

DesignCon 2004

Making S-parameter data suitable for SPICE modeling

Jan De Geest, Ph.D, FCI CDC

Stefaan Sercu, Ph.D, FCI CDC

Craig Clewell, FCI CDC

Jim Nadolny, FCI CDC

FCI
Communications, Data, Consumer Division
FCI 's-Hertogenbosch
Victorialaan 1,
5213 JG 's-Hertogenbosch,
The Netherlands
tel: +31 73 6206 922
e-mail: jdegeest@fciconnect.com
ssercu@fciconnect.com

Abstract

S-parameters are well suited to accurately represent the behavior of a passive interconnect over a broad frequency range. The need for the ability to simulate the impact of drivers and receivers on the interconnection link requires the co-simulation of these linear S-parameter models with the non-linear models for the silicon devices. This paper provides an overview of existing techniques to include S-parameter data in SPICE simulations and of their advantages and disadvantages. It focuses on the generation of accurate broad-band SPICE models for connectors, starting from measured or simulated S-parameter data.

Author's biographies

Jan De Geest received the degree in electrical engineering from the University of Gent, Belgium in 1994 and the degree in supplementary studies in aerospace techniques from the University of Brussels, Belgium in 1995. From September 1995 to December 1999 he worked as a research assistant at the Department of Information Technology (INTEC) of the University of Gent, where he received the PhD degree in electrical engineering in 2000. Since January 2000 he has been working for FCI in 's-Hertogenbosch, The Netherlands. His work focuses on the modeling and simulation of high-speed interconnection links.

Stefaan Sercu was born in Ieper, Belgium, on February 6, 1969. He received the degree in electrical engineering from the University of Gent in 1992. From 1992 to 1998, he worked as a research assistant at the Department of Information Technology (INTEC) of the university of Gent. His research concentrated on the characterization and modeling of high-speed connectors and interconnections. In 1998 he joined FCI where he is currently responsible for high-speed signal integrity simulations and measurements. In 2002 he received the PhD degree in electrical engineering from the University of Gent.

Craig Clewell was born in Reading Pennsylvania on May 19, 1964. He received a Bachelors Degree in Engineering from Penn State University in 1995 and is currently working on an MSEE from Penn State University. Before joining FCI in 1995 he had worked at AMP where he was employed as a Test Engineer. Following this employment with AMP he worked for C-MAC where he assisted in the design of crystal controlled oscillators. Initially, his focus at FCI was centered on connector design and analysis, but has been redirected to include signal integrity for the past several years as well. Craig has provided technical training at FCI, has received one patent and has another one pending. His current interests include the signal integrity of high-speed systems, boating and biking.

Jim Nadolny has 10 years experience in the connector industry and works for FCI in the Harrisburg, PA area. Jim has an MSEE from University of New Mexico and a BSEE from University of Connecticut. For the past few years Jim has been involved with multi-gigabit data transmission of imperfect channels and frequently presents at IEEE and IEC venues including DesignCon. He has over 20 publications in technical journals and trade magazines.

1. Introduction

The accuracy of traditional lumped-element or distributed SPICE models for connectors (created from a field solver) is usually only guaranteed for low frequencies and high rise times. This type of models does not normally represent the losses within the connector. Obtaining SPICE models that can accurately represent the connector impedance, the connector losses and the (multi-line) cross-talk behavior is a long and difficult task, especially if the number of pins is large. S-parameters (either measured or field solver based) are more suited to accurately represent the behavior of the connector, and of the physical phenomena behind it (skin effect, dielectric loss, dispersion), over a broad frequency range. However, when simulating or designing interconnection links the interactions between the non-linear transceiver and/or equalizer models and the linear models of the passive parts of the link must be taken into account. Models for silicon devices are either IBIS or SPICE formatted (often encrypted HSPICE models), which require specific SPICE simulators to run them. Thus, the need for SPICE models for the passive parts of an interconnection link in general, and for connectors in particular, still remains.

Integrating S-parameter data in a SPICE-like simulator can be done in several ways. One method is to include a convolution engine in a SPICE simulator that enables direct import of the S-parameter data. Certain commercial SPICE simulators like Synopsys HSPICE, Apache NSPICE and Agilent ADS offer the capability to load S-parameters directly.

A second way of integrating S-parameter data in SPICE simulations is to transfer an S-parameter model into an accurate black-box SPICE model that can then be used in a SPICE simulator. This black-box model can use either macro-modeling capabilities offered by certain SPICE dialects (like PSPICE or HSPICE), or it can be in a generic SPICE format. Commercial tools like Ansoft Full-Wave SPICE, the SPICE generator from Agilent ADS, Sigrity BroadBand Spice and Optimal WideBand Spice offer this capability.

This paper focuses on the generation of accurate broad-band SPICE models for connectors. An overview of existing techniques will be given and advantages and disadvantages of these techniques will be contrasted (regarding accuracy, simulation time, model size, etc.). A method will be proposed for generating broad-band SPICE models, using rational approximation techniques and SPICE macro-modeling capabilities. The resulting SPICE models are passive and stable. S-parameter data can come from either field solver based simulations or measurements. SPICE models will be generated using measured S-parameters of FCI's new AirMax VSTM connector and FCI's Metral® 4000 connector. The accuracy of the SPICE models in both the time domain and frequency domain will be demonstrated.

2. Existing methods for using S-parameters in SPICE simulations

There are two main methods to combine S-parameter models with SPICE models. One method is to directly manipulate the S-parameters with a convolution algorithm. The second way is to generate SPICE models that accurately represent the original S-parameters. Both methods are discussed briefly in the following two sections.

2.1 Direct import of S-parameters using a convolution engine

Suppose we have the S-parameters of a general N -port structure over a given frequency range (figure 1). The S-parameters describe the relationships between the input and output waves at the ports at given frequencies. A SPICE simulator solves for the voltages at the nodes and the currents going into the nodes at given time points. The elements in the netlist are described by a set of equations that define the relationship between the terminal currents and voltages of the elements. These equations are solved together with Kirchhoff's laws to obtain the nodal voltages and currents.

If we want to directly include S-parameters in a SPICE simulator, the S-parameters need to be converted into a set of equations containing the port voltages and currents as a function of time. This can be done in the following manner [1]. The S-parameter matrix is defined as:

$$\bar{b}(f) = \bar{S}(f)\bar{a}(f) \quad (1)$$

where the vectors a and b represent the input and output waves, defined as (with R representing a diagonal matrix containing the reference port impedances):

$$\bar{a} = (\bar{V} + \bar{R}\bar{I})/2 \quad (2)$$

$$\bar{b} = (\bar{V} - \bar{R}\bar{I})/2 \quad (3)$$

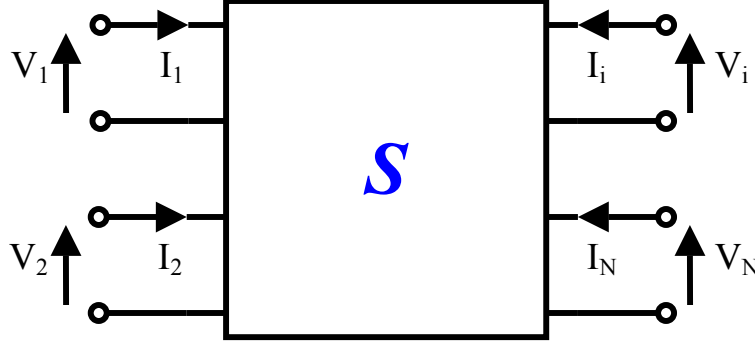


Figure 1: General N -port structure described by its S-parameter matrix.

If we apply a discrete inverse Fourier transform to equation (1) we can write the voltage vector at a discrete time point g as:

$$\bar{V}(g) = \bar{R}\bar{I}(g) + 2\sum_{h=0}^g \bar{s}(h)\bar{a}(g-h)\Delta t \quad (4)$$

The discrete time point g corresponds to the continuous time point $g\Delta t$, where Δt is the timestep. If we split the sum on the right-hand side of equation (4) in the part for $h=0$ and the part for $h=1, \dots, g$, then we can extract the term containing the input wave vector $a(g)$ from the sum and we get:

$$\bar{V}(g) = \bar{R}\bar{I}(g) + 2\bar{s}(0)\bar{a}(g)\Delta t + 2\sum_{h=1}^g \bar{s}(h)\bar{a}(g-h)\Delta t \quad (5)$$

The sum on the right-hand side of equation (5) is now a function of the voltages and currents at the previous time steps $0, \dots, (g-1)\Delta t$ only, and we define this as:

$$\bar{e}(g) = \sum_{h=1}^g \bar{s}(h)\bar{a}(g-h)\Delta t \quad (6)$$

At any time point g , $e(g)$ can be calculated from the voltages and currents at the previous time points. In other words, $e(g)$ is a constant at any given time point. The relationship between the terminal voltages and currents at a discrete time point g can be written as:

$$\bar{V}(g) = (\bar{U} - \bar{s}(0)\Delta t)^{-1}(\bar{U} + \bar{s}(0)\Delta t)\bar{R}\bar{I}(g) + 2(\bar{U} - \bar{s}(0)\Delta t)^{-1}\bar{e}(g) \quad (7)$$

or simplified as

$$\bar{V}(g) = \bar{A}\bar{I}(g) + \bar{c}(g) \quad (8)$$

The terminal voltages can be written as the sum of a term that is proportional to the terminal currents and a constant term. Solving the above equation, together with the equations imposed by the other elements in the netlist and Kirchhoff's laws results in the nodal voltages and currents at any discrete time point g .

Using the above approach requires an inverse discrete Fourier transform to convert the S-parameters from the frequency domain to the time domain, i.e. to calculate the impulse response corresponding to each of the individual S-parameters. The accuracy of the impulse response and of the transient simulation results will depend on the amount of available frequency domain data. The time domain resolution is inversely proportional to the maximum frequency and the maximum time point is inversely

proportional to the frequency step. Since the voltages and currents at a given time point are effected by the voltages and currents at previous time points, the time domain resolution needs to be small enough to control the propagation of errors from one time point to another. Furthermore, if the maximum frequency is insufficiently high compared to the spectral content of the signals generated by the sources, non-causal results will most likely be obtained. Often a lot of data points are required for accurate analysis, resulting in large S-parameter model files. E.g. if we want to accurately simulate a PRBS with a sharp edge going across a 1 meter board, both a long simulation time (small frequency step) and high resolution (high maximum frequency) will be required.

In some cases it can also be necessary to pre-process the S-parameter data before feeding it into the SPICE simulator, to make sure the S-parameter data is causal and passive [2,3]. Usually some kind of frequency domain window is applied before doing the inverse Fourier transform to avoid errors caused by ringing and aliasing in the time domain.

2.2 S-parameter based SPICE models

An alternative approach to using S-parameters in SPICE simulators is to generate an equivalent circuit that exhibits the same behavior at its ports as the original S-parameter model. Commercial tools like Ansoft Full-Wave SPICE, the SPICE generator from Agilent ADS, Sigrity BroadBand Spice and Optimal WideBand Spice offer this capability. All these tools use macro-modeling capabilities offered by certain SPICE dialects (like PSPICE or HSPICE) to generate the SPICE models, and some offer models in a generic SPICE format. The macro-modeling capabilities make it possible to perform numerical operations (like multiplications or additions) on voltages and currents through the use of voltage or current controlled voltage or current sources, and to create frequency-dependent elements through the use of voltage or current controlled sources with a frequency-dependent gain (e.g. [4]). The accuracy that can be obtained with SPICE models created using this approach is generally superior to the accuracy of conventional lumped-element or distributed SPICE models, especially for high bandwidths and low rise times. Lumped-element SPICE models can be obtained by assuming a certain circuit topology and optimizing the circuit element values to minimize the difference between the circuit response and the measured or simulated data. If a large bandwidth is required or if the number of external ports in the circuit is large, the amount of circuit elements that are necessary to achieve a given accuracy can become very large, and convergence of the optimization process can become very difficult. In such cases special techniques are required to obtain lumped-element SPICE models [5]. Another possibility is to generate a large lumped-element equivalent circuit from the three-dimensional mesh generated by a full-wave tool (like the PEEC method) and to apply a model order reduction scheme to reduce the number of internal nodes and the size of the model. Because of the nature of this technique it can only be applied to simulated responses, so it cannot be used to generate SPICE models from measured data.

The difficulty and also the challenge of the method described here lie in finding rational approximation functions that accurately represent each of the S-parameters and in ensuring the passivity and stability of the resulting SPICE models. The approach used by FCI will be discussed in more detail in the next section.

3. Accurate SPICE models from S-parameters

This section describes the method used at FCI to generate accurate SPICE models based on simulated or measured S-parameters. First the basic model format is given, and then an algorithm is presented that is used to generate rational approximation functions for the individual S-parameters. The physical properties the SPICE model has to satisfy (stability and passivity) are discussed in a third sub-section. Finally, different output formats for the SPICE model are presented.

3.1 Model description

The SPICE model that is constructed must have the same behavior at the terminals as the original S-parameter model, over a given frequency range. To synthesize such a circuit we start from the definition of the S-parameters. We can write:

$$\bar{V} = \bar{Z}\bar{I} = 50(\bar{U} + \bar{S})(\bar{U} - \bar{S})^{-1} \quad (9)$$

Here the vectors V and I contain the terminal voltages and currents respectively. Z is the impedance matrix, S is the S-parameter matrix and U is the $N \times N$ unit matrix. We assume here that the S-parameters are referenced to port impedances of 50Ω . If we define a vector X as

$$\bar{X} = \sqrt{50}(\bar{U} - \bar{S})^{-1}\bar{I} \quad (10)$$

then

$$\bar{V} = \sqrt{50}(\bar{U} + \bar{S})\bar{X} \quad (11)$$

We can write the terminal voltages and currents as:

$$V_i = \sqrt{50} \left(X_i + \sum_{n=1}^N S_{in} X_n \right) \quad \text{with } i=1, \dots, N \quad (12)$$

$$I_i = \frac{1}{\sqrt{50}} \left(X_i - \sum_{n=1}^N S_{in} X_n \right) \quad \text{with } i=1, \dots, N \quad (13)$$

Using controlled voltage and current sources several circuits can be formulated such that the behavior at the terminals is given by the equations above. One such a circuit is shown in figure 2 (where $r = \sqrt{50}$). This figure shows the components in the model for the i -th port. The terminal voltages and currents of the circuit shown in this figure satisfy equations (12) and (13). The complete SPICE model will consist of a copy of the circuit shown in figure 2 for each of the ports.

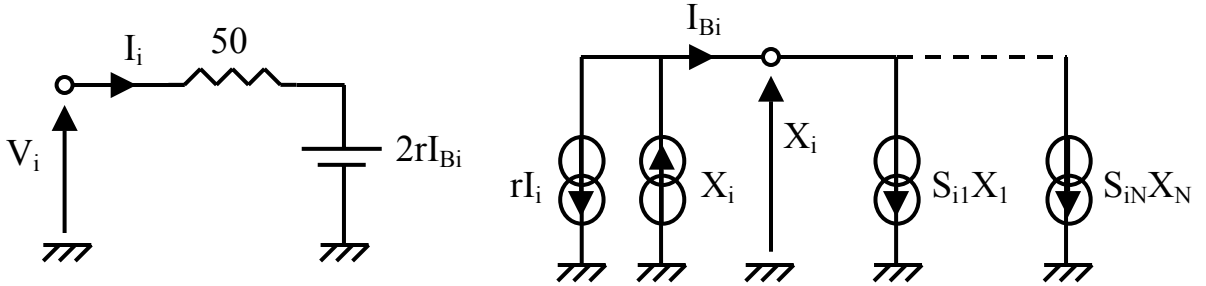


Figure 2: Schematic of synthesized circuit.

The vector X appears as a vector of nodal voltages and the S-parameters appear as the frequency-dependent gains of voltage controlled current sources (VCCS). Commercial SPICE simulators like HSPICE and PSPICE allow the use of VCCS with a frequency-dependent gain. However, in section 3.4 a model will be shown that only uses generic SPICE elements, and therefore can be used in any SPICE3 compatible simulator.

By using the circuit shown in figure 2 each S-parameter can be treated separately, which is computationally a lot easier than having to consider the S-matrix as a whole when generating a rational approximation (see section 3.2). In the literature there are methods to generate state-space models, or rational approximations for Multiple-Input Multiple-Output (MIMO) systems. The complexity and numerical difficulties associated with these methods grow drastically with the number of ports. By treating each S-parameter as a separate Single-Input Single-Output (SISO) system these difficulties can be reduced.

However, we have to keep in mind the physical properties of the circuit, which will be discussed in section 3.3. Some of the physical conditions that have to be met impose limitations on a combination of

different S-parameters. This implies that there still exists a relationship between the different rational approximation functions, and that these are not completely independent from one another.

3.2 Rational approximation of the S-parameters

Each S-parameter $S_{ij}(s)$ is approximated by a rational transfer function $H_{ij}(s)$ of the form:

$$S_{ij}(s) \approx H_{ij}(s) = \frac{a_0 + a_1s + \dots + a_M s^M}{1 + b_1s + \dots + b_N s^N} = \frac{\prod_{m=1}^M a_m (s - z_m)}{\prod_{n=1}^N b_n (s - p_n)} \quad (14)$$

with $s=j2\pi f$. z_m ($m=1, \dots, M$) and p_n ($n=1, \dots, N$) represent the zeros and the poles of $H_{ij}(s)$. The real coefficients $a_0, \dots, a_M, b_1, \dots, b_N$ are determined by fitting the rational function to a known data set consisting of K samples:

$$S_{ij,k} = S_{ij}(j\omega_k) = \frac{\sum_{m=0}^M a_m (j\omega_k)^m}{1 + \sum_{n=1}^N b_n (j\omega_k)^n} = R_{ij,k} + jI_{ij,k} \quad \text{for } k=1, \dots, K \quad (15)$$

where $\omega_k=2\pi f_k$ and f_k are the sample frequencies. The approximation error for the k -th sample can be written as (note that the subscript 'ij' was omitted to relieve the notation):

$$E(k) = \sum_{m=0}^M a_m B_m(j\omega_k) - (R_k + jI_k)(1 + \sum_{n=1}^N b_n B_n(j\omega_k)) = E_R(k) + jE_I(k) \quad (16)$$

where

$$E_R(k) = -\sum_{m=0}^M a_m P_m(j\omega_k) + \sum_{n=1}^N b_n [R_k P_n(j\omega_k) - I_k Q_n(j\omega_k)] + R_k \quad (17)$$

$$E_I(k) = -\sum_{m=0}^M a_m Q_m(j\omega_k) + \sum_{n=1}^N b_n [R_k Q_n(j\omega_k) + I_k P_n(j\omega_k)] + I_k \quad (18)$$

and

$$B_m(j\omega) = P_m(j\omega) + jQ_m(j\omega) \quad (19)$$

are the basis functions. In our case

$$B_m(j\omega) = (j\omega)^m \quad (20)$$

Equation (15) is solved in a least-squares sense by minimizing the following objective function (with $\mathfrak{g}=\{a_0, \dots, a_M, b_1, \dots, b_N\}$):

$$\min_{\mathfrak{g}} \left[0.5 \sum_{k=1}^K (E_R^2(k) + E_I^2(k)) \right] \quad (21)$$

The minimum in equation (21) is found for the parameters x satisfying the following equation (with $x \in \mathfrak{g}=\{a_0, \dots, a_M, b_1, \dots, b_N\}$):

$$\sum_{k=1}^K (E_R(k) \frac{\partial E_R(k)}{\partial x} + E_I(k) \frac{\partial E_I(k)}{\partial x}) = 0 \quad (22)$$

This results in the following system of equations:

$$\sum_{m=0}^M a_m \sum_{k=1}^K (B_m B_u^+ + B_m^+ B_u) - \sum_{n=1}^N b_n \sum_{k=1}^K (S_k B_n B_u^+ + S_k^+ B_n^+ B_u) = \sum_{k=1}^K (S_k B_u^+ + S_k^+ B_u) \quad (23)$$

$$\sum_{m=0}^M a_m \sum_{k=1}^K (B_m B_v^+ S_k^+ + B_m^+ B_v S_k) - \sum_{n=1}^N b_n \sum_{k=1}^K S_k S_k^+ (B_n B_v^+ + B_n^+ B_v) = \sum_{k=1}^K S_k S_k^+ P_v \quad (24)$$

with $u=0, \dots, M$ and $v=1, \dots, N$. The “+” superscript denotes the complex conjugate. Equations (23) and (24) form a system of $[M+N+1]$ equations in $[M+N+1]$ unknowns. This system of equations loses numerical rank very rapidly as the order of the numerator and the denominator increase, reducing the accuracy of the calculated coefficients and thus the quality of the approximation. These numerical issues can be reduced by using an orthonormal polynomial basis that spans the same space as the original basis functions given in equation (20). Such a basis can be constructed using a traditional Gram-Schmidt orthogonalization process or a Lanczos process [6,7].

For a given value of M and N the equations (23) and (24) yield the coefficients of the numerator and the denominator. From these coefficients the poles of the rational approximation function can be calculated and we can write:

$$H(s) = H_0 + \sum_{n=1}^{NR} \frac{A_n}{s - p_n} + \sum_{n=1}^{NI/2} \left(\frac{R_n}{s - p_n} + \frac{R_n^+}{s - p_n^+} \right) \quad (25)$$

with NR and NI the number of real and complex poles respectively. The estimation of the location of the poles can be improved by using a technique known as “Vector Fitting” [8]. For this a weighting function $\sigma(s)$, defined as

$$\sigma(s) = 1 + \sum_{n=1}^{NR} \frac{k_n}{s - q_n} + \sum_{n=1}^{NI/2} \left(\frac{K_n}{s - q_n} + \frac{K_n^+}{s - q_n^+} \right) \quad (26)$$

with known (initial) poles $\{q_n\}$ and unknown residues $\{k_n, K_n\}$, is introduced. This function is used to enforce the following condition:

$$\sigma(s)H(s) \cong M_0 + \sum_{n=1}^{NR} \frac{m_n}{s - q_n} + \sum_{n=1}^{NI/2} \left(\frac{M_n}{s - q_n} + \frac{M_n^+}{s - q_n^+} \right) \quad (27)$$

Since the right-hand side of equation (27) has the same poles $\{q_n\}$ as the weighting function, a cancellation between the zeros $\{z_n\}$ of $\sigma(s)$ and the poles $\{p_n\}$ of $H(s)$ must occur. Thus, the poles of $H(s)$ can be estimated by solving (27) for the unknown residues $\{k_n, K_n\}$ (and $\{m_n, M_n\}$), computing the zeros $\{z_n\}$ of $\sigma(s)$ and enforcing $z_n = p_n$. This process is called “pole relocation”. It avoids the use of ill-conditioned least squares algorithms for the direct approximation of (25). Also, the relocation of the poles can be iterated using the estimated poles as starting poles for the new iteration. This process usually converges in very few steps.

The optimal values for M and N are determined adaptively so that the best approximation is found. To determine the optimal degree for the numerator and the denominator, a number of rational approximation functions are generated for different values of M and N . The step response obtained from the rational functions $H(s)$ is compared with the step response obtained from the original S-parameters (over the given frequency range). The optimal approximation function is the function that minimizes the error between the two time domain waveforms:

$$\min_{M,N} \left[\left| IFFT(H(s)w(s)) - IFFT(S(s)w(s)) \right| \right] \quad (28)$$

The function $w(s)$ is a weighting function which is defined as the Laplace transform of a time step function $v_s(t)$ with a sinusoidal ramp at a given 10-90% risetime T_R :

$$v_s(t) = 0.5(1 - \cos(\omega_0 t)) \quad (29)$$

(with $\omega_0 = 2\pi \sin(0.8)/T_R$). The Laplace transform of this step function is:

$$w(s) = 0.5 \left(\frac{1}{s} - \frac{s}{s^2 + \omega_0^2} \right) (1 + \exp(-\pi s / \omega_0)) \quad (30)$$

When generating the rational approximation function the risetime T_R must be specified, together with the maximum sample frequency f_K . The maximum sample frequency will determine the frequency range over which the deviation between $H(s)$ and $S(s)$ is minimized, while T_R will also impact the behavior of

the SPICE model for frequencies above f_k . The lower the risetime is, the higher the bandwidth of the SPICE model will have to be. Note that in any case $M \leq N$.

3.3 Physical properties

Although the individual S-parameters are treated separately when generating a rational approximation function, those rational functions are not independent. The generated SPICE model has to satisfy a number of physical properties. The model must be stable and passive, and this will impose limitations on the rational approximation functions. In the next sections this is discussed in more detail.

3.3.1 Stability

Stability requires that the rational approximation function $H(s)$ has no poles in the right-half plane:

$$\text{Re}(p_n) \leq 0 \quad \text{for } n=1, \dots, N \quad (31)$$

If for a given degree of the numerator and the denominator the rational approximation function has poles in the right-half plane, these poles are mirrored around the imaginary axis, and the coefficients of the numerator are re-calculated to minimize the approximation error with the given denominator poles. Other ways to obtain guaranteed stable rational approximation functions from frequency domain data can be found in [6,9].

3.3.2 Passivity

Since the original physical system we are modeling (connectors, via holes, etc.) is passive, the SPICE model of this system must also be passive. Passive systems are systems that consume energy, and do not generate energy. The interconnection of two passive structures is always passive. Stable systems do not possess this closure property. Approximating a passive model by a merely stable model can result in inaccurate results or even an unstable circuit, even if the other sub-circuits in the network are stable [9,10].

Passivity requires that the magnitude of each of the S-parameters is smaller than 1 (for $0 \leq \omega \leq \infty$):

$$|S_{ij}(j\omega)| \leq 1 \quad (32)$$

and, a more strict condition, that the sum of the magnitudes of the S-parameters on each row of the S-matrix is smaller than 1 (again for $0 \leq \omega \leq \infty$):

$$P_i(j\omega) = \sum_{j=1}^N |S_{ij}(j\omega)|^2 \leq 1 \quad \text{for } i=1, \dots, N \quad (33)$$

For the SPICE model to be passive the rational approximation functions $H_{ij}(j\omega)$ also have to satisfy equations (32) and (33).

To impose the first condition the maximum magnitude $H_{ij,max}$ of $H_{ij}(j\omega)$ is calculated. If $H_{ij,max}$ is larger than 1 the rational approximation function is divided by $H_{ij,max}$ so that the maximum magnitude of the new approximation function

$$\tilde{H}_{ij}(s) = H_{ij}(s) / H_{ij,max} \quad (34)$$

is guaranteed to be smaller than or equal to 1.

The second condition is imposed by using the following procedure. All the S-parameters S_{i1}, \dots, S_{iN} on the i -th row in the S-matrix are sorted in ascending order of magnitude. After sorting the S-parameters are numbered as $S_i\{1\}, \dots, S_i\{N\}$:

$$\text{sort}(S_{i1}, K, S_{iN}) \rightarrow S_i\{1\}, K, S_i\{N\} \quad (35)$$

where

$$\max |S_i\{j\}| \leq \max |S_i\{k\}| \quad \text{for } 1 \leq j \leq k \leq N \quad (36)$$

The S-parameter with the lowest magnitude $S_i\{1\}$ is approximated first, taking care that condition (32) is satisfied. Next, the other S-parameters $S_i\{j\}$ are approximated, first making sure condition (32) is satisfied and then taking care the following condition is satisfied:

$$P_i\{j\} = \sum_{k=1}^{j-1} |H_i\{k\}|^2 + |H_i\{j\}|^2 \leq 1 \quad (37)$$

If condition (37) is violated $H_i\{j\}$ is replaced by

$$\tilde{H}_i\{j\}(j\omega) = \frac{H_i\{j\}(j\omega)}{|H_i\{j\}(j\omega_M)|} \sqrt{1 - P_i\{j\}(j\omega_M)} \quad (38)$$

where ω_M represents the pulsation for which $P_i\{j\}$ reaches its maximum. The S-parameter with the largest magnitude is approximated the last. Following this procedure the passivity of the complete model is ensured.

3.4 SPICE model output format

Commercial SPICE simulators like HSPICE or PSPICE offer the ability to use a VCCS with a frequency-dependent gain (figure 3). In HSPICE there are three possible syntaxes to use such elements (also known as “G-elements”): the “Freq”, “Laplace” and “Pole” keywords. In PSPICE similar G-elements can be used (with the “Freq” and “Laplace” keywords), however with a slightly different syntax.

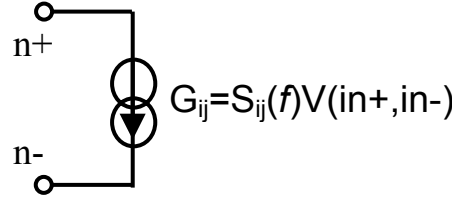


Figure 3: Voltage controlled current source with a frequency-dependent gain.

With the “Freq” keyword the gain is defined as a frequency look-up table, i.e. a table consisting of a number of frequencies and the magnitude and phase of the gain at each of the frequency points. We will not consider this approach here, since the resulting models can be very large if the tables contain a lot of frequency points (to get sufficient accuracy and resolution for the time domain waveforms), resulting in an increase in simulation time. Using the “Laplace” and “Pole” keywords an expression for the gain as a function of the frequency is specified, rather than specifying the value of the gain at specific frequencies. The “Laplace” and “Pole” keywords allow you to obtain more compact models, while not compensating on the desired resolution and accuracy. This translates into a faster simulation time when running the finished model.

With the “Laplace” keyword the coefficients of the numerator and denominator of the rational approximation functions are specified, i.e. the rational function

$$H(j\omega) = \frac{k_0 + k_1 j\omega + \mathbf{K} + k_m (j\omega)^m}{d_0 + d_1 j\omega + \mathbf{K} + d_n (j\omega)^n} \quad (39)$$

results in a G-element with the following syntax:

$$\text{Gxxx n+ n- laplace in+ in- } k_0, k_1, \dots, k_m / d_0, d_1, \dots, d_n \quad (40)$$

With the “Pole” keyword the poles and zeros of the rational function are specified, i.e. the rational function

$$H(j\omega) = \frac{a(j\omega + \alpha_{z1} - j2\pi f_{z1})\mathbf{K} (j\omega + \alpha_{zm} - j2\pi f_{zm})(j\omega + \alpha_{zm} + j2\pi f_{zm})}{b(j\omega + \alpha_{p1} - j2\pi f_{p1})\mathbf{K} (j\omega + \alpha_{pn} - j2\pi f_{pn})(j\omega + \alpha_{pn} + j2\pi f_{pn})} \quad (41)$$

results in a G-element with the following syntax:

$$\text{Gxxx n+ n- pole in+ in- } a, \alpha_{z1}, f_{z1}, \dots, \alpha_{zm}, f_{zm} / b, \alpha_{p1}, f_{p1}, \dots, \alpha_{pn}, f_{pn} \quad (42)$$

Numerically both alternatives are equivalent. The only thing that's different is the way the rational function is specified: using the coefficients of the numerator and the denominator or using the poles and zeros.

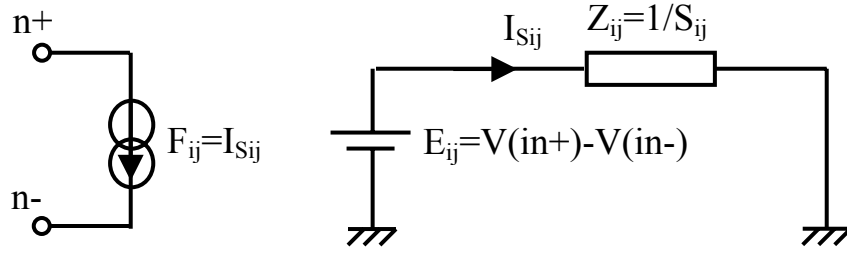


Figure 4: Equivalent circuit of a VCCS with a frequency-dependent gain.

In generic SPICE a VCCS with a frequency-dependent gain is not available. However, an equivalent circuit can be constructed that acts as a VCCS with a frequency-dependent gain (figure 4). This equivalent circuit has the same behavior at terminals 'n+' and 'n-' as the circuit shown in figure 3. In figure 4 the inverse of S_{ij} is synthesized as an impedance network. To calculate the values of the elements in the impedance network the poles, residues and direct term of S_{ij} are calculated. The poles appear as real poles or complex conjugate pole pairs:

$$\frac{1}{Z_{ij}}(j\omega) = S_{ij}(j\omega) = K + \sum_k \frac{G_k}{j\omega - P_k} + \sum_m \left(\frac{G_m}{j\omega - P_m} + \frac{G_m^+}{j\omega - P_m^+} \right) \quad (43)$$

Each of the terms on the right-hand side of equation (43) results in a smaller impedance network. These are connected in parallel to obtain the complete impedance network. The direct term K results in a resistor with resistance $R=1/K$. Each of the real poles results in a series connection of a resistor and an inductor, with the resistance given by $R_k=-P_k/G_k$ and the inductance given by $L_k=1/G_k$.

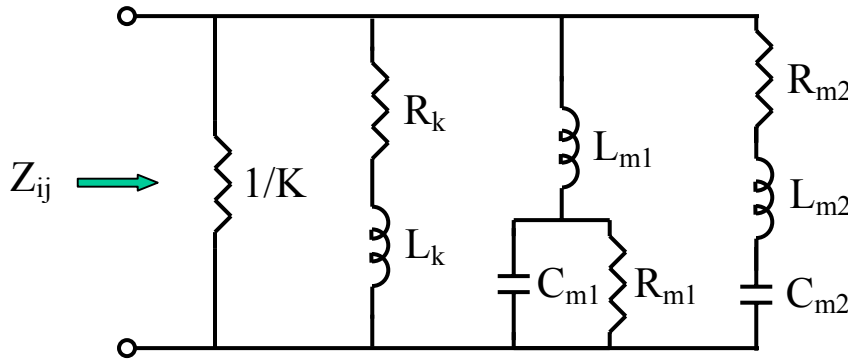


Figure 5: Schematic representation of impedance network Z_{ij} .

The complex pole pairs can be written as:

$$\frac{G_m}{j\omega - P_m} + \frac{G_m^+}{j\omega - P_m^+} = \frac{a_m + jb_m}{j\omega - u_m - jv_m} + \frac{a_m - jb_m}{j\omega - u_m + jv_m} = \frac{A_m + j\omega B_m}{D_m + j\omega C_m + (j\omega)^2} \quad (44)$$

and we can write:

$$\frac{A_m + j\omega B_m}{D_m + j\omega C_m + (j\omega)^2} = \frac{A_m + j\omega A_m/C_m}{D_m + j\omega C_m + (j\omega)^2} + \frac{j\omega(B_m - A_m/C_m)}{D_m + j\omega C_m + (j\omega)^2} \quad (45)$$

The left term on the right-hand side in equation (45) results in an inductor connected in series with a resistor and a capacitor in parallel. The inductance is given by $L_{m1}=C_m/A_m$, the resistance is given by $R_{m1}=D_m/A_m$ and the capacitance is $C_{m1}=A_m/(C_mD_m)$. The right term results in a series connection of an inductor, a resistor and a capacitor. The resistance is $R_{m2}=C_m^2/(B_mC_m-A_m)$, the inductance is $L_{m2}=-$

$C_m/(B_m C_m - A_m)$ and the capacitance is given by $C_{m2} = (B_m C_m - A_m)/(C_m D_m)$. Figure 5 shows a schematic representation of the impedance network Z_{ij} .

The disadvantage of using this equivalent network is that the complete SPICE model will contain a large amount of nodes, which will increase simulation time. The advantage is that these models can be used in any SPICE3 compatible SPICE simulator. Also, when using this type of model no inverse Fourier transform is involved (for determining the impulse response) and any numerical issues related to doing an inverse Fourier transform are avoided. Furthermore, this type of model is inherently causal, while the ‘‘Pole’’ and ‘‘Laplace’’ type models can generate non-causal results if the original S-parameter data is non-causal.

4. Example: Metral® 4000 connector

As an example of this SPICE model generation algorithm a SPICE model was generated for FCI’s Metral® 4000 connector. The Metral® 4000 connector was simulated using the 3D full-wave simulation tool HFSS [11]. The model included 5 rows (A to E) and 2 columns (figure 6). The column differential pairs are AB and DE. Row C is grounded. Figures 7 and 8 show a comparison between measured differential S-parameters and the results from HFSS. A good correlation can be observed between the measured and simulated S-parameters, both in the frequency and the time domain.

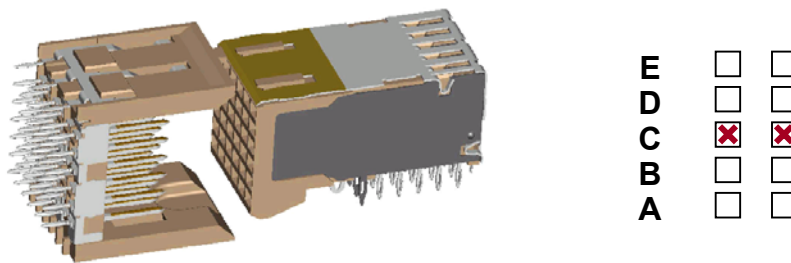


Figure 6: Metral® 4000 connector: view of the connector (left) and measured signal/ground configuration (right). The red crosses represent the ground pins.

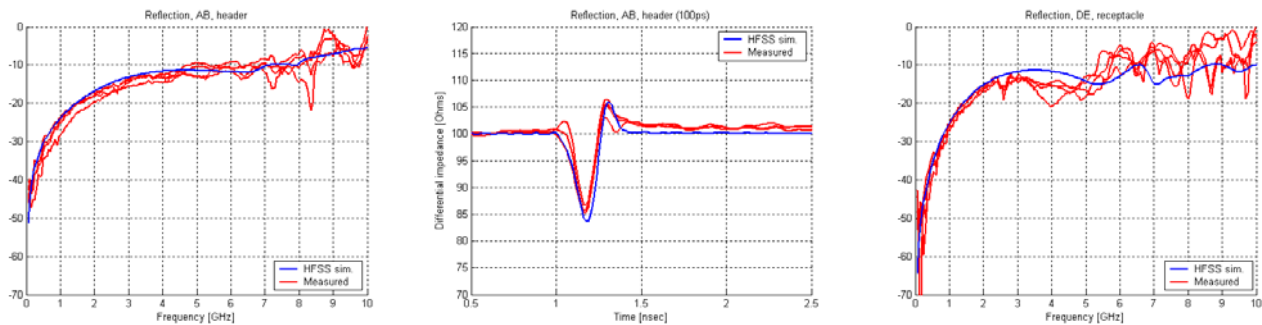


Figure 7: Comparison between measurements and simulations (HFSS) for the Metral® 4000 connector.

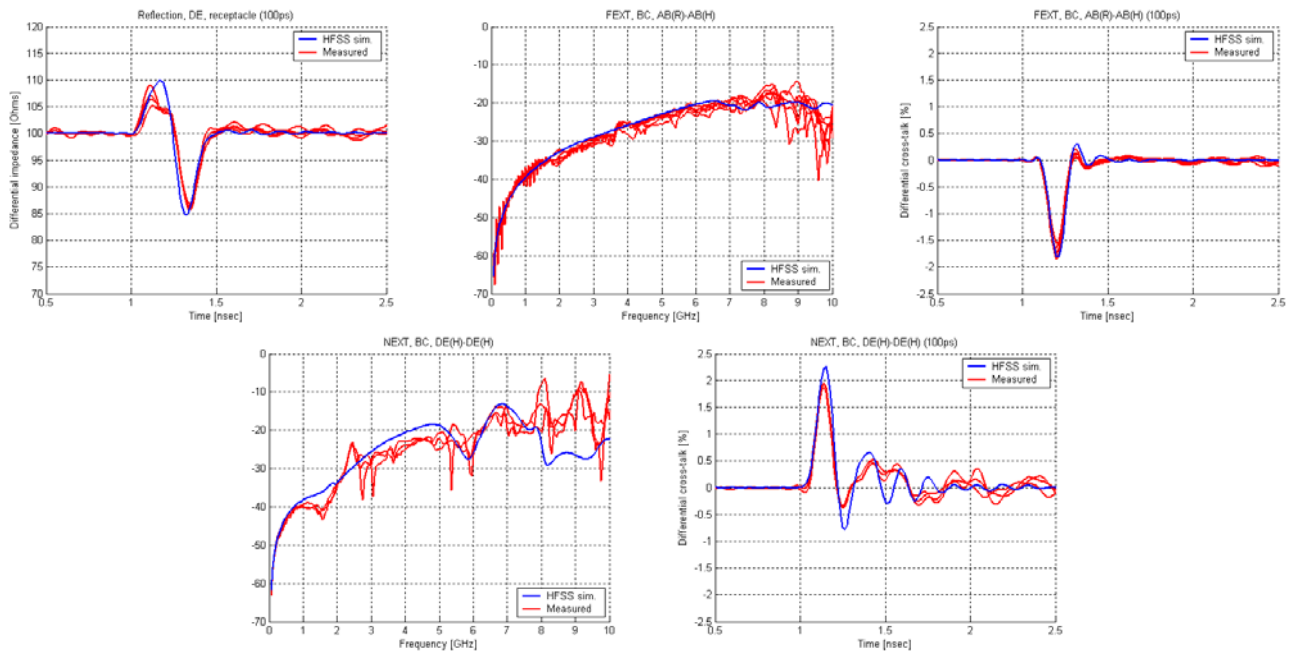


Figure 8: Comparison between measurements and simulations (HFSS) for the Metral® 4000 connector.

From the simulated single-ended S-parameters a 16-port SPICE model was generated that is valid from DC to 10 GHz. The rise time used for determining the optimal rational approximation functions was 35ps. Figures 9 to 13 show a comparison between the frequency domain and the time domain responses (i.e. the step response for a 100ps and a 35 ps input voltage step) from the SPICE model and from the original S-parameters, for a number of differential pairs. A very good agreement between the measured S-parameters and the SPICE model can be observed. Figure 14 shows the power dissipation $P_i(j\omega)$ (see equation (33)) at each of the ports. From this figure the passivity of the SPICE model can be verified.

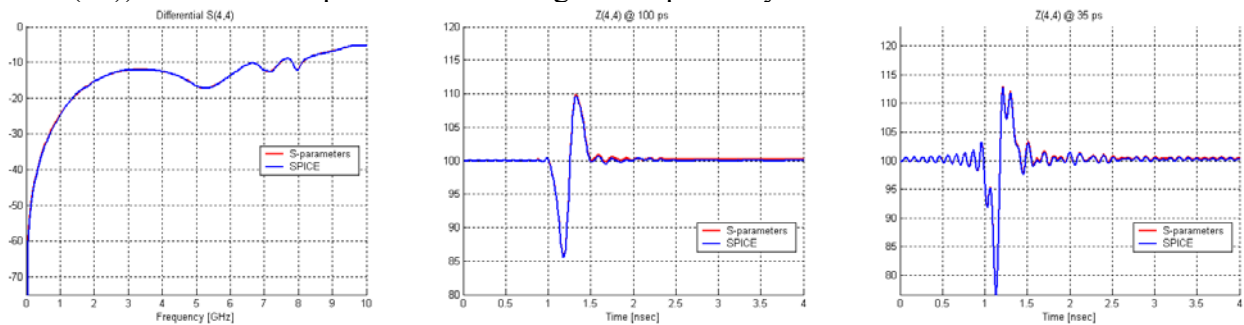


Figure 9: Metral® 4000 connector, return loss DE, header side.

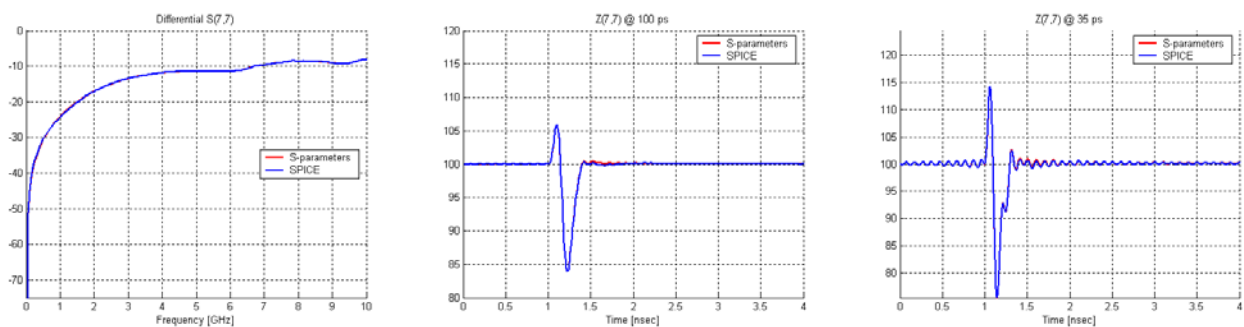


Figure 10: Metral® 4000 connector, return loss AB, receptacle side.

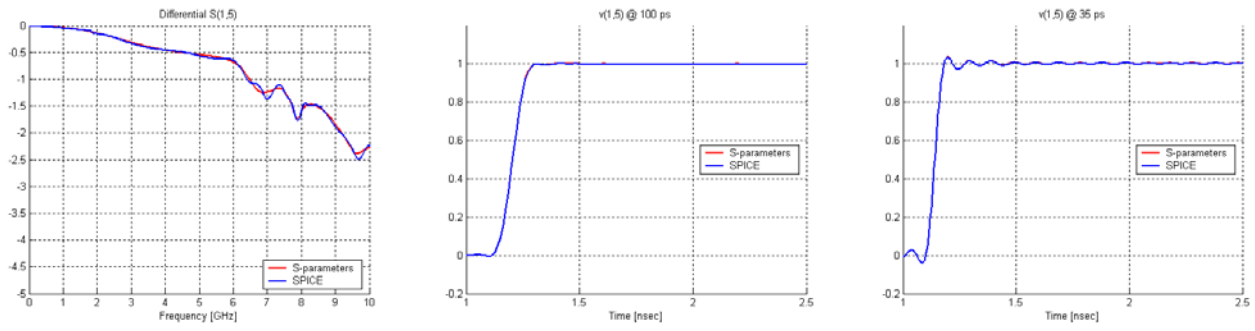


Figure 11: Metral® 4000 connector, insertion loss AB.

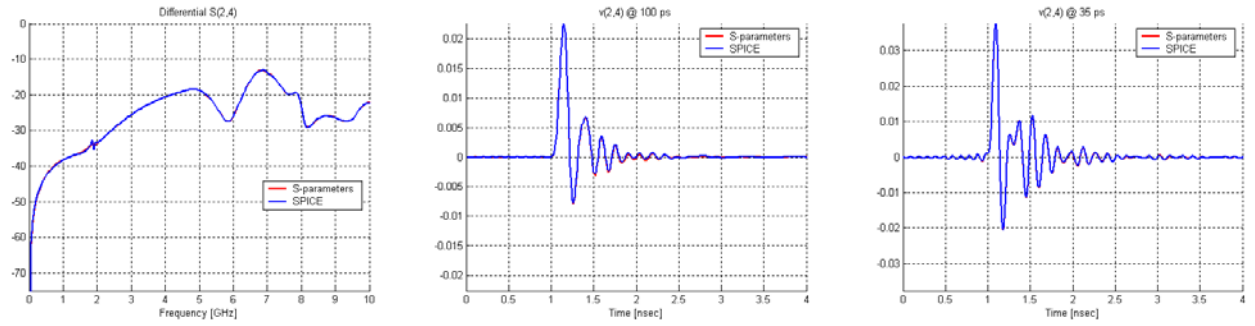


Figure 12: Metral® 4000 connector, NEXT DE-DE, header side.

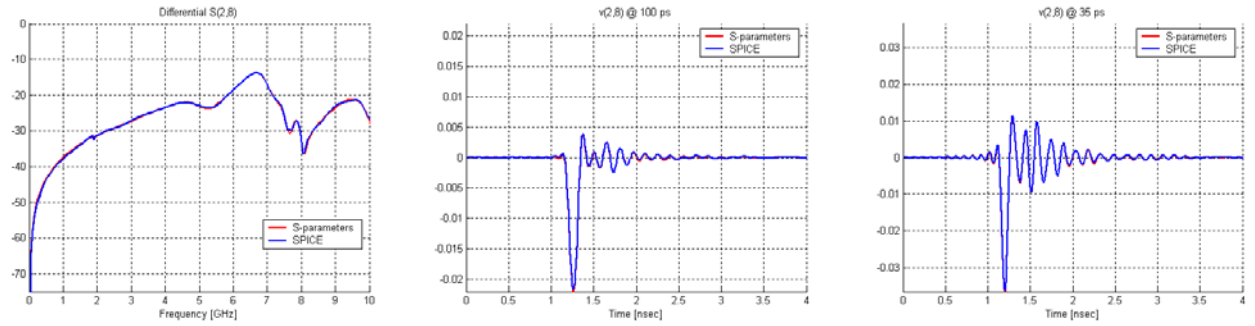


Figure 13: Metral® 4000 connector, FEXT DE-DE.

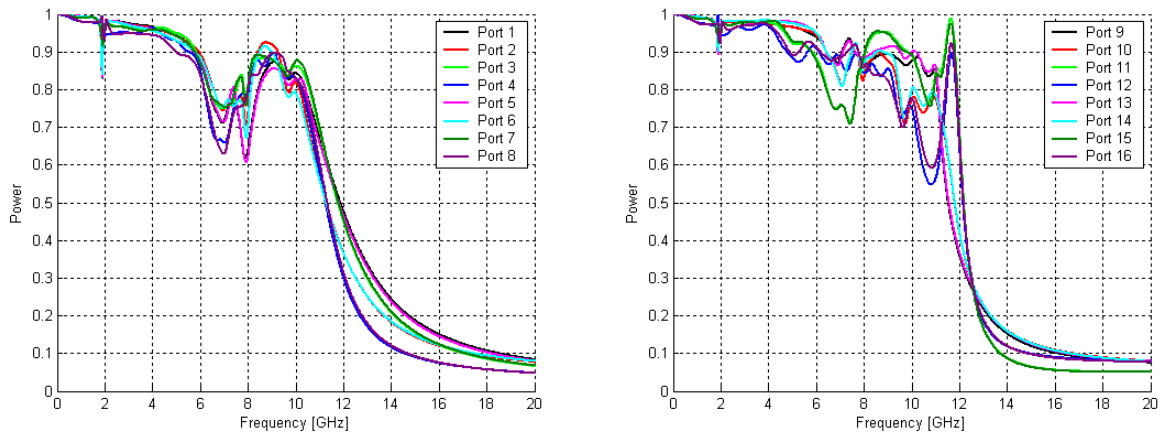


Figure 14: Metral® 4000 connector, power dissipation in SPICE model.

5. Example: AirMax VS™ connector

As a second example a SPICE model is generated for FCI's new AirMax VS™ connector (figure 15). First a description of the connector is given. Next, numerical results are shown comparing the performance of the SPICE model with the original S-parameter model, both in the frequency domain and the time domain. Finally, link simulation results are given, showing the performance of the model in an actual link. The performance of the SPICE model is compared with other models, with respect to accuracy, model size and simulation time.

5.1 AirMax VS™ description

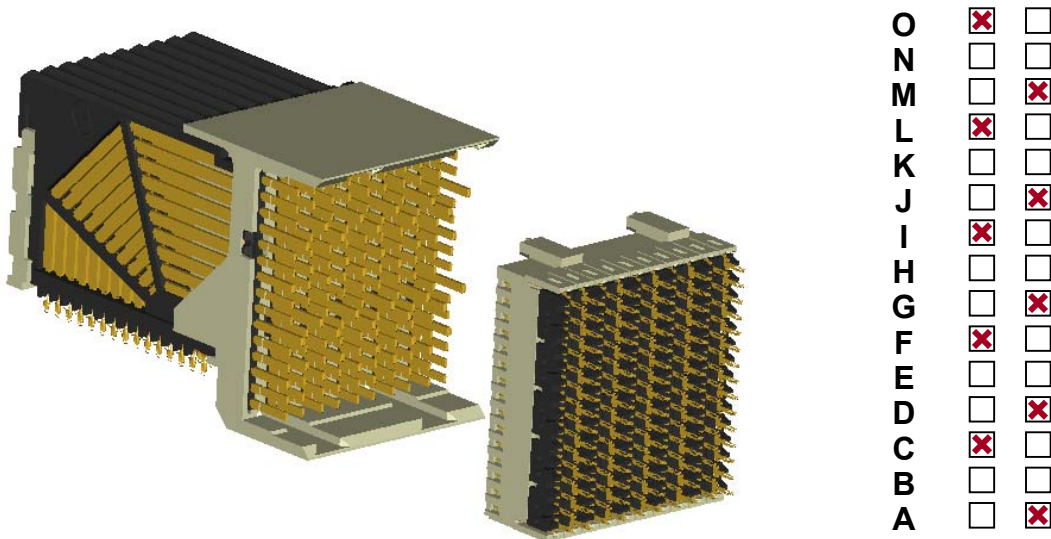


Figure 15: AirMax VS™ connector: view of the connector (left) and measured signal/ground configuration (right). The red crosses represent the ground pins.

The AirMax VS™ connector is a new connector system conforming to Hard Metric equipment practices. The connector is a scalable, inverse 2-piece backplane connector (with a vertical receptacle on the backplane and a right-angled header on the daughter card) based upon an IMLA (Insert Molder Leadframe Assembly) design that can be used for either differential pair signals, single-ended signals or power – all within the same IMLA.

The column differential pairs demonstrate low insertion loss and low cross-talk from speeds less than 2.5 Gb/s to greater than 12.5 Gb/s. The differential insertion loss is less than 0.5 dB through 6.25 Gb/s and less than 2.0 dB through 20 Gb/s. The differential impedance is $100 \pm 5 \Omega$ for a 20-80% risetime of 55ps. The differential near-end and far-end multi-active cross-talk are less than 2.5% and 3% respectively (again for a 20-80% risetime of 55ps). The single-ended insertion loss is less than 0.5 dB through 3.125 Gb/s and less than 2.0 dB beyond 6.25 Gb/s. The even-mode impedance is 60Ω nominally. The single-ended near-end and far-end multi-active cross-talk are less than 9% and 3% respectively (for a 20-80% risetime of 150ps).

The “VS” in the AirMax VS™ connector stands for “virtual shields”, as this connector design requires no interleaving shields. Elimination of the shields results in low cost, low weight (BGA advantage) and easy PCB routing. The connector utilizes edge-coupling between contacts to control the impedance of each differential pair, and it includes an optimized stagger or offset in the relative position of the contacts between the columns to minimize cross-talk both in the connector and in the via field of the connector footprint. It achieves very low loss by maximizing the use of air dielectric and minimizing the use of plastic.

5.2 AirMax VS™ SPICE model

A SPICE model was generated for the AirMax VS™ connector based on measured S-parameters. Measurements were done for 15 rows (A to O) and two columns, resulting in 40 single-ended ports in the model (figure 15). Measurement data was gathered from DC to 10.05 GHz with a 0.05 GHz step (202 data points). Rational approximation functions were calculated for the single-ended S-parameters. A risetime of 35ps was used for the weighting function (equation (30)). Figures 16 to 19 show a comparison between the measurements and the rational approximation for some of the single-ended S-parameters, both in the frequency domain and in the time domain (i.e. again the step response for a 100ps and a 35ps input voltage step). These figures show a good agreement between the original measured S-parameters and the S-parameters obtained with the rational approximation functions.

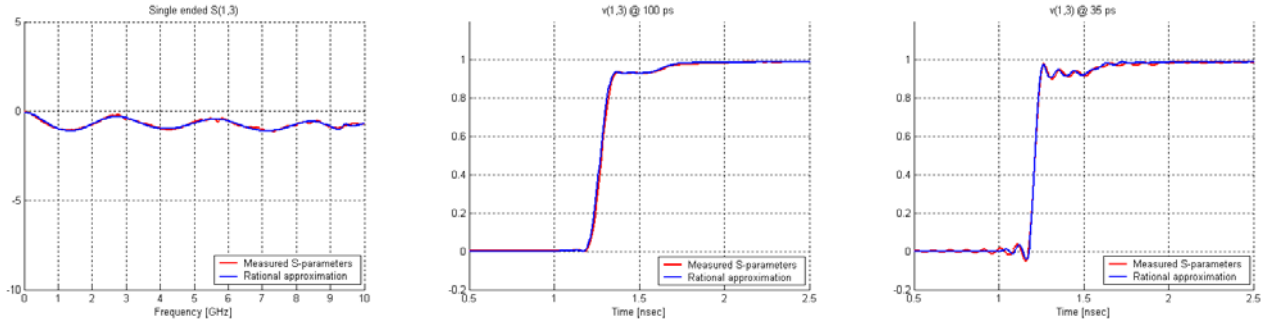


Figure 16: AirMax VS™ connector, single-ended insertion loss pin N column 2.

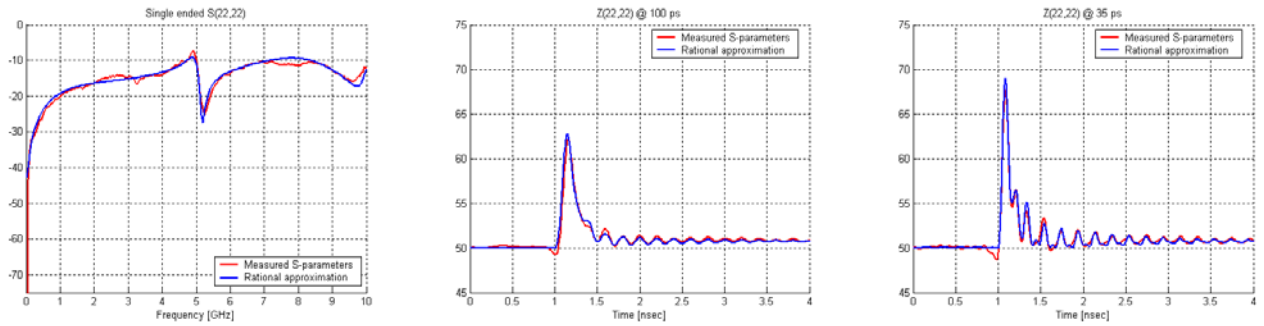


Figure 17: AirMax VS™ connector, single-ended return loss pin C column 2, receptacle side.

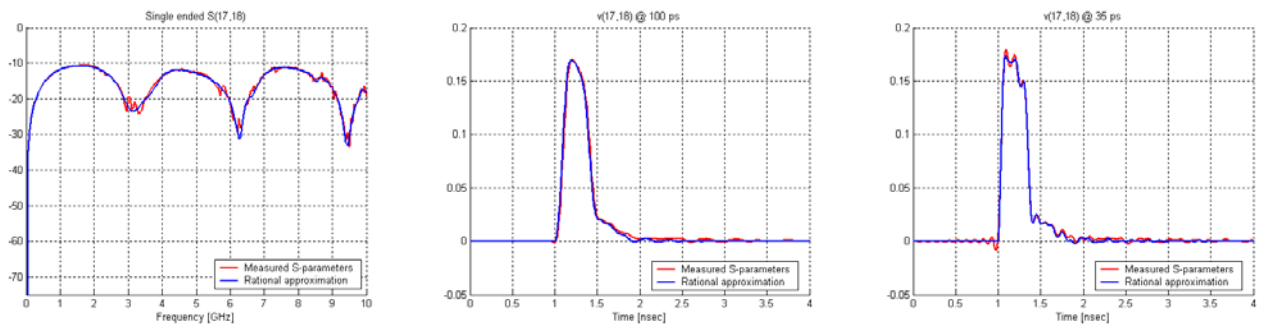


Figure 18: AirMax VS™ connector, single-ended NEXT pin K column 2 to pin L column 2, header side.

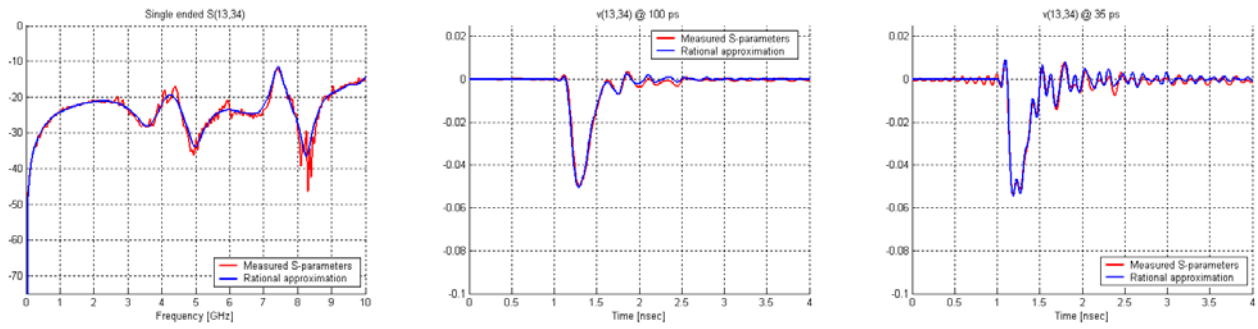


Figure 19: AirMax VS™ connector, single-ended FEXT pin E column 2 to pin F column 2.

Figures 20 to 27 show a comparison between the differential S-parameters calculated from the measurements and the differential S-parameters calculated using the approximation functions. Again, there is a good agreement between the measurements and the results from the rational approximation functions.

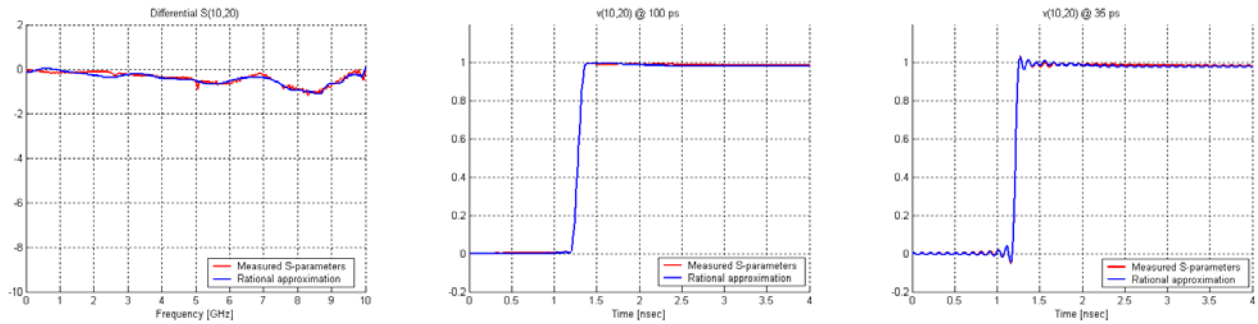


Figure 20: AirMax VS™ connector, differential insertion loss pair NO column 2.

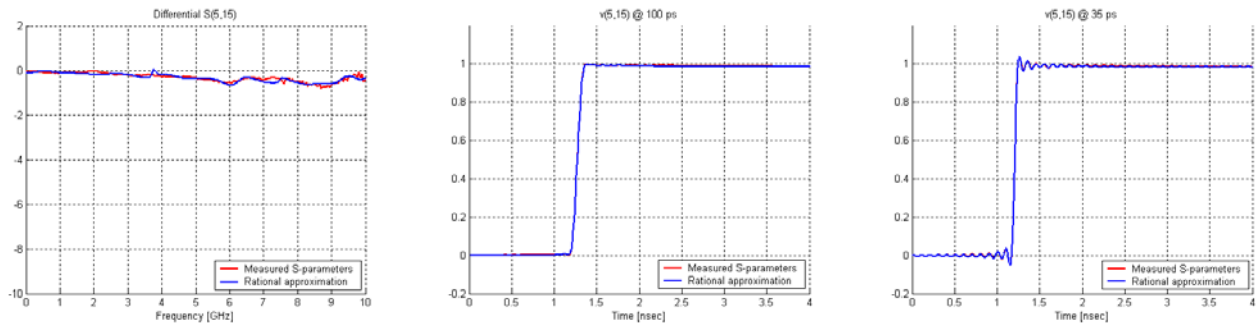


Figure 21: AirMax VS™ connector, differential insertion loss pair MN column 1.

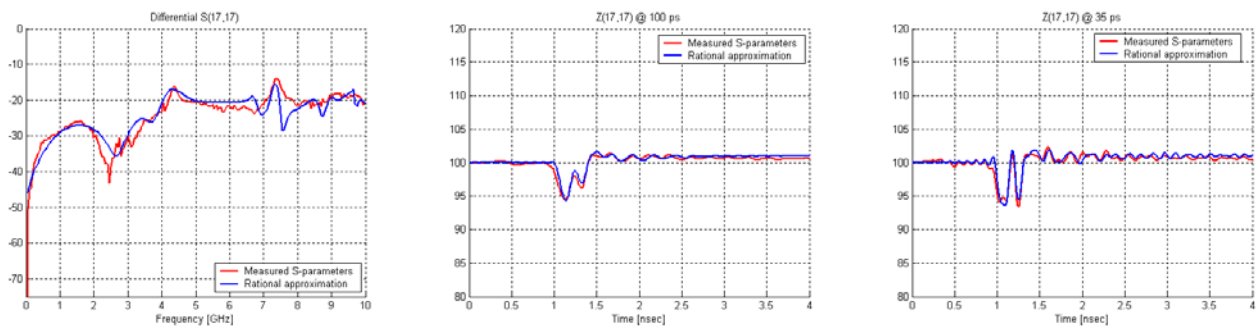


Figure 22: AirMax VS™ connector, differential return loss pair EF column 2, receptacle side.

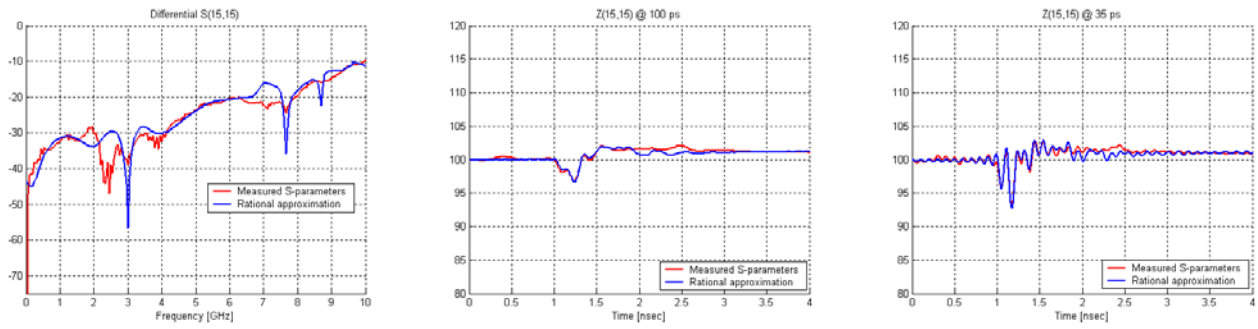


Figure 23: AirMax VS™ connector, differential return loss pair MN column 1, receptacle side.

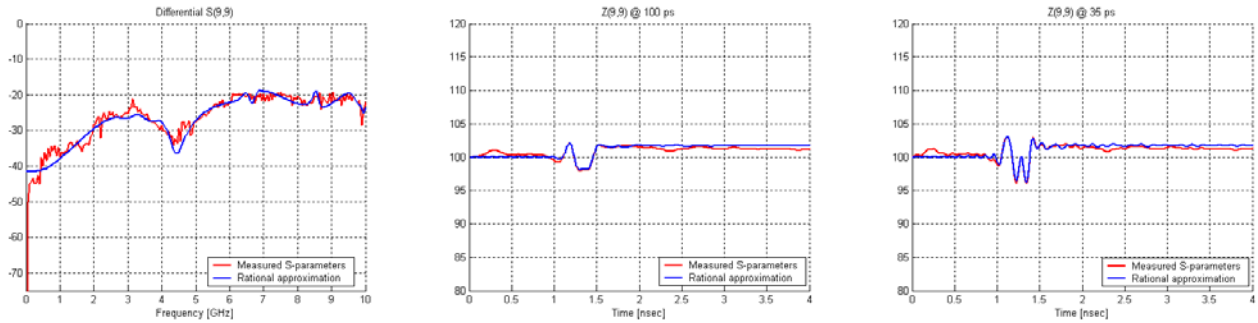


Figure 24: AirMax VS™ connector, differential return loss pair KL column 2, header side.

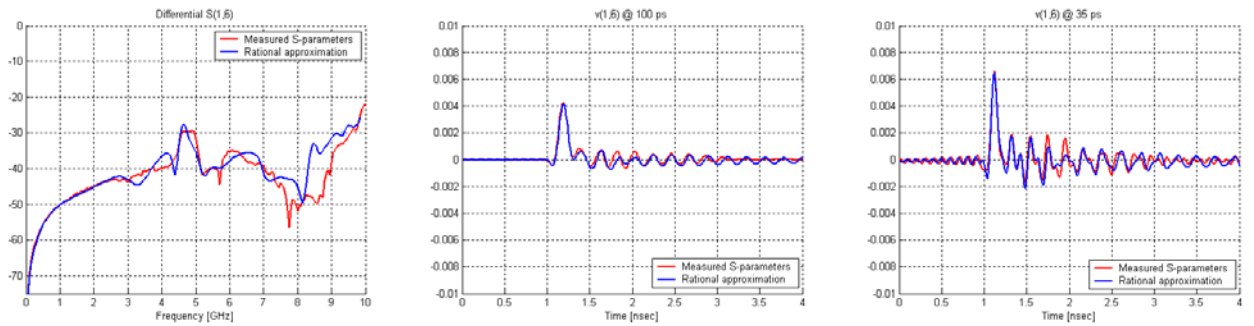


Figure 25: AirMax VS™ connector, differential NEXT pair AB column 1 to pair BC column 2, header side.

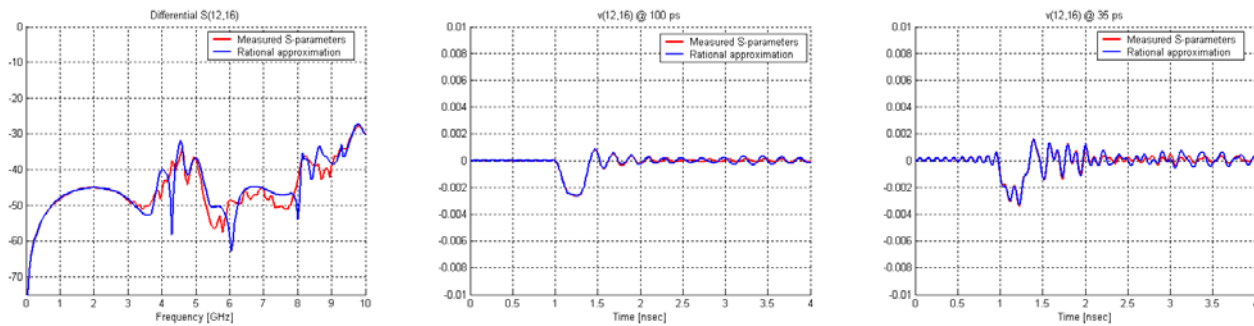


Figure 26: AirMax VS™ connector, differential NEXT pair DE column 1 to pair BC column 2, receptacle side.

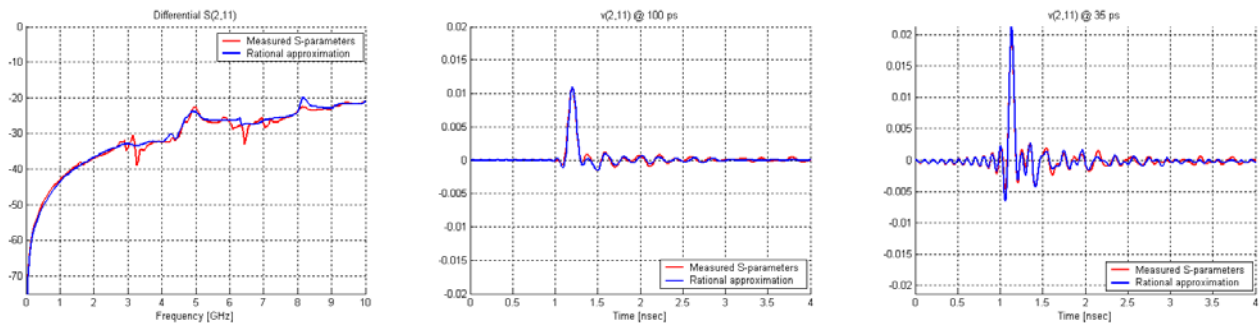


Figure 27: AirMax VSTM connector, differential FEXT pair DE column 1 to pair AB column 1.

The AirMax VSTM SPICE model was generated from measured S-parameters. Measurements suffer from measurement noise that will aggravate the approximation process. Furthermore, due to measurement errors and numerical issues in the de-embedding process the measured S-parameter model can become active, making it more difficult to satisfy the physical properties of the SPICE model. Table 1 shows the size of the SPICE model files and the number of nodes and the number of elements in the SPICE model, for different output formats. The model type indicated as “generic” is the model that makes use of the impedance network shown in figure 5, as opposed to using voltage controlled current sources like the “pole” and “laplace” models. The listed simulation time is the time to run a transient simulation from 0 to 5ns with a time step of 0.005ns. The simulations were done in HSPICE.

Model type	Model size	#Elements	#Nodes	Sim. Time
pole	382 kB	1150	200	12 sec.
laplace	706 kB	7600	200	45 sec.
generic	511 kB	38500	19100	565 sec.

Table 1: Model size, number of elements, number of nodes and simulation time for different SPICE model types.

5.3 Link simulations

The generated SPICE model was used in a backpanel link simulation. The backpanel link consists of a component board, an AirMax VSTM connector, an FR4 backplane, a second AirMax VSTM connector and a second component board (figure 28). The component boards are terminated with SMA connectors.

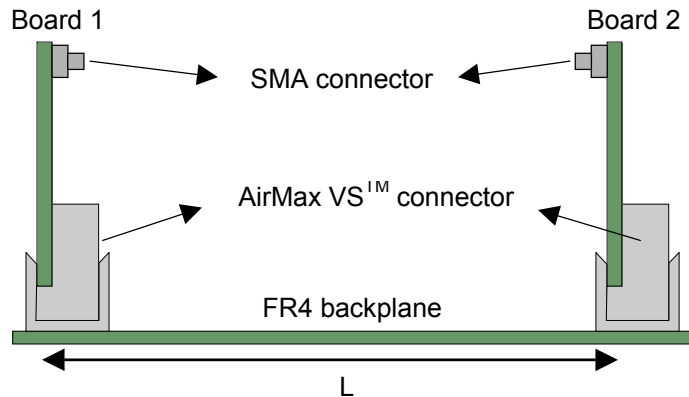


Figure 28: Schematic representation of the simulated backpanel link.

An HSPICE simulation was performed for a differential pair going through pair EF in the first connector and through pair KL in the second connector. The unused pins were terminated with 50 Ω terminations. The length of the traces on the backplane was 50 cm. For the SMA’s, the via holes and the boards,

simulated S-parameter models were used (using the “S-element” in HSPICE). For the AirMax VSTM connectors two different types of SPICE models were used. The first SPICE model was the full 40-port SPICE model described in the previous section. In this case the same model was used for both AirMax VSTM connectors. The second SPICE model was a 4-port single-pair model. This was obtained by removing all ports corresponding to the unused pins from the original full model. In this case a 4-port model for pair EF was used for the first connector, while for the second connector a 4-port model for pair KL was used.

Figure 29 shows a comparison between the measured and the modeled S-parameters.

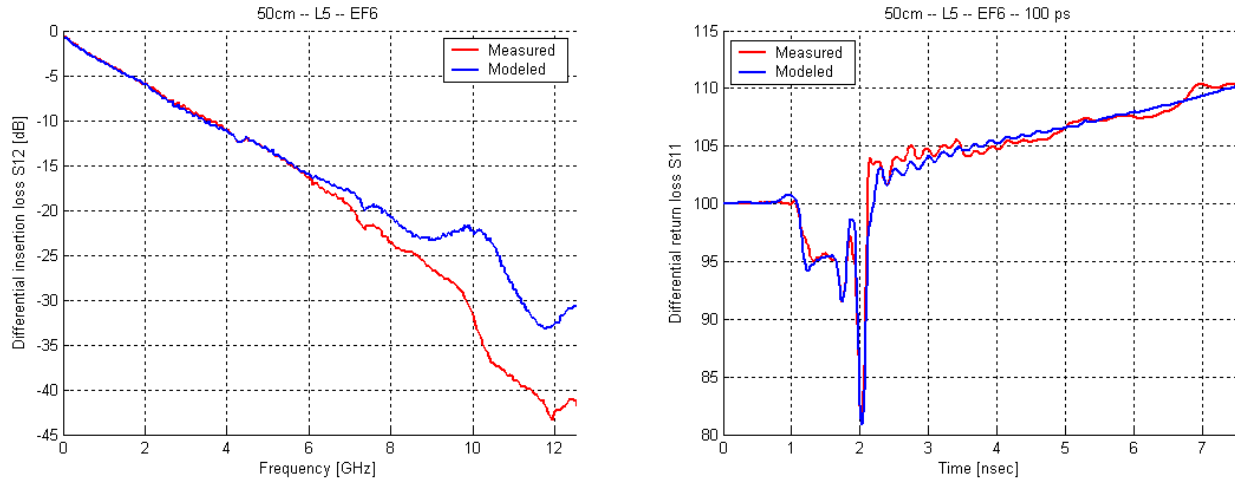


Figure 29: Measured vs. modeled S-parameters: differential insertion loss (frequency domain) and differential return loss (time domain).

Eye-pattern simulations were performed for a 6.25 Gb/s PRBS. No signal conditioning was applied. Figure 30 shows the eye-patterns in case the single-pair SPICE models are used. Figure 31 shows the eye-patterns in case the full SPICE model is used. Table 2 summarizes the simulation times and the eye openings for the different connector SPICE models. The listed simulation times are for a transient simulation from 0 to 40 ns in 2.5 ps steps (i.e. 16000 time points).

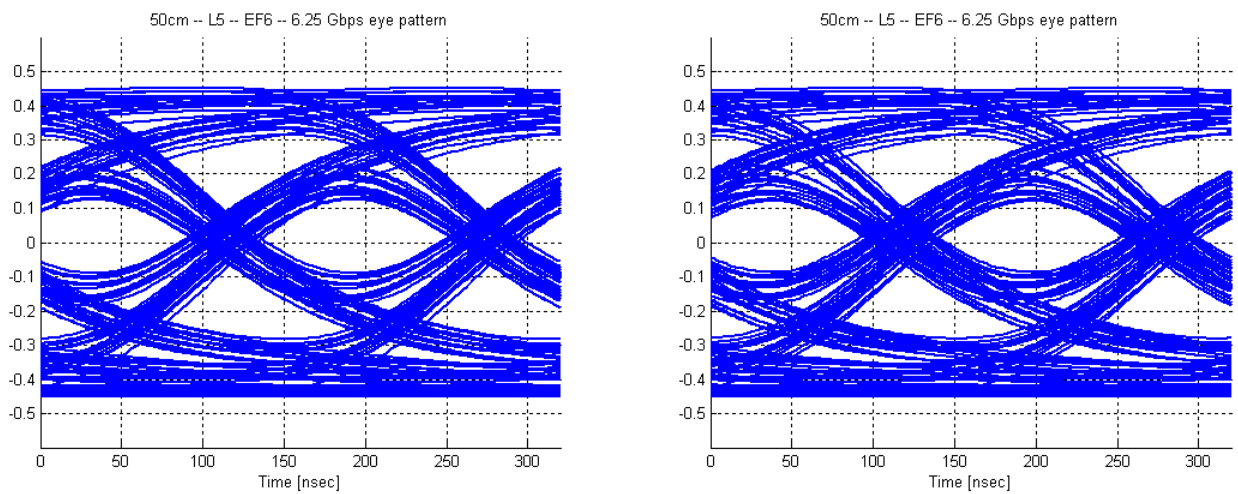


Figure 30: 6.25 Gb/s eye-pattern simulations without signal conditioning, using single-pair connector SPICE models: “pole” type SPICE model (left) and “generic” type SPICE model (right).

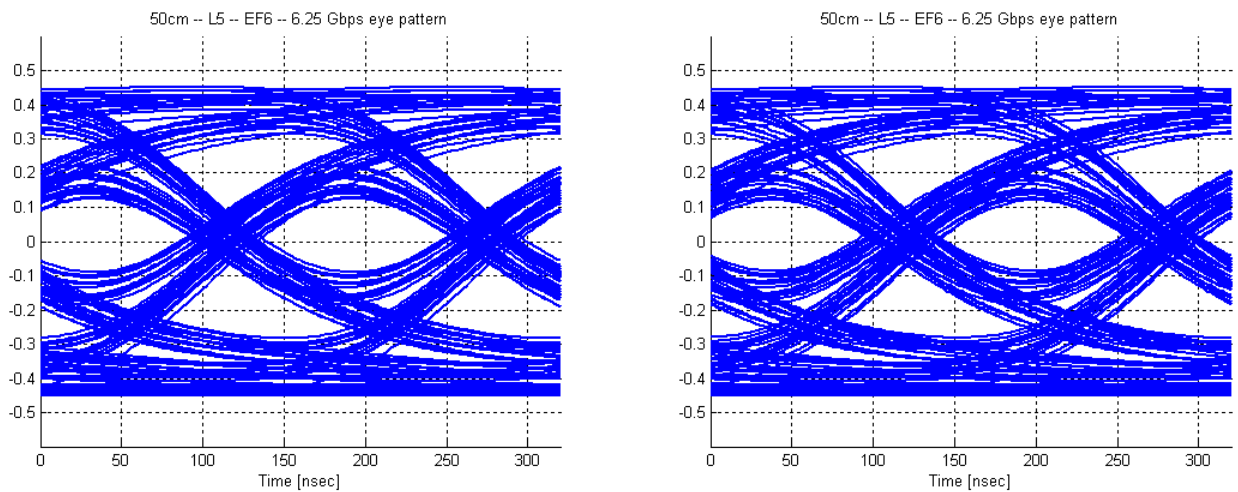


Figure 31: 6.25 Gb/s eye-pattern simulations without signal conditioning, using full connector SPICE model: “pole” type SPICE model (left) and “generic” type SPICE model (right).

Conn. SPICE Model Type	Sim. Time	Eye Height	Eye Width
Single-pair models, pole	3 min 42 sec	215 mV	115 ps
Single-pair models, generic	6 min 26 sec	208 mV	113 ps

Full model, pole	11 min 4 sec	215 mV	115 ps
Full model, generic	3 h 8 min	208 mV	113 ps

Table 2: Simulation times and eye openings (without signal conditioning) for the different connector SPICE models.

The eye openings calculated using the single-pair models and those calculated using the full model are exactly the same, both for the “pole” and the “generic” type models. Naturally, the simulation time in case of the single-pair models is a lot shorter than that in case of the full SPICE model. The eye openings calculated using the “pole” type models and those calculated using the “generic” type models are slightly different. Because of the higher number of elements and nodes in the “generic” type model (see table 1), simulation times for this type of model are longer than for the “pole” type models.

6. Conclusions

A method was proposed for generating broad-band SPICE models from measured or simulated S-parameter data, using rational approximation techniques and SPICE macro-modeling capabilities. The resulting SPICE models are passive and stable. SPICE models were generated using measured S-parameters of FCI’s new AirMax VSTM connector and using simulated S-parameters of the Metral® 4000 connector. The accuracy of the SPICE models in both the time domain and frequency domain has been demonstrated.

7. References

- [1] T. Dhaene, “Circuit modeling, transient simulation and time domain characterization of high-speed interconnection structures”, PhD thesis, Department of Information Technology (INTEC), University of Gent, 1994.
- [2] P.A. Perry and T.J. Brazil, “Forcing Causality on S-Parameter Data Using the Hilbert Transform”, IEEE Microwave and Guided Wave Letters, vol. 8, no. 11, pp. 378-380, November 1998.
- [3] T.J. Brazil, “Causal-Convolution – A New Method for the Transient Analysis of Linear Systems at Microwave Frequencies”, IEEE Trans. on Microwave Theory and Techniques, vol. 43, no. 2, pp. 315-323, February 1995.
- [4] A. Maxim, D. Andreu, M. Cousineau and J. Boucher, “A Novel SPICE Behavioral Macromodel of Operational Amplifiers Including a High Accuracy Description of Frequency Characteristics”, Proceedings of the International Symposium on Circuits and Systems (ISCAS), Orlando, Florida, May 1999.
- [5] S. Sercu, “Experimental characterization and circuit modeling of interconnections”, PhD thesis, Department of Information Technology (INTEC), University of Gent, 2001.
- [6] C.P. Coelho, J.R. Phillips and L.M. Silveira, “Robust Rational Function Approximation Algorithm for Model Generation”, Proceedings of the 36th Design Automation Conference, pp. 207-212, New Orleans, June 1999.
- [7] T.V. Nguyen and J.Li, “Multipoint Padé Approximation using a Rational Block Lanczos Algorithm”, Proceedings of the International Conference on Computer-Aided Design (ICCAD), pp. 72-75, San Jose, CA, November 1997.
- [8] S. Grivet-Talocia, “Package Macromodeling via Time-Domain Vector Fitting”, IEEE Microwave and Wireless Components Letters, vol. 13, no. 11, pp. 472-474, November 2003.
- [9] C.P. Coelho, J.R. Phillips and L.M. Silveira, “A Convex Programming Approach to Positive Real Rational Approximation”, Proceedings of the International Conference on Computer-Aided Design (ICCAD), pp. 245-251, San Jose, CA, November 2001.
- [10] S. Grivet-Talocia, “Enforcing Passivity of Macromodels via Spectral Perturbation of Hamiltonian Matrices”, Proceedings of the 7th IEEE Workshop on Signal Propagation on Interconnects, Siena, Italy, May 2003.

[11] HFSSv9, Ansoft Corporation.