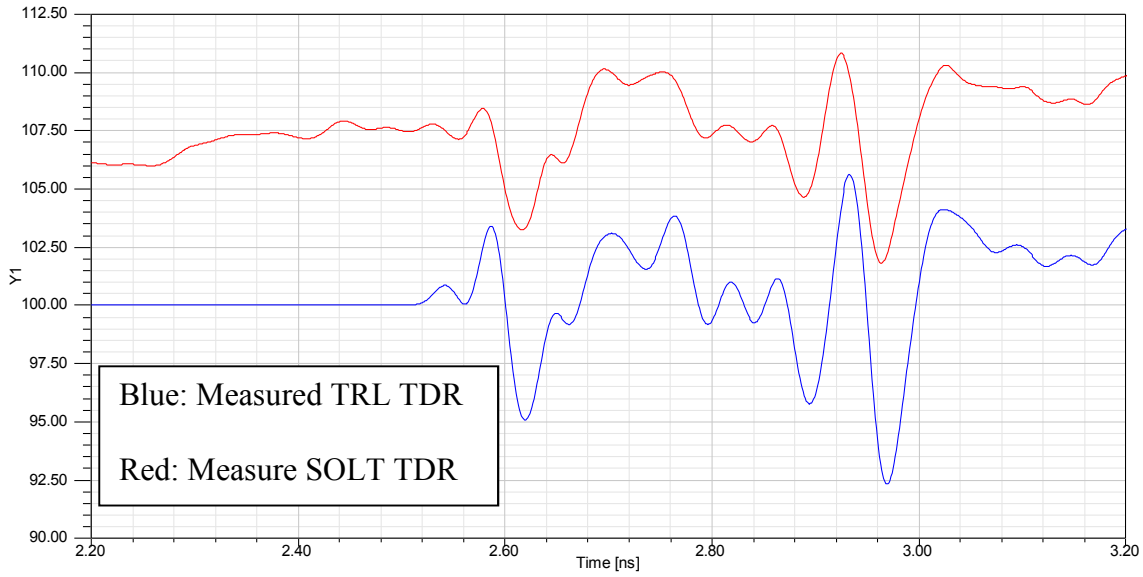


Figure 17: TDR Plots

Several details are noticed between the TRL data vs the original SOLT data.

1. Due to PCB filtering, solt rise times are slower that the TRL, noticed by the larger impedance reflections.

2. Because PCB's are not perfect, the reference pcb impedance of the solt data, is at approx 106 ohms, also modifying real connector impedance

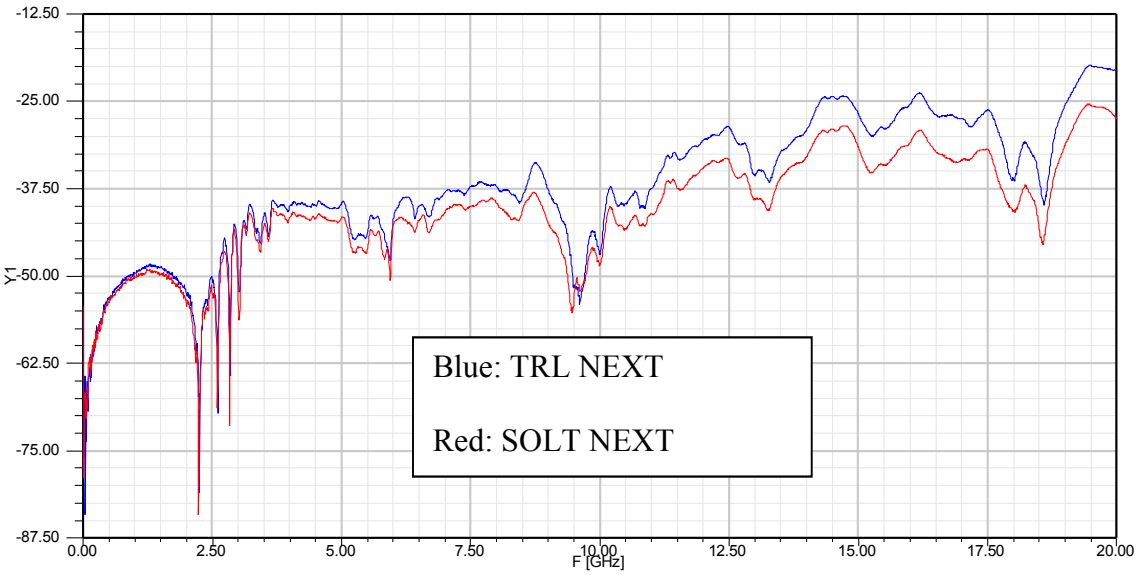


Figure 18: NEXT Freq Crosstalk

The slight increase in TRL crosstalk is expected, due to the faster rise time hitting the connector DUT.

In the next section, basics of second tier calibration method are described along with the usage of third-party TRL algorithms as a second tier calibration to perform de-embedding.

Second Tier Calibration

As will be explained below, second tier calibration can be used when the following conditions are met:

1. When partial information about the S-parameters of the fixture to de-embed is available.
2. The fixture characteristics do not drift as frequently as the network analyzer
3. It is possible to connect calibration standards at the end of the fixture.

The last condition raises an important question regarding the benefit of second tier calibration: If one can connect the calibration standards at the end of the fixture, then why not perform a regular calibration? As explained earlier, the purpose of the calibration is to account for the systematic errors in the network analyzer. The network analyzer drifts with time and/or temperature and needs to be re-calibrated - often during the course of the day. The case where second tier calibration technique is useful is when it is not always easy to connect the calibration standards, or it is time consuming to perform the usual calibration when the fixture is in place. For example, consider measuring S-parameters using a probe. As shown in *Figure **19*** the probe is connected to the network

analyzer via cables and the S-parameters of the DUT are measured by touching the tips of the probe to the DUT ports.

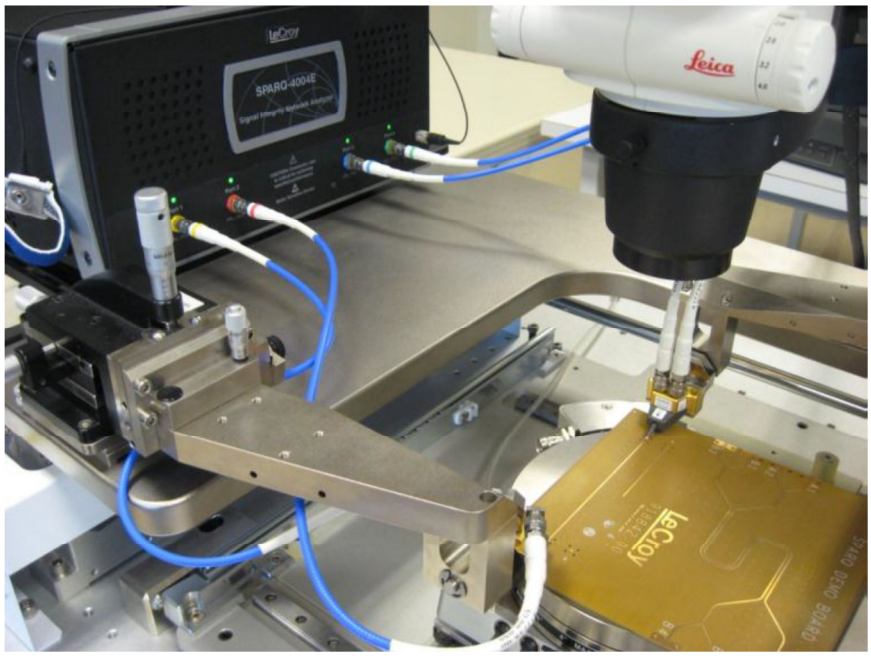


Figure 19: Board measurement using a Cascade Microtech Probe station

Probe manufacturers also provide substrate with calibration standards. As shown in *Figure **20***, to perform calibration, one needs to measure the S-parameters of the impedance standard substrates (ISSs) using the probe.

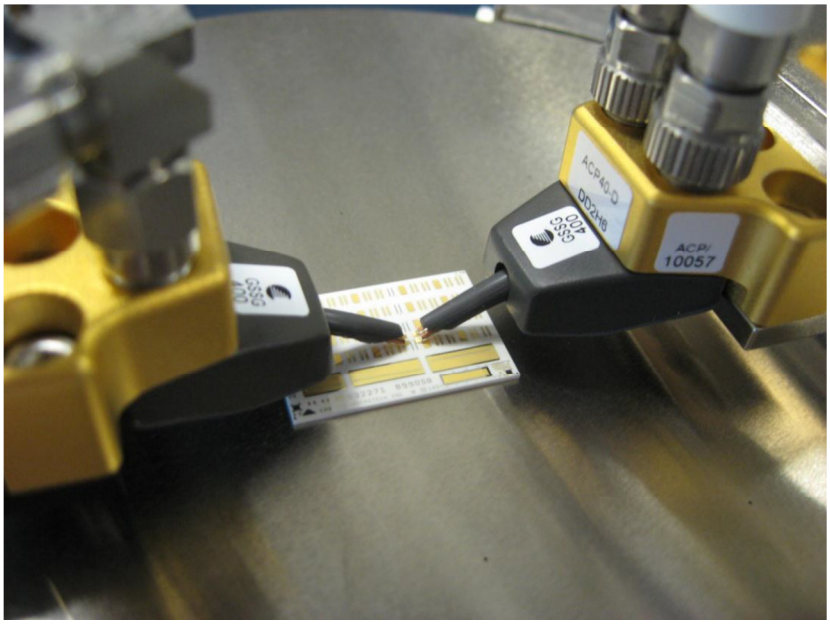


Figure 20: Cascade probes probing Impedance Standard substrates for Calibration

At the end of the calibration process, the reference plane would be at the tip of the probes and the set-up would be ready to measure S-parameters of the DUT. But now if due to drift, the network analyzer needs to be re-calibrated then the entire process needs to be repeated. Connecting the probes to different calibration standards either requires careful attention making it a time-consuming exercise or requires expensive probe positioners. Re-using the calibration substrates also reduces their life. To alleviate the above problems one can perform second tier calibration described below.

Consider the probe setup while measuring a one port DUT. As a part of SOLT second tier calibration consider measuring an open standard on the ISS. The network analyzer is already calibrated without the probe and now with the probes connected, the open standard is measured. This measurement is actually that of the probe in cascade with the open standard (see Figure 21). The loss and delay characteristics of the probe are embedded in the measurement of the open standard. Now if one were to re-measure the same open standard the next day by following the same procedure, then one should expect to get the same measurement (discounting for errors due to random noise). This will be true if the probe or the open standard didn't change its characteristics overnight. Same holds true while measuring the short (Figure 22) and the load (Figure 23) calibration standard. For all the three cases, one has the measurements of the probe in cascade with the calibration standard.

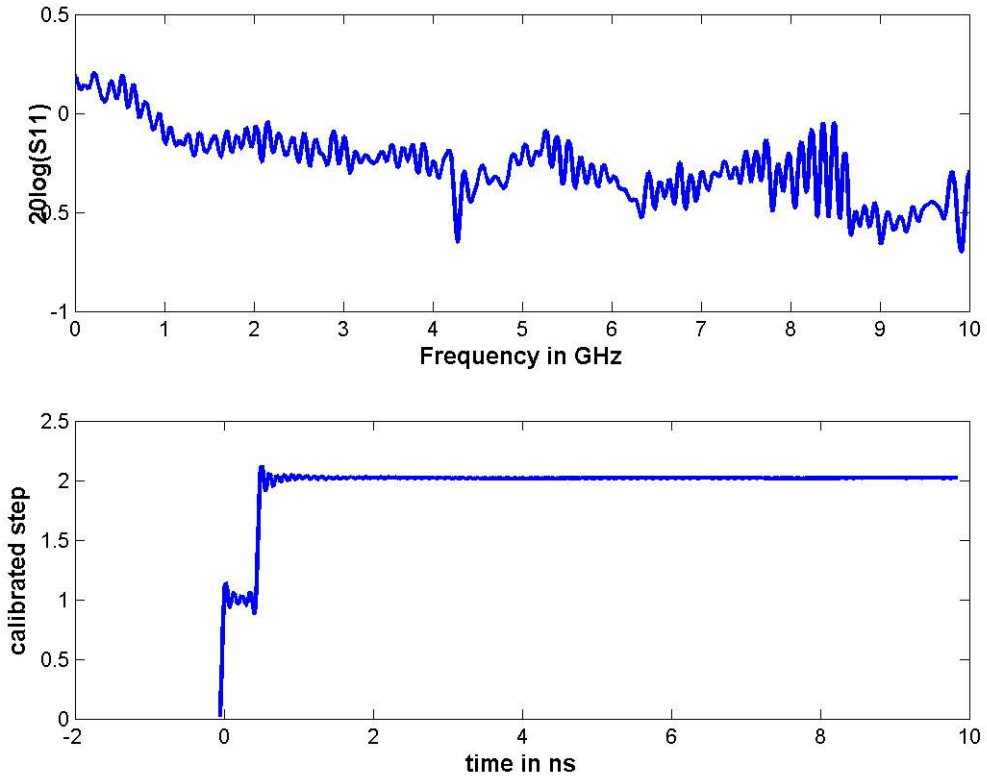


Figure 21: Measurement of open standard - S11 in dB and calibrated step

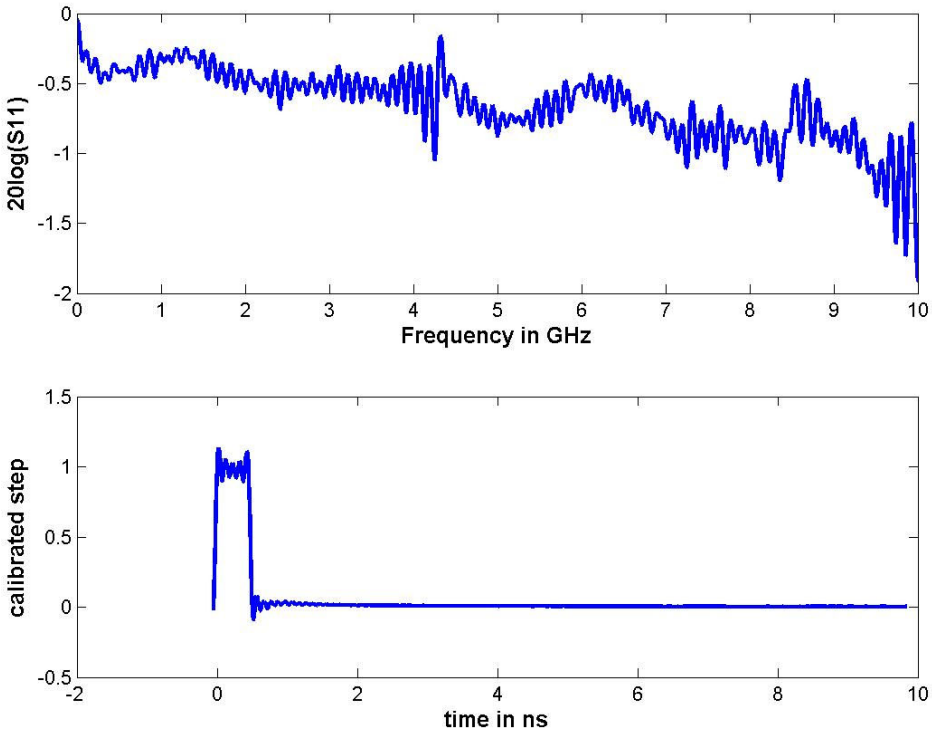


Figure 22: Measurement of short standard - S11 in dB and calibrated step

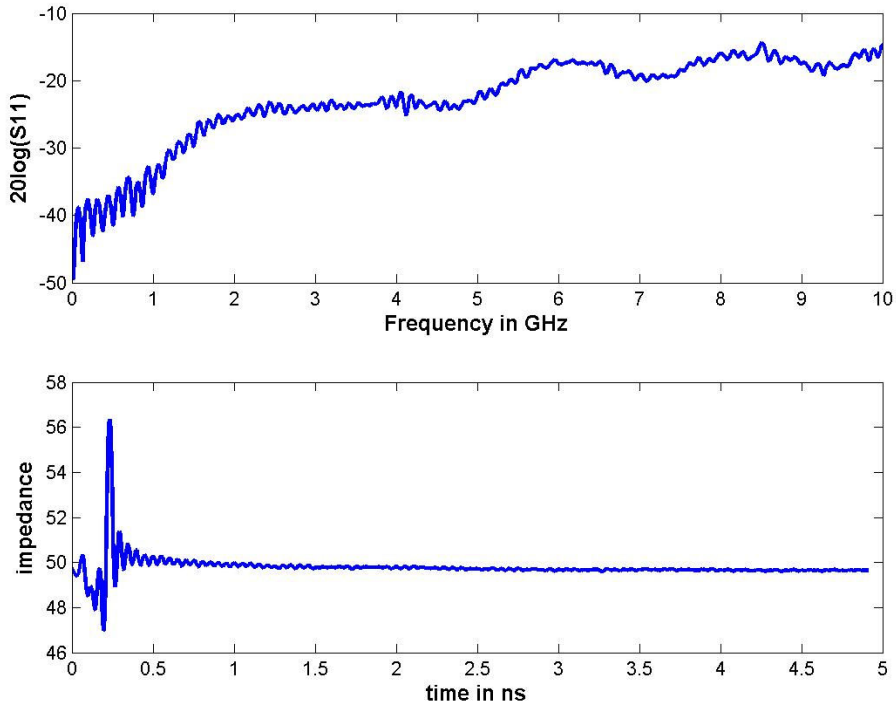


Figure 23: Measurement of load standard - S11 in dB and impedance profile

The loss in the open and short standard measurement, the delay in the calibrated steps and the reflection in the impedance profile of the load standard measurement is due to the probe. The effects of the probe are captured in the measurements of these standards. And since we know the characteristics of the calibration standards, we can calculate the error terms shown in Figure 24.

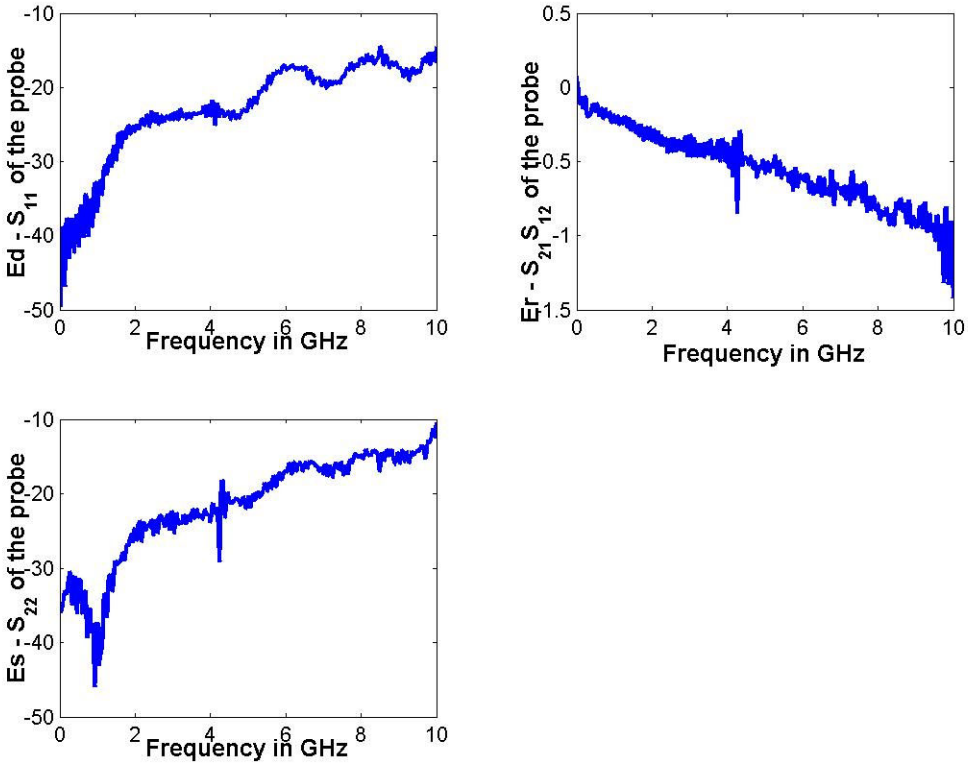


Figure 24: Error terms from the measurements of the calibration standards

These error terms capture the S-parameters of the probe. Now if a DUT measurement is made, it can be corrected by using the above error terms and the algorithm described in (Wittwer & Pupalaikis, 2008). Since the assumption is that the probe characteristics don't change as frequently as the drift in the network analyzer, each time the calibration standards are measured with a calibrated network analyzer, one must get the error terms shown in Figure 24. Hence one can avoid the measurements of the standards using the probe and perform only the regular calibration of the network analyzer. So each time the only measurement required is that of the DUT.

If the user suspects that the probe characteristics have changed then one can redo the second tier calibration procedure and get a new set of error terms. Although the procedure described here is for the measurement of a one-port DUT, it holds true for P-port DUT. The only difference would be the measurement of more calibration standards, e.g. thru standards between various pairs of ports, and SOL standards on different ports. Application note (LeCroy-SPARQ, 2012) describes the details for a four port measurement.

Although the second tier calibration procedure described here was using the standard SOLT calibration, one can use any desired calibration technique to remove the effects of probe or fixture. Next section describes the use of TRL calibration and third-party software to generate error terms and hence remove the effects of fixture.

Second Tier Calibration using TRL

Since TRL locates the Calibration Reference Plane to some point along a uniform transmission line. It is ideal for de-embedding through second-tier calibration. Not all VNAs and TDNAs provide second-tier capabilities, so third-party software is required. NIST and commercial sources provide stand-alone software to perform Multiline TRL corrections and de-embedding. Here is how the process proceeds:

1. Design and fab (or procure) Multiline Calibration Coupon as shown above
2. Complete first-tier SOLT calibration at end of VNA test cables
3. Connect test cables to Multiline standards on a test coupon (using either probes or coax connectors).
4. Measure first-tier-corrected S-parameters for all Multiline standards on Calibration Coupon
5. Use third-party Multiline software to compute second-tier correction coefficients
6. Measure first-tier-corrected S-Parameters for all DUTs using same interconnect geometry as that used in the Calibration Coupon
7. Use third-party Multiline software to de-embed DUT by correcting all first-tier DUT data with the second-tier VNA calibration

We applied second-tier TRL de-embedding using CCN's Multiline software (based on Ref [4]). Figure 25 and Figure 26 show the results of processing both VNA and SPARQ measurements. Both instruments were calibrated with SOLT as First Tier, then used to measure the TRL Calibration Coupon, and then used to measure the DUT. We note general agreement between the second-tier de-embedding and the other methods.

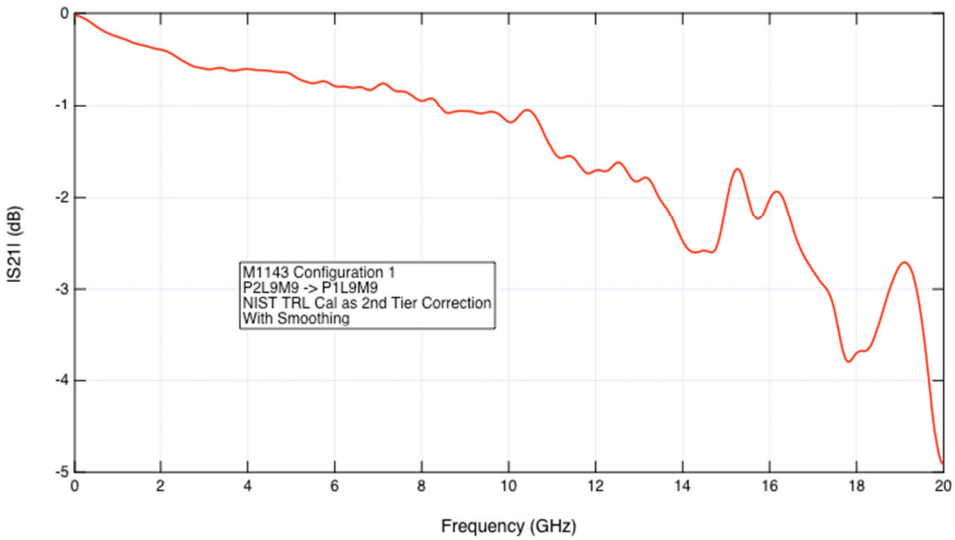


Figure 25: Insertion Loss obtained using TRL as second tier calibration

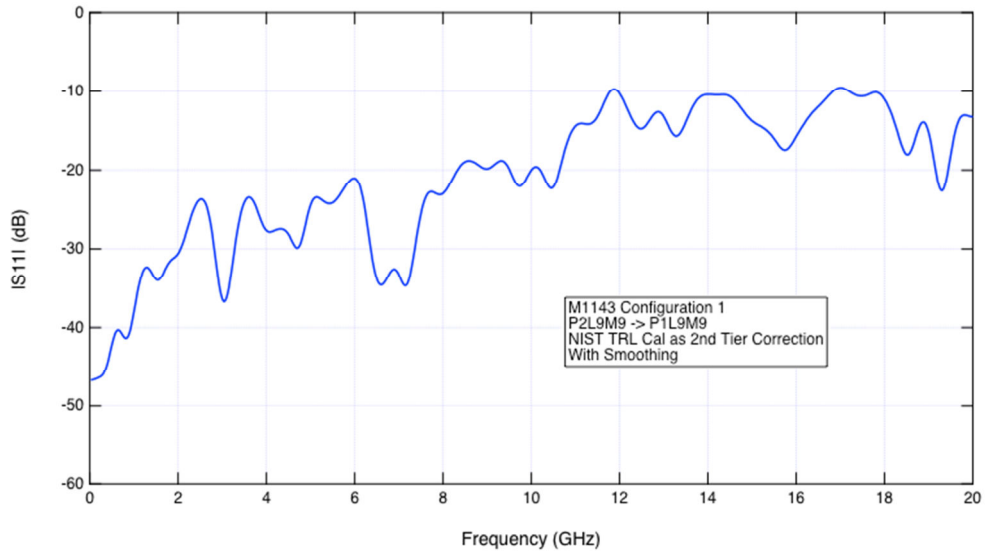


Figure 26: Return Loss obtained using TRL as second tier calibration

The next section describes the use of time domain gating method to perform de-embedding for the same measurements.

Time-domain Gating

As described in TRL section, one of the drawbacks of the algorithm is that the calibration fixture and the fixture to be de-embedded are two different structures and it is necessary for accurate de-embedding that the two structures have the same characteristics. The output of the algorithm matches the true DUT S-parameters when the two fixtures are the same. To apply the second tier calibration method one needs to be able to connect the calibration standards at the end of the fixture. In practice it might not always be possible to achieve the above requirements. In such settings time domain gating method described below can be used.

The main idea behind the gating method is that if by using system knowledge and by observing the measured S-parameters of the cascade system, if one can decide where the fixture ends and where the DUT begins, then one can use this information to generate the S-parameters of the fixture and then use equations [16], [17] and [18] to de-embed the fixture. If any additional calibration fixture measurement is available then that can be used to improve the accuracy of the fixture S-parameters to be de-embedded.

Consider the connector measurement described in the TRL section. S-parameters of the system comprising of the SMA connector – a through trace – DUT (connector) – a through trace – SMA connector is shown Figure 27. For convenience only the SDD21 and SCC21 are shown, along with the impedance profile due to SDD11 and SCC11. For the following explanation only the SDD11 port is considered. Similar reasoning applies to all the ports of the measured system. The frequency domain S-parameter measurement SDD11 can be converted into impedance profile using any of the algorithms described in

(Hayden & Tripathi, 1994), (Hsue & Pan, 1997), and (Smolyansky & Corey, 1999). Impedance Profile concepts are introduced in tutorial form in (TDR Primer).

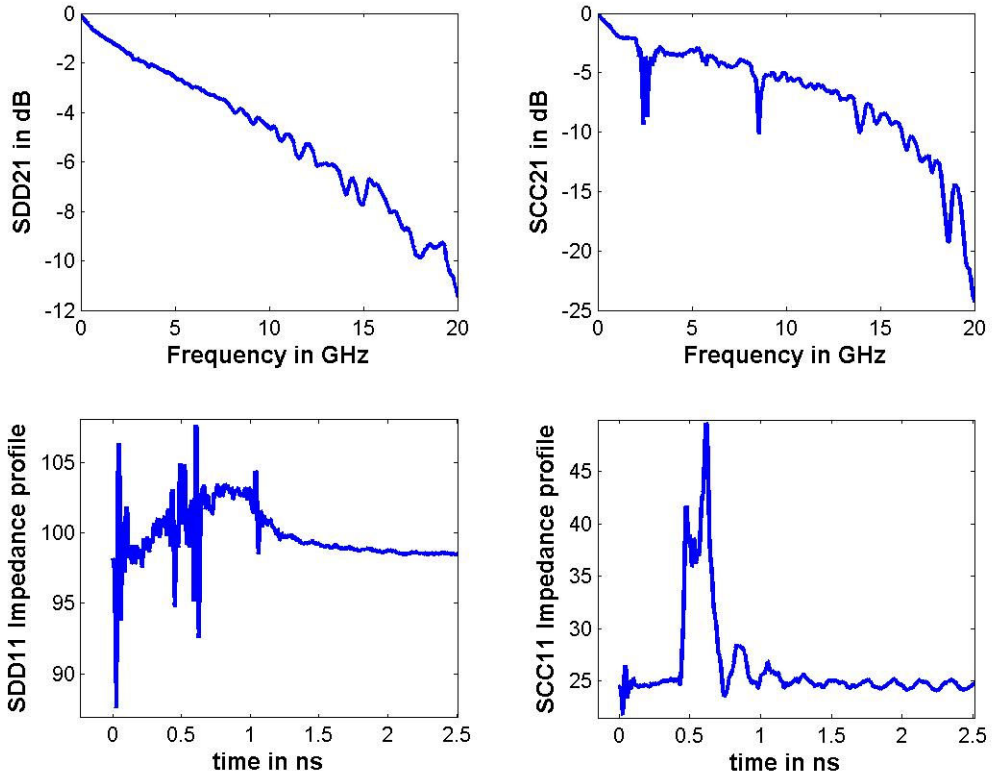


Figure 27 : Mixed mode S-parameters of the system comprising of the fixture and the DUT

The impedance profile due to SDD11 is zoomed in Figure 28. Here one can clearly see the three different impedance profiles. The one shown in the dotted line is that of the DUT. The impedance profile in Figure 28 describes how the differential impedance of the device changes as one traverses from differential port one to differential port two. The solid curve on the left of the dotted curve corresponds to the fixture on the left of the DUT, while the solid curve on the right of the dotted curve corresponds to the fixture on the right. To generate the S-parameters of the fixture on the left, the first information needed is the length of the fixture. As can be seen from the plot, it is approximately 400 ps.

To de-embed the fixture on the left we would like to obtain the two port S-parameters of this fixture from the SDD11 measurement and the knowledge of group delay characteristics and loss characteristics.

As a simple case, consider the impedance profile shown in Figure 29. Here again the dotted line indicates the impedance profile of a DUT and the solid lines correspond to the impedance profile of the fixture on the left and the right. Here for simplicity, the

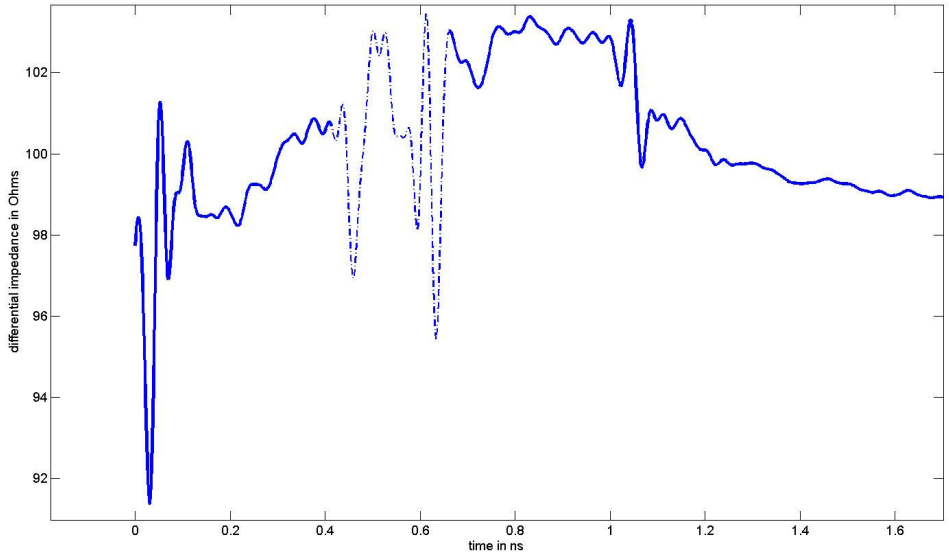


Figure 28 : Impedance profile due to SDD11

impedance profile is generated using single ended measurement of return loss. This simple example has constant characteristic impedance for the entire length of the fixture and the DUT. The fixture on the left corresponds to a transmission line of characteristic impedance of 47 ohms and a length of approximately 400 ns.

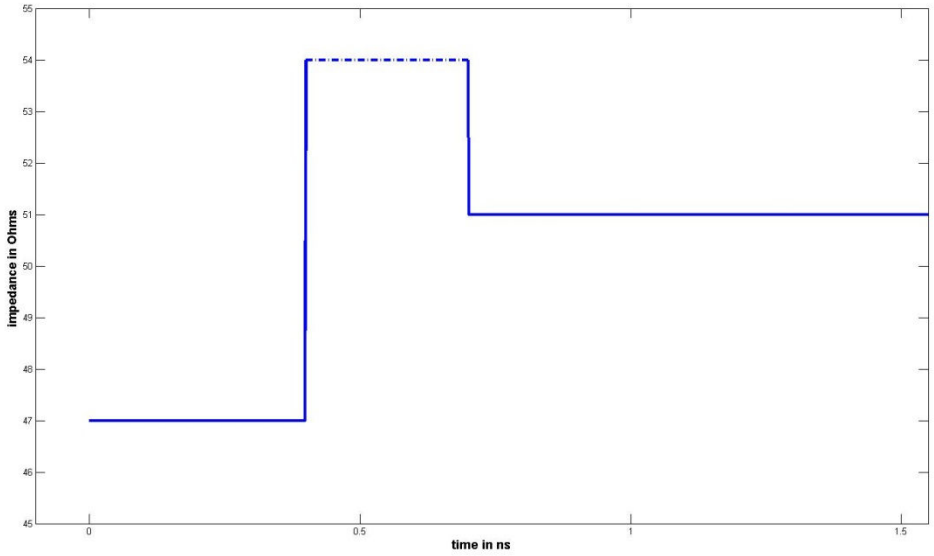


Figure 29: Impedance Profile for a simple case of fixture and DUT with constant characteristic impedance

Given the characteristic impedance one can generate the reflection coefficient ρ by using the well-known relationship given in equation [20] where Z0 is the reference impedance of the s-parameters and is arbitrary and is generally 50.

$$\rho = \frac{Z - Z0}{Z + Z0}.$$

[20]

A single section of transmission line can be approximated as shown in Figure 30. Here ρ is calculated using equation [20] and the impedance value for that particular section and z^{-1} is given as:

$$z^{-1} = Le^{-j2\pi fT},$$

[21]

where T is the delay (or the length of the transmission line) and L is the gain. In this case the gain is assumed to be one, i.e. lossless transmission line. The S-parameters of the transmission line and hence the fixture is given by equation [22]:

$$S_{tl} = \frac{1}{1 - z^{-2}\rho^2}\begin{pmatrix} \rho(1 - z^{-2}) & z^{-1}(1 - \rho^2) \\ z^{-1}(1 - \rho^2) & \rho(1 - z^{-2}) \end{pmatrix}.$$

[22]

Now instead of impedance profile of a constant value if one has the impedance profile shown in Figure 28, then one can cascade several such small transmission line structures, so that the combined impedance profile and length correspond to the actual fixture.

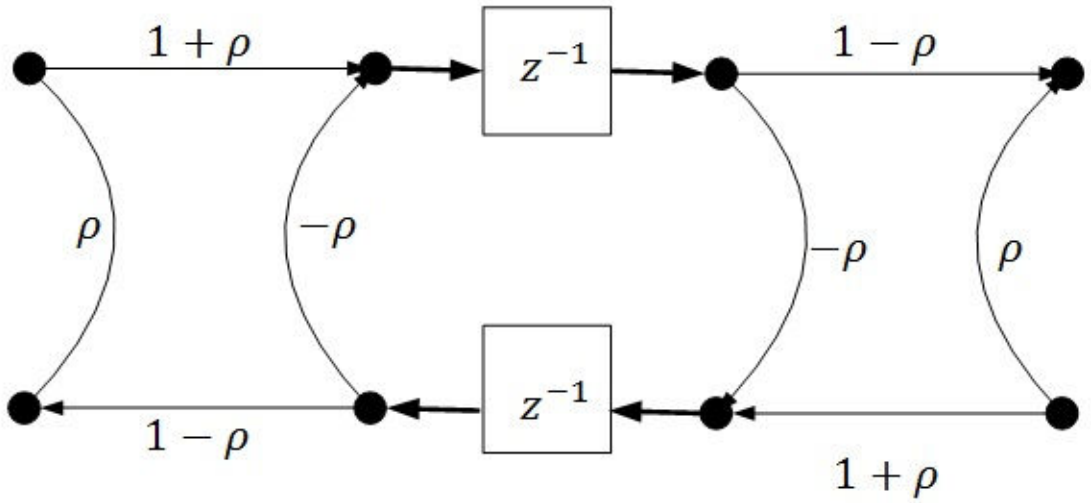


Figure 30: Signal flow diagram representation of a single transmission line

An algorithm provided in Figure 38 is utilized to generate the impedance profile, given the length of the fixture, and the SDD11 measurement. Figure 38 provides an M element vector whereby each element ρ_m , $m \in \{0, \dots, M-1\}$ contains the reflection coefficient at an interface along the line. f is a vector of frequencies whereby each element f_n , $n \in \{0, \dots, N\}$ contains a frequency corresponding to a port return loss s-parameter designated S_{11}, such that S_{11_n} is the return loss for a port at frequency f_n . The element length T is generally set to $1/4Fe$, where Fe is the last frequency point (i.e. f_N). The

algorithm operates in a loop which determines the reflection coefficient at the first interface of the return loss s-parameters SDD11, removes the effect of this first interface from the return loss s-parameters, and finds the reflection coefficient of the next interface by finding the reflection coefficient of the first interface in the s-parameters calculated with the interface effect removed. This allows the algorithm to properly account for all reflections caused by multiple impedance discontinuities.

In Figure 38, there is a loss characteristic $G(f)$ and group-delay characteristic $D(f)$ both of which are specified per section and are frequency dependent. Loss characteristic $G(f)$ and group-delay characteristic $D(f)$ can be determined by additional calibration measurements (described in the thru2l section below). If such additional measurements are not available then one has to assume ideal values for the loss and the group delay characteristics.

Given the S-parameters of M such transmission line sections, they can be cascaded together to form the S-parameters of the fixture for that particular port. Similar procedure can be applied for all the ports in the DUT. Once the S-parameters of the fixture are known, the de-embedding algorithm described in [16], [17], and [18] can be applied to obtain the DUT S-parameters.

Since the procedure outlined for gating involves impedance profile for a trace it is required that there be minimal coupling between the lines. When lines are coupled, there is no single-ended impedance profile per se and it is preferable to convert to mixed-mode prior to application. While conversion to mixed mode mitigates to a large degree the coupling effects it does introduce the possibility for mode-conversion generally shown as cross-terms in the mixed-mode s-parameters. Fortunately, these are usually small.

The next section describes a procedure to obtain loss characteristics if additional calibration measurements are made.

Thru2L measurement

Figure 31 below shows a measurement of the calibration structure – SMA cable – a through trace that is twice the length of the trace in the fixture to be de-embedded – and an SMA connector. This measurement allows us to choose the loss and length parameters needed for time domain gating.

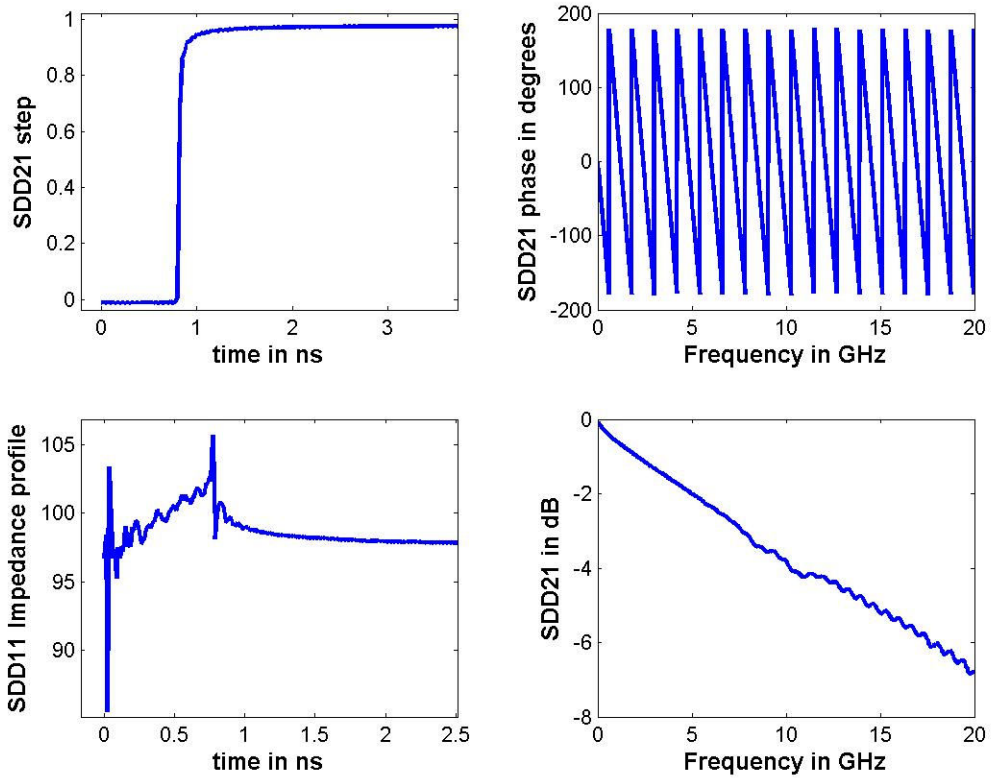


Figure 31: thru2L measurement

In the figure above, the top left picture is that of the calibrated step response due to SDD21. Top right picture corresponds to the phase response of SDD21. The bottom left plot corresponds to the impedance profile and the bottom right is the SDD21 measurement in dB. From the delay in the calibrated step or the length of the impedance profile, one can figure out the length of the fixture to be de-embedded. Since the thru here is twice the length, the delay value read from the step needs to be divided by two. This is the value to be used in the time domain gating algorithm. In this case the delay of single length thru is approximately 410 ps. The loss value (in dB\GHz\ns) needed for the gating can be calculated from the SDD21 plot. The slope of the curve gives the loss value in dB\GHz. And now since we know the length of the fixture, we need to divide the slope by the length of the fixture in ns to obtain the loss value. One point to note is that not all of SDD21 value can be attributed to loss – since there is a part of signal that does not go through because it gets reflected due to SDD11. Thus the number obtained for loss needs to be tuned using the criteria described below.

If we were to de-embed the partial fixture from the left and the other half from the right (using similar information from differential port 2), we should see zero delay in the step response, the phase plot should be flat, the impedance profile should be close to 100 ohms (because it corresponds to differential measurement) and the SDD21 should be 0 dB. So the loss and delay parameter can be tuned until the above mentioned criteria are reasonably met. Figure 32 starting from top left and moving clockwise shows the step

response, phase response and magnitude response due to SDD21 after gating and the bottom left plot is that of the impedance profile due to SDD11 when the thru 2L fixture is de-embedded from right as well as left.

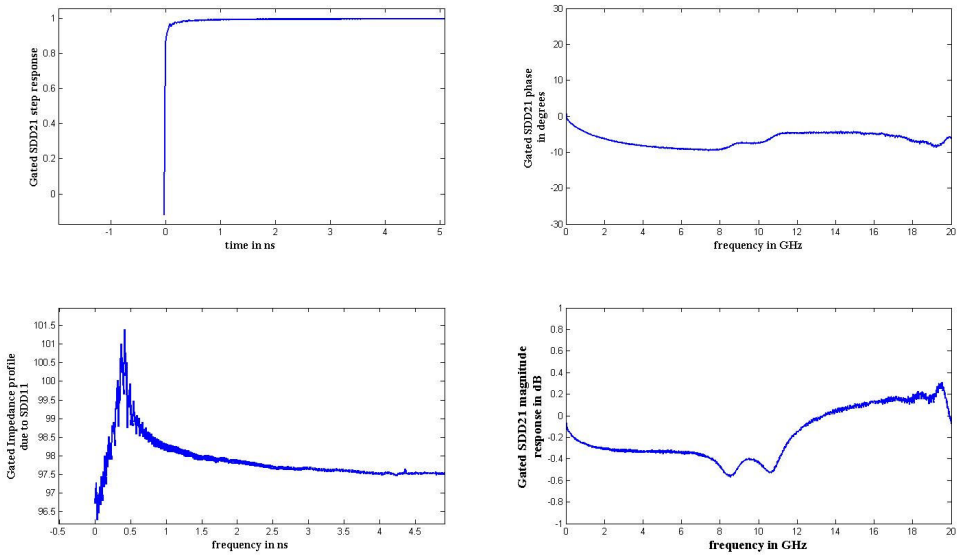


Figure 32: Thru2L after gating

Once the loss information is obtained from the calibration measurement one can now use the impedance profile from the SDD11 measurement of the DUT, and the information about the length of the fixture to obtain the transmission line model of the fixture that needs to be de-embedded. The above procedure needs to be repeated for all the ports of the DUT. The fixture can now be de-embedded by using the algorithm described in equations [16], [17], and [18]. As mentioned earlier, since gating requires minimal or no coupling between the lines, the de-embedding problem corresponds to that shown in Figure 33.

In Figure 34 the blue trace is the SDD21 before de-embedding while the red trace is the SDD21 after gating is applied. Similarly, Figure 35 shows the impedance profile before and after gating. The blue trace is the impedance profile of the fixture and the DUT which is supplied to the gating algorithm, and the red trace is the impedance profile of the DUT. As expected, the impedance profile of the DUT largely remains unchanged.

Few things to note about the gating algorithm:

1. The loss information obtained from the calibration measurement was used to generate a transmission line model with linear loss characteristics. One can expect better results if the loss as a function of frequency is used directly.
2. One improved de-embedding algorithm would be to combine gating and TRL. For TRL making the connecters repeatable for the calibration structure and the DUT is a big issue. One can gate the connectors from all the measurements and

then supply the gated results to the TRL algorithm. This way the non-repeatable connectors are not a part of the TRL de-embedding algorithm.

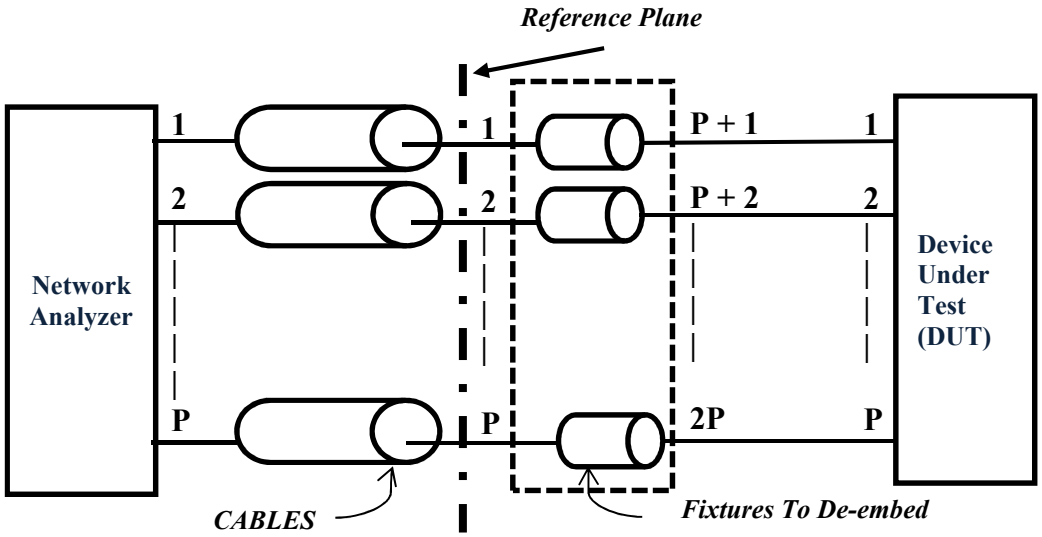


Figure 33: System diagram for de-embedding via time domain gating

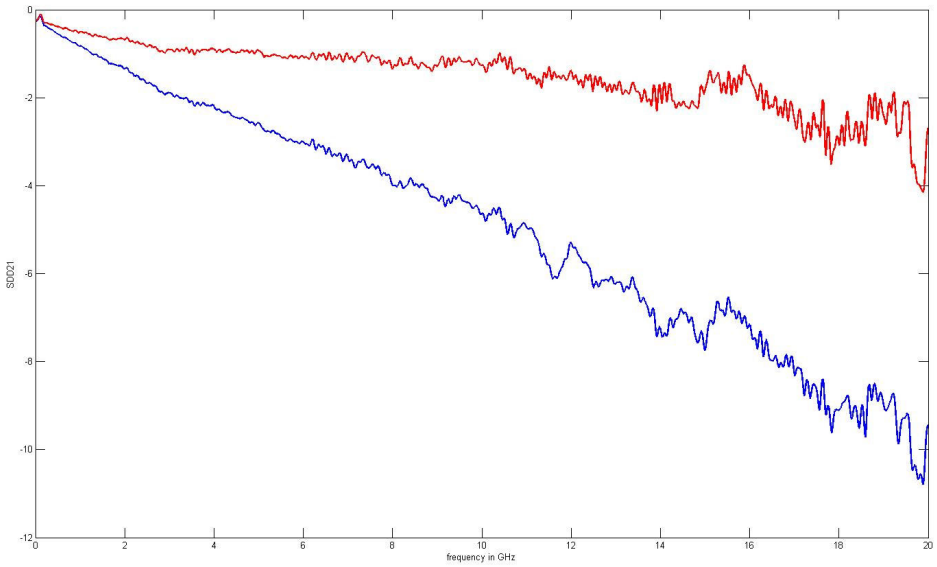


Figure 34: SDD21 – before and after gating

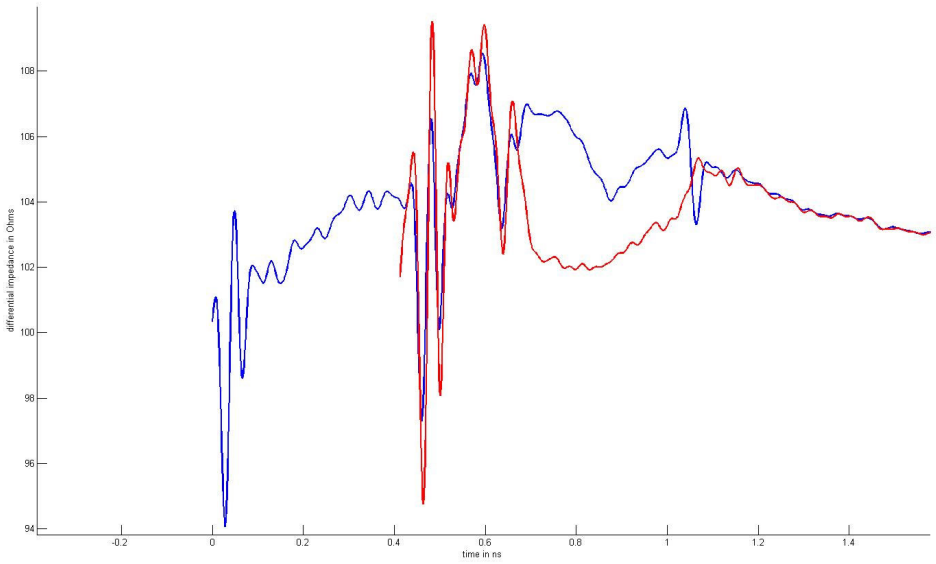


Figure 35: Impedance profile due to SDD11 – before and after gating

Figure 36 shows the de-embedded SDD21 by using the three different techniques described in the paper. The blue trace with diamonds is obtained by de-embedding using TRL as a first tiered calibration algorithm; the black trace with circles is obtained by de-embedding using TRL as a second tier calibration algorithm; and the red trace with the cross is obtained using time domain gating.

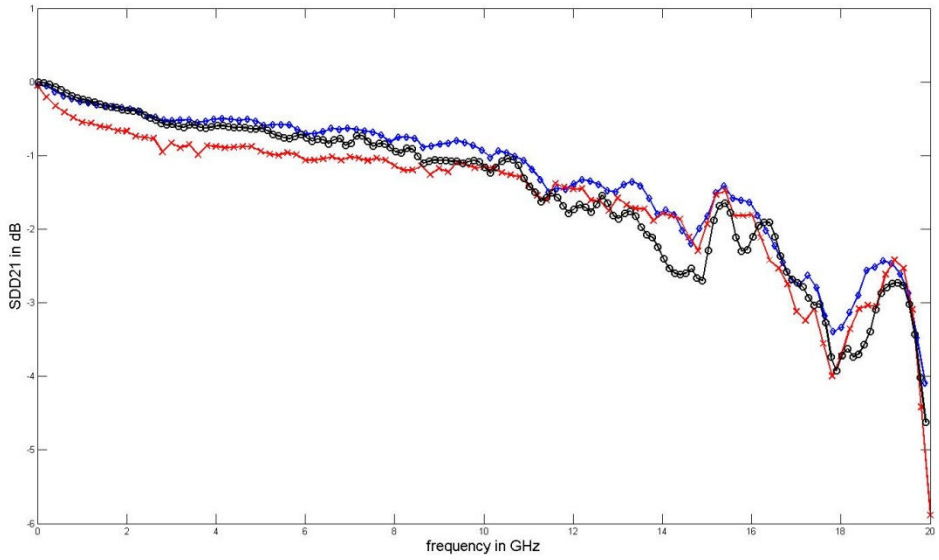


Figure 36: SDD21 comparison

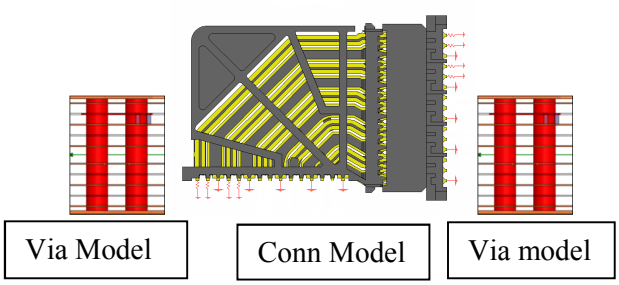
As mentioned in the gating section, the assumption of linear loss causes the slight mismatch with the other results.

Lastly, it is always a good practice to check the de-embedded results with analytical model of the DUT. The next section describes a method of generating S-parameters of the DUT by solving the models using EDM solvers.

Analytical Modeling Validation

Many component designs begin with analytical models, using EDM solvers such as Ansys HFSS or CST Microwave. In this paper connector and via models were solved in Ansys HFSS and were compared to the TRL data.

1). Models: The graphic below defines the models solved in HFSS:



IL plots comparing the measured TRL and modeled integrated via/connector model have good agreement, as shown in the plot below

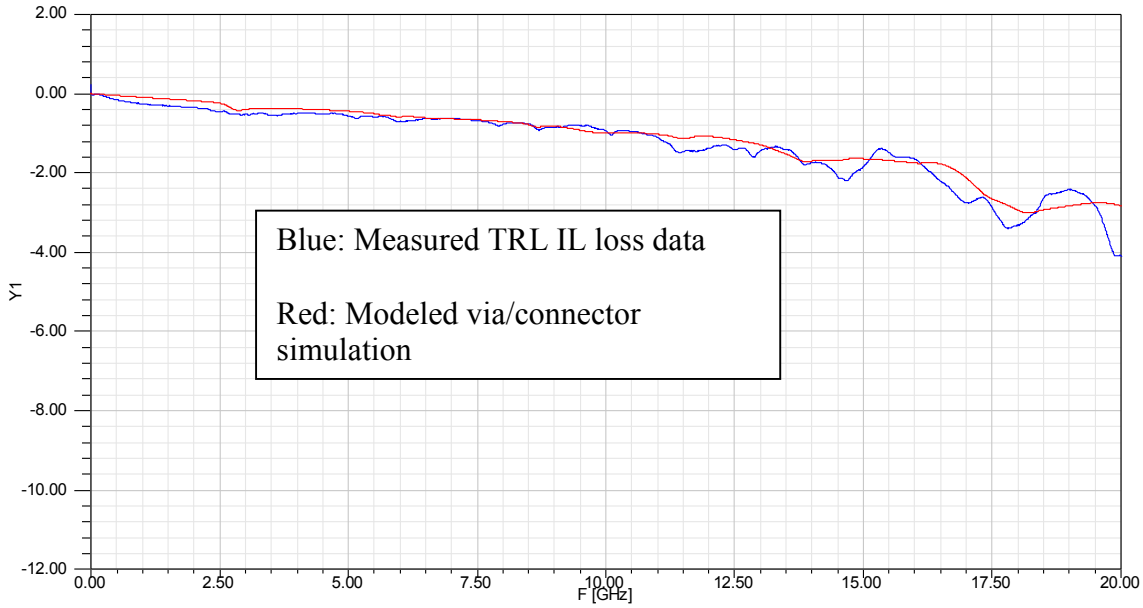


Figure 37: DUT Model vs TRL measured test data

Appendix

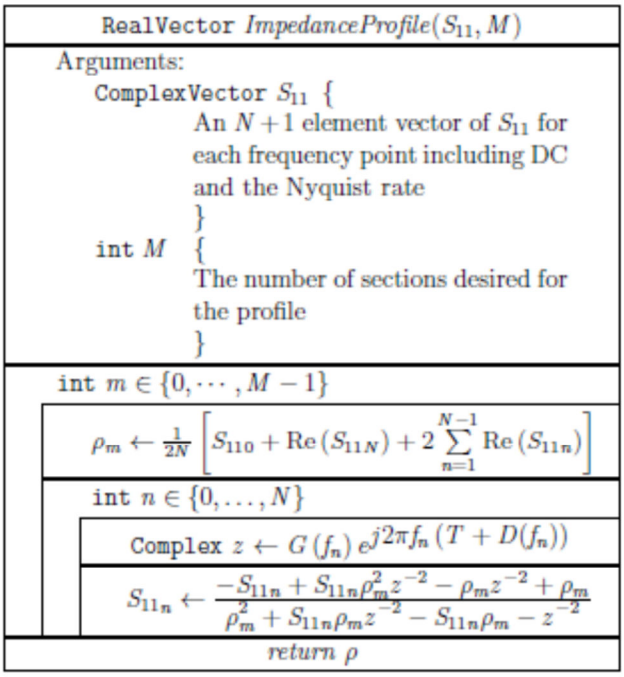


Figure 38: An algorithm for computing the impedance profile given s-parameters of the return loss at a given port and the number of desired impedance profile sections along with a loss and group-delay characteristic for each section with frequency

References

Hsue, C. W., & Pan, T. W. (1997). Reconstruction of Nonuniform Transmission Lines from Time-Domain Reflectometry. *IEEE Transactions on Microwave Theory and Techniques*, 32-38.

LeCroy-SPARQ. (2012, January). Retrieved from http://cdn.lecroy.com/files/appnotes/using_2nd-tier_calibration_for_cable_test_fixture_de-embedding_rev2.pdf

DeGroot, D. C., Jargon, J. A., & Marks, R. B. (2002). Multiline TRL revealed. *ARFTG Conf Digest*, (pp. 131-155).

DeGroot, D. C., Rolain, Y., Pintelon, R., & Schoukens, J. (2004). Corrections for nolinear vector network analyzer measurements using a stochastic multi-line/reflect method. *IEEE MTT-S Int. Microwave Symp. Digest*, (pp. 1735-1738).

Engen, G. F., & Hoer, C. A. (1979). 'Thru-reflect-line': An improved technique for calibrating the dual six-port automatic network analyzer. *IEEE Trans. Microwave Theory Tech.*, 987-993.

Hayden, L. A., & Tripathi, V. K. (1994). Characterization and modeling of multiple line interconnections from TDR measurements. *IEEE Transactions on Microwave Theory and Techniques,*, 1737-1743.

Jargon, J. A., & Marks, R. B. (1995). Two-Tier Multiline TRL for Calibration of Low-cost network analyzers. *ARFTG Conf. Dig.*, (pp. 1-8).

Marks, R. B. (1991). A Multiline method of network analyzer calibration. *IEEE Transactions on Microwave Theory and Techniques*, 1205-1215.

Meyer, C. D. (2004). *Matrix Analysis and Applied Linear Algebra.* SIAM: Society for Industrial and Applied Mathematics.

Smolyansky, D. A., & Corey, S. D. (1999, May). Printed Circuit Board Interconnect Characterization from TDR Measurements. *Printed Circuit Design Magazine*, pp. 18-26.

TDR Primer. (n.d.). *TDA Systems Application Note TDRP-0402*.

Wittwer, P., & Pupalaikis, P. J. (2008). A General Closed-Form Solution to Multi-Port Scattering Parameter Calculations. *72nd ARFTG.*