Extracting mixed mode impedance measurements

Bill Slade
20-10-2009
Table of Contents

References..................................................................................................................................................2
Purpose....................................................................................................................................................3
Background of method: the theoretical tools we need to measure device impedances.........................6
  Review of classical single line method (2-port error-boxes at reference impedance Zr).........................6
  What happens when the "through" line does match the reference characteristic impedance Zr?............8
  Four port networks, dual-coupled lines and the concept of mixed mode scattering parameters..........9
  The normal modes of dual symmetric microstrip..............................................................................11
  Estimation of “line” capacitance using "low frequency" measurement.............................................12
    Single line........................................................................................................................................12
    Dual coupled microstrip line............................................................................................................14
  How to extract true “line” characteristic impedance Zr.................................................................15
  Cascading line sections using multimode transmission parameters...............................................16
    Example: extracting parameters of "bent" line ..............................................................................17
  Finally! How do we do the de-embedding?.......................................................................................18
Procedure for extracting device-under-test impedance with examples..............................................20
  Summary of DUT parameter extraction algorithm.........................................................................20
  Characterisation of symmetric straight structure...........................................................................20
  Results of example measurement.....................................................................................................21
  Computed "line" effective permittivity and loss..................................................................................22
  The "bent line" error-box parameters...............................................................................................27
  Extraction of termination impedance 1: Even/Odd=47 ohm...........................................................28
  Extraction of termination impedance 1: Even=54 ohm, Odd=short...............................................30
Conclusions............................................................................................................................................33
Appendix A: error-box Octave extraction code..................................................................................33
  Main routine.......................................................................................................................................33
  Parameter extraction subroutine.......................................................................................................42
  Conversion from normal to mixed mode S parameters....................................................................43
  FSOLVE residual generation..............................................................................................................43
  Convert 4-port S parameters to T parameters.................................................................................44
  Convert T parameters to S parameters.............................................................................................44
  DUT De-embedding routine..............................................................................................................45

References


**Purpose**

Given the proliferation of RF integrated circuits that use differential inputs and outputs as a means to decouple common-mode noise and interference, a means of characterising the impedance and noise characteristics of these differential (multi-mode) inputs and outputs is needed. This implies that some form of mixed mode extraction of an impedance matrix is needed. Hence, we need to characterise the transmission line network from a board-mounted connector to the pins of the device we wish to test. Illustration 1 shows a set of four such structures.

One way to characterise these lines is to connect the chip-side open ends to a (soon to be) known "straight-through" structure like this:
and doing an even-odd mode analysis and measurement of the full structure.

This type of measurement is simplified by the fact that the straight through structure exhibits even-odd symmetry. The extra sections of dual coupled microstrip possess used in the "line" a natural mode structure that reflects the even-odd geometry of the transmission lines that we wish to preserve. After the straight-through lines are characterised, we can cascade the characterised lines with the "bent" RF input structure for an extra measurement and extract the parameters of the "bent" structure.

A further advantage is that the straight-through lines can be used to characterise more than one "bent" structure, if needed. This reduces the complexity of the characterisation if multiple input structures are needed.

The straight structure properties are established using a special multi-mode "through-reflect-line" (TRL) parameter extraction algorithm. As in the single line TRL, an "Open" structure is measured with the network analyser (Illustration 3 ).
The "Through" structure consists of the "Open" structure and its mirror image connected as shown in Illustration 4.

Illustration 3: Open "straight" structure to be measured as part of TRL parameter extraction.

Illustration 4: Through structure.
In order to uniquely characterise the straight structure, a set of lines slightly longer than the "Through" lines needs to be measured.

Illustration 5: "Line" structure with an extra 18mm of coupled microstrip.

**Background of method: the theoretical tools we need to measure device impedances**

**Review of classical single line method (2-port error-boxes at reference impedance Z)***

As seen in Illustration 6, the goal of this measurement is to extract the S parameters of the device under test (DUT) by measuring the S parameters of the whole structure at ports 1 and 2. This means that we need to somehow know the characteristics of the "boxes" that lie between the ports of the DUT and the ports that connect to the "outside world" (Error Box 1 and 2).

Illustration 6: Illustration of network analyser measurement. We want to extract the DUT parameters by measuring Ports 1 and 2.

One way of doing this is by removing the device under test and connecting the error boxes together in several configurations using reflective loads and lengths of transmission line. An important assumption used here is that the error boxes 1 and 2 are identical and connected back-to-back. This is the heart of the "through-reflect-line" (TRL) de-embedding scheme [1].
In Illustration 7, three measurements are carried out which allows us to characterise error box 1 and 2. The following equations relate the error box scattering parameters $E_{ij}$ to the various measurements.

**Through:**

$$\Gamma_{\text{thru}} = E_{11} + \frac{E_{21}^2 E_{22}}{1 - E_{22}^2}$$

$$G_{\text{thru}} = \frac{E_{21}^2}{1 - E_{22}^2}.$$  

Reflection measured at the port by the network analyser is designated by $\Gamma$ and transmission by $G$.

**Reflect:**

$$\Gamma_{\text{ref}} = E_{11} + \frac{E_{21}^2 \Gamma_L}{1 - E_{22} \Gamma_L},$$

where $\Gamma_L$ is the unknown reflection at the reflective load (usually an open).

**Line:**

$$\Gamma_{\text{line}} = E_{11} + E_{21}^2 \frac{L_{11} L_{22} L_{12}^2 + L_{12}^2 E_{22}}{(1 - E_{22} L_{11})^2 - E_{22}^2 L_{12}^2}$$

$$G_{\text{line}} = \frac{E_{21} L_{21}}{(1 - E_{22} L_{11})^2 - E_{22}^2 L_{12}^2}.$$
where $L_{11}, L_{21}$ are the scattering parameters of the "line" section of microstrip (looking ahead to where "line" characteristic impedance is not equal to $Z_r$). If this extra length of transmission line has a characteristic impedance equivalent to the system impedance $Z_r$ (usually 50 ohms), then

$$L_{11} = 0$$

$$L_{12} = e^{-\gamma \Delta l}$$

Therefore the "line" measurement equations reduce to the classic TRL "line" measurement:

$$\Gamma_{\text{line}} = \frac{E_{11}^2 + E_{21}^2 + 2E_{21}E_{12} e^{-2\gamma \Delta l}}{1 - E_{22} e^{-2\gamma \Delta l}}$$

$$G_{\text{line}} = \frac{E_{21}^2 e^{-\gamma \Delta l}}{1 - E_{22} e^{-2\gamma \Delta l}}$$

where the complex propagation factor of the extra length of transmission line $\gamma$ is another unknown that we solve for based on the measurement data. We now have five measurements, five equations and five unknowns ($E_{11}, E_{21}, E_{22}, \Gamma_L$, and $\gamma$), hence we can uniquely determine the characteristics of the error boxes. Keep in mind, however, that this final analysis based on the assumption of lines with characteristic impedance $Z_r$.

What happens when the "through" line does match the reference characteristic impedance $Z_r$?

If the "inward-facing" ports of the error boxes are terminated with lengths of transmission line that are not equal to $Z_r$ and we wish to continue to reference all scattering parameters to $Z_r$, then we need to account for the extra reflection caused by the length of mismatched line (because $L_{11}$ in the previous analysis is no longer zero). This is useful, because generally we do not know explicitly what the impedance of the line is, especially if we consider coupled lines where even and odd modes are present (to be covered later). Therefore, we need to solve for it based on measurement data.

If the new "line" line has unknown characteristic impedance $Z_1$, this introduces the need for another measurement or simulation to estimate the "line" characteristic impedance. This is a result of the now full set of scattering parameters for the length of unmatched transmission line, viz.:
\[ L_{11} = L_{22} = \frac{(\bar{Z}^2 - 1) \sinh (\gamma \Delta l)}{D} \]

\[ L_{21} = \frac{2}{D} \]

\[ D = 2 \cosh (\gamma \Delta l) + (\bar{Z}^2 + 1) \sinh (\gamma \Delta l) \]

\[ \bar{Z} = Z_{r}/Z_r \]

It can be clearly seen that if \( Z_1 = Z_0 \) (i.e. \( \bar{Z} = 1 \)), that the transmission line scattering parameters reduce to

\[ L_{11} = 0 \]

\[ L_{21} = e^{-\gamma \Delta l} \]

as given in the previous section.

**Four port networks, dual-coupled lines and the concept of mixed mode scattering parameters**

Scattering parameters are useful because they can be generalised to any number of ports that a linear network might have. Since the goal of this exercise is to extract impedance information for differential and common mode measurements, we need to establish what these modes are in terms of the usual S parameters measured by a network analyser.

It also so happens that the differential (odd) and common (even) modes are the natural modes of the symmetric coupled microstrip and striplines. This means that the even and odd modes are completely decoupled from each other and can be treated as independent “transmission lines” in the TRL parameter extraction algorithm.

Illustration 8 shows a block diagram of a 4 port network with incoming wave amplitudes designated by \( a_i \) and outgoing wave amplitudes as \( b_i \).

*Illustration 8: 4-port network showing incoming and outgoing wave amplitudes on each port.*
The incoming and outgoing wave amplitudes are related by the scattering (S) matrix

\[
\begin{bmatrix}
  b_1 \\
  b_2 \\
  b_3 \\
  b_4 
\end{bmatrix} =
\begin{bmatrix}
  S_{11} & S_{12} & S_{13} & S_{14} \\
  S_{22} & S_{23} & S_{24} & \\
  \vdots & S_{33} & S_{34} & \\
  \vdots & \vdots & \vdots & S_{44}
\end{bmatrix}
\begin{bmatrix}
  a_1 \\
  a_2 \\
  a_3 \\
  a_4 
\end{bmatrix},
\]

or

\[
b = S \cdot a.
\]

The S matrix of a reciprocal network is symmetric and has 10 unique parameters. Geometric symmetries will reduce the number of unique parameters still further. For example, as drawn in Illustration 8, even-odd symmetry will mean that

\[
S_{11} = S_{22} = S_{33} = S_{44} = S_{14} = S_{23}.
\]

This means that we need at least 6 independent measurements to properly characterise the network. (We will show later that we need 10 measurements using the TRL methodology to characterise the network and an external arbitrary load.)

Since we can use any linear combination of incoming and outgoing waves to model the transfer of power through the 4 port network, let us construct weighted sums and differences of the incoming and outgoing waves (depicted in Illustration 9).

Illustration 9: 4 port network showing linear combination of waves to construct even and odd (mixed) modes.

The new wave variables are [2]
The factor of \( \frac{1}{\sqrt{2}} \) is a normalisation factor to make modal power invariant as we transform from normal wave amplitudes to even-odd modal wave amplitudes. This implies the following linear transformation of variables:

\[
\begin{bmatrix}
a_{1e} \\
a_{1o} \\
a_{2e} \\
a_{2o}
\end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix}
1 & 1 & 0 & 0 \\
-1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 \\
0 & 0 & -1 & 1
\end{bmatrix} \begin{bmatrix}
a_1 \\
a_2 \\
a_3 \\
a_4
\end{bmatrix}.
\]

This means that the usual port 1-2-3-4 scattering parameters are related to the even-odd mixed-mode S-parameters by

\[
[S_{\text{mixed}}] = [R][S][R]^T,
\]

where

\[
[R] = \frac{1}{\sqrt{2}} \begin{bmatrix}
1 & 1 & 0 & 0 \\
-1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 \\
0 & 0 & -1 & 1
\end{bmatrix}.
\]

**The normal modes of dual symmetric microstrip**

The transverse cross-section geometry of the dual symmetric coupled microstrip is seen in Illustration 10.
This structure exhibits symmetry about the center line. This means that the EM field solutions (and hence, strip currents/voltages) can be decomposed into even and odd functions about the center line. There is a large literature on this type of structure. It suffices to say here that we rely on simulations [3],[4] to estimate the propagation characteristics of this structure needed as initial guesses in the solution numerical method.

It should be remembered that:

- The even and odd mode characteristic impedances will not be equal. It can be shown that $Z_e > Z_o$.
- Since there is inhomogeneous dielectric, the propagation speeds of even and odd modes will not be equal. Generally, $\varepsilon_{re} > \varepsilon_{ro}$.
- As long as the cross-section dimensions $w, h \ll \lambda/\sqrt{\varepsilon_r}$, there is little modal dispersion and the wave power remains well confined to the strip region (radiation effects are small and so-called quasi-static behaviour is a valid assumption).
- If $t$ becomes appreciable, then significant deviations from ideal behaviour can be observed. (Most simulators assume infinitesimal metal thickness.) 1 oz copper (0.035mm thick) on a 12 mil substrate (0.304mm) with $s=0.25mm$ might exhibit lower impedances as a likely result of this effect.

**Estimation of “line” capacitance using "low frequency" measurement**

**Single line**

This measurement should be carried out using a network analyser at a “low” frequency, that is, at a
frequency whose wavelength is much larger than the electrical dimensions of the structure being tested. As a general rule, the dimensions of the structure should be 1/50 to 1/100 of the wavelength. For example, a structure with an equivalent electrical length of 10cm should be measured at about 30-60 MHz (where the wavelength is 5-10 meters). This should yield a good estimate of the line's static capacitance per unit length (i.e. extra line lengths in Illustration 11 look like an extra shunt capacitance). With this in mind, the reflection coefficient measured by the network analyser is related to the “thru” and “line” capacitances by

\[
Z_{\text{thru}} = \frac{1 + \Gamma_{\text{thru}}}{1 - \Gamma_{\text{thru}}} Z_r = j \omega C_{\text{thru}}
\]

\[
Z_{\text{line}} = \frac{1 + \Gamma_{\text{line}}}{1 - \Gamma_{\text{line}}} Z_r = j \omega C_{\text{line}}
\]

The capacitance of the extra piece of line is

\[ C_1 \Delta l = C_{\text{line}} - C_{\text{thru}} \]

Given \( \gamma \), we compute the complex characteristic impedance of the line to be

\[ Z_1 = \frac{\gamma \Delta l}{j \omega \left( C_{\text{line}} - C_{\text{thru}} \right)} \]

If line losses are small, the real part of \( Z_1 \) will be much larger than the imaginary part. Generally, we will assume \( Z_1 \) to be real, while \( \gamma \) will have a small real part that accounts for the loss in the transmission line lengths (corresponding to a small imaginary component of characteristic impedance,

\text{Illustration 11: Two measurements needed for evaluating line capacitance.}
Dual coupled microstrip line

For the dual coupled microstrip, there are two capacitances that we need to find: strip-to-ground capacitance and the strip-to-strip capacitance. By exciting an even mode, the effect of inter-strip coupling capacitance is eliminated and the strip-to-ground capacitance $C_e$ can be evaluated. Exciting an odd mode allows the determination of $C_c$ (see Illustration 12).

These line capacitances can be extracted for using a low-frequency network analyser measurement of two structures as shown in Illustration 13. The extra length of line appears as extra capacitances that are proportional to the length of the lines.

Illustration 12: Equivalent capacitances for coupled microstrip for even and odd modes.

Illustration 13: Illustration of dual line measurement for extracting capacitances.
By using the even-odd S parameter transformations explained earlier, we find that the even and odd reflection coefficients in terms of the network-analyser measured S parameters are

\[
\Gamma_e = \frac{1}{2} \left( S_{11} + 2S_{21} + S_{22} \right)
\]

\[
\Gamma_o = \frac{1}{2} \left( S_{11} - 2S_{21} + S_{22} \right)
\]

Ideally, since the lines are fully symmetric \( S_{11} = S_{22} \). As a result, scattering between even and odd modes

\[ G_{eo} = S_{22} - S_{11} \]

is ignored. The line capacitance for the section of length difference is straightforward to extract.

The input impedances for the long and short lines at low frequency are dominated by line capacitance, so for the even mode we have

\[
Z_r \left( \frac{1 + \Gamma_{e,\text{long}}}{1 - \Gamma_{e,\text{long}}} \right) = j \omega C_e
\]

and for the odd mode

\[
Z_r \left( \frac{1 + \Gamma_{o,\text{long}}}{1 - \Gamma_{o,\text{long}}} \right) = j \omega \left( C_e + 2C_c \right)
\]

\( Z_r \) is the system reference impedance (usually 50 ohms). Practical measurements will not yield exactly imaginary feedpoint impedances as a result of line losses and measurement imperfection. However, the reactive component will be dominant and should be used to estimate the values for \( C_e \) and \( C_c \), whereas any small real part of feedpoint impedance can be ignored.

**How to extract true “line” characteristic impedance \( Z_r \)**

Solving the system of equations given in the previous sections requires a reasonable initial guess for the propagation factor \( \gamma \) and the “line” characteristic impedance \( Z_r \). Fortunately, (as will be discussed in more detail in the coming sections) the calculation of \( \gamma \) is insensitive to the choice of \( Z_r \). By using estimates for the propagation factor and characteristic impedance from simulation as initial guesses, the solution will refine these values based on the measurement data. The solution can be re-run using these refined values in order to generate accurate values for the error-box parameters (which are very sensitive to \( \gamma \) and \( Z_r \)).

Following [5], it is possible to compute an improved estimate for \( Z_r \) based on the propagation factor and the capacitance per unit length of the “line” microstrip \( C_1 \). If the (complex) characteristic impedance for a general lossy transmission line is given as

\[
Z_1 = \sqrt{\frac{j \omega L_1 + R_1}{j \omega C_1 + G_1}}
\]

and the propagation factor is known to be

\[
\gamma = \sqrt{(j \omega L_1 + R_1)(j \omega C_1 + G_1)}
\]
the characteristic impedance can be computed from the propagation factor, viz.
\[ Z_1 = \frac{Y}{j \omega C_1 + G_1}. \]

The trick lies in assuming that the shunt conductance is negligible with respect to the capacitive susceptance (i.e. dielectric losses are small; \( G_1 \ll \omega C_1 \)). Losses are assumed to be dominated by copper losses in the conductors. This approximation generally breaks down at low frequencies or in cases where the dielectric losses are significant. Generally extrapolation can be used to generate low-frequency approximations if they are needed. Hence, if we have an estimation of \( C_1 \) available, the line characteristic impedance is
\[ Z_1 \approx \frac{Y}{j \omega C_1}. \]

A further important result is that this also works for even and odd modes of dual coupled microstrip. For the even mode
\[ Z_e \approx \frac{Y_e}{j \omega C_{1e}} \]
and for the odd mode
\[ Z_o \approx \frac{Y_o}{j \omega C_{1o}}, \]
where
\[ C_{1e} = \frac{C_e}{\Delta l}, \]
\[ C_{1o} = \frac{(C_e + 2C_c)}{\Delta l} \]
from the analysis in the previous section.

**Cascading line sections using multimode transmission parameters**

If we attach two 4 port networks as in Illustration 14, working in terms of \( S \) parameter matrices is algebraically cumbersome. It is much more convenient to work in terms of "wave transmission" matrices [6]. These matrices are very similar in concept to the two-port ABCD parameter matrices from circuit theory [1]. In reality, they are derived from nothing more than a rearrangement of the wave amplitudes so that the left-hand port wave amplitudes are given in terms of right-hand side port amplitudes. This allows the full system transmission to be described as the simple matrix multiplication of the constituent blocks.

Illustration 14: Two 4-port networks concatenated with ports 3 and 4 of the first connected to ports 1 and 3 of the second.
Recall the definition of the scattering matrix, partitioned to highlight the left and right side wave variables:

\[
\begin{aligned}
\begin{pmatrix}
 b_1 \\
 b_2 \\
 \vdots \\
 b_3 \\
 b_4 
\end{pmatrix} &=
\begin{bmatrix}
 S_{11} & S_{12} \\
 S_{21} & S_{22} \\
 S_{31} & S_{32} \\
 S_{41} & S_{42} 
\end{bmatrix}
\begin{pmatrix}
 a_1 \\
 a_2 \\
 \vdots \\
 a_3 \\
 a_4 
\end{pmatrix},
\end{aligned}
\]

or more compactly:

\[
\begin{aligned}
\begin{pmatrix}
 b_l \\
 b_r 
\end{pmatrix} &=
\begin{bmatrix}
 S_{ll} & S_{lr} \\
 S_{rl} & S_{rr} 
\end{bmatrix}
\begin{pmatrix}
 a_l \\
 a_r 
\end{pmatrix}.
\end{aligned}
\]

Grouping right and left ports yields the mixed mode transmission matrix

\[
\begin{aligned}
\begin{pmatrix}
 b_l \\
 a_l 
\end{pmatrix} &=
\begin{bmatrix}
 T_{11} & T_{12} \\
 T_{21} & T_{22} 
\end{bmatrix}
\begin{pmatrix}
 a_r \\
 b_r 
\end{pmatrix}.
\end{aligned}
\]

Some basic matrix algebra yields the relationship between the S parameter and wave transmission matrices in terms of the 2x2 block partitions of the 4 port S matrix:

\[
\begin{bmatrix}
 T_{11} & T_{12} \\
 T_{21} & T_{22} 
\end{bmatrix} =
\begin{bmatrix}
 [S]_{ll} - [S]_{lr} [S]_{rl}^{-1} [S]_{rr} & [S]_{ll} [S]_{rl}^{-1} \\
 -[S]_{rl}^{-1} [S]_{rr} & S_{rl}^{-1} 
\end{bmatrix}.
\]

By recognising that outgoing waves on the right hand side correspond to incoming waves on the left, a system can be fully characterised using a simple concatenation rule for transmission parameters:

\[
\begin{aligned}
\begin{pmatrix}
 b_l \\
 a_l 
\end{pmatrix} &=
\begin{bmatrix}
 T_{11} & T_{12} \\
 T_{21} & T_{22} 
\end{bmatrix} \ldots \begin{bmatrix}
 T_{11} & T_{12} \\
 T_{21} & T_{22} 
\end{bmatrix}
\begin{pmatrix}
 a_r \\
 b_r 
\end{pmatrix}.
\end{aligned}
\]

This "forward" treatment can be reversed by inverting and permuting the T matrix,

\[
\begin{aligned}
\begin{pmatrix}
 b_r \\
 a_r 
\end{pmatrix} &=
\begin{bmatrix}
 0 & [I] \\
 [I] & 0 
\end{bmatrix}
\begin{bmatrix}
 T_{11} & T_{12} \\
 T_{21} & T_{22} 
\end{bmatrix}^{-1}
\begin{bmatrix}
 0 & [I] \\
 [I] & 0 
\end{bmatrix}
\begin{pmatrix}
 a_l \\
 b_l 
\end{pmatrix},
\end{aligned}
\]

where \([I]\) is the 2x2 identity matrix.

**Example: extracting parameters of "bent" line**

If we measure the parameters of the "bent" line/straight-thru line structure (see Illustration 2, repeated below for convenience), we get, after some computation, the transmission parameters of the complete test structure.
\[ [T]_{\text{struct}} = [T]_{\text{straight}} [T]_{\text{bent}} \]

The "bent" line transmission parameters are extracted by doing a simple matrix inverse and multiplication to yield:

\[ [T]_{\text{straight}}^{-1} [T]_{\text{struct}} = [T]_{\text{bent}} \]

Since we already know the "straight" line parameters, we can immediately characterise the "bent" line in terms of what we measure and what we already know! **We can now extract the parameters of anything connected to this "bent line" structure.**

**Finally! How do we do the de-embedding?**

If we connect an arbitrary unmatched mixed-mode load to the input structure (as in Illustration 15), we observe the following relationship between incoming and outgoing wave amplitudes using the network analyser:

\[
\begin{bmatrix}
  b_1 \\
  b_2 \\
  b_3 \\
  b_4
\end{bmatrix} =
\begin{bmatrix}
  [S]_{11} & [S]_{12} \\
  [S]_{21} & [S]_{22}
\end{bmatrix}
\begin{bmatrix}
  a_1 \\
  a_2 \\
  \Gamma_3 b_3 + G_{34} b_4 \\
  \Gamma_4 b_4 + G_{43} b_3
\end{bmatrix}
\]

where \( \Gamma_3, \Gamma_4 \) correspond to the return losses (reflection) from the DUT and \( G_{34}, G_{43} \) are cross-coupling terms for the DUT ports. Note that the S parameters can be either mixed mode or normal port-based variety.
In terms of the full set of measurements on ports 1 and 2, we can write the compact expressions

\[
\begin{align*}
\{ b_l \} &= [ S ]_{ll} \{ a_l \} + [ S ]_{lr} [ \Gamma ] \{ b_r \}, \\
\{ b_r \} &= [ S ]_{rl} \{ a_l \} + [ S ]_{rr} [ \Gamma ] \{ b_r \},
\end{align*}
\]

where \([ \Gamma ]\) is the port reflection matrix:

\[
[ \Gamma ] = \begin{bmatrix} \Gamma_3 & G_{34} \\ G_{43} & \Gamma_4 \end{bmatrix},
\]

and the indices "l" and "r" denote left and right ports. After some rearrangement, we can show that

\[
[S]_{lr}^{-1} [ B ] - [ S ]_{ll} [ S ]_{rl}^{-1} = [ \Gamma ] ([ I ] - [ S ]_{rr} [ \Gamma ]^{-1}].
\]

\([ I ]\) is the identity matrix, \([ B ]\) is the measured scattering parameters at the connector inputs. Let

\[
X = [ S ]_{lr}^{-1} [ B ] - [ S ]_{ll} [ S ]_{rl}^{-1}
\]

and solve for \([ \Gamma ]\) yielding

\[
[ \Gamma ] = ([ I ] + [ X ] [ S ]_{rr}^{-1})^{-1} [ X ].
\]

We now have the reflection coefficients at the pin-side of the input network. It is a simple matter to convert to the impedance matrix using:

\[
[Z]_{DUT} = Z_r ([ I ] + [ \Gamma ]) ([ I ] - [ \Gamma ])^{-1},
\]

where \( Z_r \) is the system impedance (50 ohms).

This completes the impedance extraction process!
Procedure for extracting device-under-test impedance with examples

Summary of DUT parameter extraction algorithm

1. Set up and calibrate 4 port network analyser. If only a two-port network analyser is available, you will need two 50 ohm loads that are good to the highest frequency you wish to measure. Ports are measured in pairs with loads attached to the non measured ports. See [] for more details on this.
2. Measure two-line “open”, “thru” and “line” standards, saving the scattering parameters (better yet, do several identical measurements on several boards and generate statistical averages and standard deviations of measured S parameters).
3. Convert network analyser generated S parameters to even-odd parameters as described in section 1.4 for the three standards.
4. Estimate $Z_o, Z_e, \gamma_o, \gamma_e$ using simulation of coupled strip geometry used in “line” measurement.
5. Extract even (common) mode $[E_e, \Gamma_{open_e}, \gamma_e]$ from measured reflection and transmission coefficients using estimates for impedance and propagation factors.
6. Extract odd (differential) mode $[E_o, \Gamma_{open_o}, \gamma_o]$ from measured reflection and transmission coefficients.
7. Extract even and odd mode line capacitances and generate improved approximation for even and odd mode characteristic impedances.
8. Redo steps 5 and 6 to improve error-box estimates $[E_e, E_o]$.
9. Convert even-odd error-box S-parameter data to even-odd transmission (T) parameters. Propagation factors also allow impedance transformation of (almost) arbitrary straight lengths of coupled line.
10. Connect "bent" test structure to straight structure and measure the resulting 4 port S parameters. Convert this to 4-port T-parameter matrix.
11. Do matrix inverse and multiplication to extract "bent" line T-matrix.
12. Connect "bent" line to device-under-test (DUT) ports.
13. Measure resulting 2-port S parameters at "bent" line inputs using network analyser.
14. Extract DUT reflection matrix using "bent" line T-parameters (either in normal or even-odd form).
15. Compute DUT impedance matrix (even/odd or normal two-port).

Characterisation of symmetric straight structure

The numerical method used to extract the "error box" parameters is achieved by rearranging the equations in Section 2.1 to reduce the number of equations that need to be solved as a system. Briefly, in terms of the measured quantities $\Gamma_{ref}, \Gamma_{thru}, \Gamma_{line}, G_{thru}$ and $G_{line}$, we can isolate $E_{21}, E_{11}, \Gamma_L$ from $E_{22}, \gamma$:
\[
E_{21}^2 = G_{thru} \left( 1 - E_{22}^2 \right)
\]
\[
E_{11} = \Gamma_{thru} - G_{thru} E_{22}
\]
\[
\Gamma_L = \frac{\Gamma_{refl} - E_{11}}{E_{21}^2 + \left( \Gamma_{refl} - E_{11} \right) E_{22}}.
\]

This means that we need to solve a system of two non-linear system for complex values of \( E_{22} \), \( y \):

\[
G_{line} \left( \left( 1 - E_{22} L_{11} (y) \right)^2 - E_{22}^2 L_{21}^2 (y) \right) - G_{thru} \left( 1 - E_{22}^2 \right) L_{21} (y) = 0
\]
\[
\left( \Gamma_{line} - \Gamma_{thru} + G_{thru} E_{22} \right) \left( \left( 1 - E_{22} L_{11} (y) \right)^2 - E_{22}^2 L_{21}^2 (y) \right) = 0
\]
\[
G_{thru} \left( 1 - E_{22}^2 \right) \left( L_{11} (y) - E_{22} L_{11}^2 (y) + L_{12}^2 (y) \right)
\]

The "line" scattering parameters are those from Section 2.2. Values for \( E_{22} \), \( y \) are substituted back into the first three equations to give the full error-box characterisation. The measurement of the open load can be used to establish the proper sign for \( E_{21} \) and \( y \). Note that this is done twice: once for the even modes and once for the odd. Also note that the indices 1 and 2 now denote "left" and "right" ports "l" and "r" in Section 2.8.

The extraction code is given in Appendix A.

### Results of example measurement

A test measurement was carried out using 0.25mm lines separated by 0.25mm on 12 mil (0.305mm) FR4. Copper layer is 1 oz copper (35um) coated with mask lacquer. The structures are those shown in the introduction of this report. UFL "snap on" connectors are used. The "line" section is 1.8cm (giving a measurement bandwidth of about 4.5GHz for the slower even mode).

A quasi-static microstrip simulation model [4] is used to estimate modal effective dielectric permittivity and characteristic impedance. Table 1 shows these estimated parameters and improved parameters computed using this extraction method.
Table 1: Useful results obtained from simulation and measurement of dual coupled microstrip.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Estimated (simulation)</th>
<th>Measurement results</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon_{re}$ even effective permittivity</td>
<td>3.6</td>
<td>3.5</td>
<td>Average computed well away from LF/HF limits</td>
</tr>
<tr>
<td>$\varepsilon_{ro}$ odd effective permittivity</td>
<td>3.0</td>
<td>3.0</td>
<td></td>
</tr>
<tr>
<td>$C_{1e}$ even capacitance/length</td>
<td>--</td>
<td>0.75 pF/cm</td>
<td>See Sec. 2.5</td>
</tr>
<tr>
<td>$C_{1o}$ odd capacitance/length</td>
<td>--</td>
<td>1.05 pF/cm</td>
<td>$C_e=0.15$ pF/cm</td>
</tr>
<tr>
<td>$Z_e$ even char. impedance</td>
<td>87 ohm</td>
<td>83 ohm</td>
<td></td>
</tr>
<tr>
<td>$Z_o$ odd char. impedance</td>
<td>61 ohm</td>
<td>55 ohm</td>
<td>Good match to 50 ohm system!</td>
</tr>
</tbody>
</table>

Computed "line" effective permittivity and loss

The measured values given in Table 1 are an average from approximately 1GHz to 3GHz of the data presented in Illustration 16. This is because the measurement breaks down at low frequencies (<1GHz) and is unreliable near frequencies where the "line" length is approximately 1/2 wavelength long. We rely on the assumption that mode dispersion is negligible (the quasi-static approximation is valid and we expect the permittivity curve to be essentially "flat"). Illustration 16 shows how the permittivity takes nonphysical values as the frequency falls below 1GHz and 4.5 GHz (for the even mode) and 5GHz (for the odd mode) where the 1.8 cm "line" is nearly 1/2 wavelength long (which causes a phase ambiguity, and hence poorly posed system of equations).
To avoid confusion, the permittivity is computed from the extracted propagation factors $\gamma$ for the even and odd modes:

$$\varepsilon_{\text{eff}} = c_0^2 \left( \frac{\Im (\gamma)}{2\pi f} \right)^2 \quad \text{(dimensionless)},$$

where $c_0$ is the speed of light in a vacuum and $f$ is the frequency.

In order to generate the best approximations for the effective permittivity at lower frequencies, a longer "line" section can be used at the expense of reducing the upper frequency limit. A simpler method would be to use simple extrapolation to estimate the low-frequency $\varepsilon_{\text{eff}}$ since the quasi-static approximation is assumed to be valid (yielding a constant value for $\varepsilon_{\text{eff}}$ with respect to frequency in this range).

The loss, on the other hand, will be dependent on frequency. Illustration 17 shows the loss per unit length for the even and odd modes. These curves are somewhat irregular, so a straight-line fit should be used to generate estimations. In fact, despite the variations, the straight-line trend is readily observable in the figure below for both modes.

The loss is computed from

$$\alpha = 20 \frac{\Re (\gamma)}{\ln(10)} = 8.686 \Re (\gamma) \quad \text{dB/cm}.$$
Inspection of the curves indicates that the even mode is slightly more lossy than the odd mode. Extracted scattering parameters of the "Error Box" 

An important point has been reached. From the measurements we can now characterise the straight-through microstrip test structure. Illustration 18 shows the reflection coefficient magnitudes for the test structure (terminated in 50 ohms, the system impedance).

Illustration 17: Losses in line sections computed from measurement results.

An important sanity check can be done at this point. Since the network by definition is reciprocal, the magnitudes of $E_{11}$ and $E_{22}$ must be equivalent as a result of the relationship

$$E_{21} E_{11}^* + E_{22} E_{21}^* = E_{11} E_{21}^* + E_{21} E_{22}^* ,$$

which yields

$$\Im \left( E_{21} (E_{11}^* - E_{22}^*) \right) = 0 .$$

A necessary condition for this is the the magnitudes of the reflection coefficients be equal. Hence, the $E_{11}$ and $E_{22}$ magnitude curves should coincide for each mode. In the plot above, the
curves almost coincide, indicating that the measurement most likely represents a reasonable approximation of reality.

Keep in mind that the results are not likely to be valid above 4.5GHz because of the limitations on the data used in the measurement (the length of the "line" measurement becomes nearly 1/2 wavelength, destroying the conditioning of the equations for the TRL method). This is seen in the irregular parts of the curves near 4.5-5.0GHz (where the 1.8cm section of "line" is nearly 1/2 wavelength, causing poor conditioning of TRL equations).

The phases of the reflection coefficients are shown in Illustration 19.

![Illustration 19: Phase of reflection coefficients.](image1)

The transmission coefficient magnitudes of the structure is seen in Illustration 20.

![Illustration 20: Magnitude of transmission for even and odd modes.](image2)
Odd mode transmission is better because the dual microstrip odd mode impedance is close to the system impedance of 50 ohms. Furthermore, the instability of the extraction method is clearly seen for the two modes near 4.5 and 5 GHz.

The phase of the transmission coefficient is presented in Illustration 21.

![Illustration 21: Phase of the transmission coefficient computed from measurement results.](image)
From the power symmetry condition mentioned previously, a sufficiency condition would be
\[
\arg(E_{21}) = \arg(E_{11}) + \arg(E_{22}) + n \pi,
\]
where \( n \) is an integer that accounts for the effects of the branch-cut that represents a 180 degree ambiguity in the measurement used to generate \( E_{21} \).

In fact, it is easy to verify that this is satisfied well, adding another sanity check on the calculations. Also, the phase appears linear with frequency, as would be expected for a structure whose behaviour is dominated by a non (or weakly)-dispersive transmission line.

**The "bent line" error-box parameters**

Now that we have the properties of the straight lines, we can extract the mixed mode scattering parameters of the "bent line" structure using the procedure in Section 2.7.1. Illustration 22 shows the magnitudes of the reflection and cross-coupling between even and odd modes.

Illustration 22: Reflection and cross-coupling between modes for bent-line structure. The numeric subscripts refer to left=1, right=2 ports. (The "right" port is on the "chip-pad" side.)
The letter subscripts refer to e=even, o=odd modes.

Below 3.4GHz, the even-odd mode coupling is practically negligible (less than -20dB). Above 3.4GHz, scattering between even and odd modes at Port 1 (connector side) is never higher than -19dB. Practically, this is a consequence that care was taken to ensure equal line lengths in the bent structure. Cross-mode scattering at high frequencies is likely to originate from errors in connector placement and a small amount of scattering from abrupt non-symmetric discontinuities.

Transmission through this structure is very similar to the straight line structure.
The even mode transmission is slightly worse than the odd mode as a result of a larger mismatch between the even mode line characteristic impedance (83 ohms) and the system impedance (50 ohms) than for the odd mode (55 ohms).

The phase of the transmission between left and right ports is seen in Illustration 23.

Illustration 23: Transmission coefficient magnitudes for even and odd modes.

Illustration 24: Phase of port 1 to port 2 transmission.

**Extraction of termination impedance 1: Even/Odd=47 ohm**

Using the method in Section 3.2, an attempt is made to extract the parameters a 47 ohm termination with the configuration shown in Illustration 25.
The diagonal terms of the impedance matrix are plotted in Illustration 25: Bent structure terminated so both even and odd modes see 47 ohm resistance ($Z_{ee}$ and $Z_{oo}$ will be an almost perfect match to 50 ohm).

Illustration 25: Bent structure terminated so both even and odd modes see 47 ohm resistance ($Z_{ee}$ and $Z_{oo}$ will be an almost perfect match to 50 ohm).

Illustration 26: Extracted diagonal components of impedance matrix for twin 47 ohm load.

From 0 to 3.5GHz, the real part of the extracted impedance is near 47 ohm. There is some variation that is likely from resistor non idealties, systematic (e.g. connector, soldering) and calibration errors. It also appears that there is significant capacitive reactance above 1.5GHz. However, this is a result from a single measurement and should be verified by carrying out several measurements to establish the statistical behaviour of the solution, thereby yielding a measure of confidence. It is worth noting that the reactance as shown can be accounted for by an equivalent 0.2pF shunt capacitance; a "solder blob" or resistor parasitic capacitance would be enough to account for this effect. Above 3.0GHz, the termination begins to exhibit an inductive component.
The off-diagonal impedances matrix elements provide a measure of mode-conversion between even and odd modes. Illustration 27 shows these values.

![Illustration 27: Off-diagonal impedance matrix element $Z_{oe}$ showing level of mode conversion.](image)

Small length differences, asymmetries and errors are responsible for this result. From this result, we can assume that mode conversion effects are small (likely less than 1% power exchange since we observe that $|2Z_{eo}/(Z_{oo}+Z_{ee})| \leq 0.1$).

All in all, this result is encouraging, given that the measurement is carried out using non-precision UFL PCB connectors that are hand-soldered.

**Extraction of termination impedance 1: Even=54 ohm, Odd=short**

In this experiment, we try a situation where the even and odd modes experience different reflection coefficients. Illustration 28 shows the type of termination used to achieve a near match for the even mode and complete reflection for the odd mode.
On using the extraction algorithm on the data generated by measuring the two ports of this structure, we generate the results seen in Illustrations 29 and 30 below.

Indeed, we observe the real part of the even mode impedance of the load to be near 54 ohms up to nearly 3.5 GHz. Above 3GHz, we also observe the resistor becoming somewhat inductive. The odd mode exhibits a real impedance of nearly zero, as is expected for a short circuit. It is interesting that there is a slight capacitive impedance contribution that may be a result of placement errors of the "solder blob" used. We observe that if the short is placed a little bit before the true end of the test structure, it will add a little "negative inductance" as a result of transforming through about a millimeter of "negative" length of transmission line. This will appear as a small capacitive reactance.

In Illustration 30, we see that the even and odd modes are well isolated from each other in this case as
The off diagonal impedance component is very small compared to the diagonal. This means we can treat the modes independently at the termination, as there is very little coupling between them.

*Illustration 30: Coupling between even and odd modes.*
Conclusions

This method described in this paper represents one useful method of moving the impedance measurement reference plane from a set of connectors to the pins of a device under test and extracting common- and differential mode properties of a multi-terminal RF port. If a calibrated network analyser is available, the de-embedding requires no precision standards. Repeatability, though, is important for good results. Care should be taken to avoid measurement errors.

The measurement has been described almost completely, with the exception of a proper statistical analysis. The code that has been developed is capable of reading in multiple measurement data sets and computing some basic statistics and error estimates. This work has not been completed yet (transforming measurement data errors to errors in computed parameters is something that needs to be added, for example). Despite this, we can get a feel for the error by checking the consistency of computed and measured results.

Further work on these algorithms will possibly encompass:

1. Statistical analysis and error estimations
2. Use of multiple "line" measurements with least-square solver for eliminating poor conditioning (1/2 wavelength error)
3. Improve estimates of $Z_r$
4. Curve-fitting of results and confidence intervals based on numerous measurements.
5. Graphical interface and automatic data acquisition.

Appendix A: error-box Octave extraction code

Main routine

#This program loads a series of S param measurements and generates some useful statistics about them.

clear;

global StartVec;

Rev = [0, 0, 1, 0; 0, 0, 0, 1; 1, 0, 0, 0; 0, 1, 0, 0];
Branch_e(1000)=0;
Branch_o(1000)=0;

k0 = 0;
kr = 0;
NPoints = 1601;
Nfreqs = NPoints;

#epsr = 3.38; #RO4003 diel const
epsr = 4.6; #FR4 diel const
epse = 3.3; # guesses for even/odd diel effective permittivity.
epso = 2.8;
dL = 1.8; # Length of "line"
Ze = 84.; # estimate of Ze, Zo.
Zo = 55.0;

Ze = Ze / 50.; # Normalise impedances.
Zo = Zo / 50.;

EBox(3) = 0+i;
EOcouple_refl(Nfreqs) = 0;
EOcouple_line(Nfreqs) = 0;
EOcouple_thru(Nfreqs) = 0;
S(33*Nfreqs) = 0 + i;
Gamma_refl_even(Nfreqs) = 0+i;
Gamma_refl_odd(Nfreqs) = 0+i;
Gamma_thru_even(Nfreqs) = 0+i;
Gamma_thru_odd(Nfreqs) = 0+i;
Gamma_line_even(Nfreqs) = 0+i;
Gamma_line_odd(Nfreqs) = 0+i;
Gain_thru_even(Nfreqs) = 0+i;
Gain_thru_odd(Nfreqs) = 0+i;
Gain_line_even(Nfreqs) = 0+i;
Gain_line_odd(Nfreqs) = 0+i;
E1le(Nfreqs) = 0+i;
E2le(Nfreqs) = 0+i;
E22e(Nfreqs) = 0+i;
G_refl_e(Nfreqs) = 0+i;
Prop_e(Nfreqs) = 0+i;
E1lo(Nfreqs) = 0+i;
E2lo(Nfreqs) = 0+i;
E22o(Nfreqs) = 0+i;
G_refl_o(Nfreqs) = 0+i;
Prop_o(Nfreqs) = 0+i;
Tb(16*Nfreqs) = 0+i;

f(Nfreqs) = 0;

[Databasename, count] = scanf("%s", 1);
[Databasename1, count] = scanf("%s", 1);

Files = ["s11", "s21", "s22", "s31", "s41", "s32", "s42", "s33", "s43", "s44"];
Dirs = ["Open", "Thru", "06mm", "12mm", "18mm", "Bend"];
Boards = ["Bd1", "Bd2", "Bd3", "Bd4"];
NBds = 4;

for k = 1:6 # Measurement type
    Sav = zeros(1, 12*NPoints);
    S2av = zeros(1, 12*NPoints);
    for j = 1:NBds # Board sample
        n = 0;
        for l = 1:10 # S-param (S11, S12, ...)
            FileName = [Databasename,"/",Boards(3*j-2:3*j),"/",Dirs(4*k-3:4*k),"/",Files(3*l-2:3*l),".csv"];
            printf("%s\n", FileName);
fp_in = fopen(FileName, "rt");
txt = fgets(fp_in); # move beyond junk text at beginning of file.
txt = fgets(fp_in);
txt = fgets(fp_in);
for m = 1:NPoints
    [freq, mag, phs] = fscanf(fp_in, "%e, %e, %e", "C");
    Sav(m + n * NPoints) += 10^(mag / 20.) * exp(i * pi * phs / 180.) / NBds;
    S2av(m + n * NPoints) += 10^(mag / 10.) / NBds;
    f(m) = freq;
endfor
n++;
if((k ==1) && (l ==3)) # There are only 3 Open measurements.
    break;
endif
endfor
fclose(fp_in);
endfor

FileName = [Databasename,"_",Dirs(4*k-3:4*k),"_.dat"];
fp_out = fopen(FileName, "wt");
fprintf(fp_out, "# f S11 S21 S22 S31 S41 S32 S42 S33 S43 S44
# real  imag  std_dev
");
for l = 1:NPoints
    fprintf(fp_out, "%e ", f(l));
    for j = 1:10
        fprintf(fp_out, "%e %e %e ",
            real(Sav(l + (j-1) * NPoints)), imag(Sav(l + (j-1) * NPoints)),
            sqrt(S2av(l + (j-1) * NPoints) - abs(Sav(l + (j-1) * NPoints))^2));
        if((k == 1) && (j == 3))
            break;
        endif
    endfor
    fprintf(fp_out,"\n");
endfor

# Put S param data in analysis variables.
switch(k) # Measurement type
case 1 # Reflect
    S(l) = Sav(l);
    S(l + NPoints) = Sav(l + NPoints);
    S(l + 2 * NPoints) = Sav(l + 2 * NPoints);
case 2 # Thru
    S(l + 3 * NPoints) = Sav(l);
    S(l + 4 * NPoints) = Sav(l + NPoints);
    S(l + 5 * NPoints) = Sav(l + 2 * NPoints);
endswitch

\[ S(l + 11 \times \text{NPoints}) = \text{Sav}(l + 8 \times \text{NPoints}); \]
\[ S(l + 12 \times \text{NPoints}) = \text{Sav}(l + 9 \times \text{NPoints}); \]

\text{case 5} # 18mm line (to start with. Gives best results)
\[ S(l + 13 \times \text{NPoints}) = \text{Sav}(l); \]
\[ S(l + 14 \times \text{NPoints}) = \text{Sav}(l + \text{NPoints}); \]
\[ S(l + 15 \times \text{NPoints}) = \text{Sav}(l + 2 \times \text{NPoints}); \]
\[ S(l + 16 \times \text{NPoints}) = \text{Sav}(l + 3 \times \text{NPoints}); \]
\[ S(l + 17 \times \text{NPoints}) = \text{Sav}(l + 4 \times \text{NPoints}); \]
\[ S(l + 18 \times \text{NPoints}) = \text{Sav}(l + 5 \times \text{NPoints}); \]
\[ S(l + 19 \times \text{NPoints}) = \text{Sav}(l + 6 \times \text{NPoints}); \]
\[ S(l + 20 \times \text{NPoints}) = \text{Sav}(l + 7 \times \text{NPoints}); \]
\[ S(l + 21 \times \text{NPoints}) = \text{Sav}(l + 8 \times \text{NPoints}); \]
\[ S(l + 22 \times \text{NPoints}) = \text{Sav}(l + 9 \times \text{NPoints}); \]

\text{endswitch}

\text{endfor}
\text{fclose(fp_out);}
Meas1(2,2) = S(j+5*Nfreqs);
Meas1(2,3) = S(j+8*Nfreqs);
Meas1(2,4) = S(j+9*Nfreqs);
Meas1(3,1) = Meas1(1,3);
Meas1(3,2) = Meas1(2,3);
Meas1(3,3) = S(j+10*Nfreqs);
Meas1(3,4) = S(j+11*Nfreqs);
Meas1(4,1) = Meas1(1,4);
Meas1(4,2) = Meas1(2,4);
Meas1(4,3) = Meas1(3,4);
Meas1(4,4) = S(j+12*Nfreqs);

Mxd = S2Mixed(Meas1);
E0couple_thru(j) = \sqrt(0.25 * (abs(Mxd(1,2))^2 + abs(Mxd(1,4))^2 + abs(Mxd(2,3))^2 + abs(Mxd(3,4))^2)));
   Gamma_thru_even(j) = 0.5 * (Mxd(1,1) + Mxd(3,3));
   Gamma_thru_odd(j) = 0.5 * (Mxd(2,2) + Mxd(4,4));

Gain_thru_even(j) = Mxd(1,3);
Gain_thru_odd(j) = Mxd(2,4);
endfor

for j = 1:Nfreqs
    Meas1(1,1) = S(j+13*Nfreqs);
    Meas1(1,2) = S(j+14*Nfreqs);
    Meas1(1,3) = S(j+16*Nfreqs);
    Meas1(1,4) = S(j+17*Nfreqs);
    Meas1(2,1) = Meas1(1,2);
    Meas1(2,2) = S(j+15*Nfreqs);
    Meas1(2,3) = S(j+18*Nfreqs);
    Meas1(2,4) = S(j+19*Nfreqs);
    Meas1(3,1) = Meas1(1,3);
    Meas1(3,2) = Meas1(2,3);
    Meas1(3,3) = S(j+20*Nfreqs);
    Meas1(3,4) = S(j+21*Nfreqs);
    Meas1(4,1) = Meas1(1,4);
    Meas1(4,2) = Meas1(2,4);
    Meas1(4,3) = Meas1(3,4);
    Meas1(4,4) = S(j+22*Nfreqs);

Mxd = S2Mixed(Meas1);
E0couple_line(j) = \sqrt(0.25 * (abs(Mxd(1,2))^2 + abs(Mxd(1,4))^2 + abs(Mxd(2,3))^2 + abs(Mxd(3,4))^2)));
   Gamma_line_even(j) = 0.5 * (Mxd(1,1) + Mxd(3,3));
   Gamma_line_odd(j) = 0.5 * (Mxd(2,2) + Mxd(4,4));

Gain_line_even(j) = Mxd(1,3);
Gain_line_odd(j) = Mxd(2,4);
endfor

# Extract even parameters
for j = 1:Nfreqs
    k0 = 2 * pi * f(j) / 3.0e10;
kr = k0 * sqrt(epsr);
StartVec(1) = 0.1;
StartVec(2) = 0.1;
StartVec(3) = 0.;
StartVec(4) = sqrt(epse) * k0 * dL;

[EBox, G, prop] = param_extract(Gamma_thru_even(j), Gain_thru_even(j), \Gamma_refl_even(j), Gamma_line_even(j), Gain_line_even(j), Ze, dL, k0, kr);
E11e(j) = EBox(1);
E21e(j) = EBox(2);
E22e(j) = EBox(3);
G_refl_e(j) = G;
Prop_e(j) = prop;
endfor

# Extract odd parameters
for j = 1:Nfreqs
    k0 = 2 * pi * f(j) / 3.0e10;
    kr = k0 * sqrt(epsr);
    StartVec(1) = 0.1;
    StartVec(2) = 0.1;
    StartVec(3) = 0.;
    StartVec(4) = sqrt(epso) * k0 * dL;
    [EBox, G, prop] = param_extract(Gamma_thru_odd(j), Gain_thru_odd(j), \Gamma_refl_odd(j), Gamma_line_odd(j), Gain_line_odd(j), Zo, dL, k0, kr);
    E11o(j) = EBox(1);
    E21o(j) = EBox(2);
    E22o(j) = EBox(3);
    G_refl_o(j) = G;
    Prop_o(j) = prop;
endfor

# Find branch points to correct phase of E21e, E21o.
m = 1;
n = 1;
for j = 160:1601 # Start from 160 to avoid the mess at the start.
    if(abs(arg(E21e(j-1)) - arg(E21e(j))) > pi/4.0) #detected a jump.
        Branch_e(m) = j;
        m++;
    endif
    if(abs(arg(E21o(j-1)) - arg(E21o(j))) > pi/4.0) #detected a jump.
        Branch_o(n) = j;
        n++;
    endif
endfor
n--;
m--;
for j = 1:m
    if(j == m)
        k = Nfreqs;
    else
k = Branch_e(j+1);
endif
for l = Branch_e(j):k
E21e(l) = (-1)^j * E21e(l);
endfor
endfor

for j = 1:n
if(j == n)
k = Nfreqs;
else
k = Branch_o(j+1);
endif
for l = Branch_o(j):k
E21o(l) = (-1)^j * E21o(l);
endfor
endfor

fp_out = fopen([Databasename,"_bent.out"], "wt");

for j = 1:Nfreqs
E = zeros(4);

# Build T matrix for the straight-through structure.
E(1,1) = E11e(j);
E(1,3) = E21e(j);
E(2,2) = E11o(j);
E(2,4) = E21o(j);
E(3,1) = E21e(j);
E(3,3) = E22e(j);
E(4,2) = E21o(j);
E(4,4) = E22o(j);

T1 = S2T(E);

# Build T matrix for complete structure comprising bent and straight parts.
# First, convert measured scattering parameters from normal variety to
# even-odd variety.
Meas1(1,1) = S(j+23*Nfreqs);
Meas1(1,2) = S(j+24*Nfreqs);
Meas1(1,3) = S(j+26*Nfreqs);
Meas1(1,4) = S(j+27*Nfreqs);
Meas1(2,1) = Meas1(1,2);
Meas1(2,2) = S(j+25*Nfreqs);
Meas1(2,3) = S(j+28*Nfreqs);
Meas1(2,4) = S(j+29*Nfreqs);
Meas1(3,1) = Meas1(1,3);
Meas1(3,2) = Meas1(2,3);
Meas1(3,3) = S(j+30*Nfreqs);
Meas1(3,4) = S(j+31*Nfreqs);
Meas1(4,1) = Meas1(1,4);
Meas1(4,2) = Meas1(2,4);
Meas1(4,3) = Meas1(3,4);
Meas1(4,4) = S(j+32*Nfreqs);

Mxd = S2Mixed(Meas1);
# Now, convert to T parameters
Tstruct = S2T(Mxd);

# Now, extract "bent" T parameters (reversing sense of ports 1 and 2 as a result of # Measurement direction.
Tbent = Rev * inverse(inverse(T1) * Tstruct) * Rev;

# Convert back to "bent" line S parameters (even-odd) for plotting.
Sbent = T2S(Tbent);
Tbent = T2S(Tbent);
fprintf(fp_out, "%e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e\n", f(j), \
  real(Sbent(1,1)), imag(Sbent(1,1)), \ 
  real(Sbent(1,2)), imag(Sbent(1,2)), \ 
  real(Sbent(1,3)), imag(Sbent(1,3)), \ 
  real(Sbent(1,4)), imag(Sbent(1,4)), \ 
  real(Sbent(2,2)), imag(Sbent(2,2)), \ 
  real(Sbent(2,3)), imag(Sbent(2,3)), \ 
  real(Sbent(2,4)), imag(Sbent(2,4)), \ 
  real(Sbent(3,3)), imag(Sbent(3,3)), \ 
  real(Sbent(3,4)), imag(Sbent(3,4)), \ 
  real(Sbent(4,4)), imag(Sbent(4,4)));

#Save for de-embedding operation (Tb now contains S parameter data).
Tb(j) = Tbent(1,1);
Tb(j+Nfreqs) = Tbent(1,2);
Tb(j+Nfreqs*2) = Tbent(1,3);
Tb(j+Nfreqs*3) = Tbent(1,4);
Tb(j+Nfreqs*4) = Tbent(2,1);
Tb(j+Nfreqs*5) = Tbent(2,2);
Tb(j+Nfreqs*6) = Tbent(2,3);
Tb(j+Nfreqs*7) = Tbent(2,4);
Tb(j+Nfreqs*8) = Tbent(3,1);
Tb(j+Nfreqs*9) = Tbent(3,2);
Tb(j+Nfreqs*10) = Tbent(3,3);
Tb(j+Nfreqs*11) = Tbent(3,4);
Tb(j+Nfreqs*12) = Tbent(4,1);
Tb(j+Nfreqs*13) = Tbent(4,2);
Tb(j+Nfreqs*14) = Tbent(4,3);
Tb(j+Nfreqs*15) = Tbent(4,4);
endfor
fclose(fp_out);

# Get measurement data of DUT. Enter directory that contains S param data.
for l = 1:3
    FileName = [Databasename1,"/", Files(3*l-2:3*l), ".csv"];
    fp_in = fopen(FileName, "rt");
    txt = fgets(fp_in); #move beyond junk text at beginning of file.
    txt = fgets(fp_in);
    txt = fgets(fp_in);

for m = 1:NPoints
  [freq, mag, phs] = fscanf(fp_in, "%e, %e, %e", "C");
  Sav(m + (l-1) * NPoints) = 10^(mag / 20.) * exp(i * pi * phs / 180.);
  f(m) = freq;
endfor
fclose(fp_in);
endfor

# Now, let's do the de-embedding (using even-odd modes)!!!
fp_out = fopen([Databasename1, "_DUT.out"], "wt");

for j = 1:Nfreqs
  GAMMA_even = 0.5 * (Sav(j) + 2.0 * Sav(j+Nfreqs) + Sav(j+2*Nfreqs));
  GAMMA_odd = 0.5 * (Sav(j) - 2.0 * Sav(j+Nfreqs) + Sav(j+2*Nfreqs));
  G_eo = 0.5 * (Sav(j) - Sav(j+2*Nfreqs));
  Tbent(1,1) = Tb(j);
  Tbent(1,2) = Tb(j+Nfreqs);
  Tbent(1,3) = Tb(j+Nfreqs*2);
  Tbent(1,4) = Tb(j+Nfreqs*3);
  Tbent(2,1) = Tb(j+Nfreqs*4);
  Tbent(2,2) = Tb(j+Nfreqs*5);
  Tbent(2,3) = Tb(j+Nfreqs*6);
  Tbent(2,4) = Tb(j+Nfreqs*7);
  Tbent(3,1) = Tb(j+Nfreqs*8);
  Tbent(3,2) = Tb(j+Nfreqs*9);
  Tbent(3,3) = Tb(j+Nfreqs*10);
  Tbent(3,4) = Tb(j+Nfreqs*11);
  Tbent(4,1) = Tb(j+Nfreqs*12);
  Tbent(4,2) = Tb(j+Nfreqs*13);
  Tbent(4,3) = Tb(j+Nfreqs*14);
  Tbent(4,4) = Tb(j+Nfreqs*15);

  SDUT = Extract(Tbent, GAMMA_even, GAMMA_odd, G_eo, G_eo);
  # Convert to impedance.
  ZDUT = 50.0 * (eye(2) + SDUT) * inverse(eye(2) - SDUT);

  fprintf(fp_out, "%e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e %e
",
      f(j),
      real(ZDUT(1)), imag(ZDUT(1)),
      real(ZDUT(2)), imag(ZDUT(2)),
      real(ZDUT(3)), imag(ZDUT(3)),
      real(ZDUT(4)), imag(ZDUT(4)),
      real(SDUT(1)), imag(SDUT(1)),
      real(SDUT(2)), imag(SDUT(2)),
      real(SDUT(3)), imag(SDUT(3)),
      real(SDUT(4)), imag(SDUT(4)));
  fprintf(fp_out, "%e %e %e %e %e %e %e %e %e\n",
      f(j),
      real(GAMMA_even), imag(GAMMA_even),
      real(G_eo), imag(G_eo),
      real(GAMMA_odd), imag(GAMMA_odd),
      real(SDUT(4)), imag(SDUT(4)));
endfor
fclose(fp_out);
**Parameter extraction subroutine**

# The Octave FSOLVE command is used to extract parameters
# of single line using set of TRL measurements
# Given Gamma_thru
#       Gain_thru
#       Gamma_refl
#       Gamma_line1
#       Gain_line1
#       Z1
# we extract
#       E[3]  (error box characterisation)
#       Gamma_r ("reflect" reflection coefficient)
#       gamma (line propagation factor; complex)

function [E, Gamma_r, gamma1] = param_extract(Gamma_thru, \
    Gain_thru, Gamma_refl, Gamma_line1, Gain_line1, Z1, dL1, k0, kr)
global Meas;
global res;
global StartVec;

Meas(1) = Gamma_thru;
Meas(2) = Gain_thru;
Meas(3) = Gamma_refl;
Meas(4) = Gamma_line1;
Meas(5) = Gain_line1;
Meas(6) = Z1;
niters = 0;

do

[Ans, info, msg] = fsolve("Residual", StartVec);
niters++;
StartVec(1) = 1.0 * rand(1);
StartVec(2) = 1.0 * rand(1);

until((info == 1) && ((Ans(4) / dL1 > k0) && (Ans(4) / dL1 < kr)) || (niters > 25))

#info
#msg
#res
#niters

E(3) = Ans(1) + Ans(2) * i;
E(2) = sqrt(Gain_thru * (1 - E(3)^2));
E(1) = Gamma_thru - Gain_thru * E(3);
Gamma_r = (Gamma_refl - E(1)) / (E(2)^2 + (Gamma_refl - E(1)) * E(3));
gamma1 = (Ans(3) + Ans(4) * i) / dL1;

endfunction;
**Conversion from normal to mixed mode S parameters**

# Function for converting single-ended 4-port S params to even/odd params

```matlab
function M = S2Mixed(S)

R = [1, 1, 0, 0; -1, 1, 0, 0; 0, 0, 1, 1; 0, 0, -1, 1];
M = 0.5 * R * S * R';
endfunction
```

**FSOLVE residual generation**

# This routine solves a reduced set of TRL equations
# for E22 and gamma*L
# E11, E21 and Gamma_open

```matlab
function r = Residual(x);
    global Meas;
    global res;
    r(4) = 0;
    # Meas(1) = Gamma thru
    # Meas(2) = Gain thru
    # Meas(3) = Gamma reflect
    # Meas(4) = Gamma line
    # Meas(5) = Gain line
    # Meas(6) = line characteristic impedance (normalised to 50 ohms)

    # x(1) = real(E22)
    # x(2) = imag(E22)
    # x(3) = real(gamma * dL);
    # x(4) = imag(gamma * dL);

    gamma = x(3) + x(4) * i;
    E22 = x(1) + x(2) * i;

    Gamma_thru = Meas(1);
    G_thru = Meas(2);
    Gamma_line = Meas(4);
    G_line = Meas(5);
    Z1 = Meas(6);

    D = 2 * Z1 * cosh(gamma) + (Z1^2 + 1) * sinh(gamma);
    T11 = (Z1^2 - 1) * sinh(gamma) / D;
    T21 = 2 * Z1 / D;

    r(1) = real(G_line * (1 - E22 * T11)^2 - G_line * E22^2 * T21^2 - G_thru * (1 - E22^2) * T21);
    r(2) = imag(G_line * (1 - E22 * T11)^2 - G_line * E22^2 * T21^2 - G_thru * (1 - E22^2) * T21);
    r(3) = real((Gamma_line - Gamma_thru + G_thru * E22) * ((1 - E22 * T11)^2 - E22^2 * T21^2) -
```
\[
(G_{\text{thru}} \times (1 - E22^2)) \times (T11 - E22 \times T11^2 + T21^2 \times E22));
\]
\[
r(4) = \text{imag}((\text{Gamma}_{\text{line}} - \text{Gamma}_{\text{thru}} + G_{\text{thru}} \times E22) \times ((1 - E22 \times T11)^2 - E22^2 \times T21^2) - \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \\
(G_{\text{thru}} \times (1 - E22^2)) \times (T11 - E22 \times T11^2 + T21^2 \times E22));
\]
res = r(1:4);
endfunction

Convert 4-port S parameters to T parameters

Convert 4-port S parameters to T parameters

# Convert S matrix to T matrix
function T = S2T(S)

for j = 1:2
for k = 1:2
    S11(j,k) = S(j,k);
    S21(j,k) = S(j+2,k);
    S12(j,k) = S(j,k+2);
    S22(j,k) = S(j+2,k+2);
endfor
endfor

if(abs(det(S21)) > 1e-8)
    S21inv = inverse(S21);
else
    S21inv = zeros(2);
endif

T11 = S12 - S11 * S21inv * S22;
T12 = -S21inv * S22;
T21 = -S21inv * S22;
T22 = S21inv;

for j = 1:2
for k = 1:2
    T(j,k) = T11(j,k);
    T(j+2,k) = T21(j,k);
    T(j,k+2) = T12(j,k);
    T(j+2,k+2) = T22(j,k);
endfor
endfor
endfunction

Convert 4-port S parameters to T parameters

Convert T parameters to S parameters

# Convert T matrix to S matrix
function S = T2S(T)

for j = 1:2
for k = 1:2
    S11(j,k) = T(j,k);
    S21(j,k) = T(j+2,k);
    S12(j,k) = T(j,k+2);
    S22(j,k) = T(j+2,k+2);
endfor
endfor

endfunction
endfor
endfor
if(abs(det(T22)) > 1.0e-8)
    T22inv = inverse(T22);
else
    T22inv = zeros(2);
endif
S11 = T12 * T22inv;
S12 = T11 - T12 * T22inv * T21;
S21 = T22inv;
S22 = -T22inv * T21;
for j = 1:2
    for k = 1:2
        S(j,k) = S11(j,k);
        S(j+2,k) = S21(j,k);
        S(j,k+2) = S12(j,k);
        S(j+2,k+2) = S22(j,k);
    endfor
endfor
endfunction

DUT De-embedding routine

# This routing extracts reflection coefficient matrix of two-port structure of DUT using characterisation data.

function G = Extract(Sin, Gamma_1, Gamma_2, G_12, G_21)
    for j = 1:2
        for k = 1:2
            S11(j,k) = Sin(j,k);
            S21(j,k) = Sin(j+2,k);
            S12(j,k) = Sin(j,k+2);
            S22(j,k) = Sin(j+2,k+2);
        endfor
    endfor
if(abs(det(S21)) > 1e-8)
    S21inv = inverse(S21);
else
    S21inv = zeros(2);
endif
if(abs(det(S12)) > 1e-8)
    S12inv = inverse(S12);
else
    S12inv = zeros(2);
endif
B(1,1) = Gamma_1;
B(1,2) = G_12;
B(2,1) = G_21;
B(2,2) = Gamma_2;
X = S12inv * (B - S11) * S21inv;
G = inverse(eye(2) + X * S22) * X;
endfunction