SPICE LINE 2.23
with Telecom Line Predictor

User's Guide

March 2000

This document includes 39 pages.

TABLE OF CONTENTS

1. SpiceLine and the SpiceLine User's guide

1.1 SpiceLine

SpiceLine 2.23 with Telecom Line Predictor (usually called "SpiceLine" later on) is an easy-to-use software for multiconductor transmission line modeling. It can be implemented with any Spice 3F.2 compatible simulator, but it is best suited for the ICAP/4 Windows version 7.x simulation package edited by Intusoft, because of the parameter passing syntax choosen in SpiceLine.

SpiceLine is edited by Excem, a company specialized in EMC and in the design and simulation of electronic circuits. The copyrights of SpiceLine are owned by Excem, and SpiceLine is a registered trademark of Excem.

1.2 This User's guide, copyrights, trademarks

This User's guide contains most information needed to understand how to use SpiceLine, and what are the definitions we use for the parameters describing the multiconductor transmission line problem. It contains a theoretical presentation and implementation examples. Another document, the Reference manual, contains a detailed description of every features of SpiceLine.

Note that this User's Guide is not a tutorial on EMC nor on multiconductor line modeling, and we will refer to the available litterature, when appropriate. Numbers between square brackets, like [1] indicate one of the references listed at § 5.

Excem
12, chemin des Hauts de Clairefontaine,
78580 Maule
France

tel: 33 1 34 75 13 65
fax : 33 1 34 75 13 66
e-mail : info@excem.fr
site : excem.fr

Please note that :

- "Intusoft" and "ICAP/4" are trademarks of Intusoft,
- "Adobe" and "Acrobat" are trademarks of Adobe Systems Incorporated,
- "Microsoft" and "Windows" are trademarks of Microsoft Corporation,
- "SpiceLine" is a trademark of Excem société anonyme.

1.3 Terms of sale, warranty, license for the SpiceLine software

Unless otherwise specified in your contract with Excem, our "General Terms of Sale" applicable at the date of the invoice shall apply to the SpiceLine Software. These terms of sale ("conditions générales de vente") are printed on the back of the invoice. They include the definition and the limitation of warranty.

This software is supplied "As Is". However Excem will make its best efforts to solve any problem related to a malfunction of the software.

The Licence is subject to Excem's "General Terms of Sale". The Licensee may use the SpiceLine software on a number of computers equal to the number of copies purchased.

Please note that the software licences are NOT transferable.

1.4 Updates and printing history

Each technical document emanating from Excem carries a single and individual document number. This number is printed on the left of the bottom line of each page. Please keep track of additional relevant documents pertaining to your SpiceLine software package.

The "User's guide" dated November 1996, document 95104301B is cancelled,
The "User's guide" dated July 1998, document 98042103B is cancelled,
The "User's guide Update" dated July 1999, document 99072101A is cancelled,
The "Reference Manual" dated March 2000, document 00012108B is applicable to SpiceLine 2.23.

....
....
....
....
....
....
....
....
....
....
....
....

2. What is SpiceLine ?

2.1 A software for emission, immunity and crosstalk predictions

SpiceLine is intended for computing emission, immunity and crosstalk of multiconductor transmission lines. Each problem starts with the description of the multiconductor transmission line with two or three matrices of suitable per-unit-length parameters. SpiceLine then generates a SPICE model of the transmission line.

This SPICE model can be used directly in a SPICE simulator for frequency domain simulation (.AC simulation in the SPICE wording) or time domain simulation (.TRAN) of crosstalk.

From the results of an appropriate frequency-domain simulation (the *.OUT file), SpiceLine is also able to compute the frequency-domain emission by radiation of the transmission line, considering type 1 coupling and type 2 coupling.

Immunity of a system to radiated disturbances can also be assessed from a frequency domain simulation.

2.2 Description of the transmission line and crosstalk predictions

SpiceLine 2.23 with Telecom Line Predictor is able to compute SPICE compatible models for lossless multiconductor transmission lines from line inductance and capacitance matrices provided by the user. SpiceLine is also able to include some losses in those lossless model, according to an empirical model. If one wishes to include losses, a resistance matrix must be provided.

SpiceLine is able to generate 4 different lossless SPICE models for any given transmission lines :
- **L-Model** : implements lumped inductors, coupling between inductors, and capacitors ;
- **M-Model** : implements lumped inductors and capacitors, and B-elements ;
- **D-Model** : implements lossless transmission lines and B-elements ;
- **E-Model** : implements lossy transmission lines and B-elements.

where “B-element” refers to the non-linear dependant source of the Spice 3F.2 syntax. Losses can also be included in any of those models (i.e. the resistance matrix can be taken into account).

The inductance, capacitance and resistance matrices are considered frequency independant.

The models created by SpiceLine can be copied into any existing or new SPICE library. If the models are meant to be used with ICAP/4 Windows, no further editing is required, because SpiceLine and ICAP implement the same parameter passing syntax for the line length (in meter). In addition to the models themselves, SpiceLine users have also access to some additional information on the computed quantities related to the eigenmodes of the line : the propagation velocities vector, the transfer matrices, the characteristic impedance matrix...

The Telecom Line Predictor of SpiceLine offers several important additional features :
- Entering line parameters (inductance, capacitance, resistance) not referenced to ground, by accepting user-defined change of variables for inputs,

▪ Modifying the line parameters with a model suitable for taking into account a modification of a shielded cable height above a ground plane.

This Telecom Line Predictor is what is needed for a proper treatment of multipair cables.

The SPICE models so obtained can be used for .AC or .TRAN SPICE simulations, using accurate circuit models of the transmission line terminations.

2.3 Capabilities of SpiceLine: emission predictions

Assuming the transmission line is a cable or bundle installed above an ideal ground plane, if a suitable change of variables was implemented it is possible to compute the radiated emission of the line at a given point in space.

This capability is offered for .AC SPICE simulations only: non-linear behaviours of terminations are therefore not taken into account.

The computation takes into account type 1 coupling (i.e. common mode current on the cable), and type 2 coupling (i.e. common mode voltage). Higher order coupling phenomena are not taken into account. The radiated fields computed by SpiceLine are valid in the near field.

Cable with one or several screens are taken into account as far as the screen is well described by the frequency-independant inductance and resistance matrices. Of course, the effect of various screen grounding techniques can be easily assessed. The decrease in radiated emission resulting from the twisting of pairs is also taken into account in the kind of model implemented.

2.4 Capabilities of SpiceLine: immunity predictions

Assuming the transmission line is a cable or bundle installed above an ideal ground plane, it is also possible to compute the voltage and current induced on the transmission line by external field sources at a given point in space.

The source can be an infinitesimal electric dipole or an infinitesimal magnetic dipole. The induced voltage and currents are given for 3 different orthogonal orientations, so that the polarisation of sources is taken into account.

In fact the computation method used for immunity is based on reciprocity and a computation of emission. For this reason, the characteristics of emission and immunity calculation done by SpiceLine are the same. For instance, immunity predictions are offered for .AC SPICE simulations only. As is the case for the computation of emission by radiation, the computation takes into account type 1 coupling, and type 2 coupling. Also, the induced voltages and currents computed by SpiceLine are valid in the near field of the source.

3. Theory of operation

3.1 The inductance, capacitance and resistance matrices

SpiceLine is meant for the simulation of multiconductor cables having *n* conductors. A reference conductor being always implied, the problem is in fact a *n* +1 conductor transmission line.

The key input parameters of the SpiceLine software are the inductance matrix **L** and the capacitance matrix **C** (eventually the resistance matrix **R**) of a transmission line, with respect to a reference conductor, for instance a ground plane. They are square $n \times n$ matrices, containing per-unit-length quantities respectively in H/m, F/m and Ω/m. All voltages are considered with the reference conductor as ground reference. The properties and limitations of such a transmission line model are widely presented in the literature, and will not be discussed here. Let us only note that the conductance matrix **G** of the transmission line is assumed negligible, and that the **L**, **C** and **R** matrices are assumed frequency independant.

These matrices may be directly entered by the user. Alternatively, the user will enter other matrices (see § 3.3, § 3.4 and § 3.5) and SpiceLine will compute the **L**, **C** and **R** matrices. In any case, the matrices entered by the user are called “input matrices”, and the **L**, **C** and **R** matrices finally used to create the transmission line model will be the “output matrices”.

The output matrices must obey a certain number of basic properties. They can be verified by SpiceLine. For the **L** matrix these properties are :

- the **L** matrix is square of size *n* ;
- the **L** matrix is symetric :

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad L_{ij} = L_{ji} \tag{1}$$

- all coefficients of **L** are positive :

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad L_{ij} \ge 0 \tag{2}$$

- **L** is the matrix of a positive definite forme, therefore :

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad L_{ii} \cdot L_{jj} > L_{ij}^2 \tag{3}$$

and, if we denote $(L)_i$ the submatrix of **L** containing only the lines 1 to *i* and the columns 1 to *i* we have for the jacobi determinants :

$$\forall i \quad 1 \le i \le n \quad \left|(L)_i\right| > 0 \tag{4}$$

For the **C** matrix these properties are :

- the **C** matrix is square of size *n* ;
- the **C** matrix is symetric :

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad C_{ij} = C_{ji} \tag{5}$$

▪ the diagonal coefficients of **C** are positive while the non diagonal coefficents are negative :

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad C_{ii} \ge 0 \quad \text{and} \quad i \ne j \Rightarrow C_{ij} \le 0 \tag{6}$$

▪ **C** is the matrix of a positive definite forme, therefore:

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad C_{ii} \cdot C_{jj} > C_{ij}^2 \tag{7}$$

and, if we denote $(C)_i$ the submatrix of **C** containing only the lines 1 to i and the columns 1 to i we have for the jacobi determinant :

$$\forall i \quad 1 \le i \le n \quad \left|(C)_i\right| > 0 \tag{8}$$

▪ The sum of the coefficients of any line (respectively any column) of **C** is positive :

$$\begin{cases} \forall i & 1 \le i \le n & \displaystyle\sum_{\alpha=1}^{n} C_{i\,\alpha} \ge 0 \\ \forall j & 1 \le j \le n & \displaystyle\sum_{\beta=1}^{n} C_{\beta\,j} \ge 0 \end{cases} \tag{9}$$

For the **R** matrix these properties are :

▪ the **R** matrix is square of size n ;
▪ the **R** matrix is symetric :

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad R_{ij} = R_{ji} \tag{10}$$

▪ all coefficients of **R** are positive :

$$\begin{cases} \forall i & 1 \le i \le n \\ \forall j & 1 \le j \le n \end{cases} \quad \text{we have} \quad R_{ij} \ge 0 \tag{11}$$

▪ for the DC resistance matrix, all non-diagonal coefficient are equal and smaller than any diagonal coefficient.

3.2 Theory of transmission line modeling with SPICE

The lossless L-model created by SpiceLine (see § 2.2) only implements lumped inductors, coupling between inductors, and lumped capacitors. A lossy L-model also includes the required lumped resistors at both ends (see § 3.6), assuming that the dielectric losses are negligible (i.e. the per-unit-length conductance matrix is not accounted for). This equivalent circuit is directly derived from the output matrices. The limitation of the lossless L-model and lossy L-model do not deserve much discussion.

The M-model, D-model and E-model created by SpiceLine are based on the eigenmodes of a lossless transmission lines. The theory for creating eigenmode-based Spice models for lossless multiconductor transmission lines is well known and presented in [1] and [2]. Let us summarize it. Because of the properties of **L** and **C**, the product matrix **CL** can be diagonalised with a real **T** matrix:

$$\mathbf{T}^{-1}\mathbf{C}\mathbf{L}\mathbf{T} = \begin{pmatrix} 1/c_1^{\,2} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1/c_n^{\,2} \end{pmatrix} \tag{12}$$

where the quantity (c_1, ..., c_n) can be shown to be the propagation velocities of each of the n modes. We define the **S** matrix as:

$$\mathbf{S} = k\,\mathbf{C}^{-1}\mathbf{T} \tag{13}$$

where k is an arbitrary positive constant. **S** diagonalizes the **LC** matrix :

$$\mathbf{S}^{-1}\mathbf{L}\mathbf{C}\mathbf{S} = \begin{pmatrix} 1/c_1^{\,2} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1/c_n^{\,2} \end{pmatrix} \tag{14}$$

An important quantity is the characteristic impedance matrix $\mathbf{Z}_0$ defined as:

$$\mathbf{Z}_0 = \mathbf{C}^{-1}\mathbf{T}\begin{pmatrix} 1/c_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1/c_n \end{pmatrix}\mathbf{T}^{-1} \tag{15}$$

It has the property of being the impedance of a n-pole which, when connected to an end of the multiconductor transmission line, would absorb any incoming signal without reflection. One should also introduce the modal characteristic impedance matrix defined as $\mathbf{Z}_{0m} = \mathbf{S}^{-1}\mathbf{Z}_0\mathbf{T}$. It is diagonal:

$$\mathbf{Z}_{0m} = \mathbf{S}^{-1}\mathbf{Z}_0\mathbf{T} = \begin{pmatrix} z_{0m\,1} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & z_{0m\,n} \end{pmatrix} \tag{16}$$

Let us note (i_1,..., i_n) the currents flowing into the conductors 1 to n of the cable, measured at a given distance z from the near end of the cable and **i** the corresponding column vector. Let us note (v_1,..., v_n)

the voltages of the conductors 1 to n of the cable, measured at the same distance z from the near end between each conductor and the reference conductor and $\mathbf{v}$ the corresponding column vector.

The modal transform is defined as :

$$\begin{cases} \mathbf{v}_{\mathrm{m}} = \mathbf{S}^{-1}\mathbf{v} \\ \mathbf{i}_{\mathrm{m}} = \mathbf{T}^{-1}\mathbf{i} \end{cases} \tag{17}$$

where $\mathbf{v}_{\mathrm{m}}$ is the vector of the n modal voltages and $\mathbf{i}_{\mathrm{m}}$ the vector of the n modal currents.

In the modal domain, each voltage and current of index α propagates at the velocity c_α with the modal characteristic impedance $z_{0\mathrm{m}\ \alpha}$. This short theoretical sketch contains all that is needed for the computation of propagation and crosstalk of a multiconductor transmission line. Using (17), the problem of n+1 coupled conductors is changed into a problem with n uncoupled 2-conductors transmission lines with a known propagation velocity and a known characteristic impedance.

The lossless SPICE model created by SpiceLine is based on this computation.

Let us finally note that some authors do not use (13), but another condition. Our practice of defining a matrix $\mathbf{S}$ in this manner, gives the relationship

$$z_{0m\,\alpha} = \frac{1}{k\,c_\alpha} \tag{18}$$

for the mode characteristic impedance for mode i. In SpiceLine, we adopt $k = 10^{-10}$. This keeps the modal impedance within the physical bounds of a characteristic impedance, say between 10 Ω and 1000 Ω, values for which SPICE may be expected to work properly.

Practically, eigenmode transformation is implemented with B-elements.

It is difficult to introduce losses in such models, and we have therefore introduced an heuristic approach [2] to include some losses in SPICE models for multiconductor transmission lines, with lumped elements at both ends of a lossless multiconductor transmission line models. This approach is explained at § 3.6. Of course, such lossy models are inappropriate if the line is too long (because in this case those lumped elements bring a mismatch in the model, that is not present in the real world).

The M-model implements an eigenmode transformation, but it is based on lumped elements for the propagation of the eigenmodes. The purpose of the eigenmode transform is to remove all coupling between inductors, which have sometimes been reported to create convergence problems in some SPICE-like simulation software.

For the D-model, the propagation of the (uncoupled) eigenmode is described with distributed lossless transmission line models (the T-elements of SPICE), whereas for the E-model, the propagation of the eigenmode is modeled with distributed lossy transmission line model (the O-elements of SPICE) without losses (per-unit-length resistance and per-unit-length conductance of the lossy transmission line model set to 0).

L-models, M-models, and D-models have only one parameter L, for the length of the line in meter, which defaults to L = 1 meter. E-models have two parameters, the line length L in meter, which defaults to L = 1 meter, and the R parameter which is the dimensionless REL breakpoint control parameter of the lossy transmission line SPICE model, which defaults to 1.

3.3 Additional features for twisted pair cables: change of variables

The Telecom Line Emission Predictor of SpiceLine offers the possibility of implementing a transformation (i.e. a change of variables) for the **L**, **C** and **R** matrices.

Twisted pair cables are designed for symetry. If their symetry was ideal, and if they were implemented with perfectly symetrical terminations, they would not have any crosstalk, nor any emission. It is therefore essential to take their deviation from perfect symetry into account. Unfortunately, these deviations are accounted for by tiny differences between elements of the **L** and **C** matrices.

It has been shown in [3] that a convenient way of measuring or describing these deviations from symetry is to implement a transformation on voltages and currents. In that case, instead of entering the **L**, **C** and **R** matrices of the cable, one will enter the transformed matrices $\mathbf{L_T}$, $\mathbf{C_T}$ and $\mathbf{R_T}$ matrices.

Let us number the conductors of the cable in the following manner :
conductor 1 : conductor 1 of pair 1
conductor 2 : conductor 2 of pair 1
conductor 3 : conductor 1 of pair 2
conductor 4 : conductor 2 of pair 2
etc...

In the case of a UTP (unshielded twisted pair) cable with p pairs, we can thus number the $n = 2p$ conductors of the cable. In the case of a STP (shielded twisted pair) cable with p pairs, we in addition give the number $n = 2p+1$ to the cable shield.

Let us note $(i_1,..., i_n)$ the currents flowing into the conductors 1 to n of the cable, measured at a given distance z from the near end of the cable and **i** the corresponding column vector. Let us note $(v_1,..., v_n)$ the voltages of the conductors 1 to n of the cable, measured at the same distance z from the near end between each conductor and the reference conductor and **v** the corresponding column vector.

Let us note $(I_1,..., I_n)$ the transformed currents and **I** the corresponding column vector. Let us note $(V_1,..., V_n)$ the transformed voltages and **V** the corresponding column vector. The transformed current and voltages are defined by the transformation matrices **A** and **B** :

$$\begin{cases} \mathbf{V} = \mathbf{A}\,\mathbf{v} \\ \mathbf{I} = \mathbf{B}\,\mathbf{i} \end{cases} \tag{19}$$

whence the transformed inductance, resistance and capacitance matrices are respectively

$$\begin{cases} \mathbf{L}_T = \mathbf{A}\mathbf{L}\mathbf{B}^{-1} \\ \mathbf{R}_T = \mathbf{A}\mathbf{R}\mathbf{B}^{-1} \\ \mathbf{C}_T = \mathbf{B}\mathbf{C}\mathbf{A}^{-1} \end{cases} \tag{20}$$

SpiceLine allows that the user introduce two changes of variables : one for the currents, and one for the voltages. The changes of variables are defined by specifying a **A** matrix and a **B** matrix. Four pre-defined **A** matrices and four pre-defined **B** matrices are also provided in Spice Line.

- The utp1a.cov and utp1b.cov are respectively a **A** matrix and a **B** matrix for a UTP cable with one pair. The transformations are :

$$\begin{cases} V_1 = v_1 - v_2 \\ V_2 = \dfrac{1}{2}(v_1 + v_2) \end{cases} \qquad \begin{cases} I_1 = \dfrac{1}{2}(i_1 - i_2) \\ I_2 = i_1 + i_2 \end{cases} \tag{21}$$

■ The utp4a.cov and utp4b.cov are respectively a **A** matrix and a **B** matrix for a UTP cable with four pairs. The transformations are :

$$\begin{cases} V_1 = v_1 - v_2 \\ \ldots \\ V_4 = v_7 - v_8 \\ V_5 = \dfrac{1}{2}(v_1 + v_2) - \dfrac{1}{2}(v_3 + v_4) \\ V_6 = \dfrac{1}{4}(v_1 + \ldots + v_4) - \dfrac{1}{2}(v_5 + v_6) \\ V_7 = \dfrac{1}{6}(v_1 + \ldots + v_6) - \dfrac{1}{2}(v_7 + v_8) \\ V_8 = \dfrac{1}{8}(v_1 + v_2 + \ldots + v_8) \end{cases} \qquad \begin{cases} I_1 = \dfrac{1}{2}(i_1 - i_2) \\ \ldots \\ I_4 = \dfrac{1}{2}(i_7 - i_8) \\ I_5 = \dfrac{1}{2}(i_1 + i_2) - \dfrac{1}{2}(i_3 + i_4) \\ I_6 = \dfrac{1}{3}(i_1 + \ldots + i_4) - \dfrac{2}{3}(i_5 + i_6) \\ I_7 = \dfrac{1}{4}(i_1 + \ldots + i_6) - \dfrac{3}{4}(i_7 + i_8) \\ I_8 = i_1 + i_2 + \ldots + i_8 \end{cases} \tag{22}$$

■ The stp1a.cov and stp1b.cov are respectively a **A** matrix and a **B** matrix for a STP cable with one pair. The transformations are:

$$\begin{cases} V_1 = v_1 - v_2 \\ V_2 = \dfrac{1}{2}(v_1 + v_2) - v_3 \\ V_3 = v_3 \end{cases} \qquad \begin{cases} I_1 = \dfrac{1}{2}(i_1 - i_2) \\ I_2 = i_1 + i_2 \\ I_3 = i_1 + i_2 + i_3 \end{cases} \tag{23}$$

■ The stp4a.cov and stp4b.cov are respectively a **A** matrix and a **B** matrix for a STP cable with four pairs. The transformations are :

$$
\left\{
\begin{array}{l}
V_1 = v_1 - v_2 \\
\dots \\
V_4 = v_7 - v_8 \\
V_5 = \frac{1}{2}(v_1 + v_2) - \frac{1}{2}(v_3 + v_4) \\
V_6 = \frac{1}{4}(v_1 + \dots + v_4) - \frac{1}{2}(v_5 + v_6) \\
V_7 = \frac{1}{6}(v_1 + \dots + v_6) - \frac{1}{2}(v_7 + v_8) \\
V_8 = \frac{1}{8}(v_1 + v_2 + \dots + v_8) - v_9 \\
V_9 = v_9
\end{array}
\right.
\qquad
\left\{
\begin{array}{l}
I_1 = \frac{1}{2}(i_1 - i_2) \\
\dots \\
I_4 = \frac{1}{2}(i_7 - i_8) \\
I_5 = \frac{1}{2}(i_1 + i_2) - \frac{1}{2}(i_3 + i_4) \\
I_6 = \frac{1}{3}(i_1 + \dots + i_4) - \frac{2}{3}(i_5 + i_6) \\
I_7 = \frac{1}{4}(i_1 + \dots + i_6) - \frac{3}{4}(i_7 + i_8) \\
I_8 = i_1 + i_2 + \dots + i_8 \\
I_9 = i_1 + i_2 + \dots + i_9
\end{array}
\right.
\tag{24}
$$

3.4 Additional features for twisted pair cables : change of cable height

The Telecom Line Predictor of SpiceLine offers a model for implementing a modification of cable height above a ground plane. The modifications apply only to the $\mathbf{L_T}$ and $\mathbf{C_T}$ matrices of a **STP cable**, for a transform having the following properties :
property 1 : all elements of the last row of the **B** matrix have the same value,
property 2 : for any row of the **A** matrix but the last line, the sum of all elements is zero.

We note immediately that the transformation defined by (14) to (17) have these two properties.

If :
h_R is the reference height, that is to say the height (geometrical mean distance) for which the $\mathbf{L_T}$ and $\mathbf{C_T}$ matrices have been entered,
h_N is the nominal height, that is to say the height (geometrical mean distance) for which the SPICE models will be computed,
h is a height, taking on the value h_R or h_N as appropriate,
we assume that only the last row and last column of the $\mathbf{L}_T$ matrix is modified by the change of height, in such a manner that, for any integer i between 1 and n :

$$
\left\{
\begin{array}{l}
L_{T\,i\,n}(h) = \frac{\mu_0}{2\pi} \ln \frac{h}{h_R} + L_{T\,i\,n}(h_R) \\
L_{T\,n\,i}(h) = \frac{\mu_0}{2\pi} \ln \frac{h}{h_R} + L_{T\,n\,i}(h_R)
\end{array}
\right.
\tag{25}
$$

and that only the last row and last column of the $\mathbf{C}_T^{-1}$ matrix is modified by the change of height, in such a manner that, for any integer i between 1 and n :

$$\begin{cases} C_{T\ in}^{-1}(h) = \dfrac{1}{2\pi\varepsilon_0} \ln\dfrac{h}{h_R} + C_{T\ in}^{-1}(h_R) \\ C_{T\ ni}^{-1}(h) = \dfrac{1}{2\pi\varepsilon_0} \ln\dfrac{h}{h_R} + C_{T\ ni}^{-1}(h_R) \end{cases} \tag{26}$$

Of course, this model gives only an estimate of what the L_T and C_T could be for this new height, and let us recall that it is limited to STP cables.

3.5 Sequence of operations on input data

The user enters the $\mathbf{L_T}$, $\mathbf{C_T}$ and $\mathbf{R_T}$ matrices, referred to as **input matrices** in SpiceLine.

SpiceLine then implements the modification for the new cable height, according to § 3.4 : it computes the **modified input matrices**. If no change of cable height is specified, the modified input matrices are of course equal to the input matrices.

SpiceLine then implements the inverse transform defined by the **A** and **B** matrices according to §3.3, in order to recover the **L**, **C** and **R** matrices. These matrices are called the **inverse-transformed input matrices (referenced to ground)**. If no change of variable is specified, the modified input matrices are of course equal to the modified input matrices.

Some manual editing can then be done on the inverse-transformed input matrices, using a “Diagonal multiplying factor” and a “Non-diagonal multiplying factor”. This leads to the generation of the **output matrices**. This manual editing is sometimes necessary in order to pass the criteria defined in § 3.1, when the matrix values were obtained from inaccurate measurements.

The next operation consists in computing the eigenmodes of the cable assumed lossless. One can then create SPICE models which will eventually take losses into account, to some extend (see § 3.6).

3.6 Taking losses into account

The explanations given in this paragraph are derived from [2]. A lossless model of a cable is generally expected to over-estimate the induced voltages and currents. This is usually acceptable to the user, because it leads to an additionnal margin in the design process. There is nevertheless two special cases where a lossless model usually gives rise to an over-optimistic assessment of unwanted signal. The first case is related to the real part of the reference conductor impedance ; the second case occurs if conductors of the multiconductor transmission line (MTL) are terminated at both ends to the reference conductor with low impedances.

The first problem can be described as the consequence of neglecting the off-diagonal terms of the **R** matrix. This can be cured by adding a current controlled voltage source (an H-element according to the SPICE syntax), in series with each relevant conductor in the MTL SPICE model. It can be done at one or both ends of the conductor, in such a way that the total transimpedance (or transresistance in this case, i.e. the ratio of output voltage of the H-element to its input current) is equal to the off-diagonal term of **R**(0 Hz). Let us call this "cosmetic measure # 1".

The second problem mentionned above occurs when conductors other than the reference conductor are grounded or connected to ground with a low-impedance termination. For example, a common practice is to have many grounded-at-both-ends conductors on flat unshielded ribbon cables, for improved EMC characteristics. In the case of an MTL including one or several shields, the shields are also normally grounded at both ends. Each conductor with low impedance to ground at both ends provide some shielding. Neglecting losses over-estimates their shielding effectiveness at low frequencies. This is because in the lossless model, their per-unit-lenght impedance decreases to zero at zero frequency, instead of being limited by their per-unit-length DC resistance. This per-unit-length resistance is equal to the corresponding diagonal term of **R**(0 Hz) minus the per-unit-length DC resistance of the reference conductor. We therefore advocate the practice of adding in the MTL SPICE model, an appropriate lumped resistor at one or both ends of conductors potentially subject to this problem. Let us call this "cosmetic measure # 2".

The two cosmetic measures proposed are equivalent to artificially adding the whole or part of the DC resistance matrix of the MTL to the lossless model.

Note also that the "cosmetic measures" proposed above are only applicable when the added impedance and transimpedance remain much smaller than the corresponding terms of the characteristic impedance matrix of the MTL (assumed lossless). If this was not the case, they would generate a parasitic impedance mismatch in the problem, and hurt the accuracy of high frequency coupling calculation. Also, the shields in the example above are characterized by a per-unit-length transfer impedance with frequency independant resistive and inductive terms only: as is well known, this is only accurately valid for some types of screen, for instance non-optimized single-braided shields.

SpiceLine allows model to be created either with all terms of the resistance matrix taken into account, for a model including the cosmetic measures #1 and # 2: at each end of the SPICE model a circuit equivalent to the resistance matrix **R** times the half cable length is added. It is also possible to only take into account the diagonal terms of the resistance matrix, for a model including the cosmetic measure # 2.

3.7 Theory of crosstalk problems

Once a SPICE model of the cable is created and implemented into a SPICE simulation software, the current and voltages may be computed during the simulation. Cable crosstalk is described by the SPICE model, which can be used in .AC and .TRAN simulation with SPICE.

Usually, the emission by conduction (also called conducted emission by some authors) of signal cables is measured with the total current flowing on the cable, as a function of frequency. This quantity can be easily obtained from an .AC simulation, at one of the cable termination. With ICAP/4 version 7.x, one could for instance issue an ICL block such as:

```
.control
save allcur
alias imc mag(i(v1)+i(v2)+i(v3)+i(v4)+i(v5)+i(v6)+i(v7)+i(v8))
.endc
.PRINT AC IMC
```

3.8 Theory of emission by conduction and by radiation problems

SpiceLine is able to compute the emission by conduction (also called conducted emission by some authors) at a given abscissa along the cable, and the emission by radiation (also called radiated emission by some authors) of a cable at a given point in space in the frequency domain. This feature is only available if a change of variable is selected, such that the last element of the transformed current vector is the common mode current, and such that the last element of the transformed voltage vector is a common mode voltage. The changes of variable given by formulas (21) to (24) meet this requirement.

The steps for the emission computation are the following :

1) create a model of the cable
2) prepare the .AC simulation, making sure that at the near end of the cable, conductor 1 of the cable is at node N1, conductor 2 at node N2..., and that a source V1 (meant for current measurement) has its negative node connected to node N1, a source V2 has is negative node connected to node N2....
3) simulate the model, making sure that the **only** .PRINT statements are in the following order :
.PRINT AC V(1) VP(1)
.PRINT AC I(V1) IP(V1)
.PRINT AC V(2) VP(2)
.PRINT AC I(V2) IP(V2)
etc.
4) Select the output file of this simulation (*.OUT file) in SpiceLine so that the voltage and current values at the near end of the cable, computed by SPICE is known from SpiceLine.

Note that the for_cir directory of disk 3 contains the utp1p1, stp1p1, utp4p4 and stp4p4 SPICE projects that meet the requirements of point 2) above.

For each frequency in this output file, SpiceLine will establish the chain supermatrix. This supermatrix allow the straightforward computation of all voltages and currents along the cable. Then according to [3] the type 1 and type 2 emission mecanisms are used to compute the emission of the cable at a point P specified by its right-hand cartesian coordinate (X, Y, Z), X being a vertical distance above the horizontal ground plane, Z being an horizontal distance parallel to the cable, the origin being the orthogonal projection of the near end of the cable on the ground plane.

More precisely, at a given abscissa z along the cable, the value of the chain supermatrix is :

$$\begin{pmatrix} \mathbf{S}\cosh(\gamma\, z)\mathbf{S}^{-1} & -\mathbf{Z}_0\mathbf{T}\sinh(\gamma\, z)\mathbf{T}^{-1} \\ -\mathbf{T}\sinh(\gamma\, z)\mathbf{T}^{-1}\mathbf{Z}_0^{-1} & \mathbf{T}\cosh(\gamma\, z)\mathbf{T}^{-1} \end{pmatrix} \tag{27}$$

Therefore, the voltage and currents at z can be computed from the voltage and current computed by the SPICE software at the near-end ($z = 0$), as:

$$\begin{pmatrix} \mathbf{v}(z) \\ \mathbf{i}(z) \end{pmatrix} = \begin{pmatrix} \mathbf{S}\cosh(\gamma\, z)\mathbf{S}^{-1} & -\mathbf{Z}_0\mathbf{T}\sinh(\gamma\, z)\mathbf{T}^{-1} \\ -\mathbf{T}\sinh(\gamma\, z)\mathbf{T}^{-1}\mathbf{Z}_0^{-1} & \mathbf{T}\cosh(\gamma\, z)\mathbf{T}^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{v}(0) \\ \mathbf{i}(0) \end{pmatrix} \tag{28}$$

where $\mathbf{v}(z)$ and $\mathbf{i}(z)$ are the voltage vector and current vector at abscissa z along the cable, where $\mathbf{v}(0)$ and $\mathbf{i}(0)$ are the voltage vector and the current vector at the near-end. This formula does not take any "cosmetic measure" into account, and therefore only applies to the lossless case. In fact, SpiceLine computes the voltage and currents at z as:

$$\begin{pmatrix} \mathbf{v}(z) \\ \mathbf{i}(z) \end{pmatrix} = \begin{pmatrix} \mathbf{S}\cosh(\gamma\, z)\mathbf{S}^{-1} & -\mathbf{Z}_0\mathbf{T}\sinh(\gamma\, z)\mathbf{T}^{-1} \\ -\mathbf{T}\sinh(\gamma\, z)\mathbf{T}^{-1}\mathbf{Z}_0^{-1} & \mathbf{T}\cosh(\gamma\, z)\mathbf{T}^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{v}_{\text{Spice}} - \dfrac{L_c}{2}\mathbf{R}\,\mathbf{i}_{\text{Spice}} \\ \mathbf{i}_{\text{Spice}} \end{pmatrix} \tag{29}$$

where $\mathbf{v}_{\text{Spice}}$ and $\mathbf{i}_{\text{Spice}}$ are the voltage vector and the current vector computed by SPICE, and where L_c is the total cable length.

SpiceLine will deliver the $\mathbf{v}(z)$ and $\mathbf{i}(z)$ vectors at an abscissa z specified by the user.

The computation of the radiated fields is based on the assumption that the reference conductor is an ideal (i.e. infinite and perfectly conducting) ground plane. The fields radiated by the cable above the ideal ground plane, if one considers only type 1 and type 2 emission mechanisms, are determined by the per-unit-length dipole moments :

$$\mathbf{M}_T = -2h_N\, I_{CM}\, \mathbf{e}_y \tag{30}$$

and

$$\mathbf{P}_T = 2h_N\; C_{CM}\; V_{CM}\, \mathbf{e}_x \tag{31}$$

where $I_{CM} = I_n$ is the common mode current, where $V_{CM} = V_n$ is the common mode voltage, and where C_{CM} is the common mode capacitance equal to the last diagonal term of the transformed capacitance matrix $\mathbf{BCA}^{-1}$. The fields are computed as:

$$\begin{aligned} \mathbf{E} = & \int_0^{L_c} \left\{ \left(\frac{1}{r^3} + \frac{\mathrm{j}k}{r^2} \right) \left[3(\mathbf{u}.\mathbf{P}_T)\mathbf{u} - \mathbf{P}_T \right] - \frac{k^2}{r} \left[\mathbf{u} \times (\mathbf{u} \times \mathbf{P}_T) \right] \right\} \frac{\mathrm{e}^{-\mathrm{j}\mathbf{kr}}}{4\pi\,\varepsilon_0}\ \mathrm{d}z \\ & + \int_0^{L_c} \left\{ \left(\frac{\mathrm{j}k}{r^2} - \frac{k^2}{r} \right) \mathbf{u} \times \mathbf{M}_T \right\} \frac{\eta_0\, \mathrm{e}^{-\mathrm{j}\mathbf{kr}}}{4\pi}\ \mathrm{d}z \end{aligned} \tag{32}$$

and

$$\begin{aligned} \mathbf{H} = & \int_0^{L_c} \left\{ \left(\frac{1}{r^3} + \frac{\mathrm{j}k}{r^2} \right) \left[3(\mathbf{u}.\mathbf{M}_T)\mathbf{u} - \mathbf{M}_T \right] - \frac{k^2}{r} \left[\mathbf{u} \times (\mathbf{u} \times \mathbf{M}_T) \right] \right\} \frac{\mathrm{e}^{-\mathrm{j}\mathbf{kr}}}{4\pi}\ \mathrm{d}z \\ & + \int_0^{L_c} \left\{ \left(-\frac{\mathrm{j}k}{r^2} + \frac{k^2}{r} \right) \mathbf{u} \times \mathbf{P}_T \right\} \frac{\mathrm{e}^{-\mathrm{j}\mathbf{kr}}}{4\pi\,\varepsilon_0\eta_0}\ \mathrm{d}z \end{aligned} \tag{33}$$

The integrals are computed as Rieman sums, with a maximum number of terms ranging between 16383 and 32766. As a consequence, the calculation will become inaccurate when the line length is not much smaller than 16383 times the wavelength, or when the point at which the fields are computed is too close to a very long line.

3.9 Theory of voltages and currents induced by external field sources

SpiceLine is also able to compute the voltage and current induced on the transmission line by external field sources at a given point in space. This computation is based on reciprocity. The theory is the following.

case a : voltage induced by an electric dipole

Let us consider the emission by radiation of a cable excited at the near-end termination by a unique current source delivering the current I_{T1}. With the method and the limitations presented at § 3.8, we are able to compute the electric field $\mathbf{E}_\mathbf{P}$ and the magnetic field $\mathbf{H}_\mathbf{P}$ at a given point P.

If we now install a small electric dipole made of two aligned segments of length d_E, at the point P, we know that the voltage induced in this dipole by the fields produced by the current I_{T1} is:

$$V_{P1} = d_E \ \mathbf{u} \cdot \mathbf{E}_\mathbf{P} \tag{34}$$

where **u** is a unit vector aligned with the dipole.

Let us now assume that we replace the current source which produced the current I_{T1} with an open circuit and that we measure the voltage V_{T2} across this open circuit with the sign convention of generator associated with the sign convention choosen for measuring I_{T1}. If we install a current source sourcing a current I_{P2} into the electric dipole, measured with the generator's sign convention associated with the sign convention choosen for measuring V_{P1}, the reciprocity theorem tells us that we will measure:

$$V_{T2} = \frac{V_{P1} I_{P2}}{I_{T1}} = j\omega \ \frac{\mathbf{u} \cdot \mathbf{E}_\mathbf{P}}{I_{T1}} P \tag{35}$$

where $P = \mathbf{P} \cdot \mathbf{u}$ is the electric dipole moment measured positive in the direction of **u**.

If one creates a SPICE simulation for computing the electric field $\mathbf{E}_\mathbf{P}$ radiated for I_{T1} = 1 A, SpiceLine will compute in this manner the voltage V_{T2} induced by an electric dipole with an amplitude of 1 Cm.

case b : voltage induced by a magnetic dipole

As for the case a, let us consider the emission by radiation of a cable excited at the near-end termination by a unique current source delivering the current I_{T1}. We are able to compute the electric field $\mathbf{E}_\mathbf{P}$ and the magnetic field $\mathbf{H}_\mathbf{P}$ at a given point P.

If we now install a small magnetic dipole made of a circular loop of surface S_H at the point P, we know that the voltage induced in this dipole by the fields produced by the current I_{T1} is:

$$V_{P1} = j\omega\mu_0 \, S_H \, \mathbf{u} \cdot \mathbf{H}_\mathbf{P} \tag{36}$$

where **u** is a unit vector orthogonal to the plane of the loop.

Let us now assume that we replace the current source which produced the current I_{T1} with an open circuit and that we measure the voltage V_{T2} across this open circuit with the sign convention of generator associated with the sign convention choosen for measuring I_{T1}. If we install a current source sourcing a current I_{P2} into the electric dipole, measured with the generator's sign convention associated with the sign convention choosen for measuring V_{P1}, the reciprocity theorem tells us that we will measure:

$$V_{T2} = \frac{V_{P1} I_{P2}}{I_{T1}} = j\omega \, \mu_0 \ \frac{\mathbf{u} \cdot \mathbf{H}_\mathbf{P}}{I_{T1}} M \tag{37}$$

where $M = \mathbf{M} \cdot \mathbf{u}$ is the magnetic dipole moment measured positive in the direction of **u**.

If one creates a SPICE simulation for computing the electric field $\mathbf{H_P}$ radiated for I_{T1} = 1 A, SpiceLine will compute in this manner the voltage V_{T2} induced by a magnetic dipole with an amplitude of 1 Am².

case c : current induced by an electric dipole

Let us consider the emission by radiation of a cable excited at the near-end termination by a unique voltage source delivering the voltage V_{T1}. With the method and the limitations presented at § 3.8, we are able to compute the electric field $\mathbf{E_P}$ and the magnetic field $\mathbf{H_P}$ at a given point P.

If we now install a small electric dipole made of two aligned segments of length d_E, at the point P, the short-circuit current of this dipole excited by the fields produced by the voltage V_{T1} is:

$$I_{P1} = \frac{d_E}{Z_E}\, \mathbf{u} \cdot \mathbf{E_P} \tag{38}$$

where Z_E is the dipole's impedance and where $\mathbf{u}$ is a unit vector aligned with the dipole.

Let us now assume that we replace the voltage source which produced the current V_{T1} with a short-circuit in which we measure the current I_{T2} using generator's sign convention associated with the sign convention choosen for measuring V_{T1}. If we install a voltage source delivering a voltage V_{P2} to the electric dipole, measured with the generator's sign convention associated with the sign convention choosen for measuring I_{P1}, the reciprocity theorem tells us that we will measure:

$$I_{T2} = \frac{I_{P1} V_{P2}}{V_{T1}} = j\omega\, \frac{\mathbf{u} \cdot \mathbf{E_P}}{V_{T1}}\, P \tag{39}$$

where $P = \mathbf{P} \cdot \mathbf{u}$ is the electric dipole moment measured positive in the direction of $\mathbf{u}$.

If one creates a SPICE simulation for computing the electric field $\mathbf{E_P}$ radiated for V_{T1} = 1 V, SpiceLine will compute in this manner the current I_{T2} induced by an electric dipole with an amplitude of 1 Cm.

case d : current induced by a magnetic dipole

As previously, let us consider the emission by radiation of a cable excited at the near-end termination by a unique voltage source delivering the voltage V_{T1}. We are able to compute the electric field $\mathbf{E_P}$ and the magnetic field $\mathbf{H_P}$ at a given point P.

If we now install a small magnetic dipole made of a circular loop of surface S_H at the point P, we know that the short-circuit current induced in this dipole by the fields produced by the voltage V_{T1} is:

$$I_{P1} = \frac{j\omega\mu_0 S_H}{Z_H}\, \mathbf{u} \cdot \mathbf{H_P} \tag{40}$$

where Z_H is the dipole impedance, and where $\mathbf{u}$ is a unit vector orthogonal to the plane of the loop.

Let us now assume that we replace the voltage source which produced the current V_{T1} with a short-circuit in which we measure the current I_{T2} using generator's sign convention associated with the sign convention choosen for measuring V_{T1}. If we install a voltage source delivering a voltage V_{P2} to the electric dipole, measured with the generator's sign convention associated with the sign convention choosen for measuring I_{P1}, the reciprocity theorem tells us that we will measure:

$$I_{T2} = \frac{I_{P1} V_{P2}}{V_{T1}} = j\omega \mu_0 \frac{\mathbf{u} \cdot \mathbf{H_P}}{V_{T1}} M \tag{41}$$

where $M = \mathbf{M} \cdot \mathbf{u}$ is the magnetic dipole moment measured positive in the direction of **u**.

If one creates a SPICE simulation for computing the electric field $\mathbf{H}_\mathbf{P}$ radiated for V_{T1} = 1 V, SpiceLine will compute in this manner the current I_{T2} induced by a magnetic dipole with an amplitude of 1 Am².

Normalization of the response to external fields

When studying the immunity of a circuit to external radiated disturbances, it is customary to relate the effect (in our case the induced current I_{T2} or the induced voltage V_{T2}) to the value of the incident field at a given point. We have choosen to normalize the effect of a dipole having any orientation, lying at the point P of coordinates (X, Y, Z) in the following way :
— If it is an electric dipole, it will take on the electric dipole moment value which would produce an electric field E_0 = 1 V/m at the cable near end in an empty space, if the electric dipole was oriented in such a direction that the cable near-end would lie in the plane of maximum far-field radiation of the dipole,
— If it is a magnetic dipole, it will take on the magnetic dipole moment value which would produce a magnetic field H_0 =1 A/m at the cable near-end in an empty space, if the magnetic dipole was oriented in such a direction that the cable near-end would lie in the plane of maximum far field radiation of the dipole.

Thus, the electric dipole moment takes on the value:

$$P_0 = \frac{4\pi \varepsilon_0}{\sqrt{\left(\frac{1}{r^3} - \frac{k^2}{r}\right)^2 + \left(\frac{k}{r^2}\right)^2}} E_0 \tag{42}$$

or the magnetic dipole moment takes on the value:

$$M_0 = \frac{4\pi}{\sqrt{\left(\frac{1}{r^3} - \frac{k^2}{r}\right)^2 + \left(\frac{k}{r^2}\right)^2}} H_0 \tag{43}$$

the value of r in (42) and (43) being:

$$r = \sqrt{\left(X - h_N\right)^2 + Y^2 + Z^2} \tag{44}$$

for a cable near-end at the coordinates (h_N, 0, 0)

4. Getting started

4.1 Initial inspection and parts list

4.1.1 Unpacking

You should have received the present User's Guide, the Reference Manual, and five floppy disks.

4.1.2 Disk 1

The floppy disk1 contains the files :
spline23.exe
~mat.vts
~mat.bak
~matout.vts
~vect.vts

4.1.3 Disk 2

The floppy disk2 contains the files :
bc450rtl.dll
bds52.dll
bids45.dll
bivbx11.dll
owl250.dll
vtssdll.dll
vtss.vbx

4.1.4 Disk 3

The floppy disk3 contains the files :
bc520rtl.dll
bivbx11.dll
owl52.dll

4.1.5 Disk 4

The floppy disk 4 contains files in the root and in four other directories :
\exemples
\for_sn
\for_pr
\for_cir

The root of floppy disk4 contains the files describing several changes of variables (see § 3.3) :

ident.cov
stp1a.cov
stp1b.cov
stp4a.cov
stp4b.cov
utp1a.cov
utp1b.cov
utp4a.cov
utp4b.cov

The directory \examples of the disk4 contains SpiceLine files for 5 SpiceLine projects :

IEEE0296
UTPD1
UTPD8
STPD1
STPD8

The file names of the file belonging to these projects are made with the name of the project and the following extensions.

*.pjt	project file containing general information on the project
*.inc	file containing the input capacitance matrix
*.inl	file containing the input inductance matrix
*.inr	file containing the input resistance matrix
*.ouc	file containing the output capacitance matrix
*.oul	file containing the output inductance matrix
*.our	file containing the output resistance matrix
*.cva	file containing the A matrix
*.cvb	file containing the B matrix
*.erc	file containing the error report on the output capacitance matrix
*.erl	file containing the error report on the output inductance matrix
*.err	file containing the error report on the output resistance matrix

for a total of 70 files in this directory.

The directory \for_sn of disk 4 contains, in the \for_sn\symsplne subdirectory, some SpiceNet symbols:

demec.sym	this is a “braid” sub-symbol used as a cable screen in our STP symbols,
masl.sym	this is a reference conductor sub-symbol used in other cable symbols,
pair.sym	this is a twisted pair symbol, also used as sub-symbol in our STP and UTP symbols,
fredline.sym	a symbol for a cable with two shielded pairs and two conductors in the middle
stp1p.sym	a symbol for a single twisted pair cable, screened,
stp4p.sym	a symbol for a cable with 4 twisted pairs, screened,
utp1p.sym	a symbol for a single twisted pair cable, not screened,
utp4p.sym	a symbol for a cable with 4 twisted pairs, not screened,

The directory \for_pr of disk 4 contains two files :

splne223.lib is a library file containing various Spice subcircuits created with SpiceLine.
mycables.lib is an empty library file for storing your own Spice subcircuits.

The splne223.lib library file contains 6 SPICE models :

FRE1E is the original model used in [2], created with an earlier version of SpiceLine. This is a lossless E-model for a flat cable with one shielded pair, two conductors, and an other shielded pair.

FLA1D is a model created with SpiceLine, using the same data as the one used for FRE1E, contained in the IEEE0296 SpiceLine project. This is a D-model, for which all terms of the resistance matrix have been taken into account. No change of variables was implemented.

UT12E is a model for a single pair UTP cable, created with SpiceLine, using the data contained in the UTPD1 SpiceLine project. We implemented the utp1a.cov and utp1b.cov changes of variable, respectively for A and B. This is a E-model, for which all terms of the resistance matrix have been taken into account.

UT42E is a model for a 4 pairs UTP cable, created with SpiceLine, using the data contained in the UTPD8 SpiceLine project. We implemented the utp4a.cov and utp4b.cov changes of variable, respectively for A and B. This is a E-model, for which all terms of the resistance matrix have been taken into account.

ST11E is a model for a single pair STP cable, created with SpiceLine, using the data contained in the STPD1 SpiceLine project. We implemented the stp1a.cov and stp1b.cov changes of variable, respectively for A and B. This is a E-model, for which only the diagonal terms of the resistance matrix have been taken into account.

ST12E is a model for a single pair STP cable, created with SpiceLine, using the data contained in the STPD1 SpiceLine project. We implemented the stp1a.cov and stp1b.cov changes of variable, respectively for A and B. This is a E-model, for which all terms of the resistance matrix have been taken into account.

ST42E is a model for a 4 pairs STP cable, created with SpiceLine, using the data contained in the STPD8 SpiceLine project. We implemented the stp4a.cov and stp4b.cov changes of variable, respectively for A and B. This is a E-model, for which all terms of the resistance matrix have been taken into account.

The directory \for_cir of disk 4 contains ICAP/4 files for 6 projects :

ieeeter
ieeequar
utp1p1
stp1p1
stp4p4
utp4p4

The file names of the file belonging to these projects are made with the name of the project and the following extensions: *.c1, *.cir, *.ckt, *.cnt, *.con, *.d1. Each circuit *name*.cir has therefore 6 files in the subdirectory:

name.cir, *name*.c1 (netlists created by SpiceNet),
name.cnt, *name*.con (control statement for SpiceNet),
name.ckt (SPICE compatible netlist),
name.d1 (schematics).

4.1.6 Disk 5

Disk 5 contains two Acrobat Reader compatible files

maspli2e.pdf	for the User's guide,
maspli2f.pdf	for the Reference manual.

These files are intended to be used with Acrobat reader 3.0 or later versions. Acrobat Reader 4.0 or later version is recommended because of its superior printing capabilities.

4.2 Installation

SpiceLine is designed to work under Microsoft Windows 95/98.

Please create on your hard disk a \Sline directory and, if you wish, a subdirectories like \Sline\files. Copy the contents of Disk 1, Disk 2 and Disk 3 in \Sline. Copy the contents of the root of Disk 4, and if you wish, the contents of the \examples directory of Disk 4 into \Sline\files.

Please also copy the mycable.lib file of the for_pr directory of disk 4 into \Sline\files.

We assume that ICAP/4 is installed in the \Spice or \Spice4 directory of the current drive. Please follow the six steps below applicable to ICAP/4 Windows (Please note that some deviation from this procedure might be needed according to your version: please check your manual) :

Step 1 : copy the contents of the for_pr directory of disk 4 into the PR directory (\Spice\PR or \Spice4\PR).
Step 2 : copy the contents of the for_sn directory of disk 4 (that is to say the symsplne subdirectory and its files) into the SN\SYMBOLS directory (\Spice\SN\SYMBOLS or \Spice4\SN\SYMBOLS).
Step 3 : copy the contents of the for_cir directory of disk 4 into the CIRCUITS directory or into an other subdirectory of \Spice\ or \Spice4
Step 4 : If you are using ICAP/4 Windows version 7.6, using a text editor add
..\sn\symbols\symsplne
at the end of the Spice4\sn\sym.@@@ file.
Step 5 : compile an updated SpiceNet database. If you are using ICAP/4 Windows version 7.6, this is simply accomplished by running MakeDb.exe in SN (\Spice\SN or \Spice4\SN): double-click MakedB.exe in your Windows Explorer, then click on "Run Sindex", then click on "Run DBMake".
Step 6 : test that you can access your new cable models under the "SpiceLine2.23" heading in the SpiceNet "Parts Browser" dialog.

For earlier versions of ICAP, step 5 may require 3 substeps, for instance:
Substep 5.1 : update the Lib.@@@ file that contains the list of libraries. This is simply accomplished by adding the text "splne223.lib", at the right spot in SN\Lib.@@@ (\Spice\SN\Lib.@@@ or \Spice4\SN\Lib.@@@) using a text editor.

Substep 5.2 : with a text editor, add the appropriate lines at the end of \Spice\DBASE.SRC.
Substep 5.3 : run SN\MAKEDB.BAT, which compiles an updated SpiceNet database.

If you want to know more about this six steps procedure, please read the "Updating the data base" entry in your ICAP documentation.

Please note that we found that the simulator “IsSpice version 7.6" delivered with ICAP/4 version 7.6 has a major bug when one directly measures a voltage at the pin of a subcircuit. The solutions to this problem are:
— Try to always keep in mind that you should not do that (for instance, inserting a 0 V DC source solves the problem).
— Or (better) ask your Intusoft reseller to supply - free of charge - a simulator delivered with ICAP/4 version 7.51, for instance the simulator “IsSpice version 4f3.6". Then, save the cml.dll, real.dll and Spice4.exe file of ICAP/4 version 7.6 somewhere if you think you might need them again, and install the cml.dll, real.dll and Spice4.exe of the earlier simulator version into the IS directory (\Spice\IS or \Spice4\IS).

For launching SpiceLine, use the Windows 95/98 file manager and doubleclick on Spline23.exe.

4.3 Files produced by SpiceLine

The different matrices used in a project are saved under the following names, where *filename* is the name of the project :

the L input matrix is in *filename*.INL,
the C input matrix is in *filename*.INC,
the R input matrix is in *filename*.INR,
the L output matrix is in *filename*.OUL,
the C output matrix is in *filename*.OUC,
the R output matrix is in *filename*.OUR,
filename.ERL is a text file describing errors found during the verification of the L output matrix,
filename.ERC is a text file describing errors found during the verification of the C output matrix,
filename.ERR is a text file describing errors found during the verification of the R output matrix,
filename.CVA contains the A matrix used in the project.
filename.CVB contains the B matrix used in the project.

4.4 Sequence of operations with SpiceLine

A full SpiceLine session for obtaining SPICE models contains the following three steps :
1) creating a new project or opening an existing project,
2) defining the line parameters,
3) creating one or several ICAP4/Windows-compatible line models.

These models are stored in a *libname*.lib file, and can later be used in any circuit that you want to simulate.

If you also want to compute a radiated emission, you need to accomplish three additional steps:
4) running a ICAP4/Windows version 7.x simulation meeting the requirements of § 3.8,

5) computing the radiated fields with SpiceLine, using the *name1*.out file generated by ICAP4/Windows,
6) displaying the results stored in the *name2*.out file generated by SpiceLine, for instance with the IntuScope data post-processor of ICAP4/Windows.

If you also want to compute a voltage or current induced by a field source, you need to accomplish three steps after step 3:
4bis) running a ICAP4/Windows version 7.x simulation meeting the requirements of § 3.8,
5bis) computing the induced currents or voltages with SpiceLine, using the *name1*.out file generated by ICAP4/Windows,
6bis) displaying the results stored in the *name2*.out file generated by SpiceLine, for instance with the IntuScope data post-processor of ICAP4/Windows.

4.5 Example 1 : solving a crosstalk problem

■ Use *File/New project* to create the project *trial1.pjt*.
■ Use *Line parameters/Number n* to specify the number of conductor in the cable including shields. Select $n = 2$ for a multiconductor transmission line with two conductors above a ground plane.
■ Use *Line parameters/Line geometry* for specifying the reference height for which the inductance, capacitance and resistance matrices that you are going to enter have been established ; note that this is not a STP cable and the nominal height for which the calculation will be performed **must** therefore be equal to the reference height (in other words, you cannot use the tools for modifying the matrices described in § 3.4). In the present case, you can leave the nominal height and the reference height to 0.1 m.
■ Use *Line parameters/Select change of variables* and select *ident.cov* for the A matrix and the B matrix, because you are going to directly introduce the **L** matrix, the **C** matrix and the **R** matrix for this cable, as they are defined in § 3.1.
■ Use *Line parameters/Edit input matrix* to enter the **L** matrix, the **C** matrix and the **R** matrix. Do not press *Esc* while editing a matrix, because it will be lost. You first select *Line parameters/Edit input matrix/L*, in order to use the **Input Matrix Editor - LT** to enter the **L** matrix, taking care that you must press the **enter** key after typing each number, so that the entered number becomes justified on the right of each cell. Enter the following **L** matrix (in µH/m):

$$\begin{pmatrix} 0.816 & 0.27 \\ 0.27 & 0.816 \end{pmatrix} \tag{45}$$

You **must** use the decimal separator specified in your Windows system. When your matrix is entered, press *save* in the **Input Matrix Editor - LT**, don't take the warning message into account, then press *Create output*, then *Verify output*. Important: do not click on *Save* after after pressing *Create output* (this would save the values of the output matrix as the input matrix). Proceed in the same manner to enter the following **C** matrix (in pF/m):

$$\begin{pmatrix} 48.9 & -12.1 \\ -12.1 & 48.9 \end{pmatrix} \tag{46}$$

In this example, we are not interested in taking losses into account, and we therefore do not enter any **R** matrix.
■ Use *Line parameters/View output matrix* to check your output **L** matrix and output **C** matrix. In the present case, they are equal to the input matrices.
■ Use *Line model/Compute eigenmodes* to compute the mode propagation velocities, the eigenvectors, the characteristic impedance matrix, etc (see § 3.2).

■ Use *Line model/View modes*, if you wish, in order to inspect the computed quantities. You can compare them with those obtained in [1] for the symetrical case.
■ Use *Line model/Compute Spice models*, for storing the SPICE model. You can choose an existing *name.lib* file (in which case the new models will be appended to the library file), or create a new one. Type *mycables.lib* as file name. Select L an M and D and E models, with names like *pu1l*, *pu1m*, *pu1d* and *pu1e* in the four **Name** boxes. Select *Add*.

You just added four models to the mycable.lib library file. file of \Sline\files. This library mycable.lib should now be copied in the PR directory (\Spice\PR or \Spice4\PR) in order to be used by the ICAP/4 SPICE simulator.

You are done. The new SPICE model can be directly used with ICAP/4 Windows version 7.x for the simulation of crosstalk problems, provided a suitable *INCLUDE MYCABLES.LIB is inserted in the *File/Edit controls* window of SpiceNet. The length of the cable is the L parameter (note: this is not an inductance). If you wish to implement them in a convenient way, you may choose to manually associate these models with proper SpiceNet symbols, which you can call with the *Draw/Get symbol* window of SpiceNet. These actions are documented in your ICAP/4 documentation. The length of the cable can in this case be specified using the L parameter of the Label dialog of the cable symbol in SpiceNet.

Note that for the SPICE model just created, the utp1p.sym model included in \Spice\SN\SYMBOLS\SYMSPLNE is appropriate.

Following the appropriate steps, you can also edit the new library by adding appropriate *SRC and *SYM lines to each model (see the supplied splne223.lib library), then compile a new data base, and thus become able to access your models created with SpiceLine in the SpiceNet *Part/X part/Browser* window.

4.6 Example 2 : solving an emission problem

The possibility of solving an emission by radiation problem is now implemented in SpiceLine, when suitable changes of variables have been implemented.(see § 3.8).

■ Use *File/New project* to create the project *trial2.pjt*.
■ Use *Line parameters/Number n* to specify the number of conductor in the cable including shields. Select $n = 3$ for a multiconductor transmission line with two conductors and an a shield above a ground plane.
■ Use *Line parameters/Line geometry* for specifying the reference height for which the inductance, capacitance and resistance matrices that you are going to enter have been established ; note that this is a STP cable and the nominal height for which the calculation will be performed could therefore be different from the reference height (in other words, you can use the tools for modifying the matrices described in § 3.4). In the present case, you can leave the nominal height and the reference height to 0.1 m.
■ Use *Line parameters/Select change of variables* and select *utp1a.cov* for the A matrix and *utp1b.cov* the B matrix, because you are going to introduce transformed inductance, capacitance and resistance matrices ($\mathbf{L_T}$, $\mathbf{C_T}$ and $\mathbf{R_T}$) as input matrices (see § 3.5)
■ Use *Line parameters/Edit input matrix* to enter the **L** matrix, the **C** matrix and the **R** matrix. Do not press *Esc* while editing a matrix, because it will be lost. You first select *Line parameters/Edit input matrix/L*, in order to use the **Input Matrix Editor - LT** to enter the **L** matrix, taking care that you must press the **enter** key after typing each number, so that the entered number becomes justified on the right

of each cell. Enter the following $\mathbf{L_T}$ matrix (in µH/m) :

$$\begin{pmatrix} 0.693 & 0.002 & 0 \\ 0.002 & 0.543 & 0 \\ 0 & 0 & 0.996 \end{pmatrix} \tag{47}$$

You **must** use the decimal separator specified in your Windows system. When your matrix is entered, press *save* in the **Input Matrix Editor - LT**, don't take the warning message into account, then press *Create output*, then *Verify output*. Important: do not click on *Save* after after pressing *Create output* (this would save the values of the output matrix as the input matrix). Proceed in the same manner to enter the following $\mathbf{C_T}$ matrix (in pF/m) :

$$\begin{pmatrix} 48 & -0.08 & 0 \\ -0.08 & 76.6 & 0 \\ 0 & 0 & 20.2 \end{pmatrix} \tag{48}$$

Proceed in the same manner to enter the following $\mathbf{R_T}$ matrix (in Ω/m) :

$$\begin{pmatrix} 0.112 & 0 & 0 \\ 0 & 0.061 & -0.033 \\ 0 & -0.033 & 0.035 \end{pmatrix} \tag{49}$$

■ Use *Line parameters/View output matrix* to look at your output **L**, **C** and **R** matrices. In the present case, they are equal to the input matrices.
■ Use *Line model/Compute eigenmodes* to compute the mode propagation velocities, the eigenvectors, the characteristic impedance matrix...
■ Use *Line model/View modes*, if you wish, in order to inspect the computed quantities.
■ Use *Line model/Compute Spice models*, for storing the SPICE model. Again, type *mycables.lib* as file name. Select and D and E models, with names like *ps1d* and *ps1e* in the four **Name** boxes. Select "All terms" in order to take the resistance matrix into account. Select *Add*.

You just added two models to the mycable.lib library file. file of \Sline\files. This library mycable.lib should now be copied in the PR directory (\Spice\PR or \Spice4\PR) in order to be used by the ICAP/4 SPICE simulator. Note that for the SPICE model just created, the stp1p.sym model included in \Spice\SN\SYMBOLS\SYMSPLNE is appropriate.

Following the appropriate steps, you can also edit the new library by adding appropriate *SRC and *SYM lines to each model (see the supplied splne223.lib library), then compile a new data base, and thus become able to access your models created with SpiceLine in the SpiceNet *Part/X part/Browser* window.

■ You should now perform a SPICE simulation of a *name*.cir SPICE project including for instance the **ps1d** subcircuit according to the requirements of § 3.8. You may for instance:
— copy the 6 files stp1p.* found initially in the for_cir directory of disk 4 (see step 3 of § 4.2) under the name pscirc.*,
— open a ICAP/4 simulation for pscirc.cir
— in SpiceNet, change the subcircuit name of the cable for ps1d, give the appropriate line length, and in the Edit Control dialog, add *INCLUDE MYCABLE.LIB.

■ Simulate this circuit.
■ Start SpiceLine again and use *File/Open project* to select the *trial2.pjt* project.
■ Use *Radiated fields/Describe position*, to specify the line length, the coordinate X, Y, Z of the point at which the fields will be computed, and the abscissa Zx at which the last element *V* and *I* vectors will be computed (see § 3.8).
■ Use *Radiated fields/Select excitation and compute*, and select *name*.OUT (for instance pscirc.out). The calculation begins first with the calculation of *I* and *V*, then with the calculation of the radiated fields **E** and **H**. The results can be saved, after the calculation is completed, into a file *result*.OUT that will contain the field values and the current and voltage values, as a function of frequency,.

This file can be viewed in a text editor, or used in the IntuScope post processor of the ICAP/4 Windows package.

Please note that this example is built with the STPD1 SpiceLine project and with stp1p1 SPICE project, which originally implements the ST12E model (see § 4.1.5). The result obtained with these projects are shown in Appendix A.

5. References

[1] F. BROYDE, E. CLAVELIER, C. HYMOWITZ :
"Simulation Crosstalk and Field to Wire Coupling with a Spice Simulator", *IEEE Circuits and Devices*, Vol. 8, No. 5, September 1992, pp. 8-16.

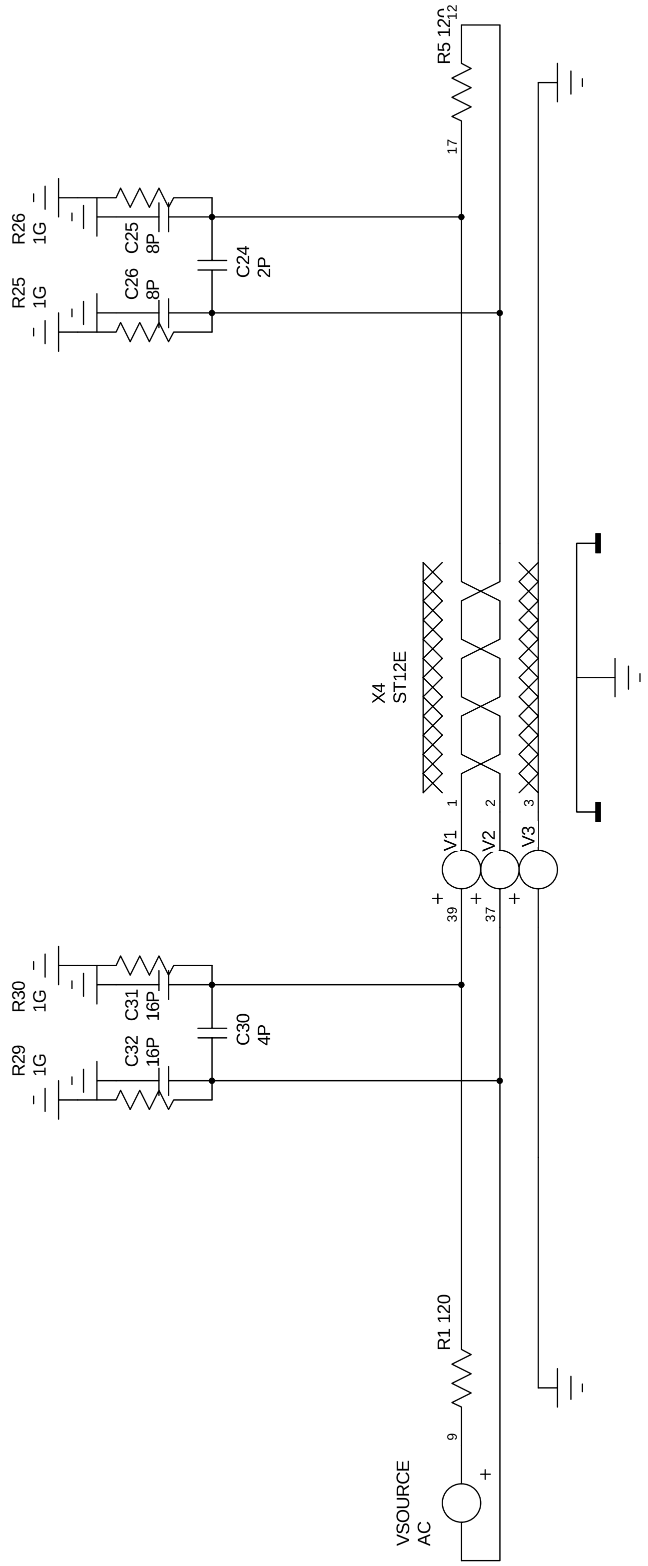
[2] F. BROYDE, E. CLAVELIER, L. HOEFT :
"Comments on “A SPICE Model for Multiconductor Transmission Lines Excited by an Incident Electromagnetic Field", *IEEE Transactions on EMC*, Vol. 38, No. 1, February 1996, pp. 104-108.

[3] F. BROYDE, E. CLAVELIER, O. DAGUILLON, A. ZEDDAM :
“Comparaison de l’émission des câblages de télécommunication”, actes du 9ième colloque international et exposition sur la CEM, Brest 8-11 juin 1998, pp B5-13 à B5-18.

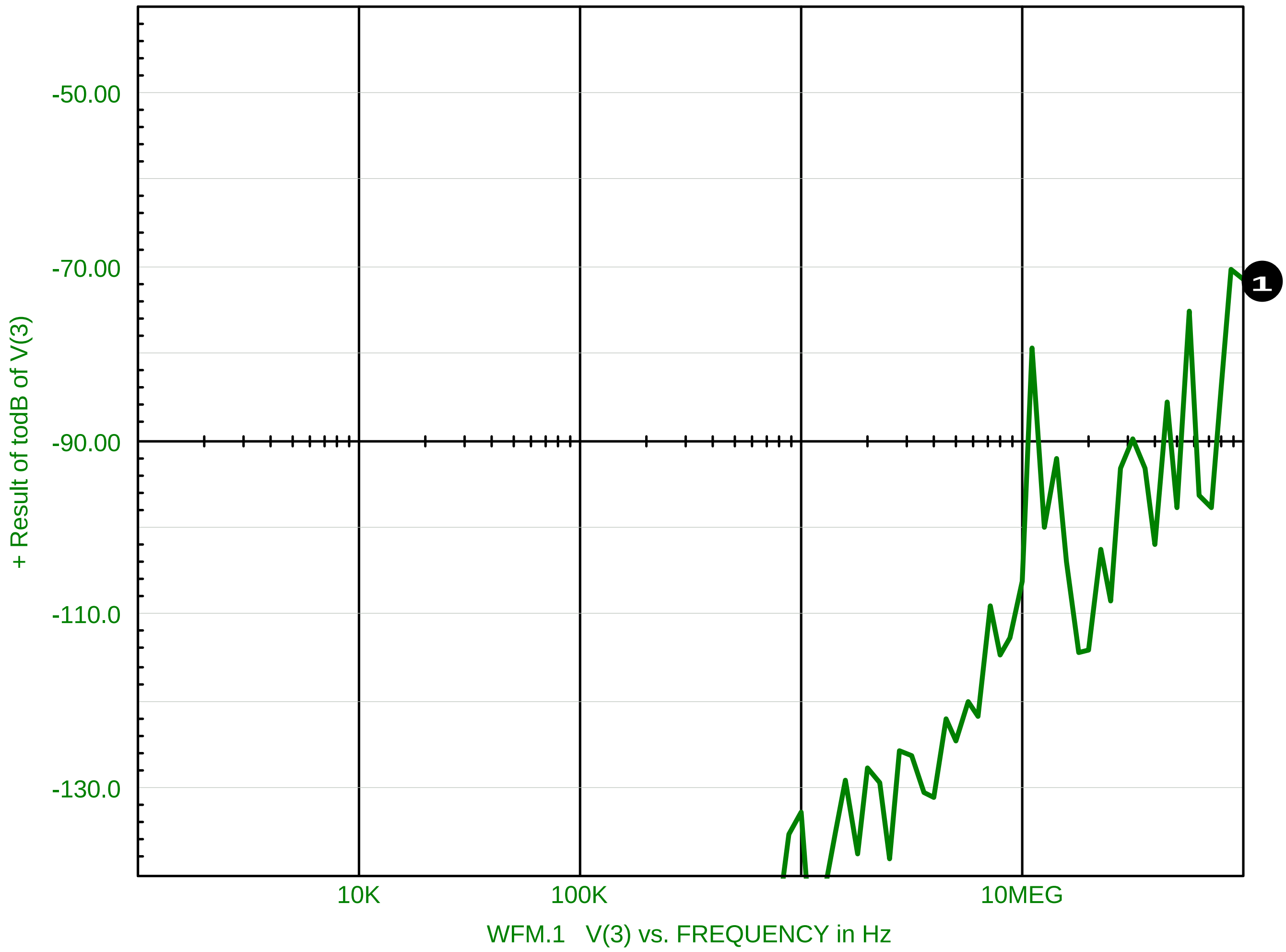
6. Index

7. Appendix: Results obtained with Example 2

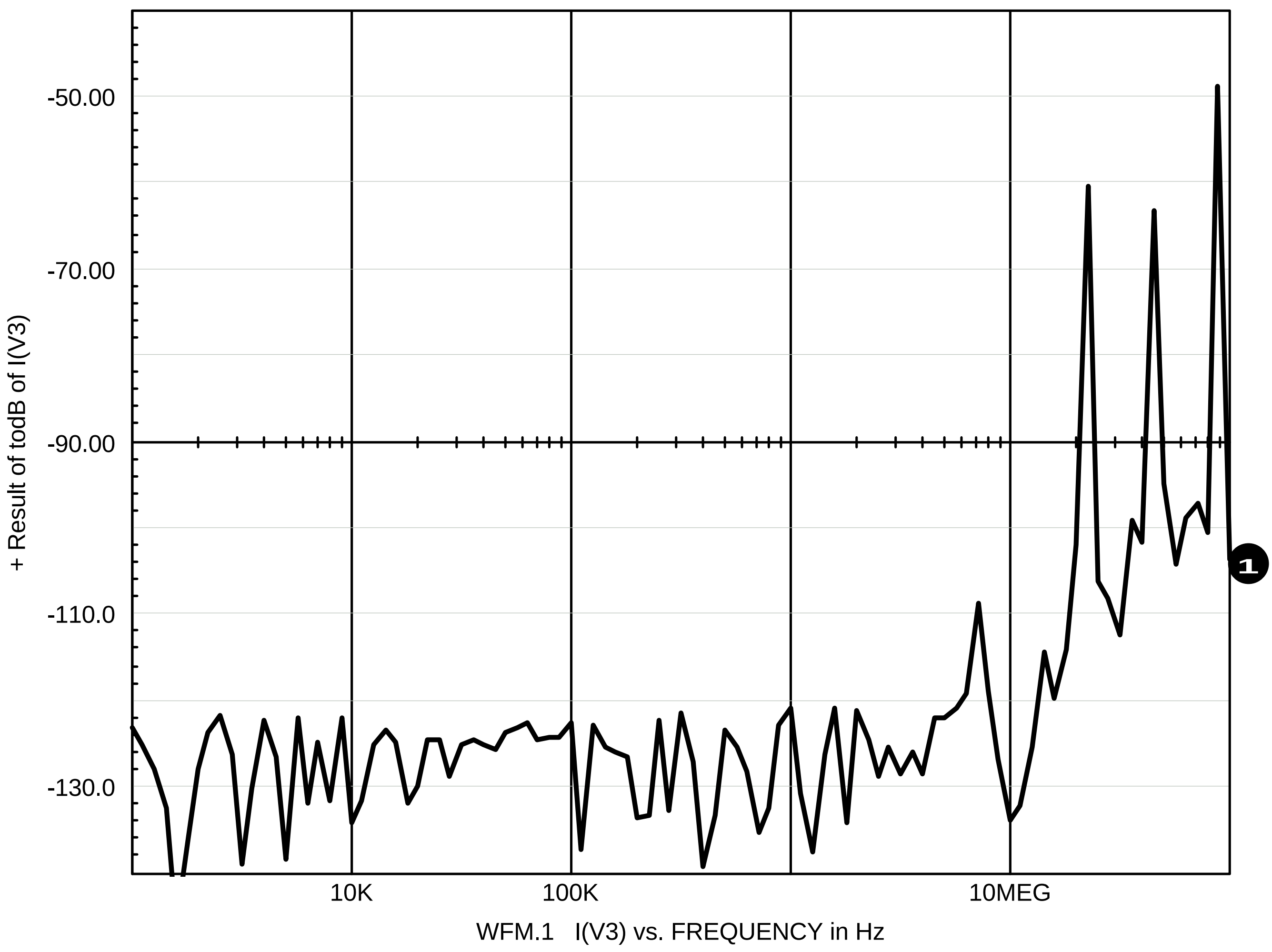
The next pages present the results one could obtain if one follows the steps of § 4.6. Page A1 shows the stp1p circuit. The length of the cable is 10 m. The characteristic impedance for the differential mode is close to 120 Ω and the source amplitude is 1 V. Page A2 shows the stp1p1.cir file. Page A3 shows the common mode voltage in the middle of the cable, in dB above 0,5 V. Page A4 shows the common mode current in dB above 4.167 mA at the same point. Page A5 shows the radiated electric field at the point of coordinate (0.1 m, 1 m, 0 m), in dB above 1 µV/m. Page A6 shows the magnetic field at the same point, in dB above 1 µA/m. Page A7 shows. the current in dB above 1 mA induced by a field normalized to 1 V/m at the origin, produced by an electric dipole at the point of coordinate (0.1 m, 1 m, 0 m), for the *x* polarization. Page A8 shows the current in dB above 1 mA induced by a field normalized to 1 A/m at the origin, produced by a magnetic dipole at the same point, for the *y* polarization.


R5 120
12
17
R26
1G
C25
8P
R25
1G
C26
8P
C24
2P
X4
ST12E
1
2
3
V1
V2
V3
39
37
R30
1G
C31
16P
R29
1G
C32
16P
C30
4P
R1 120
9
VSOURCE
AC

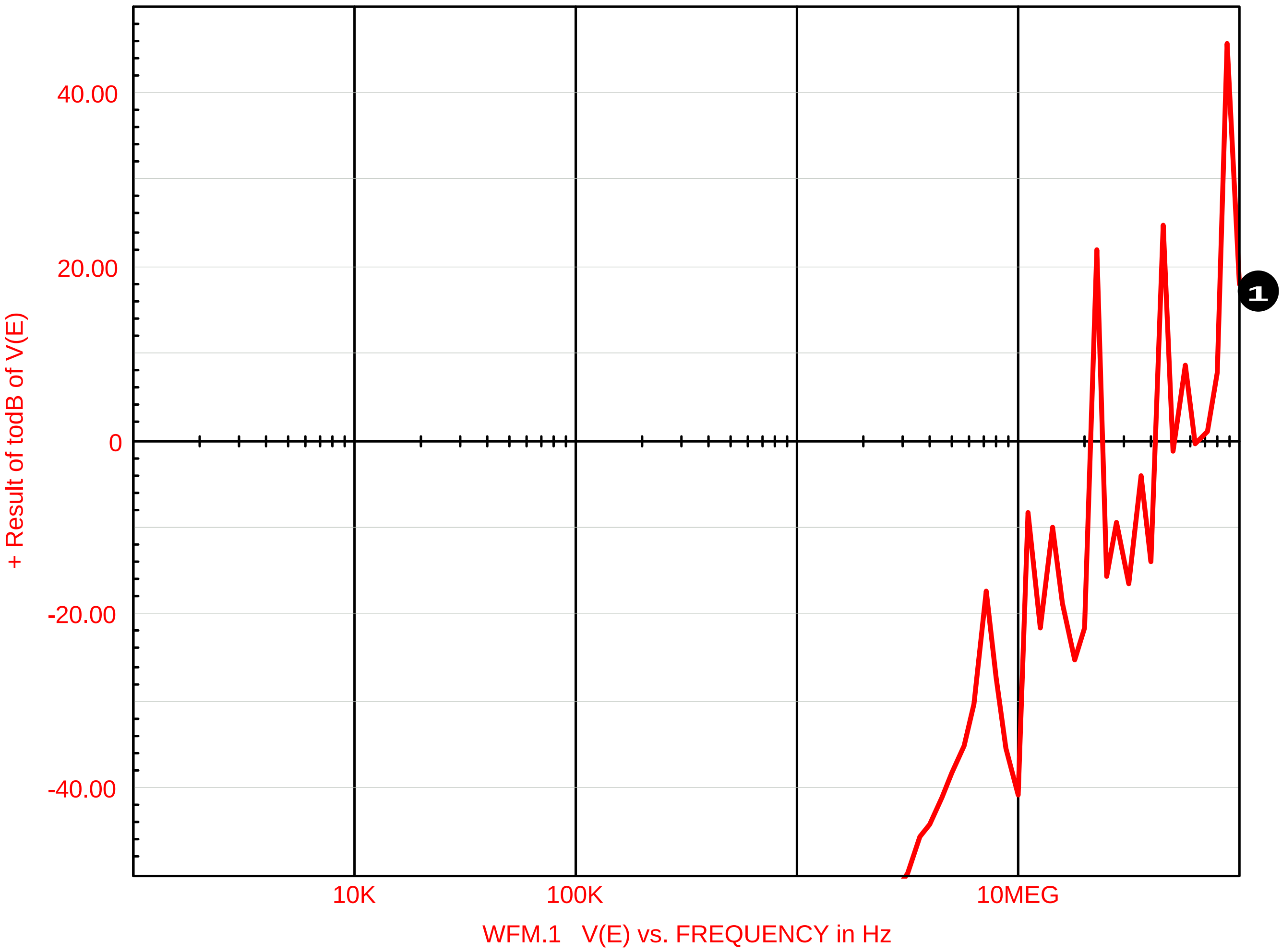
```
E:\SPICE4\CIRCUITS\utp4p1
*SPICE_NET
.AC DEC 20 1000HZ 100MEGHZ
.PRINT AC V(1) VP(1)
.PRINT AC I(V1) IP(V1)
.PRINT AC V(2) VP(2)
.PRINT AC I(V2) IP(V2)
.PRINT AC V(3) VP(3)
.PRINT AC I(V3) IP(V3)
*INCLUDE SPLNE223.LIB
V2 37 2
R1 9 39 120
VSOURCE 9 37 AC 1
R5 17 12 120
C24 12 17 2P
R25 0 12 1G
R26 0 17 1G
C25 0 17 8P
C26 0 12 8P
C30 37 39 4P
R29 0 37 1G
R30 0 39 1G
C31 0 39 16P
C32 0 37 16P
V3 0 3
X4 1 2 3 17 12 0 0 ST12E {L=10 }
V1 39 1
.END
```



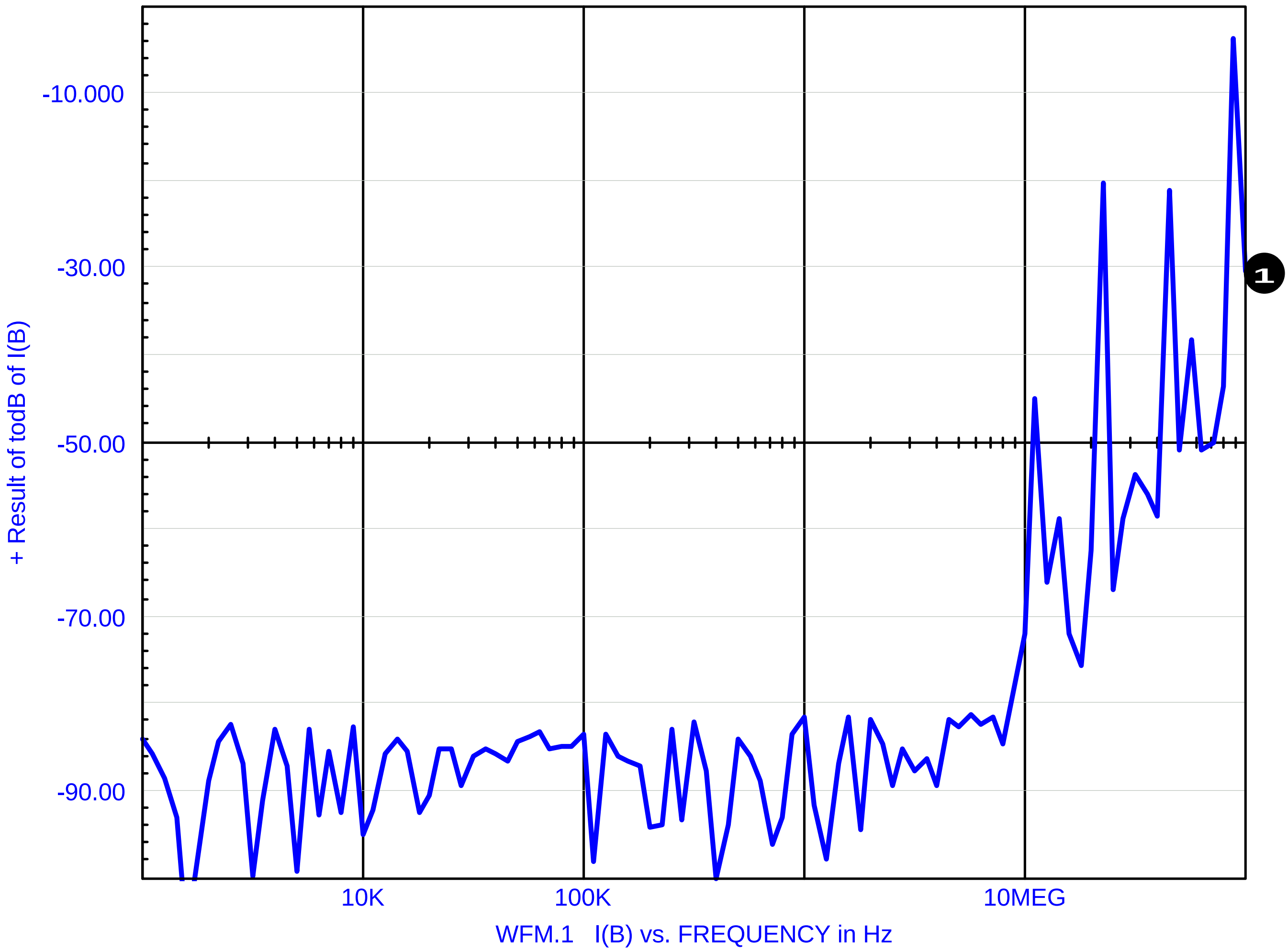
Common mode voltage



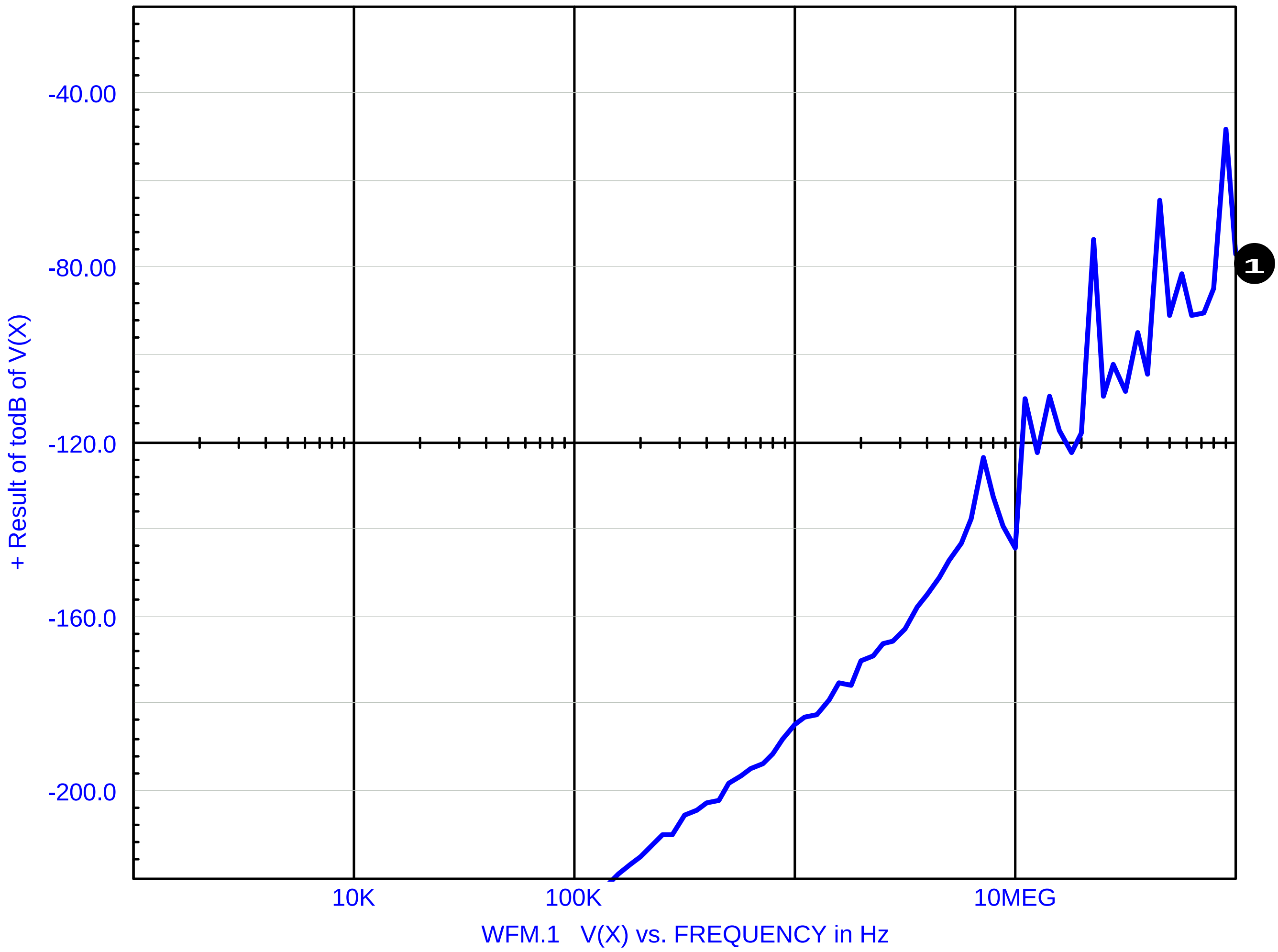
Common mode current



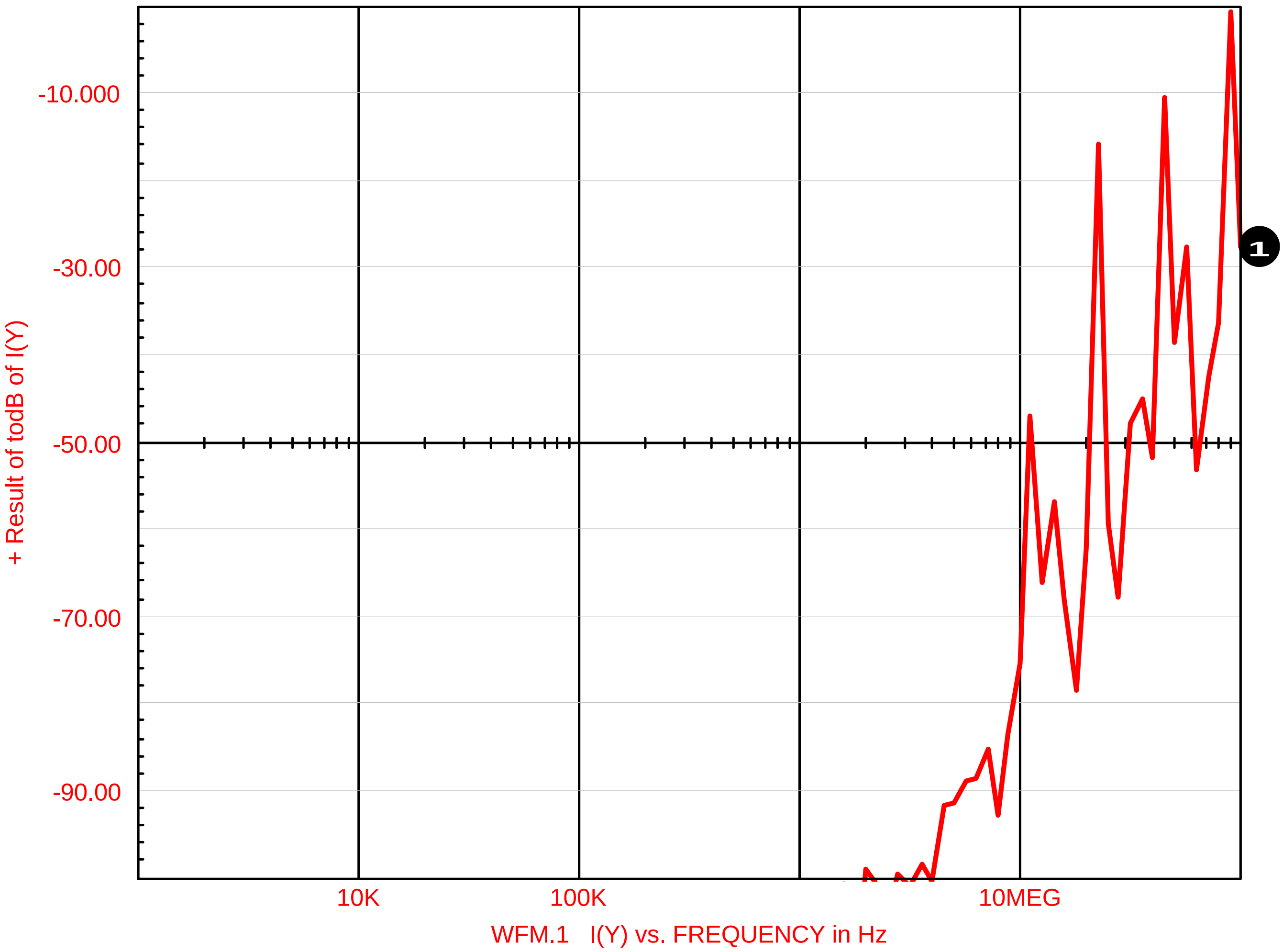
Radiated electric field



Radiated magnetic field



Current induced by an electric dipole



Current induced by a magnetic dipole