

NOTICE: This document contains references to Agilent Technologies. Agilent's former Test and Measurement business has become Keysight Technologies. For more information, go to **www.keysight.com**.





Agilent Technologies

**EMPro 2012
May 2012
FDTD Simulation**

© Agilent Technologies, Inc. 2000-2011

5301 Stevens Creek Blvd., Santa Clara, CA 95052 USA

No part of this documentation may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Agilent Technologies, Inc. as governed by United States and international copyright laws.

Acknowledgments

Mentor Graphics is a trademark of Mentor Graphics Corporation in the U.S. and other countries. Microsoft®, Windows®, MS Windows®, Windows NT®, and MS-DOS® are U.S. registered trademarks of Microsoft Corporation. Pentium® is a U.S. registered trademark of Intel Corporation. PostScript® and Acrobat® are trademarks of Adobe Systems Incorporated. UNIX® is a registered trademark of the Open Group. Java™ is a U.S. trademark of Sun Microsystems, Inc. SystemC® is a registered trademark of Open SystemC Initiative, Inc. in the United States and other countries and is used with permission. MATLAB® is a U.S. registered trademark of The Math Works, Inc.. HiSIM2 source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code in its entirety, is owned by Hiroshima University and STARC.

The following third-party libraries are used by the NlogN Momentum solver:

"This program includes Metis 4.0, Copyright © 1998, Regents of the University of Minnesota", <http://www.cs.umn.edu/~metis> , METIS was written by George Karypis (karypis@cs.umn.edu).

Intel@ Math Kernel Library, <http://www.intel.com/software/products/mkl>

SuperLU_MT version 2.0 - Copyright © 2003, The Regents of the University of California, through Lawrence Berkeley National Laboratory (subject to receipt of any required approvals from U.S. Dept. of Energy). All rights reserved. SuperLU Disclaimer: THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

AMD Version 2.2 - AMD Notice: The AMD code was modified. Used by permission. AMD copyright: AMD Version 2.2, Copyright © 2007 by Timothy A. Davis, Patrick R. Amestoy, and Iain S. Duff. All Rights Reserved. AMD License: Your use or distribution of AMD or any modified version of AMD implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. AMD Availability:

<http://www.cise.ufl.edu/research/sparse/amd>

UMFPACK 5.0.2 - UMFPACK Notice: The UMFPACK code was modified. Used by permission. UMFPACK Copyright: UMFPACK Copyright © 1995-2006 by Timothy A. Davis. All Rights Reserved. UMFPACK License: Your use or distribution of UMFPACK or any modified version of UMFPACK implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included.

UMFPACK Availability: <http://www.cise.ufl.edu/research/sparse/umfpack> UMFPACK (including versions 2.2.1 and earlier, in FORTRAN) is available at <http://www.cise.ufl.edu/research/sparse> . MA38 is available in the Harwell Subroutine Library. This version of UMFPACK includes a modified form of COLAMD Version 2.0, originally released on Jan. 31, 2000, also available at <http://www.cise.ufl.edu/research/sparse> . COLAMD V2.0 is also incorporated as a built-in function in MATLAB version 6.1, by The MathWorks, Inc. <http://www.mathworks.com> . COLAMD V1.0 appears as a column-preordering in SuperLU (SuperLU is available at <http://www.netlib.org>). UMFPACK v4.0 is a built-in routine in MATLAB 6.5. UMFPACK v4.3 is a built-in routine in MATLAB 7.1.

Errata The ADS product may contain references to "HP" or "HPEESOF" such as in file names and directory names. The business entity formerly known as "HP EEsof" is now part of Agilent Technologies and is known as "Agilent EEsof". To avoid broken functionality and to maintain backward compatibility for our customers, we did not change all the names and labels that contain "HP" or "HPEESOF" references.

Warranty The material contained in this document is provided "as is", and is subject to being changed, without notice, in future editions. Further, to the maximum extent

permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this documentation and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license. Portions of this product include the SystemC software licensed under Open Source terms, which are available for download at <http://systemc.org/> . This software is redistributed by Agilent. The Contributors of the SystemC software provide this software "as is" and offer no warranty of any kind, express or implied, including without limitation warranties or conditions or title and non-infringement, and implied warranties or conditions merchantability and fitness for a particular purpose. Contributors shall not be liable for any damages of any kind including without limitation direct, indirect, special, incidental and consequential damages, such as lost profits. Any provisions that differ from this disclaimer are offered by Agilent only.

Restricted Rights Legend U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in Software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

FDTD Simulation

FDTD Simulation Overview	6
Specifying FDTD Simulation Setup	13
Defining a Grid and Mesh	23
Defining Grid Settings	24
Defining Mesh Settings	35
Using Conformal FDTD Meshing	44
Viewing FDTD Simulation Results	47
Viewing 2D Plots	48
Viewing 3D Fields	58
Viewing FDTD Default Output	71
Adding TDR Waveforms	78
GPU Acceleration for FDTD Simulations	81
Example: How to Simulate using FDTD	84
Creating a Microstrip Patch Antenna Simulation	85
Creating a Low Pass Filter Simulation	103
Creating a Monopole Antenna on a Conducting Box Simulation	119
Creating a Simple SAR Calculation Simulation	133
Validating SAR Calculations	145

FDTD Simulation Overview

You can create an FDTD simulation by using the *Setup FDTD Simulation* (fdd) window. In this window, specify the FDTD setup by specifying the required values. After setting up an project, you can run calculations on the geometry and store project simulations in the Simulations window. From this window, you can create, queue and run simulations. After you have run simulations and completed the required calculations, you can view results. The Simulations workspace window provides an interface to queue projects in the calculation engine.

This window lists the name of every simulation that is for a project. You can choose how to run the simulation on your machine by selecting an option from the **Queue** drop-down list. From this list, choose a value to specify whether to run the simulation on the CPU or on the hardware card with Acceleware hardware acceleration.

Starting the Calculation Engine

There are two ways to start the EMPrFDTD engine. The first way is to launch it from EMPro in the *Simulations* workspace window. The second way is to start the calculation from a command line. This is preferable for calculations that require a large amount of memory in order to free memory used by the interface.

Running EMPrFDTD Within the EMPro Interface

Calculations launched from EMPro are sent to the EMPrFDTD engine from the *Simulations* workspace window. A list of every simulation that has been created within the *New Simulation* dialog is listed in the main window. When a simulation is ready to run, send it to EMPrFDTD by selecting the simulation and clicking the *Add To Queue* button, located to the right of *Selected Simulation* at the top of the window. Although multiple simulations can be queued at once, only one simulation is run in EMPrFDTD at a time. Once the *Play* button is pressed, each simulation will run one at a time until all of the queued simulations have terminated.

Running EMPrFDTD From the Command Line

In some cases, it is useful to run EMPrFDTD from a command line. This will prevent EMPro from overloading since it will not be allocating memory for EMPrFDTD in addition to its normal memory requirements. This is not an issue for most calculations, but it may cause problems when the memory required to run a calculation approaches the limitations of computer memory. In this case, closing EMPro and running EMPrFDTD from the command line will free up any memory used by EMPro.

To run EMPrFDTD, navigate to the appropriate project folder and run the command:

Windows: `empr fdd.exe [options]`

Mac OS X/Linux: `empr fdd [options]`

For normal operation no [options] need be specified, but the table below lists several

options that may be of interest.

Command Line Options

option	Function
v	Verbose mode, prints progress of calculation.
proc N	Sets the number of processors to use for a calculation with the multi-processor module (optional).
loadstatic	Loads fields saved by the static voltage solver.
savestaticonly	Saves fields by the static voltage solver and terminates before time-stepping.
savestatic	Saves fields by the static voltage solver and continues timestepping.
hardware	Uses Acceleware hardware acceleration; in the case that an error occurs initializing Acceleware, the engine uses the software to run calculation.
hardwareonly	Uses Acceleware hardware acceleration; in the case that an error occurs initializing Acceleware, the calculation is terminated.
forcetemporarydatatomemory	Forces steady-state data to be stored in memory rather than in an external file.
saronly	Calculation skips time-stepping and only performs post-processing of SAR data.

Running the Calculation Remotely

Running a remote calculation is an alternative way to potentially speed up calculation time and free local computer resources for other operations. This section will explain how to copy files and launch the calc engine on a remote computer.

Note

If your EMPro project is stored on a file system which is remotely accessible, you can avoid the process of copying files back and forth from the remote machine. Simply log into the remote machine, navigate to the simulation directory and launch the calc engine.

Running Simulations Remotely

1. **Create the simulation.** After creating the simulation from the *Simulations* workspace window, click the *Create Simulation Only* button to save the simulation to the *Simulations* folder where your EMPro project is stored.

You can find the path to this directory in the first line of text under the *Summary* tab

2. **Copy files to the remote machine.** For simplicity, you can copy the whole *Simulations* directory (you'll need to copy it recursively, since it contains subdirectories). However, only a few files in this directory are required by the calculation engine:

```
project.xsim
Run*/geometry.input
Run*/mesh.input
Run*/project.input
```


**Note**

The * in the file name represents the simulation number, (e.g., Run0001/geometry.input, Run0002/geometry.input, etc).

For Windows users, it may be easiest to zip each file individually and transfer them to the remote machine over an ftp connection.

For Mac OS X users, you can save these files in a compressed archive with the following command:

```
tar -czf inputFilesForCalcEngine.tar.gz project.xsim Run*/geometry.input
Run*/mesh.input Run*/project.input
```

Then copy the *inputFilesForCalcEngine.tar.gz* file to the remote machine and extract it using:

```
tar -xzf inputFilesForCalcEngine.tar.gz
```

3. **Run the simulation** Log in to the remote machine, change directories to the *Simulations* folder, and run EMPrFDTD for the project.

**Note**

For more information on running the calc engine from the command line, see [Running EMPrFDTD From the Command Line](#).

4. **Copy files back to local machine.** In order to view your results from the EMPro GUI, you'll need to copy all of the *Simulations* files from the remote machine to your local machine, *with the exception of the files listed above*. That is, after running the simulation on the remote machine, all *new* files should be copied back into the *Simulations* folder on your local machine. The `tar` utility makes this easy. On the remote machine, change directories to the *Simulations* directory and run

```
tar -czf outputFilesFromCalcEngine.tar.gz -exclude=project.xsim -
exclude=Run*/geometry.input -exclude=Run*/mesh.input -
exclude=Run*/project.input *
```

For Windows users, zip the *outputFilesFromCalcEngine.tar.gz* file and send it back to your local machine over the ftp connection. You can then extract the files using Winzip. Overwrite any old files with newer versions of the same file.

For Mac OS X users, copy the *outputFilesFromCalcEngine.tar.gz* file to your local machine, change directories and extract it using:

```
tar -xzf outputFilesFromCalcEngine.tar.gz
```

5. **Refresh results.** To make your results available from the EMPro GUI, click *Refresh* in the *Results* workspace window.

Remote SAR Post-processing

1. **Setting up post-processing.** After requesting post-processing from the RESULTS workspace window, in the subsequent dialog box, tell the application to not run the calculation right away.
2. **Copy raw SAR data, calc engine input files, and SAR request files.** The following files are required for post-processing in the *Simulations* folder on the

remote machine:

```
project.xsim
Run*/geometry.input
Run*/mesh.input
Run*/project.input
Run*/SteadyStateOutput//SAR_Raw_Sensor..sar.gz
Run*/SteadyStateOutput*/SAR_Raw_Sensor.stats
Run*/request.sar
```

The * in the file name represents the simulation number, (e.g., Run0001/geometry.input, Run0002/geometry.input, etc).

For Windows users, it may be easiest to zip each file individually and transfer them to the remote machine over an ftp connection.

For Mac OS X users, change the directory to *Simulations* and run the following command on your local machine:

```
tar -czf sarRawData.tar.gz project.xsim Run*/geometry.input Run*/mesh.input
Run*/project.input Run*/SteadyStateOutput//.sar.gz Run*/request.sar
```

Then copy the *sarRawData.tar.gz* file to the remote machine and extract it using:

```
tar -zxvf sarRawData.tar.gz
```

3. **Run SAR averaging on the remote machine.** On the remote machine, change directories to the *Simulations* directory, and execute the calc engine application using the `-saronly` command line flag.



Note

For more information on running the calc engine from the command line, see [Running EMPrFDTD From the Command Line](#).

4. **Copy SAR results back to the local project.** If you've already copied your other results back to the local project, you'll need to package up your SAR results and copy them back. From the remote machine, copy the following files to the *Simulations* directory on the local machine:

```
Run*/status
Run*/SteadyStateOutput/*/*gsar.gz
Run*/SteadyStateOutput//.infosar1g
Run*/SteadyStateOutput//.infosar10g
Run*/SteadyStateOutput//.infoseq
Run*/SteadyStateOutput//.*gssq
Run*/SteadyStateOutput//.stats
```

Then run the following command:

```
tar -czf sarResults.tar.gz Run*/status Run*/SteadyStateOutput/*/*gsar.gz
Run*/SteadyStateOutput//.infosar1g
Run*/SteadyStateOutput//.infosar10g Run*/SteadyStateOutput//.infoseq
Run*/SteadyStateOutput//.gssq Run*/SteadyStateOutput//.stats
```

**Note**

Notice that this command sequence retrieves all of the SAR averaging results, including those done with your initial calc engine run.

For Windows users, zip the *sarResults.tar.gz* file and send it back to your local machine over the ftp connection. You can then extract the files using Winzip. Overwrite any old files with newer versions of the same file.

For Mac OS X users, copy the *sarResults.tar.gz* file to your local machine, change directories and extract it using:

```
tar -xzf sarResults.tar.gz
```

- 5. Refresh results.** To make your results available from the EMPro GUI, click **Refresh** in the *Results* workspace window.

Calculation Considerations

The calculation portion of EMPro may be quite lengthy depending on the application. A few guidelines are provided here for estimating computer resources, monitoring the progress of the calculation, and avoiding calculation instability.

**Note**

For information about defining proper termination criteria to ensure that the calculation has finished, see *Specifying Termination Criteria* (fddd).

Computer Resources Estimation

EMPrFDTD will give a time estimate while the calculation is running. This is recalculated every time EMPrFDTD updates its status based on how much time passed since the last update and the remaining number of timesteps. It is not a completely accurate estimation since it does not consider data such as near-field samplings which may only be saved during certain portions of the calculation. Also, the estimate does not include any post-processing which may occur. Of special note are the SAR averages since they are computed in post-processing and may require a significant amount of calculation time.

**Note**

A quick way to estimate the amount of memory that EMPrFDTD will need for a given problem is to multiply the number of cells in the geometry by 27. If magnetic materials are included in the geometry (any material with non-free space permeability), multiply by 30, rather than 27. The resulting number is the approximate number of bytes needed to calculate the project.

Multi-Processing Modules

EMPrFDTD has the ability to do both threaded and Message Passing Interface (MPI) calculations. Both of these capabilities are optional features which may be added to the calculation engine.

Multi-threaded calculations use shared memory and are intended for computers with multiple processors and/or cores on a single motherboard. The overhead for the multi-threading routines can cause very small calculations to run slower when more than one

thread is selected.

Note

In general, the multi-threaded option should not be used when the number of FDTD cells in the geometry is less than one million.

The number of cells is computed simply from the **X**, **Y**, and **Z** dimensions (in cells) of the geometry space. The number of threads used for a calculation is defined in the *Queue* drop-down box of the *Simulations* workspace window.

Specify the number of processors that are to be used for the calculation under *Maximum # Of Threads*.

To specify the number of threads from the command line, the "-proc N" option should be used, where N is the number of threads.

Note

For a summary of command-line options, refer to [Running EMPrFDTD From the Command Line](#).

MPI calculations can be executed on cluster of computers that are connected by a network. At present, only Linux computers are supported and each computer must have the same MPI libraries installed. To run an MPI calculation, see the documentation for the MPI library that is installed on all machines in the cluster. This documentation will give instructions on how to start an application using the MPI tools.

Acceleware Hardware Acceleration

EMPro has optional Acceleware hardware acceleration. The hardware is available in the form of graphics cards, which can replace existing graphics cards or act from a stand-alone computer. When this hardware is available, it may be enabled within the *Simulations* workspace window under the *Queue* drop-down box. To enable it from the command line, the "-hardware" option should be used.

Monitoring Calculation Progress

While EMPrFDTD is running, its progress will be updated periodically. When launched from the EMPro interface, the progress of the calculation will be printed in the *Output* tab located within the *Simulations* workspace window, as seen in the figure below. When running from a command line, the progress will be printed to the window that was used to start the calculation if the "-v" option is used.

Calculation Stability

Improper application of the outer boundaries can lead to unstable calculations. Typically the stability of a calculation depends on a few simple guidelines involving boundaries and source placement. Because an absorbing boundary condition like LIAO tries to simulate free space, it requires that a certain amount of continuity be present in the cells leading to the boundary. The cross section at a boundary must be the same for at least 10 cells in from the boundary.

Note

For many problems, a free-space border of 10-20 cells is the best way to ensure stability and accurate performance of the outer boundary.

Another rule for stability is that no source can be placed within 10 cells of an absorbing boundary. An unstable calculation is easily determined by viewing the line plots of time domain data or by viewing the field snapshots. When automatic convergence is enabled, EMProFDTD can automatically detect an unstable simulation and terminate it.

In some cases instability can be introduced by a frequency-dependent material. If such materials are used in the calculation and an instability results, it may be necessary to change the material parameters or reduce the calculation timestep.

See Also


FDTD Overview (appendixb)

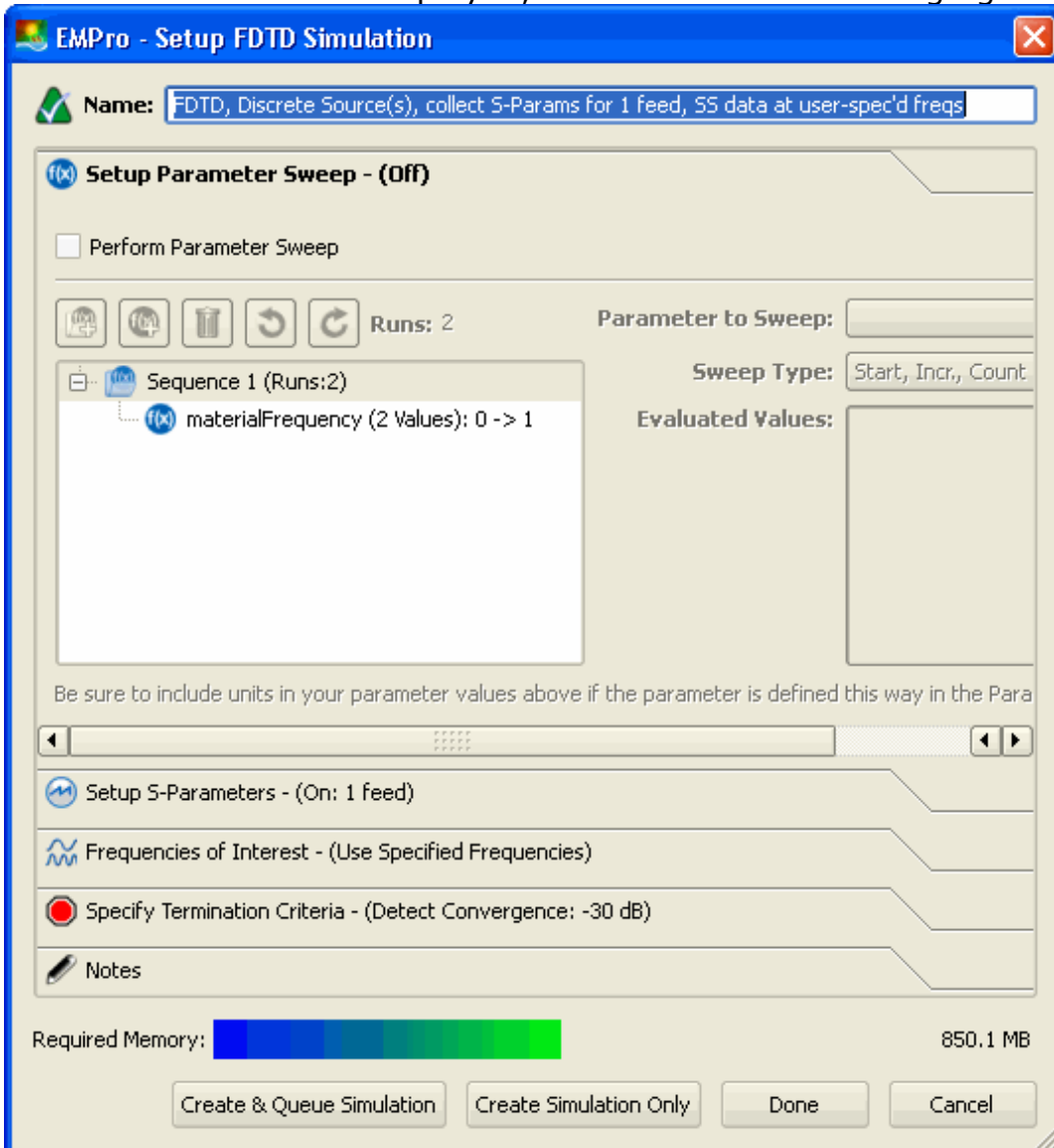
Specifying FDTD Simulation Setup

Before running an FDTD simulation, you need to specify the simulation options that are specific to the FDTD simulator. You can create an FDTD setup by specifying the required values in the Setup FDTD Simulation window. This section provides information about how to create an FDTD simulation setup.

Selecting the FDTD Simulator

You can specify simulation options that are specific to the FDTD simulator in the Setup FDTD Simulation window. To open the Setup FDTD Simulation window:

1. Select **FDTD Simulation** from a drop-down list on the Simulations toolbar, as shown in the following figure:
2. Click **Setup** () to create and edit the setup of a new simulation. The **Setup FDTD Simulation** window is displayed, as shown in the following figure:



Specifying Parameter Sweep Values


A parameter sweep can be set up so that a script will loop through a particular parameter multiple times and run a calculation at each iteration. For example, if the parameter is antenna length, a script can loop through various antenna lengths, and run a calculation at each length. The *Sweep Type* is defined by the values it will collect *Start, Incr., Count* values, or *Start, End, Count* values, to generate the *Ending Value* or *Increment* values, respectively. A third option is to select *Comma Separated* for the parameter sweep, which do not necessarily have to be at evenly spaced intervals.

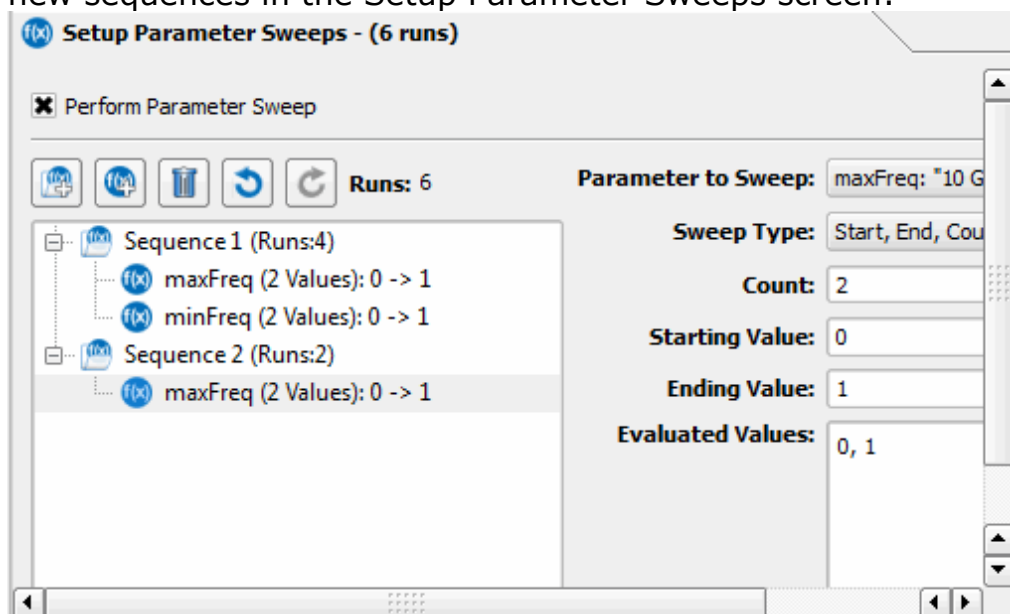
The Setup Parameter Sweep screen enables you to specify the sweep parameter, sweep type, and values. You can specify the following parameters:


Option	Description
Parameter to sweep	Specify the required parameter. You can specify multiple parameter sweep values.
Sweep type	Select the required sweep type.
Count	Specifies the number of variables to be used in a simulation.
Starting value	Specify the start value of the parameter.
Ending Value	Specify the end value of the parameter.
Evaluated Values	Displays all the values that are used between the start value and end value.

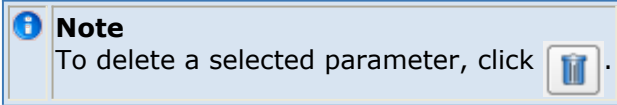
Adding New Parameters

To add new parameters:

1. Select the **Perform parameter Sweep** check box in the Setup Parameter Sweeps screen.
2. Click  to add a new parameter sequence. The following figure displays a list of new sequences in the Setup Parameter Sweeps screen:



3. Select a sequence and click  to add new parameters to the sequence.



To specify parameter sweeps:

1. Select the **Perform parameter Sweep** to enable the options available in the Setup Parameter Sweeps screen.
2. For each parameter, specify values for the following options:
 - Select a value from the **Parameter to sweep** drop-down list.
 - Select a value from the **Sweep type** drop-down list.
 - Specify a count value.
 - Type a value in the **Starting Value** text box.
 - Type a value in the **Ending Value** text box.
 - The **Evaluated Values** box automatically displays values.
3. Select other parameters in **Sequence** and specify the required values for each option.

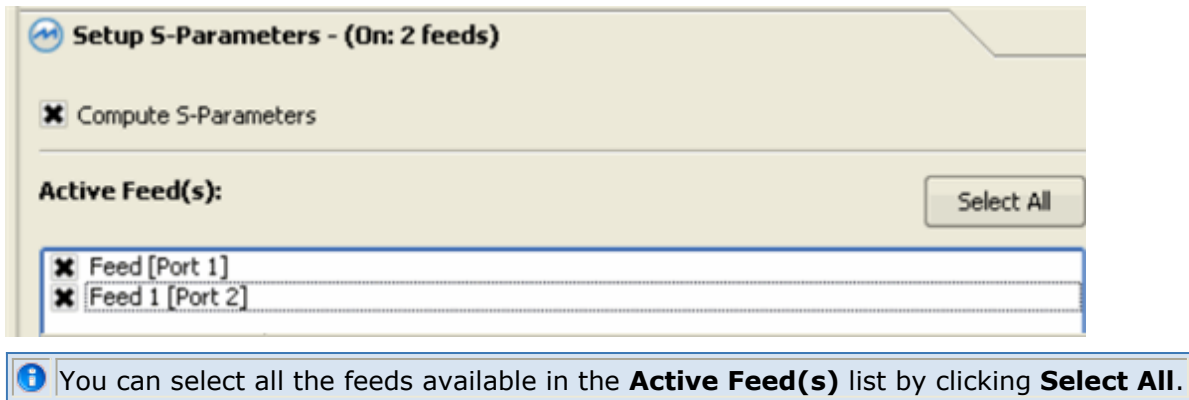
After running a simulation, the sequence of parameters are displayed in the **Summary** tab of the Simulation window.

Setting up S-parameters

To calculate a S-parameter for a project with multiple ports defined, the simulation will consider one port at a time to be the active port. The computation with that source active will provide data for a column of the S-parameter matrix. For example, if port one is excited in a three-port circuit, EMPro will determine S_{11} , S_{21} and S_{31} . If different ports are to be excited, a separate calculation must be performed with each port active. For example, if the full S-parameter matrix for a two-port problem is desired, two calculations must be performed with a different port active in each. EMPro will save the S-parameters for each run in separate files, differentiated by the active port number. If a parameter sweep is specified in the *Setup Parameter Sweep* tab for multiple parameters in addition to specifying multiple ports within the *Setup S-Parameters* tab, the parameter sweep will be performed for each individual port.

You can specify the feeds for calculating S parameters in the Setup FDTD Simulation window. To set up S parameters:

1. Click **Setup S Parameter** in the Setup FDTD Simulation window.
2. The **Compute S-Parameters** check box is selected, by default.
3. Select the check box for required feeds in the **Active Feed(s)** list, as shown in the following figure:



Specifying Frequencies

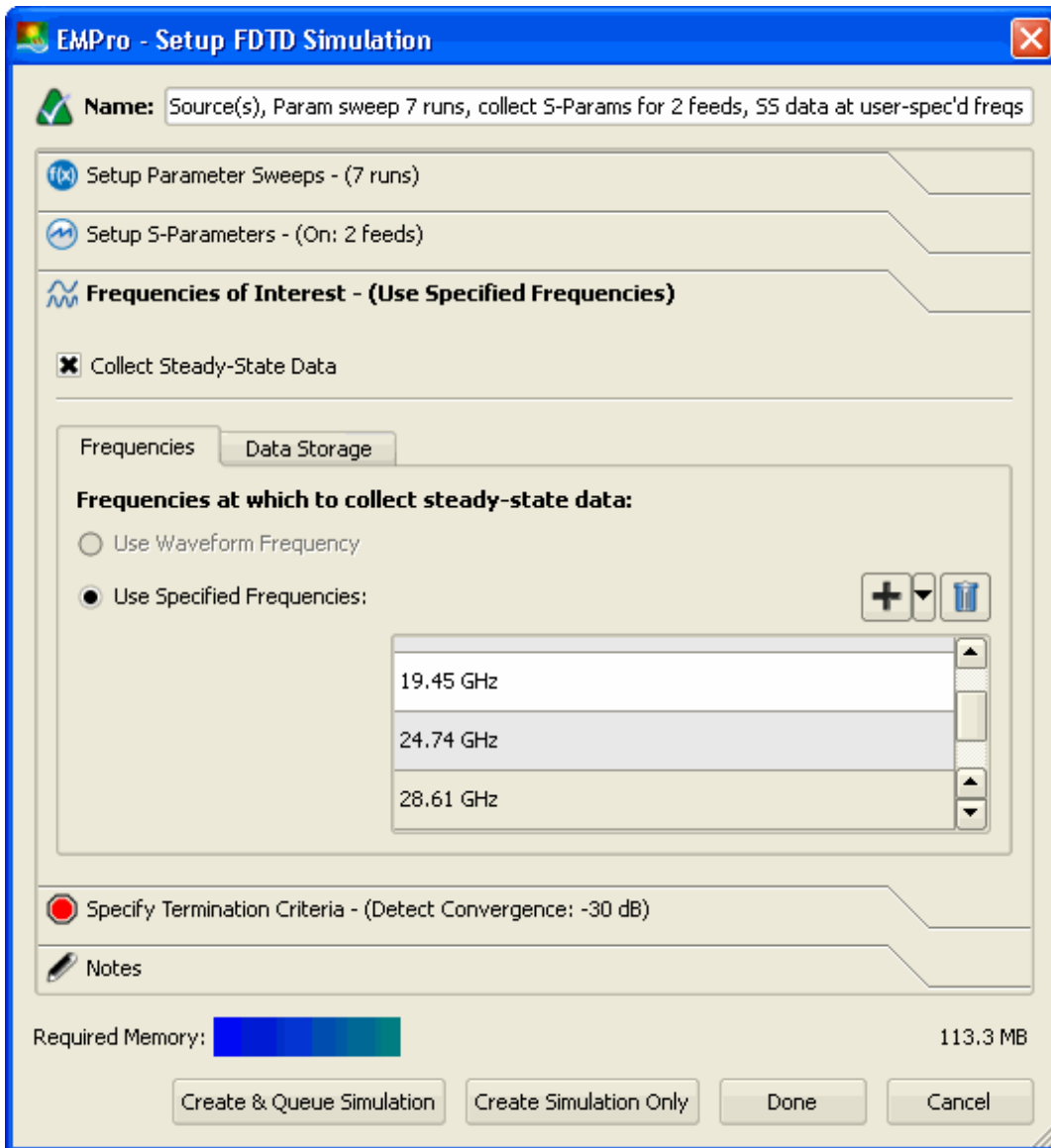
Within this section, the *Frequencies* tab specifies whether the simulation is a broadband (transient) or steady-state calculation. For broadband calculations, uncheck the *Collect Steady-State Data* box at the top of this tab. For steady-state calculations, check this box and choose whether the calculation is to only *Use Waveform Frequency* or to *Use Specified Frequencies* of interest. By specifying more than one frequency of interest, the calculation engine will essentially run a separate calculation at each discrete frequency by running DFT, saving each as its own run. This will therefore increase the calculation time in comparison to using only the waveform frequency.

In the *Data Storage* tab, the user has the ability to specify whether to save temporary data *In Memory* or *On Disk*. Saving the data in memory will speed up the calculation because there is no file saving or loading from disk, but it increases the memory requirements.

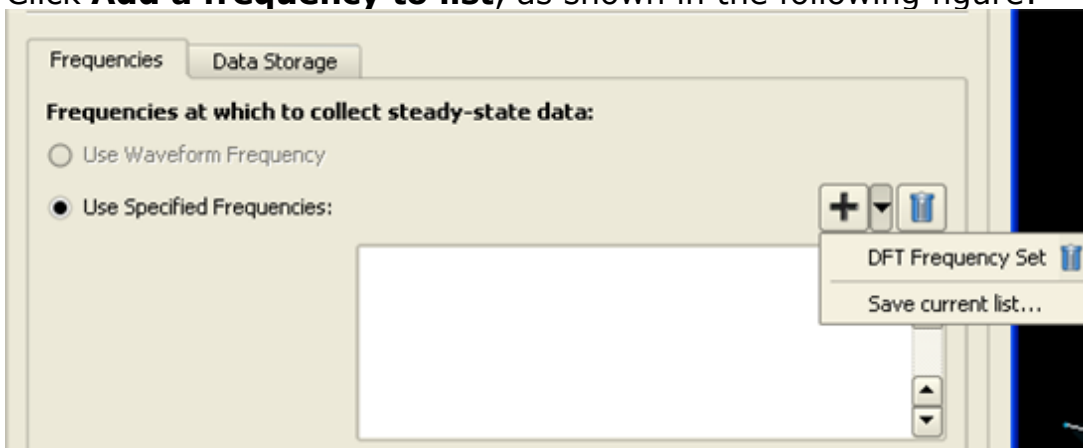
Also use this tab to designate which data to save for steady-state far zone post-processing. For a broadband excitation, the user has several options. Checking *Normalize Fields* will match calculated values to a sinusoidal run. Checking *Compute Dissipated Power* will calculate dissipated power based on electric field and magnetic field samplings. It is recommended to leave this box unchecked unless there is specific interest in dissipated power, because it can increase run time significantly due to sampling data over the entire geometry. *Sampling Interval* specifies how often to sample a data type. A sampling interval of one provides the most accurate results because it reduces the effects of aliasing. Timestep sampling options allow you to specify the frequency to resolve and sampling interval.

Using the Setup FDTD Simulation window, you can specify the frequency settings for your FDTD simulation.

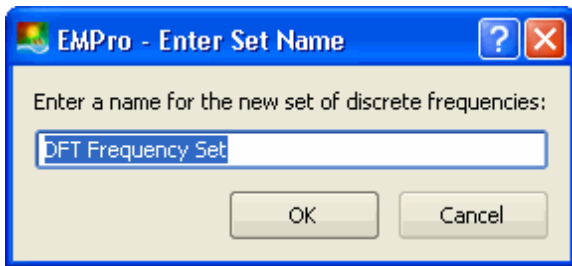
1. Click **Frequencies of Interest** in the Setup FDTD Simulation window.
2. Select the **Collect Steady State** check box.
3. Select **Use specified frequencies**, as shown in the following figure:



4. Click **Add a frequency to list**, as shown in the following figure:



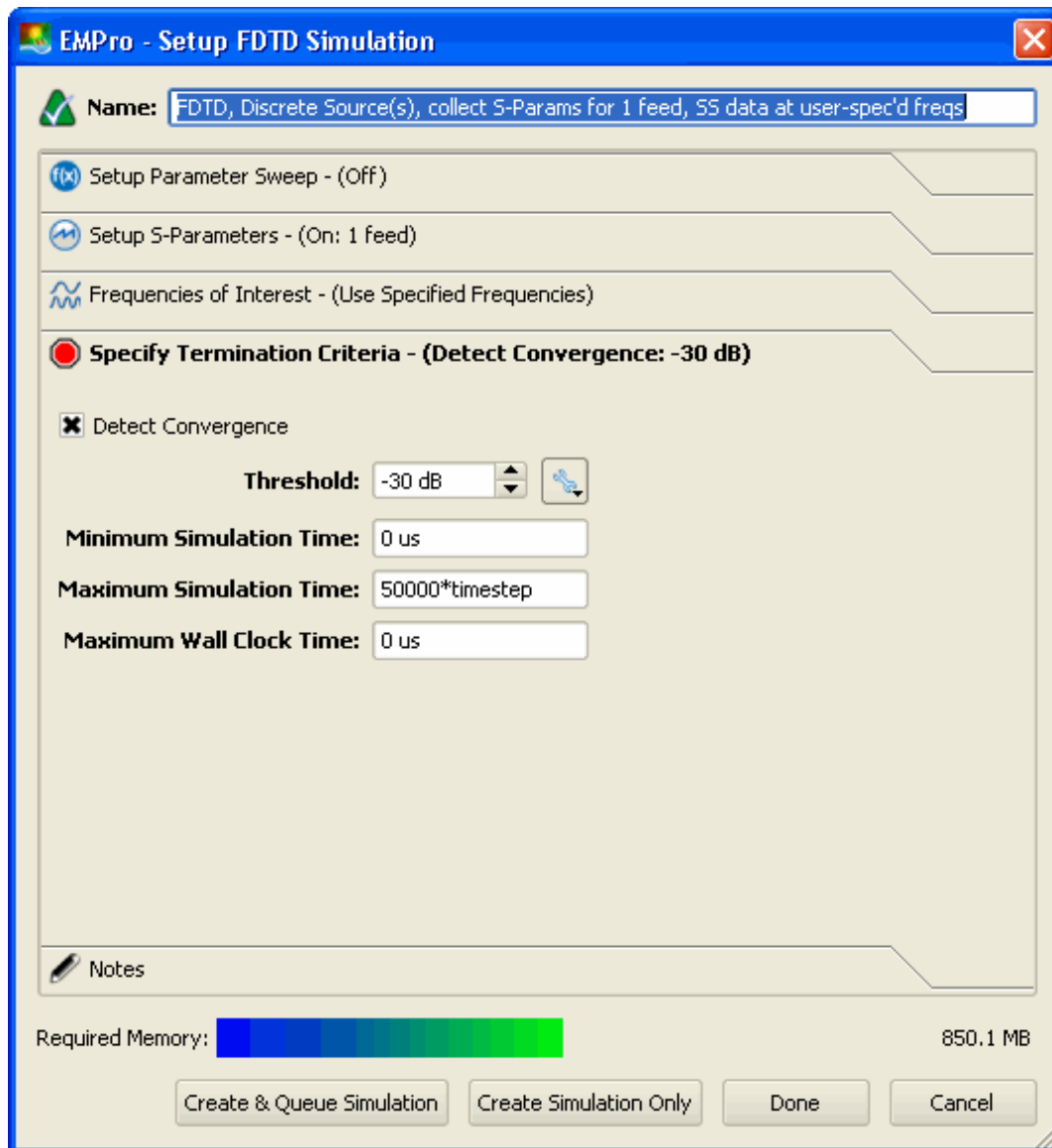
5. Click **Save current list**. The **Enter Set Name** dialog box opens.
 6. Type a name for the new set of discrete frequencies.



7. Click **Data Storage**.
8. Select the required option for storing data.
9. Type a value in **Frequency to Resolve** text box in **Timestep Sampling**.
10. Specify a value in **Sampling Interval**.

Specifying Termination Criteria

Convergence and stability are essential in determining whether a calculation will yield usable results. Convergence in a broadband calculation is met when all electromagnetic energy has dissipated to essentially zero. There are several options in EMPro to define termination criteria to ensure proper convergence had been reached.



If you specify a value of 0 for the Minimum Simulation Time field, it has no effect on the simulations. If you specify a value different other than 0, it indicates that the time stepping algorithm will not stop until the specified minimum time step is reached. This means that if the number is not zero, the FDTD timestepping algorithm will run at least until the specified minimum simulation time is reached and will continue after that until convergence is reached.

Example:

Simulation setup 1: Minimum simulation time = 0 and simulation converges after 10000 timesteps

Simulation setup 2: Minimum simulation time = 12000*timestep -> simulation will not stop after 10000 timesteps but will continue and do at least 12000 timesteps.

The most basic method of ending a calculation is defining a value in the *Maximum Timesteps* definition. Once the defined number of timesteps has completed, the calculation will stop. It is important to note that the calculation will terminate regardless of whether or not convergence has been met, so setting this definition to a proper value is important. If it is too low, results will be of no use.

Selecting the *Detect Convergence* check-box will automatically stop the calculation if slow convergence is detected, regardless of whether the number of maximum timesteps have completed. Due to numerical noise in the calculation, there may be a trivial amount of electromagnetic energy, even after the calculation has converged. The value defined in the *Threshold* dialog box specifies when the calculation has reached an acceptable value to assume convergence.




Note

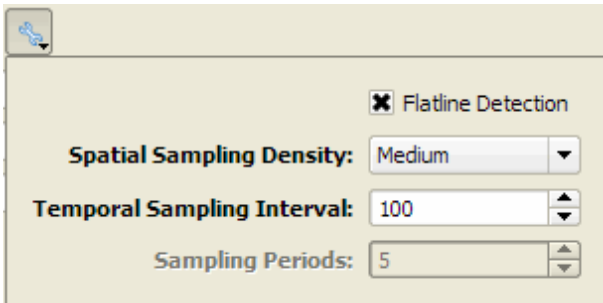
A general rule of thumb is that the values should have diminished by at least 30 dB or 1/1000th from the peak values.

For sinusoidally-excited problems, typical values for this setting range from -55 dB to -25 dB depending on the level of accuracy versus runtime desired. For instance, if high accuracy S-parameters are the goal, then the convergence threshold should be set to lower than -30 dB. If however, the user wants to view antenna patterns, -30 dB to -35 dB is suitable. The trade-off here is run time for accuracy. In general, for sinusoidally excited problems, using automatic convergence with a threshold of about -35 dB will produce very accurate results and will run in the shortest possible time to reach this level.

If a calculation is finished but convergence has not been reached, the output from most calculations will be meaningless. The only option is to decrease the convergence *Threshold* or increase the number of *Maximum Timesteps* in the calculation and run it again. If a resonance is occurring at a frequency beyond the range of interest, and a broadband input is used, the input waveform can be modified to limit the frequency content and eliminate the resonance. If the resonance is in band, or a sinusoidal input is used, then this is not applicable and more timesteps must be run or a lower convergence threshold must be chosen.

Specifying Advanced Options

You can also specify several advanced options by clicking the *Advanced* button (). These options allow you to set a specific termination criteria, as shown in the following figure:



Selecting the *Flatline Detection* check-box will stop the calculation if a slow convergence is detected. This may occur if the user sets the convergence threshold to a very low value (e.g. < -50 dB). In this case the calculation may converge but to a level higher than specified.

Note
To prevent false convergence, "slow" convergence can only be detected once the convergence level has reached at least -40 dB.

For a steady-state calculation, convergence is reached when near-zone data shows a constant amplitude sine wave - when all transients have died down and the only variation left is sinusoidal. In this case "convergence" is tested on the average electric field in the space for its deviation from a pure sine wave. If *Detect Convergence* is turned on, EMPro automatically places points throughout the space for this purpose. It is particularly important to monitor the results inside high permittivity dielectrics since the field propagation in these materials is much slower than in free space.

To ensure that steady-state calculations converge, EMPro will enable you to control the *Sampling Interval* and *Sampling Density* of the sample points. The temporal *Sampling Interval* definition is used to control the interval for which convergence is tested during calculations with broadband (pulse) excitations. The Spatial Sampling Density is linked directly to the convergence criterion. To detect convergence, a number of sampling points are selected and the time variation of the fields in these sampling points is measured to monitor convergence. The Spatial Sampling Density field allows you to control how many sampling points are chosen. You can accept the default value, **Medium**. Switching between Low, Medium, and High value is similar to switching between a sampling rate as shown below:

- Low: Every 4th Yee cell in each direction: so if your grid has 400x400x400 cells in the x, y and z-direction, the sampling will have 100x100x100 points.
- Medium: Every 2nd Yee cell in each direction
- High: 3 all Yee cells

Note

Setting this value to 100 or 200 timesteps is typical. Setting this value to much less than that increases the computational overhead a small amount.

Sampling Density is used to control how many spatial samples are used to determine convergence. The sample points are equally spaced in all three dimensions of the grid. *Low* density samples every 4th point in each dimension, while *High* density samples every point in each dimension. For very low frequency problems or where the number of timesteps per RF cycle is greater than 200 (e.g., very small cells with a low frequency excitation), this should be set to *LOW*. For moderate frequencies or where the number of timesteps per RF cycle is less than 200 but greater than 100 this should be set to *Medium*. For high frequency problems where the number of timesteps per RF cycle is less than 100, a *High* setting gives the best accuracy. This setting is for both broadband as well as sinusoidal excitations.

Adding Notes

The Notes section enables you to attach any notes to a project. Click **Notes** in the **Create FDTD Simulation** dialog box to display the **Notes** screen. The notes will be available in the *Simulations* workspace window after the simulation is created.

Queuing and Running Simulations

After defining the necessary components of the simulation in the *New Simulation* dialog described above, select *Create And Queue Simulation* and the main window will show all of the queued and completed simulations as seen in the figure below. Only one simulation can be run at one time, so as soon as a calculation is complete, another queued simulation will begin.

In this window, you can view information in the following tabs:

- **Summary:** Provides a summary of the calculation.
- **Notes:** Documents any notes that were added by the user in the *New Simulation* dialog.
- **Output:** provides the output generated by EMPrFDTD. Statistics such as the percentage of completion, current timestep, convergence, time elapsed, etc, are listed for every simulation in this tab (including output information for every parameter swept).
- **Diagnostics:** Provides an analysis of the calculation.

Completing the Process of Specifying Setup

After you have completed entering your FDTD Simulation options, click the **Done** button to apply the current settings in the *Create FDTD Simulation* dialog box, or click **Create Simulation Only** to accept the settings. You can also click **Create and Queue Simulation**.

Running the FDTD Simulation

After completing the FDTD simulation setup, you can run calculations on the geometry.

FDTD Simulation

The Simulations window stores the project simulations. You can create, queue, and run simulations using the Simulations workspace.

To run an FEM simulation, click **Create and Queue** in the Simulations workspace.

Defining a Grid and Mesh

This section provides information about the following topics:

- *Defining Grid Settings (fDTD)*
- *Defining Mesh Settings (fDTD)*
- *Using Conformal FDTD Meshing (fDTD)*

Defining Grid Settings

The EMPro grid provides the main interface between a dimension-based geometry and the finalized FDTD mesh space. EMPro provides several tools for creating and customizing an effective grid so that accurate results can be retrieved without allocating excessive memory and calculation time. This section provides information about the tools available for specifying the properties of the EMPro grid and how to set localized grid properties.

Note

For more information about grid theory, concepts, and recommendations within EMPro, see *Grid Appendix* (appendixb).

General grid definitions are assigned to the entire grid within the *Grid Tools* window of the *Geometry* workspace window. Here definitions such as target cell size, bounding parameters, and limits are defined. Customizable grid definitions such as fixed points and grid regions can also be added here.

The *Gridding Properties Editor* governs grid definitions that are applied to individual objects. This editor is useful for objects whose characteristics are not adequately considered by the general grid definitions set within the *Grid Tools* editing tabs.

The following sections explain how to enter the grid properties and specify the various grid elements using these editors.

Using Grid Tools

There are two ways to access the *Grid Tools* window. With the *geometry* workspace window open, either select *Grid Tools* from the drop-down menu in the window, or double-click on the *FDTD: Grid* branch of the *Project Tree*. Once this window opens, click *Edit Grid* to define the grid parameters for the project.

The tabs located within grid tools follow the step-by-step editing process that is used to define the grid. The first tab, *Size*, specifies general definitions that are applied to the whole grid, whereas the second and third tabs, *Fixed Points* and *Grid Regions*, enable the user to customize the grid by manually placing gridlines and changing the cell characteristics of different regions in the mesh.

The fourth tab, *Limits*, applies constraints to the grid so that memory requirements and adjacent cell size ratios do not exceed a reasonable value. Since grid definitions largely influence the memory requirement of a particular calculation, two bars are always visible in order to monitor how much memory the intended grid definitions will utilize. The *Current* bar shows how much memory the last applied mesh required, while the *New* bar shows how much memory a new mesh requires before changes are applied to the project. The *Limits* tab also enables you to specify the smallest desired cell size. If a grid definition requires a cell smaller than the smallest desired size, the grid becomes invalid and a warning message will appear.

The fifth tab, *Info*, provides data about both the current and the proposed grid. (The two sets of values will be the same if any proposed changes have been applied.) The grid's dimensions, upper and lower boundaries, cell counts, and minimum and maximum edge

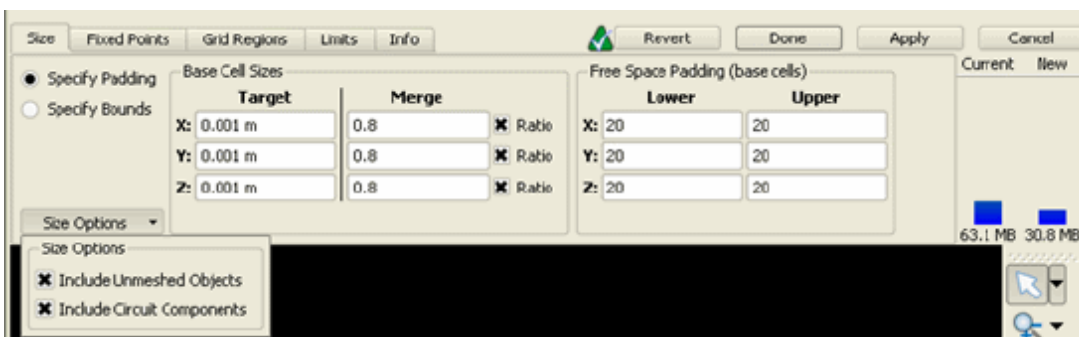
lengths are shown. A value is shown for each axis, **X**, **Y** and **Z**.

Size Tab

The *Size* tab enables users to define the cell size and the placement of the grid at the boundaries of the project. There are two options for defining the outer boundaries. The first option is to specify the padding around the project's lower and upper boundaries in terms of discrete numbers of cells. The second option is to specifying a grid bounding box by its physical coordinates, independent of the geometry. Each option is detailed below.

Specify Padding

The next figure displays the available options for specifying the padding of the outer boundaries of an EMPro cell.



Selecting *Specify Padding* enables you to add fixed points for the bounds of a geometry. The actual padding cells will begin at those fixed points and proceed outward. Within the geometry, the area between the fixed points will be divided into grid cells. If there are no manual grid regions, manual fixed points, part grid regions, or part-associated automatic fixed points in the project, the area between the fixed points bounding the geometry will be divided into cells with even spacing. Each evenly-spaced cell will have an edge length no greater than the target cell size.

Note

To select the fixed points automatically, choose [Use Automatic Fixed Points](#), which is found in *Project Tree* under the *Parts* section of a geometry, in the *Gridding Properties Editor*.

The *Base Cell Sizes* window allows entry of the target cell size when generating the EMPro mesh in areas that are not part of a grid region. Note that this is the target cell size, and may not reflect the actual cell size in the mesh. The actual cell size may be slightly lower depending on the constraints placed on the mesh, but it will never be higher, ensuring that the desired level of detail is not compromised. When there are two fixed points or grid region bounds of any type, the region between them will be divided into a number of cells. The number of cells that is required depends upon the target cell size. No cells larger than the target will be generated. The target cell size may not divide evenly into the distance between the fixed points or grid region bounds, The gridding process prefers to generate cells of a constant size. First the minimum number of cells required to span the distance is calculated and then a cell size, less than or equal to the target size, is calculated so that the distance is spanned exactly.

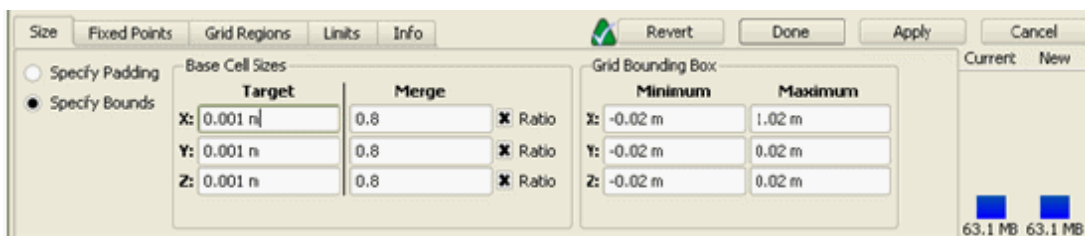
Note
For more information about exceptions to the general rules, see *Grid Appendix* (appendixb).

The *Free Space Padding (In Cells)* window enables you to define the extent of the padding space around the geometry. The extent is specified as a number of cells, each of which is taken to be the target cell size. In simple projects this means that the number of cells specified will be added as freespace around the geometry. In more complex projects, grid regions close to the boundary may cause transition regions containing smaller cells to intrude into the freespace padding. The freespace padding's extent will remain the same, whether it is made up of the specified number of cells each exactly the target cell size, or whether the padding extent contains some number of smaller-sized transition region cells.

There are also two checkboxes available in the *Size Options* drop-down list: *Include Unmeshed Objects* and *Include Circuit Components*. Selecting either of these options will ensure that any unmeshed object and/or circuit component is included within the padded region. Otherwise, only meshed objects will be used to determine the size of the padded region. These options may also affect the size of the region bounded by free space padding.

Specify Bounds

The following figure displays the available options for defining the outer bounding box. In the following figure, a project boundaries are defined by a bounding box within the *Size* tab:



The *Base Cell Sizes* dialog box is analogous to the *Base Cell Sizes* definition under *Specify Padding*.

There are some instances when defining a *Grid Bounding Box* rather than a padding of free space cells may be advantageous in placing the outer limits of the grid. The grid bounding box, unlike cell padding, is independent of the geometry, so it is possible to define a grid bounding box that does not include the entire geometry.

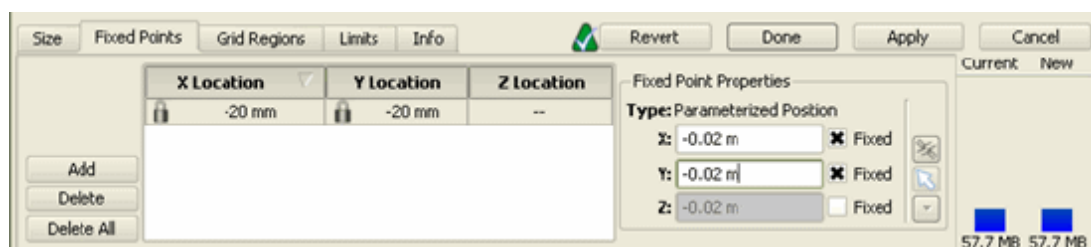
Fixed Points Tab

Fixed points are essentially gridlines placed at discrete locations in the mesh, and therefore control the placement of cell boundaries in the mesh. They are used to ensure that important parts of the geometry are considered at cell edges which cannot be controlled by uniform meshing. For example, it may be useful to place a fixed point at the edge of a *Parts* object in the geometry so that its edge does not fall between cell edges.

The *Fixed Points* tab stores the definitions associated with the placement of fixed points in the geometry. To add a fixed point to the mesh, select the *Add* button and define the

Fixed Point Properties by typing in the physical coordinates of interest. Note that any of the three principle directions may be checked or unchecked (in the *Fixed* check box) depending on how many points are to be defined. For example, if only the X-coordinate is defined, a fixed point will be added that intersects at that location on the X-axis, which defines a plane in YZ. Defining fixed points in all three principle directions will result in three fixed points, intersecting at the defined point.

The following figure displays the window associated with defining fixed points. In this instance, only X and Y are checked, so two fixed points spanning the X- and Y-planes will be added in the X (at $X = -0.2$ m) and Y (at $Y = -0.2$ m) plane, respectively.



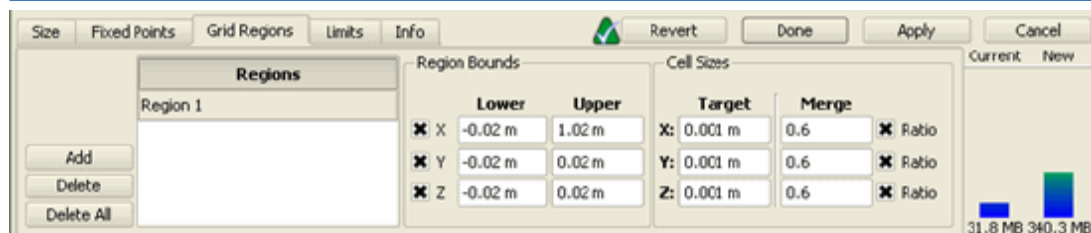
Note
A fixed point can also be defined by checking the X, Y, and/or Z coordinate check boxes of interest and clicking on the desired location of the plane in the simulation space with the *Selection* tool.

Grid Regions Tab

A *Grid Region* is a bound region in the mesh that is assigned a cell size different from the cell size assigned in the *Size* tab. Grid regions may be desirable in locations that require finer meshing and thus require smaller cells. It is also advantageous to use them in areas that are not important to the EMPro calculation by defining them with larger cells to save memory and calculation time. Simply define the *Region Bounds* and the *Cell Sizes* in each direction of interest, and a region will be placed on the grid.

The following figure displays the window associated with defining grid regions. In this example, the cells within the particular grid region are defined to be 0.001 m. Automatic fixed points will be merged to be spaced 60% of this distance as defined in the *Merge* boxes.

Note
For more information about fixed point merging, see *Grid Appendix* (appendixb).



Limits Tab

The *Limits* tab contains several required and several optional limits used to restrict characteristics of the grid.

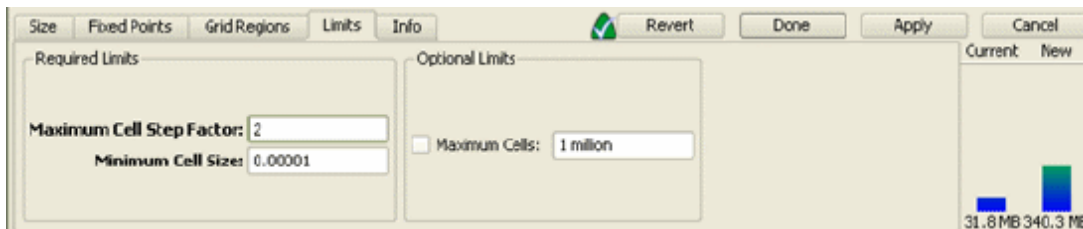
The *Maximum Cell Step Factor* definition restricts the rate at which the cell size may increase between two adjacent cells. For example, if a cell is 1 mm long and the maximum cell step factor is set to 2, the next cell cannot exceed 2 mm, because that would cause the cell ratio between these two cells to exceed 2. Therefore, if a grid region has cells of 1 mm, and the default global grid target is 5 mm, additional cells are needed, forming a transition region between the grid region and the Global grid, since 1 mm:5 mm exceeds the 1:2 ratio.

Note
The ratio of 1:2 is used in this example because it is the largest ratio recommended in Chapter 11, Section 2 of the Taflov and Hagness in Bibliography text.

The second required limit, *Minimum Cell Size*, represents the global minimum cell size allowed in the project. No cell in the project can fall below this limit or the project will not be valid. There are other *Minimum Cell Size* definitions within the *Grid Tools* and *Gridding Properties Editor*, but they must be greater or equal to this global definition to be considered a valid definition.

The two optional limits aid in restricting the size of the EMPro project so that the calculation time and memory requirements do not exceed a defined limit. These limits can be applied to the project by restricting the number of *Maximum Cells* that are contained within the mesh. If this definition is applied to a project and exceeded, an error message will appear to indicate this.

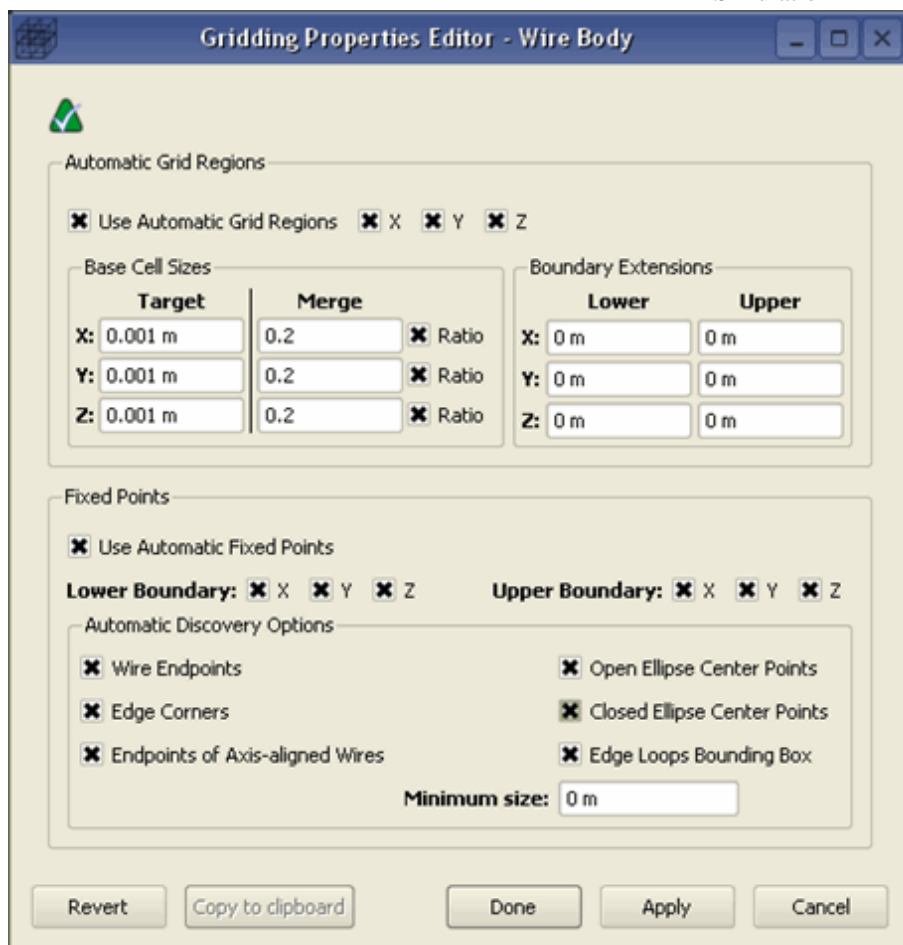
The following figure displays the *Limits* tab.



Using the Gridding Properties Editor

The *Gridding Properties Editor* window is used to define grid regions and fixed points associated with an individual object. To open the *Gridding Properties Editor* window, right-click on the appropriate object in the *Parts* branch of the *Project Tree* and select *Gridding Properties*.

The *Use Automatic Grid Regions* and *Use Automatic Fixed Points* checkboxes enable either of these definitions when they are selected. The figure below shows the *Gridding Properties Editor* window with both grid regions and fixed points enabled. The following sections detail the definitions associated with grid regions and fixed points.



Use Automatic Grid Regions

A grid region is a region assigned a specific cell size within a specified lower and upper boundary. A custom cell size can be applied in this region to any or all of the three principal directions. They are desirable in instances in which the *Cell Sizes* defined to the entire grid under *Grid Tools: Grid Regions* is not sufficient for a particular object.

Grid regions can be added to the grid within *Grid Tools* by defining each region's physical bounds, or a grid region can be applied to a single object within this editor. In this case, the grid region will be applied everywhere within the object (plus any extended region defined in the *Boundary Extensions* definition).

Note

For more information about how to define a region physical bounds, see [Grid Regions Tab](#).

Cell Sizes

The *Cell Sizes* definition refers to the maximum target cell size that may occur between the lower and upper boundaries. The actual cell size between these boundaries may be slightly lower depending on the location of these boundaries. For instance, if a *Cell Size* is defined to be 1 m in the **X**-direction and the upper and lower boundaries are defined at **X** = 1 m and **X** = 4.9 m, respectively, the actual cell size would be 0.975 m since the distance between the upper and lower boundaries would not permit four 1 m cells within 3.9 m. The actual cell size will never be adjusted to a value higher than the *Cell Sizes*

definition, however.

Boundary Extensions

The upper and lower boundaries of the grid region usually occur at the edges of the object, but in the case that these bounds are not sufficient, they can be moved inward or outward in any X, Y, and/or Z direction by defining values in the *Lower Boundary Extensions* and *Upper Boundary Extensions* windows. If a positive distance is defined, the boundary will be shifted at that distance away from the object and conversely, if a negative distance is defined, the boundary will be shifted into the object.

Use Automatic Fixed Points

Fixed points are analogous to gridlines whose locations are explicitly defined to control where the edges of meshed cells fall in the geometry. Automatic fixed points are given that name because their positions are automatically determined by examining the geometry of the part with which the automatic fixed points are associated. Cell sizes are adjusted to flow as evenly as possible between the fixed points while never exceeding the target cell size at the location of each fixed point. For more about the Limits tab, see [Limits Tab](#).

Fixed points can be placed anywhere in the grid by specifying their location within the *Fixed Points* tab of the *Grid Tools* window, or they can be added within this editor to a particular object based on the object's geometry. For more information about the *Fixed Points* tab, see [Fixed Points Tab](#).

The following settings control where fixed planes are placed in reference to the object geometry.

Lower and Upper Boundaries

Places fixed point(s) at the lower or upper boundary of the object in the direction specified.

Wire Endpoints

Places fixed point(s) at any discontinuity existing in the object. Refer to the following figure for an example of a fixed point placed at an edge.

Edge Corners

Places fixed point(s) anywhere that there is a discontinuity in the derivative of a object's edge (i.e., points that do not have tangents).

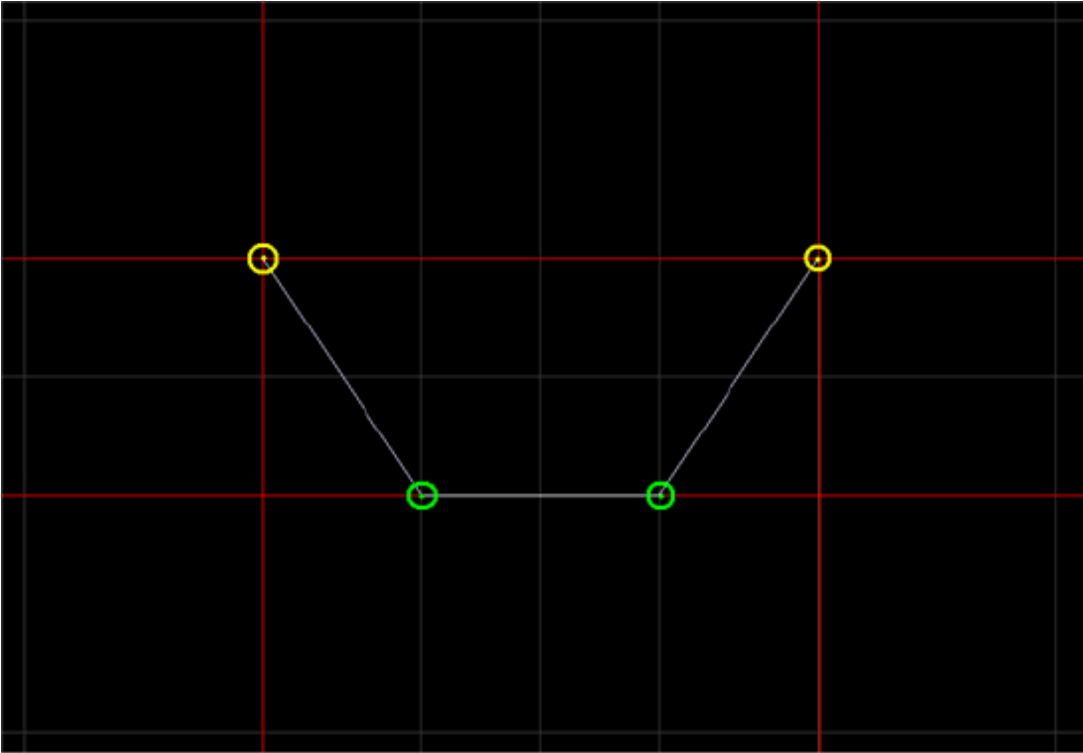
The figure below also shows the fixed points placed at wire endpoints (yellow points) and edge corners (green points) that occur within the simple wire body (shown in white).



Note

Notice how the cell sizes vary to accommodate the placement of the four fixed planes (shown in red).

Fixed points placed at wire endpoints and edge corners

Fixed points placed at wire endpoints and edge corners**Endpoints of axis-aligned wires**

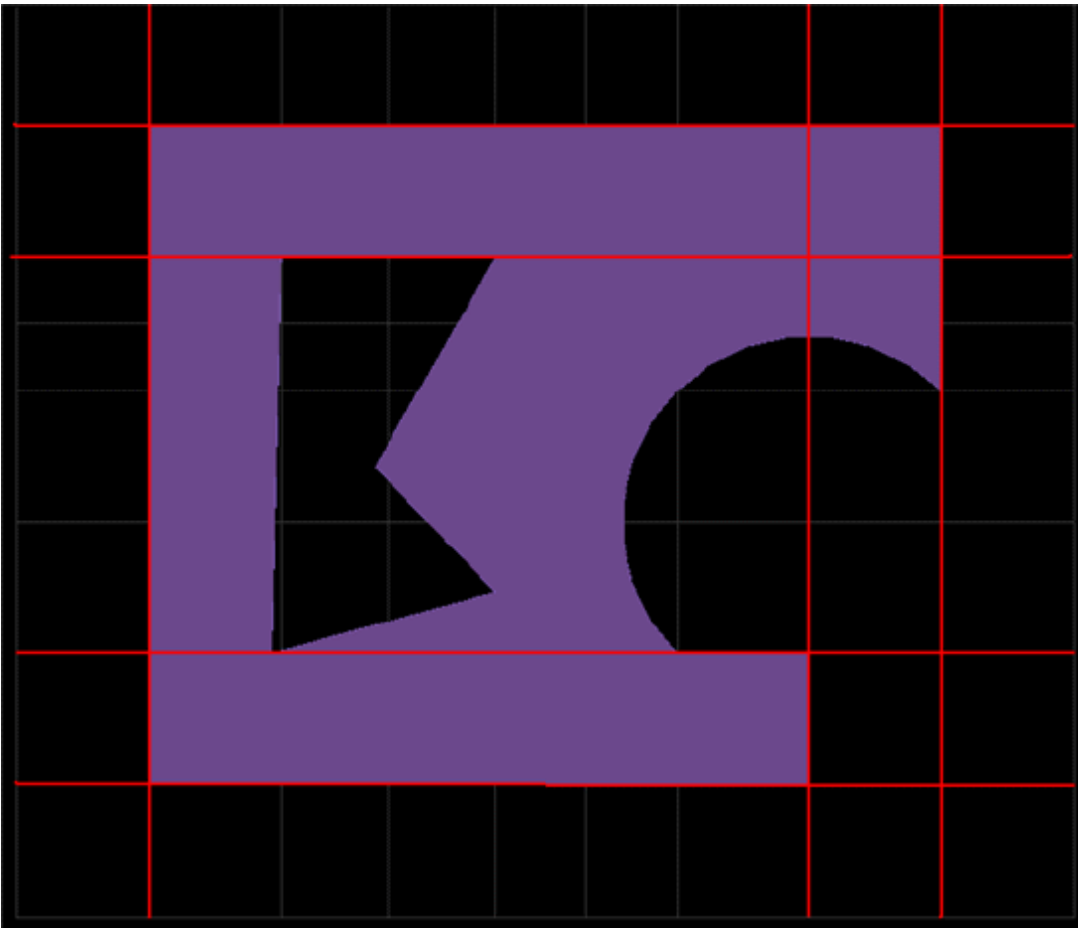
Places fixed point(s) along any axis-aligned straight edges that exist in the object geometry.

Below, fixed planes (in red) are placed at axis-aligned straight edges.

**Note**

Notice that straight edges that are not axis-aligned do not have fixed planes.

Fixed points placed at axis-aligned straight edges



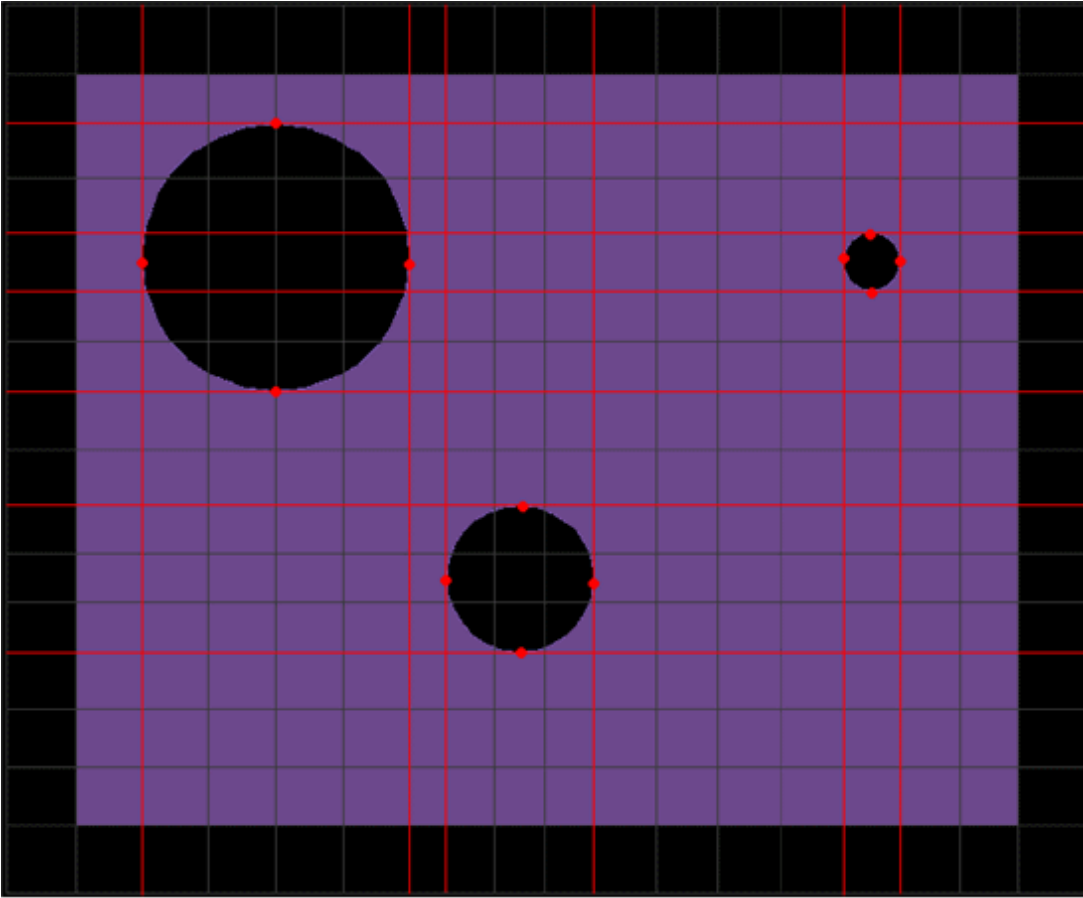
Edge Loops Bounding Box

Places fixed point(s) at the edges of *loops* or closed circuit of edges. Any loop that smaller than the *Minimum Size* will be ignored. Thus, to consider all loops that exist in the object's geometry, simply define the *Minimum Size* as 0.

In the figure below 12 fixed points placed at the edges of loop boundaries.

Note Notice that the *Minimum Size* has been set to 0 so all loops are considered. If the minimum size was larger than one or more loops, no fixed points would be placed at its edges.

Fixed points placed at edges of loops (minimum size = 0)

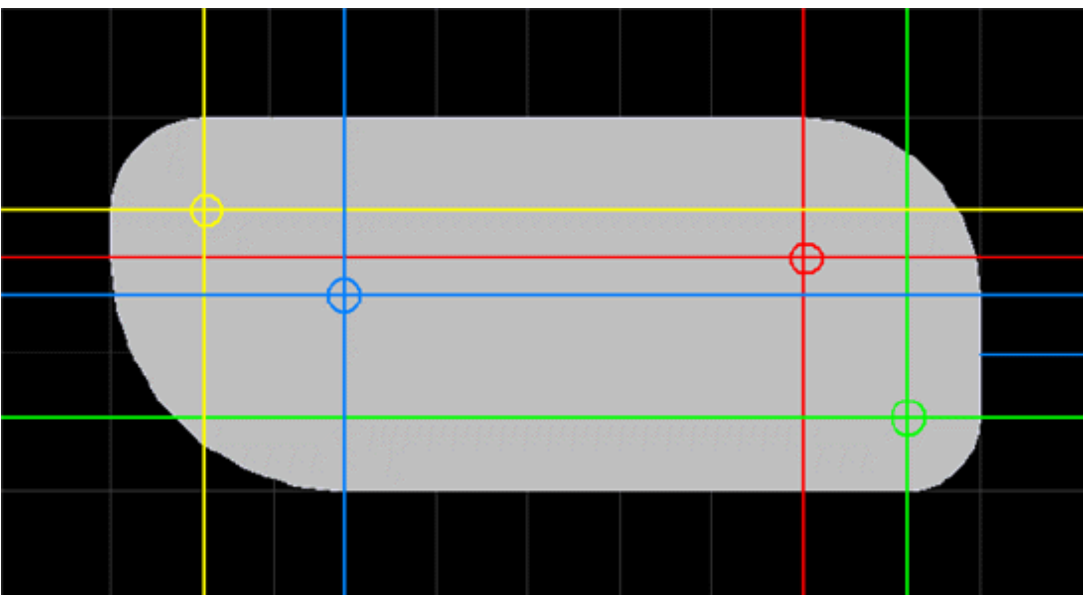


Open Ellipse Center Points

Places fixed point(s) at the center of open ellipses.

The figure below shows four fixed planes placed at the center of four filleted edges, which are treated as open ellipses.

Fixed points placed at the center of open ellipses



Closed Ellipse Center Points

Places fixed point(s) at the center of closed ellipses.

Defining Mesh Settings

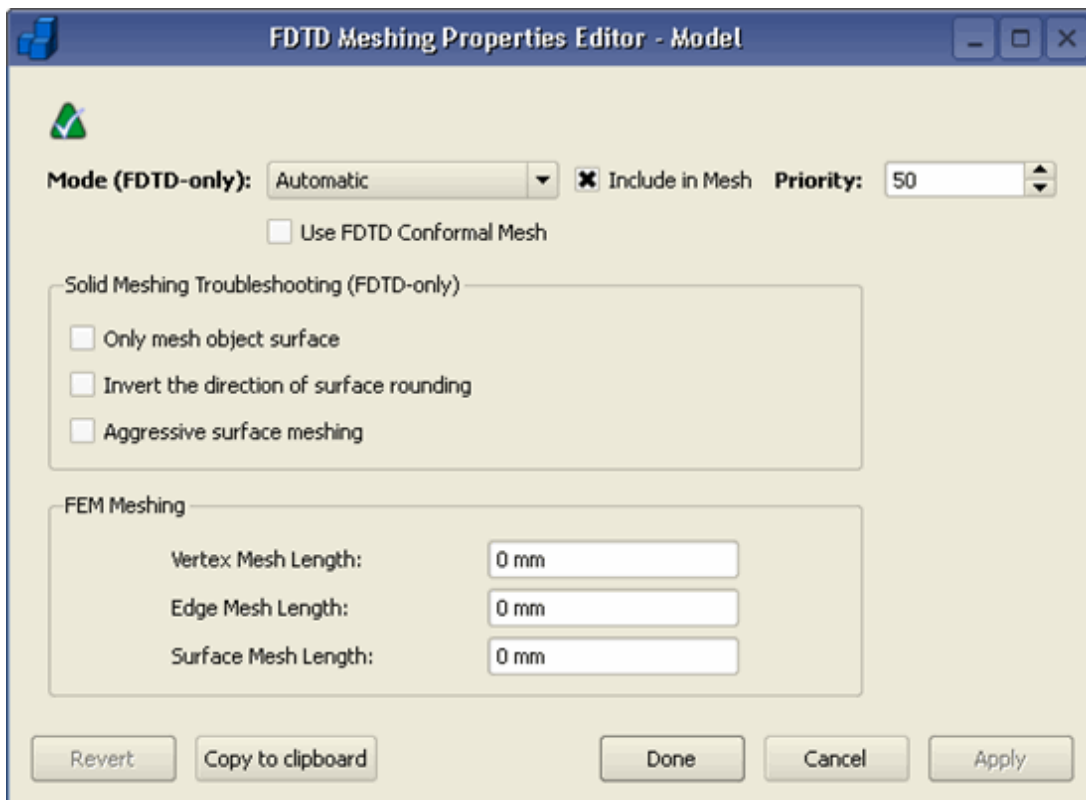
You can determine how each object within the boundaries of the defined grid is meshed by using the Meshing Properties Editor window. By default, every object in the project is meshed with uniform priority. However, you can change these settings to mesh selected objects carefully. This section provides information about how to:

- Specify the objects to be included in the mesh.
- Adjust the way touching objects are meshed.

Accessing the Meshing Properties Editor

You can use the Meshing Properties Editor window to adjust the properties of the mesh for individual objects. This enables you to control the meshing process, when necessary, to create an accurate mesh. You can use the default settings. A mesh is generated automatically when calculation files are written, or when grid definitions are applied within the Grid Tools dialog box and the project is viewed with the *Mesh View* button in the *View Tools* menu.

To open the **FDTD Meshing Properties Editor** dialog box, right-click the required *Modelname* in the **Parts** list and select **Gridding/Meshing > Meshing Properties**. The FDTD Meshing Properties Editor displays different setting configurations depending on the type of part being modified, as shown in the following figure:



Note while editing a grid, a preview of the grid is created in the Mesh View mode, but it is not actually meshed until the grid definitions have been applied.

When *Automatic Remeshing* is turned on, the mesh will be updated automatically, so that the cutplanes visible within *Mesh View* will be current. If this feature is not turned on, the icon will indicate the mesh is out-of-date. In this case, click the Automatic Remeshing button and select *Remesh Now* to generate the current version of the mesh. The FDTD meshing properties editor displays different setting configurations depending on the type of part being modified.

Specifying Meshing Modes

EMPro provides the following types meshing modes in EMPro:

- **Meshing Disabled:** Select *Meshing Disabled* to disable meshing for objects that do not need to be considered during meshing operations. Selecting this option will reduce memory usage.
- **Electrical Meshing:** Select *Electrical Meshing* to include the object in the electric mesh only.
- **Magnetic Meshing:** Select *Magnetic Meshing* to include the object in the magnetic mesh only.
- **Dual Meshing:** Select *Dual Meshing* to force the object to be included in both electrical and magnetic meshes.
- **Automatic Meshing:** The Automatic Meshing mode is selected by default. It enables EMPro to automatically determine which parts are included/excluded from the electric and magnetic meshes. This determination is based on the electric and magnetic properties of the material assigned to the part. When Automatic Meshing is selected:
 - Parts with electric material, other than free space, are included in the mesh.
 - Parts with magnetic material, other than free space, are included in the mesh.
 - Parts made entirely of free space material are not included in the mesh.

You can assign a meshing mode either from the FDTD Meshing Properties Editor dialog box or by right-clicking an object in the Parts list and then selecting **Gridding/Meshing > Mesh Type**.

The following table describes the various meshing modes:

Mesh Mode	In Electric Mesh	In Magnetic Mesh
Automatic	All parts with non-free space electric material properties	All parts with non-free space magnetic material properties
Electric	Yes	No
Magnetic	No	Yes
Both	Yes	Yes
Disabled	No	No

To include a part in the mesh where both the Electric and Magnetic material types are set to Free Space, select either the Electric Mesh mode or the Dual Mesh mode. To select a different mesh mode, right-click an object in the Project Tree and select Meshing Properties or Mesh Type. A window containing the available mesh modes appears. Under most circumstances, it is best to keep the default mode because it enables the software to automatically apply the most memory-efficient meshing option.

Specifying the Meshing Order

The *Meshing Order* setting specifies the level of priority that each object has. If objects overlap in a particular region of space, the object with a higher meshing priority will overwrite the other objects. Each object is assigned a default priority of 50. This should be changed to a higher or lower value depending on whether the object should be given more or less consideration, respectively, when meshed. Choosing *Set Priority* prompts the user to enter a numerical priority definition.

Additionally, where numerical priority values are equal, priority can be specified within the *Project Tree*, as displayed below. Moving the selected *Parts* object priority UP or DOWN, increases or decreases its importance in the meshing calculation, respectively. Likewise, selecting *Move To Top* assigns that object the highest priority of all of the objects in the project; *Move To Bottom* will assign it the lowest priority.

How Do I Set Mesh Priority

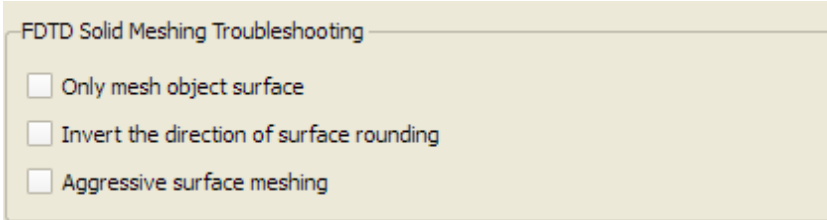
1. Right-click **Materials** and choose the **Select From Default Material Library** option.
2. Select a material and click **Add**.
3. Assign materials to the objects.
4. Double-click a material to view its default priority.
5. Right-click **Parts** and choose **View Parts List (All Parts)**.
6. Double-click the mesh priority column and type the required mesh priority.
OR
7. Right-click an object in the **Parts** list and choose **Gridding/Meshing, Gridding-Meshing\Meshing Order> Set Priority**.
8. Type the mesh priority.
9. Click **OK**.

Specifying the Include in Mesh Setting

The Include in Mesh setting specifies whether the selected part is considered while calculating the mesh. This setting does not affect the mesh mode or mesh order of the part. It enables you to remove an object from the mesh or add it back in without resetting the mesh mode or order associated with the part each time they make a change. This option can be set by right-clicking a part and selecting the **Gridding/Meshing>Include in Mesh**, or by selecting the Meshing Properties Editor. To add or remove multiple parts from the mesh, right-click the selected parts and select **Include in Mesh** or **Exclude from Mesh**.

Specifying the Solid Meshing Options

Under most circumstances, the solid meshing options are not necessary. These options may be advantageous in projects that have small features that are not adequately considered by default meshing operations. The following figure shows the solid meshing options in the *Meshing Properties Editor*.



Only Mesh Object Surface

The *Only Mesh Object Surface* option meshes the shell of the physical object rather than its entire volume.

Invert The Direction Of Surface Rounding

If the edge of an object falls in the center of a cell rather than at a cell edge, the default behavior of EMPro is to round the object away from the center of the object. Selecting *Invert The Direction Of Surface Rounding* will invert this behavior and round the edge inward, towards the center of the object. This usually is not necessary but may be useful in cases where an object is very close to another. In this instance, rounding a surface outward can cause the objects' edges to touch and result in a short circuit that may cause inconsistent results.

Specifying the Aggressive Surface Meshing

Aggressive Surface Meshing should only be enabled for an object that is very small compared to the projects' cell size. This option will create a more detailed mesh around smaller features to ensure that they are surrounded with cell edges. This would be necessary for small feature whose electrical conductivity may be compromised without detailed meshing.

Specifying the Volume Meshing Options

Several additional settings are available when assigning mesh properties for volumetric parts, such as voxels. There are two *Volume Meshing Method* algorithms available for converting the volumetric data to cell edge data. Both methods use an intersection method to determine how much of a given volume is contained inside of a cell edge. Therefore, it is applied to every volume that intersects the cell edge. The data is combined to measure the percentage that each material fills in each cell edge. These percentage fill values are in increments of a half of a percentage point. This information is then used based on which algorithm is selected.

- The first method, referred to as the *Dominate Material* meshing algorithm, evaluates the list of percentage fill values, and then chooses the material that has the largest percent fill to apply as the final material for that cell edge.
- The second method, referred to as the *Averaged Material* meshing algorithm, uses the percent fill values for each cell edge to generate an "averaged material." These averaged materials are maintained in a separate list from the original materials of the geometry.

Degradation Level

The *Degradation Level* is available in the *Averaged* volume meshing method. It dictates how non-averaged materials intersecting the volume of the cell edge are to be grouped and considered for averaging, by defining the size of a set of bins that represent a percentage of the total cell edge volume. For example, a degradation level of 4 subdivides the total volume (100%) by 4, meaning each bin represents 25% of the volume. Therefore, for consideration in the averaged material, a non-averaged material must fill at least half of the bin, or 12.5% of the total volume. By defining a course resolution for material consideration we are able to condense the number of averaged materials, keeping memory and calculation speed reasonable while providing a user-defined level of accuracy.

Volume Calculation Method

The *Volume Calculation Method* determines the accuracy of the rendered volume. It directly relates to the *Degradation Level*, as it determines the amount of material volume that fills a cell. *Approximate* is sufficient for most calculations, and runs most efficiently. *Exact* is a better choice when it's important to resolve structures that are changing rapidly within the size of a cell. By definition, it is more accurate but also requires more memory.

Touching Objects

The FDTD mesh is a discretization of a continuous 3-D space into cells of user-specified size. The process of transferring these geometric representations into this discretized space (meshing) can lead to unintended artifacts. One common situation that arises is that two objects in the geometry that are very close together, but not touching, end up intersecting in the mesh. Such an artifact can lead to inaccurate results due to a short-circuiting effect in the calculation. The user traditionally has two options in this case. The first is to move the two objects further apart, which may not be possible given the design of the geometry. The other is to create cells small enough in that region to resolve the separation between the objects, which may lead to an extremely large number of cells and a dramatic increase of computation resources required to run the simulation.

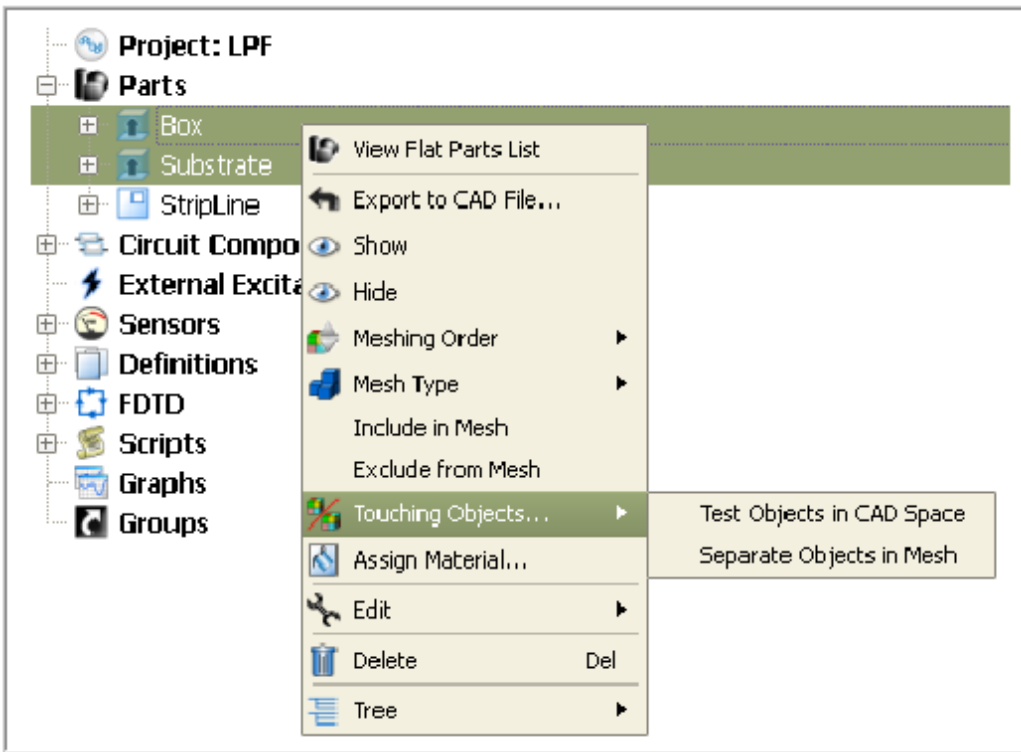
EMPro provides a third option to alleviate the work required by the user. This is the role of the *Touching Objects* functionality. There are two tools available for the user to identify and correct any unintended artifacts caused by meshing or geometric positioning.

- The first, *Test Objects in CAD Space*, tests two objects for a CAD space intersection. It will report to the user if two objects are actually touching in non-discretized space.
- The second, *Separate Objects in Mesh*, enable you to create a relationship between two objects that will ensure they are separated in the final mesh by a user-defined number of cells.

These tools enable the user to maintain a separation while not sacrificing computational complexity or geometric design.

Using the Test Objects in CAD Space Option

In order to find out whether two objects are actually touching, EMPro provides the *Test Objects in CAD Space* function to test the selected objects. This can be done by holding | **Ctrl**| and selecting two objects in either the *Geometry* workspace window or under the *Parts* branch of the *Project Tree*. Then test the objects by selecting **Touching Objects > Test Objects in CAD Space**, as shown in the following figure.



A dialog box is displayed with the results. The user will also have the option of ensuring that the objects are separated in the mesh, as seen in the following figure. If **Yes** is selected, the *Separate Objects in Mesh* dialog will appear, as described in the following section.

Separating Objects in Mesh

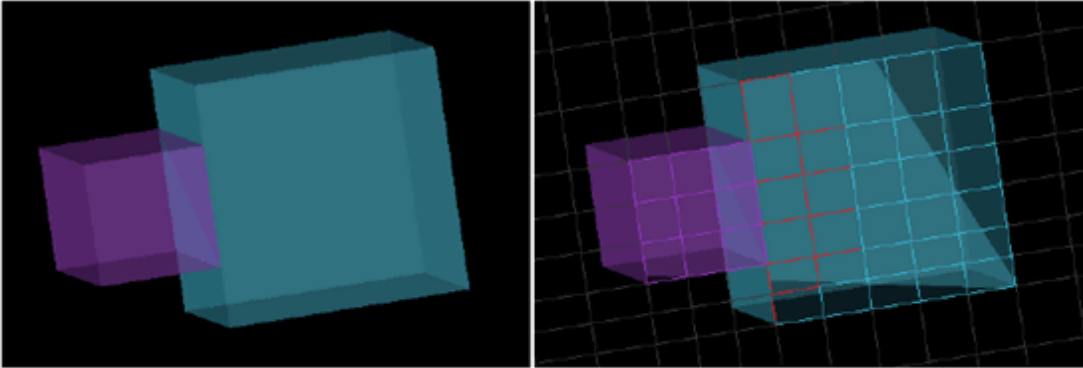
Defining a separation operation between two objects ensures that a gap of user-defined size is maintained despite discretization or geometric positioning. To perform this operation, select two objects and navigate to **Touching Objects > Separate Objects in Mesh**, (seen in the figure above). The dialog shown in the following figure will appear.

- The *Primary Model* is separated from the *Secondary Model*. Ensure they are identified appropriately. Pressing *Swap Inputs* will reverse their labels.
- Define the *Separation Distance (cells)* between the two objects. This is the number of cells that is "taken away" from the *Secondary Model* in the meshing operation.
- Specify the *Fill Material* to use, if any. If no material is specified then the material existing in the mesh under the removed object will be used.

Note

This is a very powerful concept, as it means that any complex configuration of materials may exist in the gap between two objects, and that information is not lost by performing the operation.

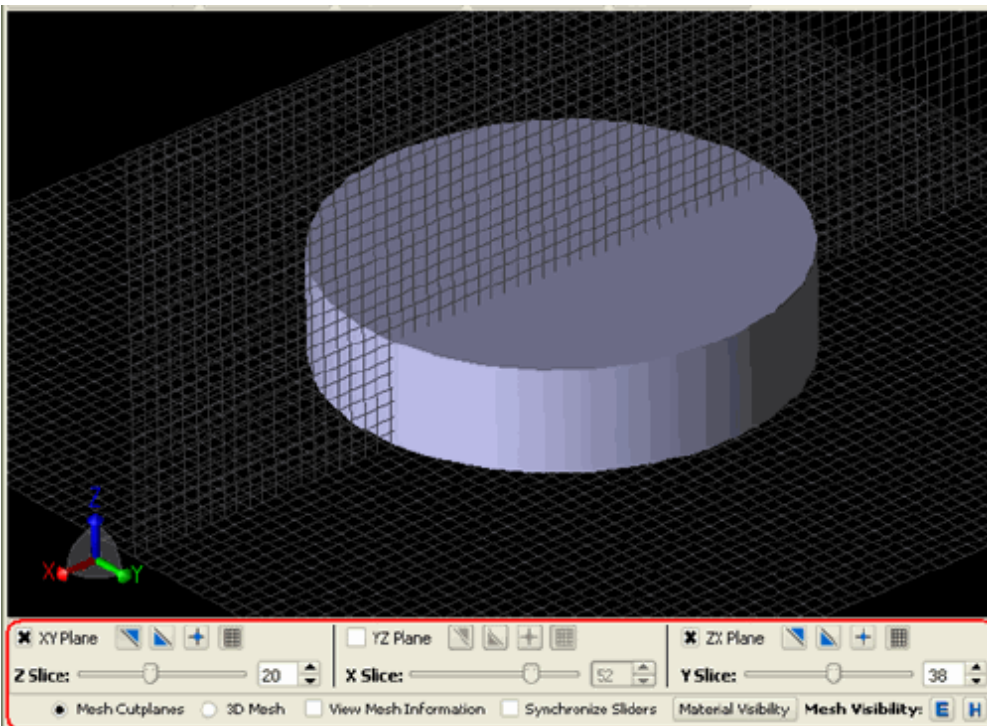
To illustrate this concept, consider the case of two touching cubes made of different materials. A mesh separation of 2 cells and a different (red) material is applied, with the smaller (purple) cube as the *Primary Model* and the larger (blue) cube as the *Secondary Model*. Though the geometry is unaffected, the mesh separation is visible when viewing the geometry in Mesh View. The following figure displays touching objects geometry (left) and mesh view (right) with a 2-cell mesh separation:



Once this relationship is specified it will be maintained and updated appropriately in the mesh and nothing further is required by the user. If the user decides to delete one of the operands, the operation will be automatically removed; and if the user re-orient or changes materials on the operands, the operation will reflect that appropriately. The mesh separation is managed from the **FDTD > Mesh** branch of the **Project Tree**.

Displaying the Mesh View

This button toggles between Mesh View and Normal View. Alternatively, double-clicking **Mesh** in the FDTD list enables Mesh view, as shown highlighted in the following figure:



When in Mesh View, there are two main viewing modes, *Mesh Cutplanes* and *3D Mesh*, that are controlled by radial buttons along the bottom of the *Geometry* workspace window. A valid mesh must be generated to use these viewing options.

The first mode, *Mesh Cutplanes*, creates cutplanes of the mesh in any or all of the three primary planes. Toggle any of these cutplanes on or off by checking or unchecking their respective boxes. The sliders associated with each of these planes are enabled when its respective plane is turned on. The slider moves the cutplane throughout the slices in the mesh. Additionally, each checked plane will activate the following icons, which aid in manipulating the cutplanes view:



- Cuts Away solid geometry on the -Z side of the mesh slice.



- Cuts Away solid geometry on the +Z side of the mesh slice.



- Toggles viewing edges normal to the slice.



- Toggles viewing the grid.



- Toggles rendering of all the electric components.



- Toggles rendering of all the magnetic components.

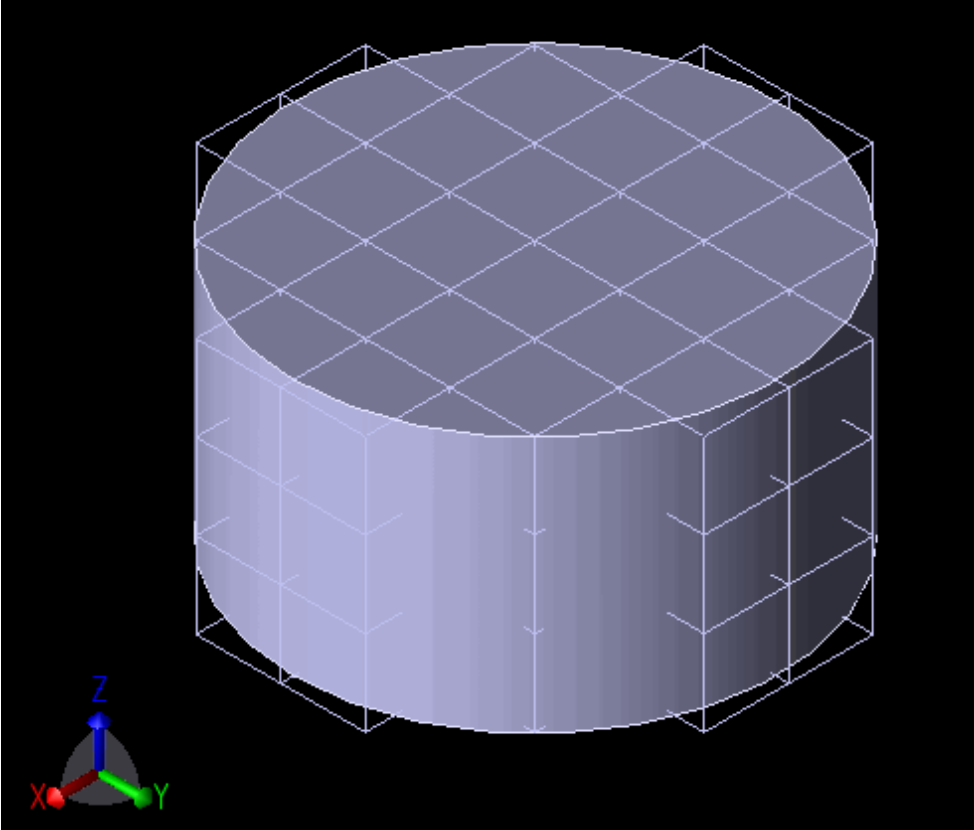
There are also two check boxes available:

- **View Mesh Information** - Displays a dialog box with information about the mesh at the location of the mouse.
- **Synchronize Sliders** - Moves the cutplane simultaneously with a slider adjustment while the mouse button is still pressed.

The following illustration shows this first mode that is displayed when the Mesh View icon is selected. Note that this is only a preview of the mesh when it is shown while editing the grid within the Grid Tools dialog box. Any other time, it is a representation of the most recently generated mesh.

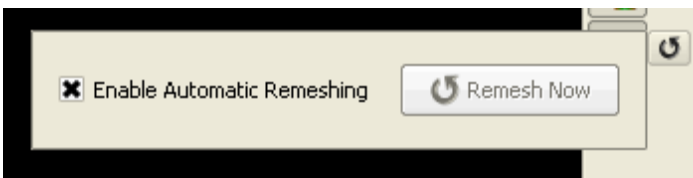
The second mode, *3D Mesh*, provides several different options to view the mesh. Select any of the radial buttons within this option to create a customized three-dimensional view of the mesh. The Mesh Visibility icons are available in this mode, which enables you to toggle the Electrical and Magnetic components of the mesh. Additionally, the Material Visibility button controls which specific materials are visible in the view.

The following figure below displays an object in 3D Mesh mode with All Edges displayed.



Meshing Options

The *Automatic Remeshing* feature is located in the *View Tools* toolbar. When this feature is enabled, remeshing is automatically performed any time a change is made to the geometry. If this feature is not enabled, remeshing must be performed manually. Automatic Remeshing is not desirable when large meshes are imported because of their memory and performance demands.



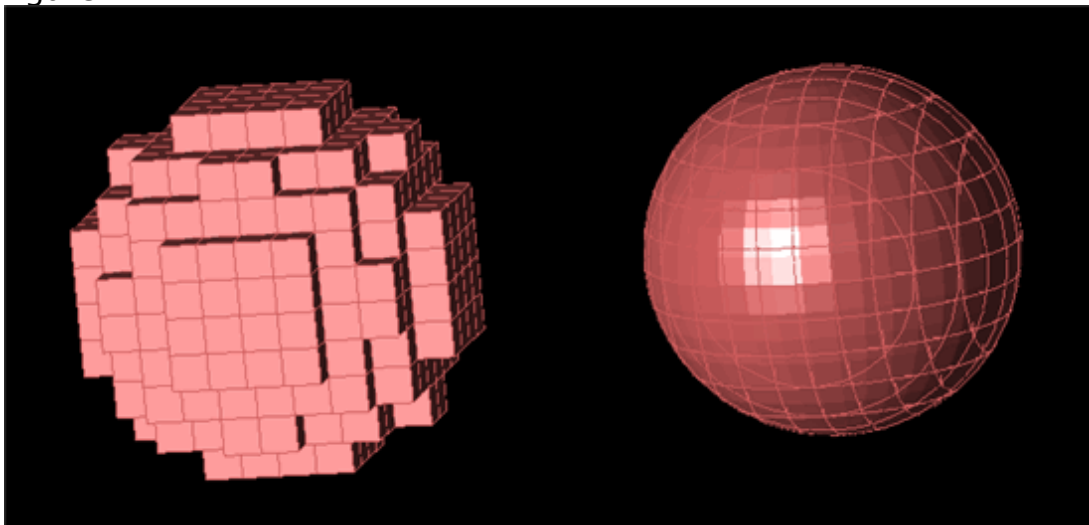
Using Conformal FDTD Meshing

You can control the discretization of the domain in FDTD with standard meshing by setting appropriate values for the cell sizes. The cell size is often limited by the dimensions and the features of the geometric structure. This in turn determines the runtime cost and the memory consumption of the FDTD simulation. The new conformal meshing feature enables you to specify a coarser grid without compromising the accuracy too much. Conformal meshing consists of a set of techniques that use geometric information to provide a subcellular discretization of the computational domain in order to obtain accurate simulation results for a given grid resolution. In other words, one does not have to specify smaller cell sizes in order to increase the accuracy, or one can reduce the overall simulation time and memory requirements for a given level of the desired accuracy.

For example, a highly curved radiating body requires a fine grid with the standard meshing technique in order to obtain accurate results. Appropriately placed fixed points and grid regions make this grid refinement straightforward, and ultimately provide good accuracy in the simulation results. However, increasing the number of cells in each of the X, Y, and Z directions by a factor of N leads to a factor of N^3 increase in memory consumption and up to an N^4 increase in simulation time. Therefore, it can be difficult with the standard meshing technique to sufficiently refine the computational grid while keeping the simulations fast and small. In this case FDTD with conformal meshing allows you to specify a coarser grid and thus reduces the computational costs while obtaining a similar level of accuracy.

Using conformal meshing:

- It is possible to represent gaps between conductors in the mesh without having a full grid cell distance between the conductors. This allows you to increase the cell size and timestep, resulting in shorter simulation times.
- Curved structures can be modeled more accurately than can be done with standard meshes only, since the conformal mesh represents the boundary of the object as actually curved inside of a grid cell. Thus, using conformal meshing can improve the accuracy of your simulations for a given grid resolution, as shown in the following figure:



A conformal mesh uses more geometry information than required by a conventional mesh for computing fields near the boundary of parts. This incurs execution time and memory usage penalties when conformal mesh is enabled. In order to avoid these penalties, you can refer the following guidelines while using the conformal mesh features:


- The conformal meshing feature is most useful for parts with high conductivity or Perfect Conductor electric material properties.
- Conformal meshing is useful on parts with high curvature or where the size of small geometric features is significant.
- Axis-aligned geometry that is already well-represented by a conventional mesh will not benefit by using conformal meshing.
- Conformal meshing is not intended for use on parts with cross-sectional size that is smaller than a single grid cell. In such cases, it may not be possible to generate a conformal mesh that maintains connectivity without enabling aggressive surface meshing as well.

Enabling Conformal Meshing

To enable the conformal meshing feature on a single part, perform one of the following steps:

- Right-click the *Parts: Modelname* branch of the *Project Tree* and select *Meshing Properties* to open the **Meshing Properties Editor** window.
- Right-click the *Parts* in the *Project Tree* and select **Enable Conformal Mesh** from the **Gridding/Meshing** sub menu.
- If an assembly is selected in the *Project tree*, the select **Subparts > Gridding/Meshing > Enable conformal Mesh on all Subparts**.

In order to mesh a part using the conformal meshing feature, the assigned material part must be classified as either an electric Perfect conductor, as a Good conductor, or as otherwise having low conductivity. This classification is controlled by the nature of the material, and also by Good Conductor choice in the Material editor.

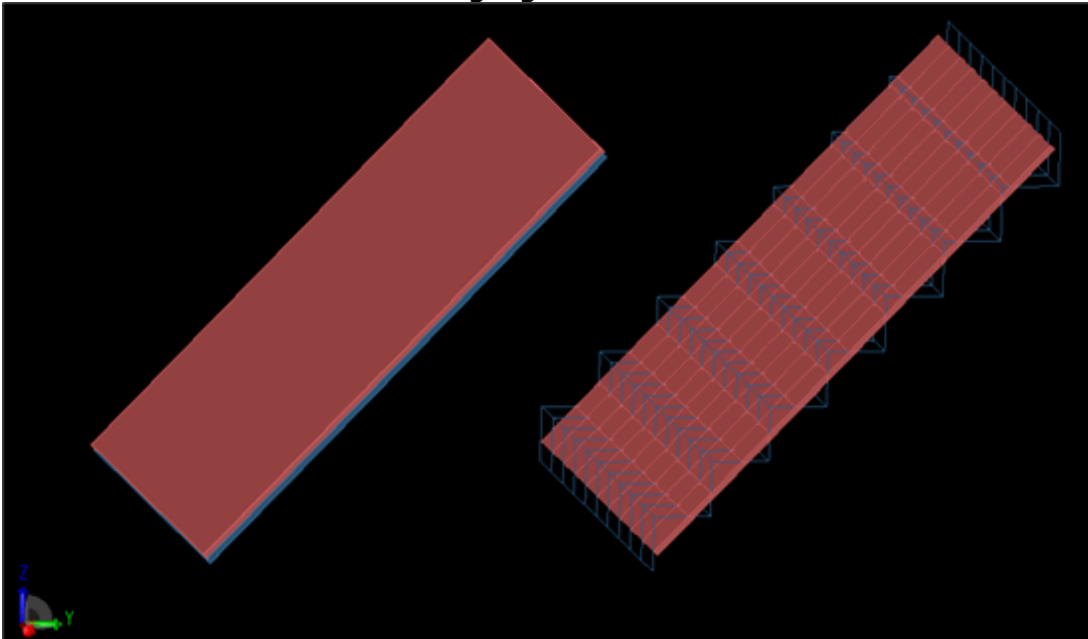
 The Conformal formulation for parts meshed with Good Conductor materials assumes a minimum of five skin depths thickness to the part of all locations, thus blocking all energy transfer through the part.

When the conformal meshing feature is enabled for any part in a simulation, EMPro, by default, reduces the simulation timestep by as much as 30% in order to maintain numeric stability in the computation. The maximum amount of timestep reduction is controllable from the Conformal Accuracy setting on the Advanced tab of the Project Properties Editor. This setting allows you to control the simulation speed against the accuracy of the subcellular representation of geometry. Large values of this setting (near 1.0) will cause the simulation timestep to be reduced very little, thus giving the fastest simulation runs. Small values of this setting (near 0.1) may cause the simulation timestep to be significantly reduced (as low as 10% of its usual value) but will most accurately represent your geometry. The default value of 0.7 should be reasonable in most cases.

Limitations of Conformal Meshing

The conformal meshing feature has the following limitations:

- You can use the conformal meshing feature only with simulations excited by Discrete Sources and Total Field Plane Waves. You cannot use Scattered Field Plane Wave, Gaussian Beam, and Static Voltage Points with conformal meshing.
- You can enable the conformal meshing feature only on parts using nondispersive and isotropic electric materials. Some material types cannot exist on or near a part meshed with the conformal meshing feature in EMPro. These include Dispersive, Thin Wire, and Nonlinear electric materials, and any magnetic material type other than Magnetic Freespace. Furthermore, when an Averaged Electric Material is encountered nearby a part meshed with conformal feature, the Dominant Material of the average is used in its place.
- The conformal-enabled parts cannot come within one cell of the simulation boundary.
- You should take care when a thin part with an Conformal mesh is present less than a grid cell away from a part that is conventionally meshed. As the conventionally meshed part finds the best fitting set of Cartesian mesh edges to describe its surfaces, it may choose some mesh edges on the opposite side of the conformal meshed parts. In such cases, the best recourse may be to enable Conformal feature for both parts, although judicious use of fixed points may correct this behavior in some situations. The following figure illustrate this scenario:



- Parts having conformal meshing use a more faithful meshed representation of the geometry than a conventional Cartesian mesh can provide. This can have unexpected consequences in case where there are small gaps in your parts which a conventional mesh would typically short together. The conformal mesh will never do this. Thus, it is critical when using conformal meshing that the CAD representation of your parts contains no unwanted gaps.

Viewing FDTD Simulation Results

This section provides information about the following topics:

- *Viewing FDTD Default Output* (fDTD)
- *Viewing 2D Plots* (fDTD)
- *Viewing 3D Fields* (fDTD)

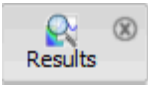
Viewing 2D Plots

You view FDTD simulation in the form of plots XY plots, polar plots, and smith charts. The following types of plot categories are available, depending on the **X**-axis (abscissa) type:

- **XY Plots:** Enables you to visualize data in a Cartesian coordinate system.
- **Polar Plots:** Represents data with an angle-independent axis.
- **Smith Charts:** Displays complex data as per the required frequency, such as S-parameters and reflection coefficient.

Generating Plots

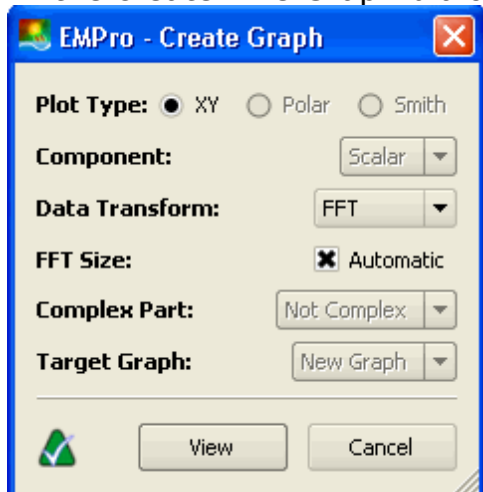
To generate 2D plots:

1. Click **Results** () to open the workspace.
2. Select a project from the **List Project** drop-down list.
3. Right-click the required result type in the **Results** workspace window and select **Create Line Graph**, as shown in the following figure:

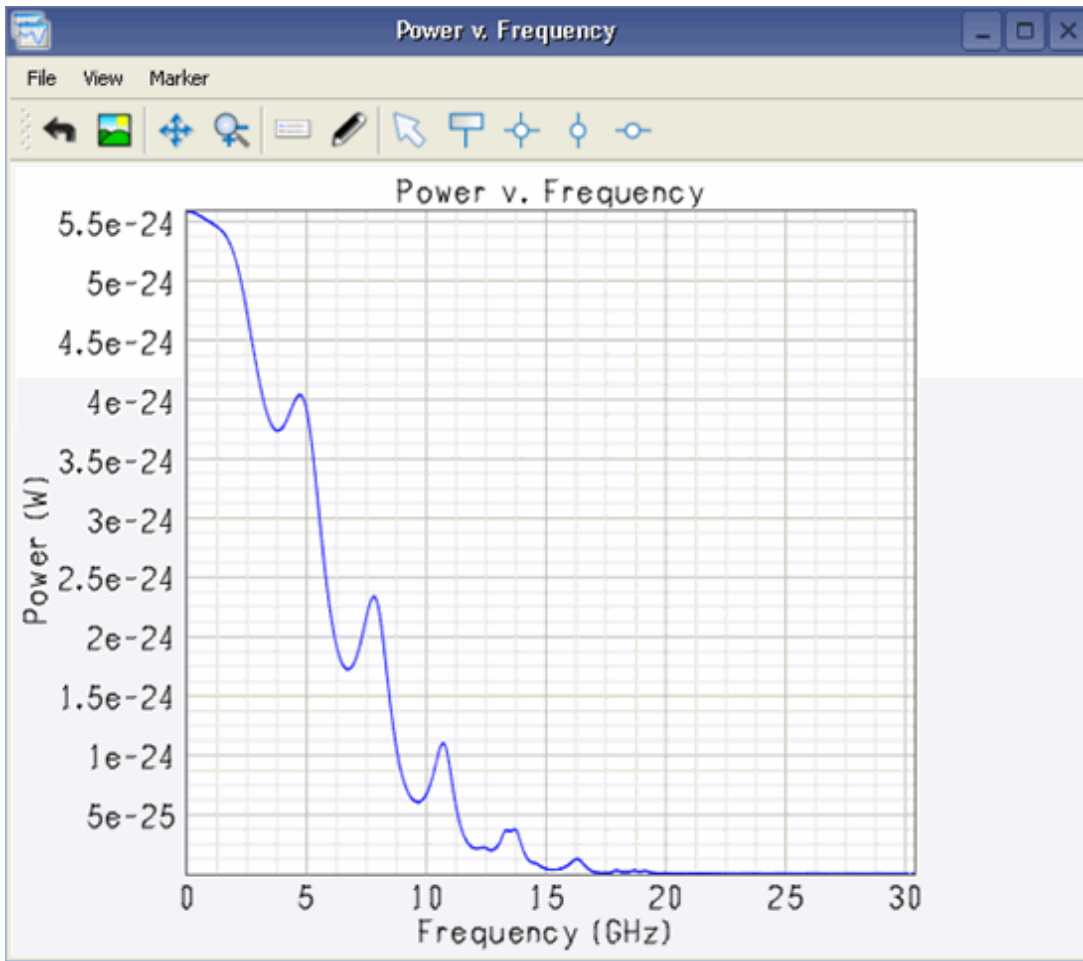
Sensor Type	Domain	Field Type	Status	Misc Info
System Sensor	Frequenc	...	Complete	Active port: 1
System Sensor	Frequenc	...	Complete	Active port: 1
Circuit Component	Time	...	Complete	Active port: 1
Circuit Component	Time	...	Complete	Active port: 1
Circuit Component	Frequenc	...	Complete	Active port: 1

View (default)
 View Smith Chart
Create Line Graph...
 Postprocess Results...
 Export ▶
 Unlist Project
 Copy Python Expression

4. In the create Line Graph dialog box, you can select the required plot type



5. Specify the graph properties such as the component, data transform, and complex part. The Target Graph option enables you to view and edit plots that were previously created (from the same data selected in the Results window).
6. Click **View**. The 2D plot is displayed, shown in the following figure:



For all calculations, the most important quantities are the time-domain plots of the fields in the problem space. Always perform a quick review of these values to ensure that a calculation has converged. Without convergence, most other results will be meaningless, particularly any plots converted to the frequency domain such as S-parameters or impedance.

Note You can plot input impedance and S-parameters in rectangular form versus frequency or as a Smith chart.

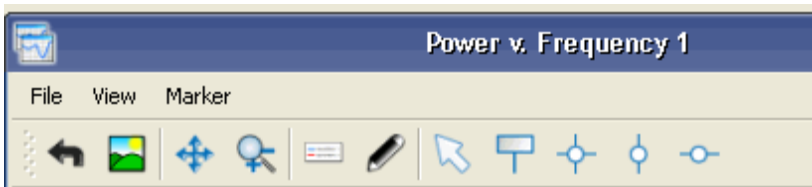
The following table describes the result data plots:

FDTD Simulation



Sensor Type	Time Dependence	Result Type
Port Sensor	Broadband	Current
		Impedence
		Input Power
		Instantaneous Power
		Reflection Coefficient
		S-Parameters
		VSWR
Point Sensor/Surface/Volume	Broadband	E-Field (E)
		H-Field (H)
		B-field (B)
		Conduction Current (Jc)
		Poynting Vector (S)
System Sensor	Broadband	Net Feed Loss
		Net Input Power
Far Zone Sensor	Broadband	Radar Cross Section
		Gain
		E-Field (E)
Far Zone Sensor	Single-Frequency	Radar Cross Section
		Axial Ratio
		E Theta, E Phi
		Circular Polarization
		Ludwig-2 Az, EI
		Ludwig-2 AI, Ep
		Ludwig-3
Far Zone Post Processor	Single-Frequency	E-Field (E)

Customizing Plots



You can customize the data plots in the displayed plot window. The tools available within the plot window depend on the type of graph you are viewing, such as XY Plots, Polar Plots, or Smith Charts. You can use the Properties toolbar to customize plots. The following figure displays the Properties toolbar:



Use the following options available on the Properties toolbar:

- Export Data (): Select this option to export graphical data point values to a text file in a specified directory.
- **Export Image Tool** (): Select this icon to save an image of the current plot to a

specified directory. |

- **Pan Tool** (): Select this tool, and drag the mouse within the plot to pan to the required view of the plot. Press the **|Ctrl|** key and drag to pan along the independent axis. Press the **|Shift|** key and drag to pan along the dependent axis. |
- **Zoom Tool** (): Select this tool to zoom-in and to zoom-out of the plot. The mouse wheel as well as the right and left mouse buttons, can be used to perform zoom operations as described below. Using mouse-wheel:

Roll the center wheel of the mouse forward to zoom-out of both axes simultaneously.

Press **|Ctrl|** and roll the center wheel forward to zoom-out of the independent axis.

Press **|Shift|** and roll the center wheel forward to zoom-out of the dependent axis.

Roll the center wheel of the mouse backward to zoom-in to both axes simultaneously.

Press **|Ctrl|** and roll the center wheel backward to zoom-in to the independent axis.

Press **|Shift|** and roll the center wheel backward to zoom-in to the dependent axis.

Using mouse buttons:

Right-click and drag the the mouse anywhere in the plot window to zoom-out of both axes simultaneously.



Press **|Ctrl|** and right-click/drag to zoom-out of the independent axis.

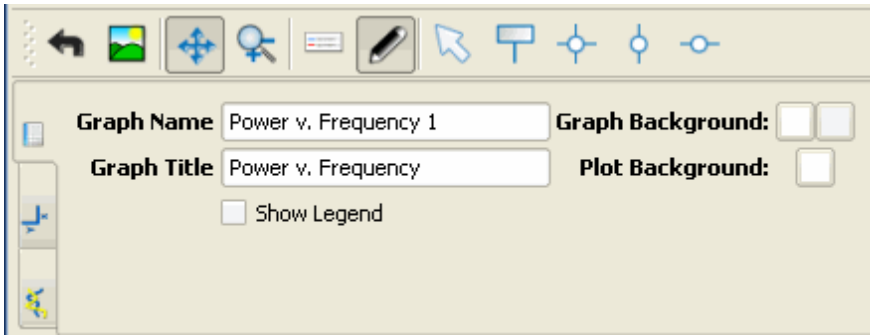
Press **|Shift|** and and right-click/drag to zoom-out of the dependent axis.

Left\click and drag to define a rectangular view-window in the plot window to zoom-in to both axes.

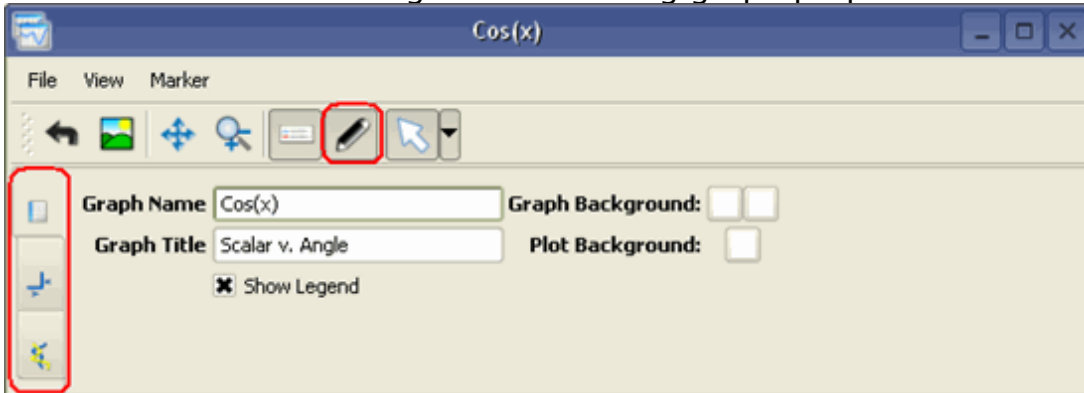
Press **|Ctrl|** and left-click/drag to zoom-in to the selected domain of independent axis values.

Press **|Shift|** and and left-click/drag to zoom-in to the selected range of dependent axis values.

- **Legend Visible** (): Enables you to toggles the display of the legend within a graph.
- **Graph Properties Tool** (): Enables you to define graph name, title of the axes, and characteristics of the plotted lines. Clicking **Graph Properties Tool** displays options for customizing the properties of any graph, as shown in the following figure:



It consists of the following tabs for editing graph properties:

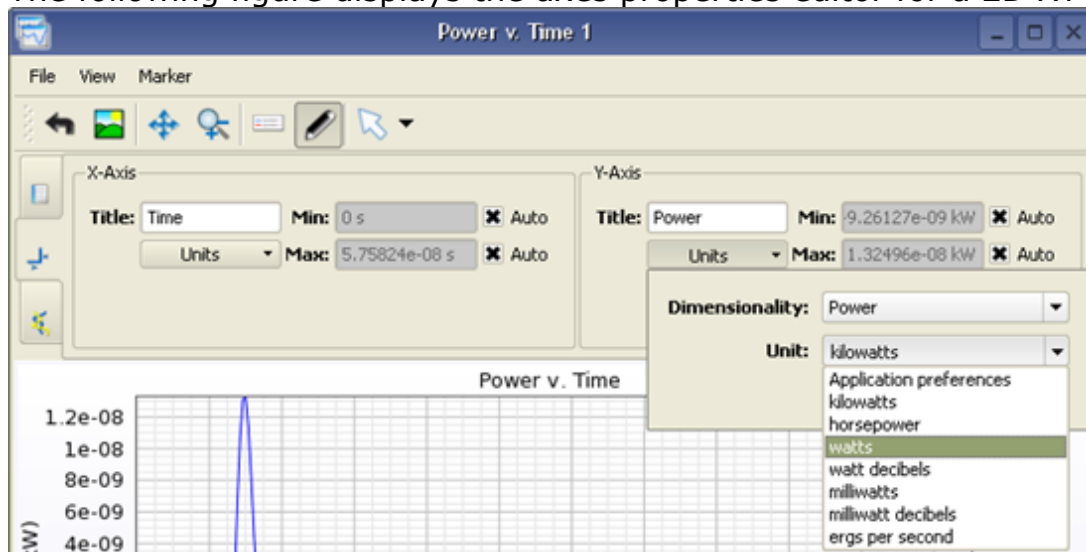


- Graph Properties: Enables you to define the graph name, title, and background color of the graph. You can select the Legend Visible check box to toggle the legend display on and off. This tab is displayed in the figure above.
- Axes properties: Enables you to define the title of the axes and the limits of the axes in this tab. The *Auto* checkbox may be selected to auto-select these limits.

Note
If a graph only contains continuous data, the **X**-axis must be specified (this will not happen automatically).


You can use the **Units** drop-down list to specify the units and to apply a log scale if necessary. For few Smith plots, you can modify the *Reference Impedance*.

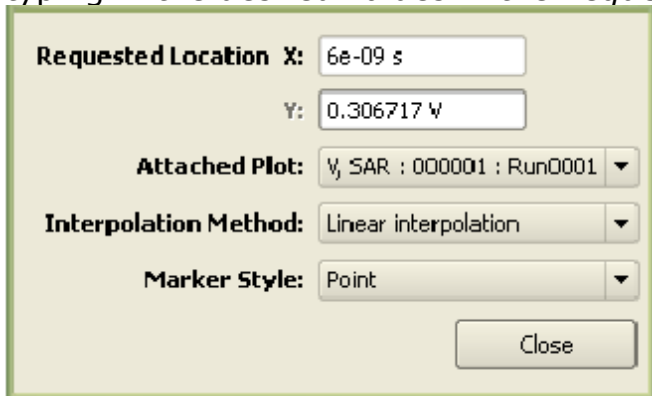
The following figure displays the axes properties editor for a 2D XY graph:



- Plot properties: Define the characteristics of the plotted lines in this tab. A list of every dependent variable is listed with a customizable line color, line width, and

line style. Any unwanted variables can also be deleted in this tab.

- Selection Tool** (): Select this tool to move, delete, or edit a marker's properties. To move a marker, click on the marker with the selection tool (once selected, it will turn yellow) and roll the mouse-wheel forward or backward to move an attached marker along its plot. Pressing **|Ctrl|** and rolling the mouse-wheel will speed up the movement of the marker. To delete any marker, simply select the unwanted marker, right-click and press *Delete Marker* . All markers may be deleted at one time by right-clicking anywhere in the plot area and choosing *Delete All Markers* . To edit a marker's properties, click on the desired marker so that it turns yellow, then right-click and select *Marker Properties* . A window will appear with several different editing options. The location coordinates of the marker can be adjusted by manually typing in the desired values in the *Requested Location* section of the dialog box.

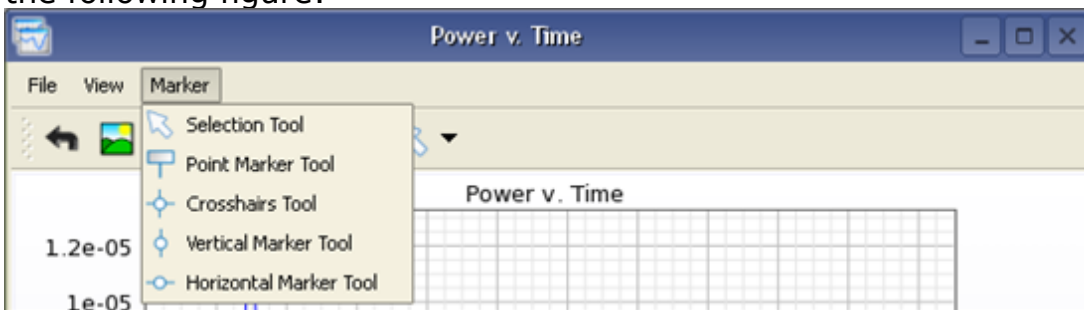



Note

If the coordinate box you want to edit is disabled, select the appropriate option in the *Attached plot* drop-down menu.

The marker may also be attached to a particular plot by selecting its name in the *Attached Plot* drop-down menu and selecting an *Interpolation Method* for EMPro to use in order to place the point. Depending on this definition, the marker will be shifted to the nearest point on the selected function or linearly interpolated based on the independent axis that is entered by the user in the *Requested Location* dialog box. Finally, the type of marker can also be redefined in this editor window in the) *Type* drop-down list.


Additionally, this tool can be used to move or close the legend in the graphical space. Select this tool by clicking its icon or select it from the *Markers* menu, as shown in the following figure:

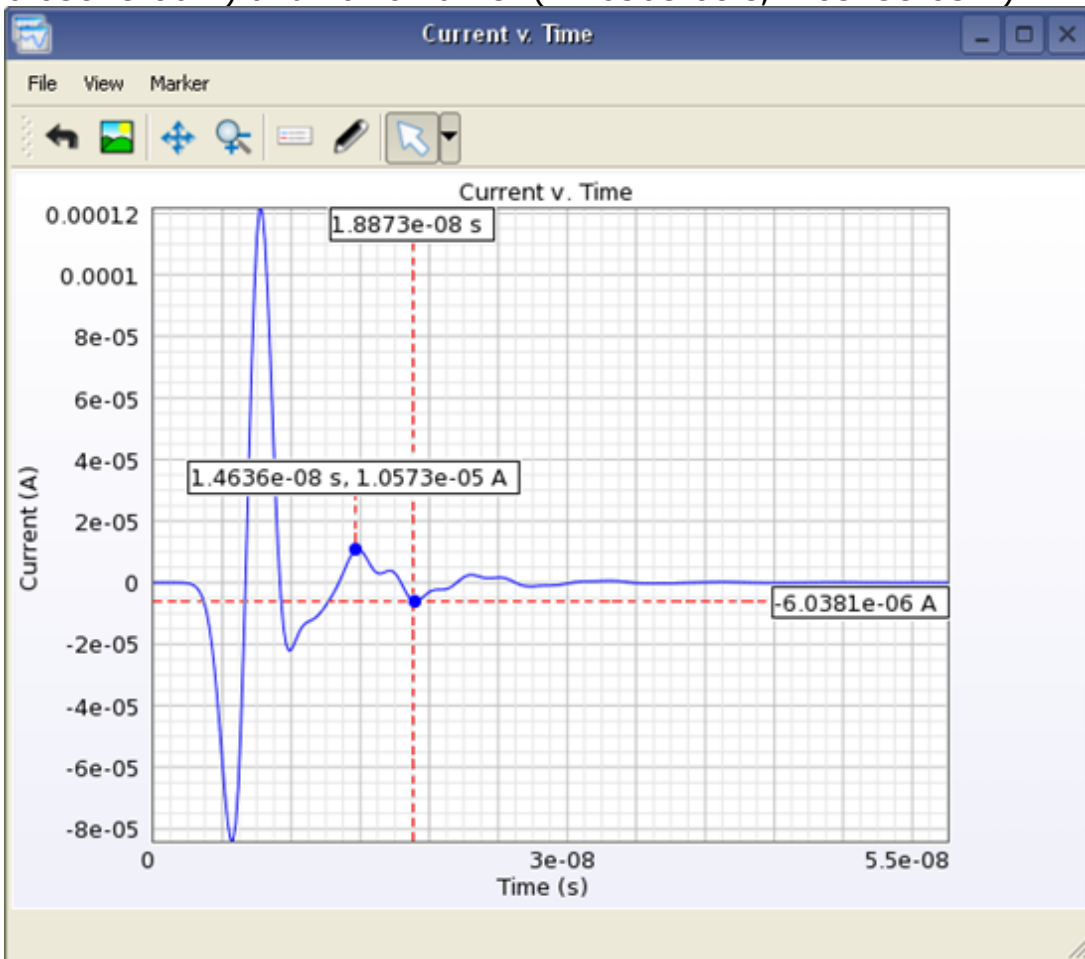


- Point Marker Tool** (): Select this tool to mark any point on the plot by clicking on the desired marker location. A marker with the location coordinates appears above

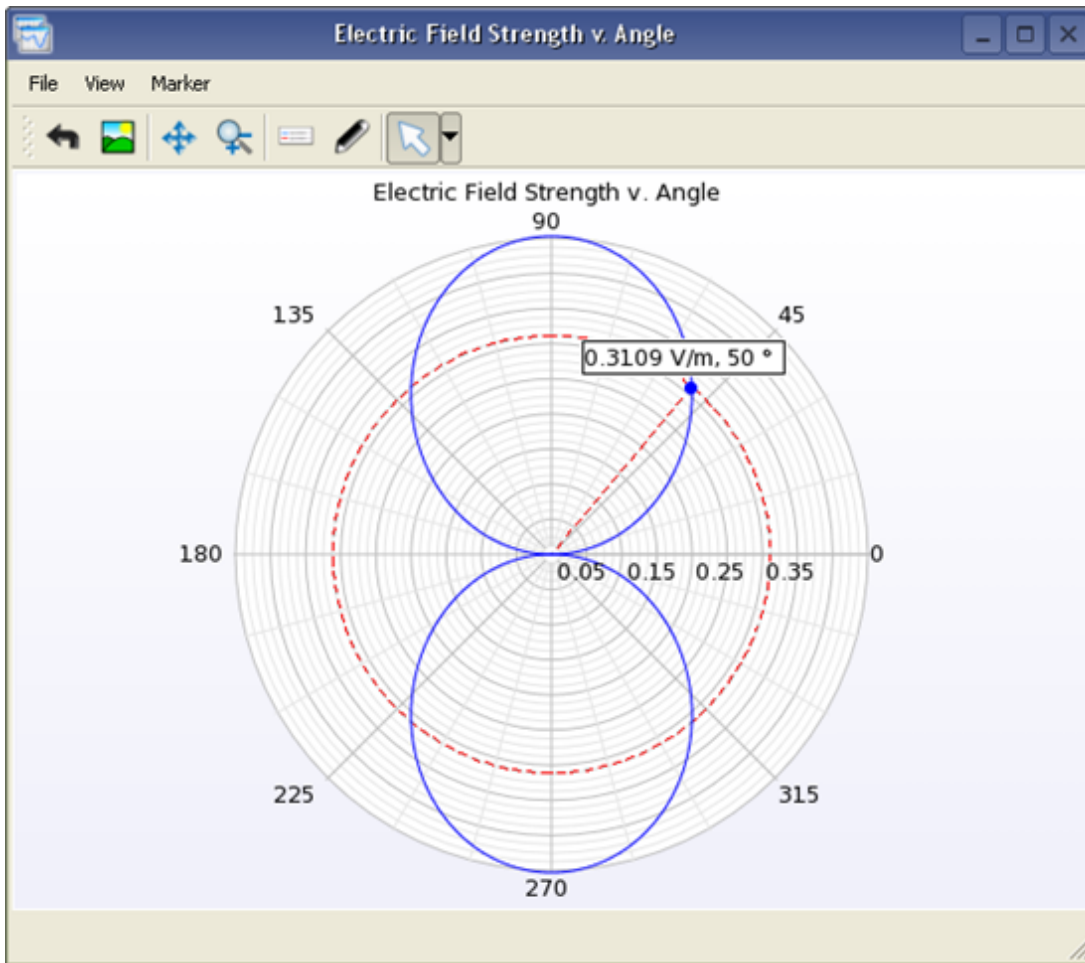
the point, which depend on the type of plot:

- **XY Plot:** (*X* -location, *Y* -location)
- **Polar Plot:** (radius, angle)
- **Smith Plot:** (real part of location, imaginary part of location, frequency)
 When the mouse moves close to the plotted curve, it is snapped to the closest location on the interpolated line or sampled point. Holding the **[Ctrl]** key will disable the snapping action, allowing a point to be placed anywhere. Holding the **[Shift]** key will snap the marker to sampled points only. Note that markers placed on sampled points are blue, and markers placed on interpolated points are black.

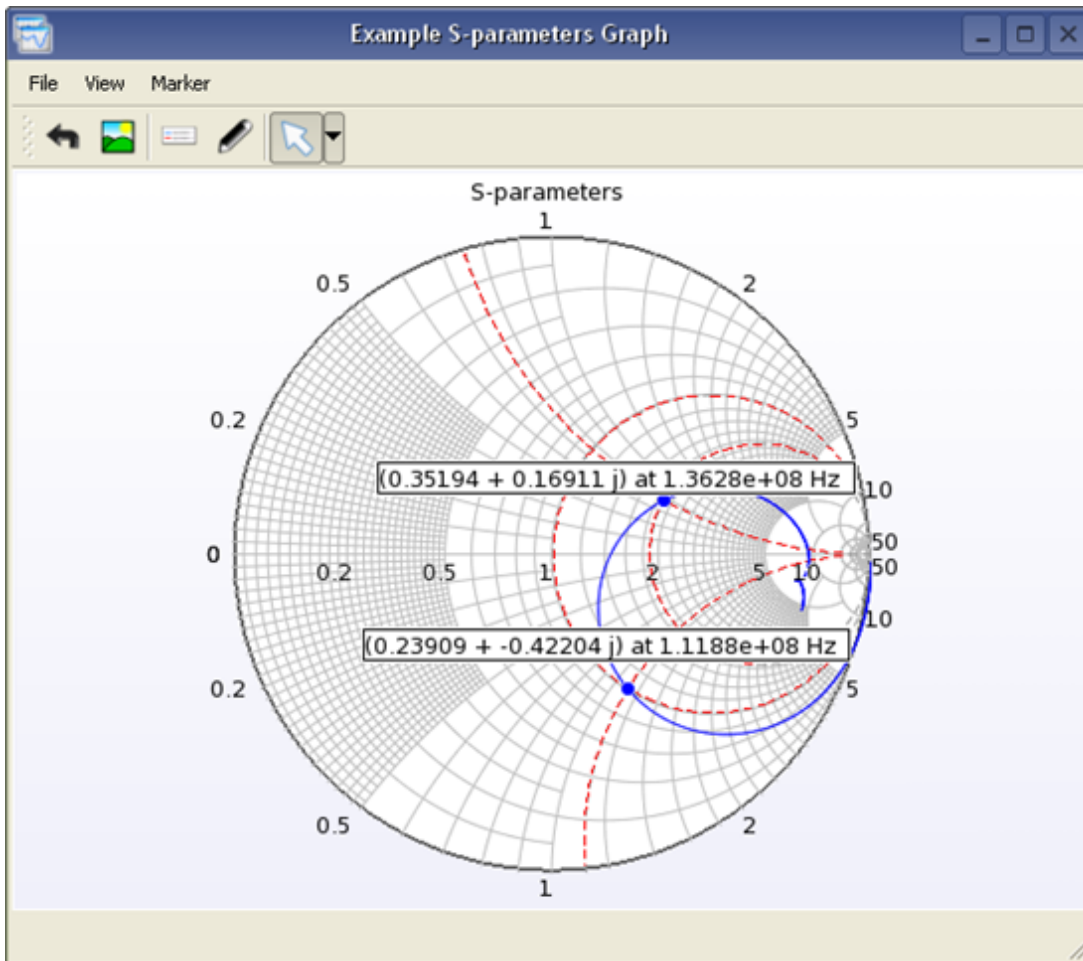
- **Crosshair Marker Tool** (): Select this tool to mark the location of a single point by two intersecting cross-hairs. The marker is placed at the right edge of the plot. (Snapping actions are the same as the Point/Tracker Marker described above.) The following figure highlights an *XY Plot* with a Crosshair Marker ($1.8873e-08$ s, $-6.0381e-06$ A) and Point Marker ($1.4636e-08$ s, $1.0573e-05$ A).





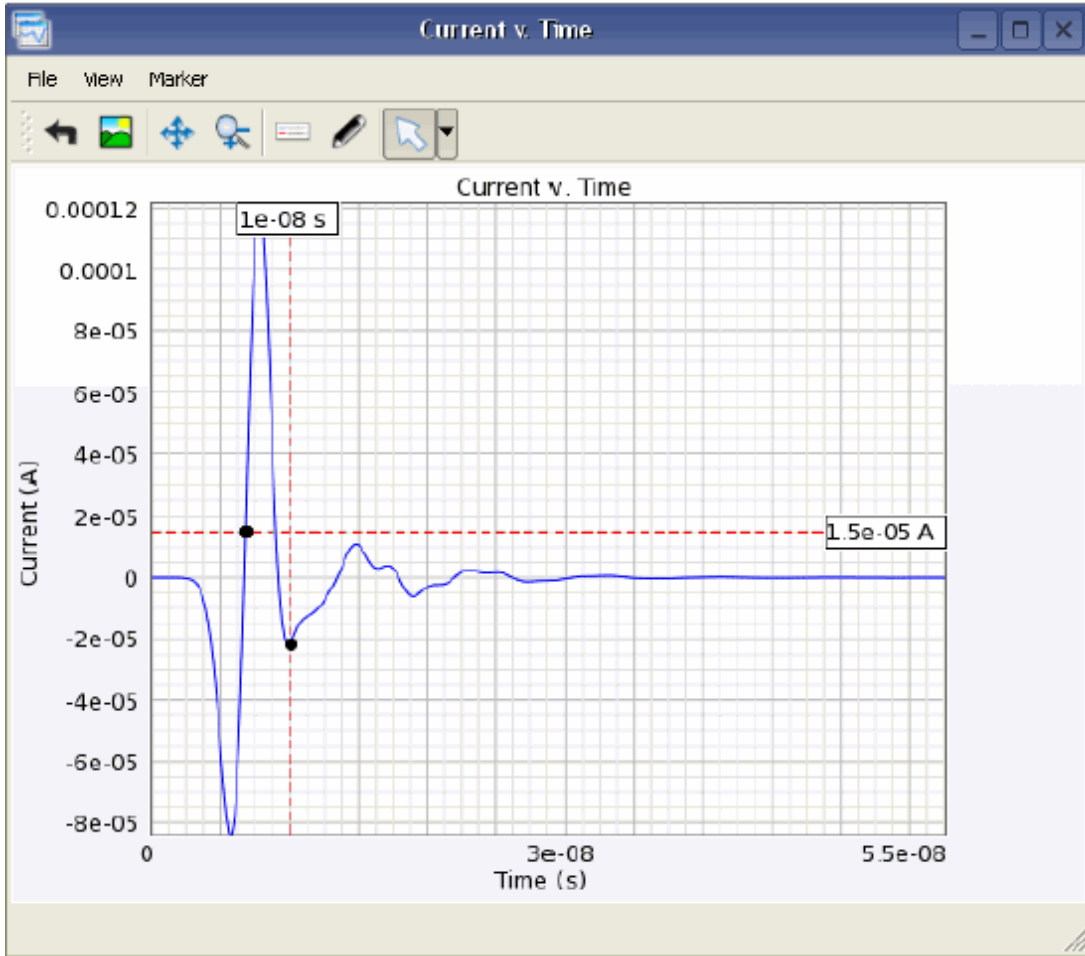
The following figure highlights a polar graph with a Crosshair Marker along a radius.



The following figure shows a smith graph with a *Crosshair Marker* along a radius.



- **Vertical Marker Tool** (): Select this tool to place a vertical line that intersects with the **X**-axis. The marker (**Y** = constant) will be placed along the top-edge of the plot area. (Snapping actions are the same as the Point/Tracker Marker described above.)
- **Horizontal Marker Tool** (): Select this tool to place a horizontal line that intersects with the **Y**-axis. The marker (**X** = constant) will be placed along the right-edge of the plot area. (Snapping actions are the same as the Point/Tracker Marker described above.) The figure below shows a 2-D XY Graph with a Vertical Marker (at **X** = 1e-08 s) as well as a Horizontal Marker (at **Y** = 1.5e-05 A).



Viewing 3D Fields

You can view colored 3D field displays by saving either broadband or single-frequency field quantities in the geometry (solid or meshed). You can view the following types of fields with the geometry:

- **Time Domain Snapshots** - available for any calculation as they are simply snapshots of the near-zone fields at specific steps in time.
- **Complex Fields or Derived Quantities** (such as SAR) - available at specific frequencies
- **Three Dimensional Far-Zone Fields** - can be requested either before the calculation with a far-zone sensor, or after a calculation through post-processing.

Viewing Data Collected With Near-Field Sensors

Field snapshots are listed in the field control panel as either single slices or as field sequences. The field sequences are movies of the individual slices as the fields progress with time in a particular slice of the geometry. Depending on what was saved, each field snapshot may have electric and magnetic fields, current densities, Poynting vectors stored for each direction (**X**, **Y**, and **Z**) and a display of the combined magnitude. Field snapshots are collected by surface sensors and solid volume sensors for broadband data.

**Note**

In order to view single-frequency data, the *Collect Steady-State Data* check box must be selected in the *Simulations* workspace window (present in the *Frequencies Of Interest* tab).

Results Plots Available For Each Sensor Type

Sensor Type	Result Type	Time Domain	Discrete-Frequency	Broadband
Point Sensor	E-Field (E)	X		X
	H-Field (H)	X		X
	B-field (B)	X		X
	Poynting Vector (S)	X		X
	Conduction Current (Jc)	X		X
	Scattered E	X		X
	Scattered H	X		X
	Scattered B	X		X
	Average Power			X
	Surface/Solid Sensor	E-Field (E)	X	X
	H-Field (H)	X	X	
	B-field (B)	X	X	
	Conduction Current (Jc)	X	X	
	Scattered E	X	X	
	Scattered H	X	X	
	Scattered B	X	X	
	Poynting Vector (S)	X		
	Average Power		X	
SAR Sensor	SAR (Specific Absorption Rate)		X	
HAC Sensor	E-Field (E)		X	
	H-Field (H)		X	
	HAC max E-Field (E)		X	
	HAC max H-Field (H)		X	

(Scattered) Electric Fields (E): magnitude, normal, or **X**, **Y** or **Z** components of the (scattered) electric field data at each cell edge.

(Scattered) Magnetic Fields (H): magnitude, normal, or **X**, **Y** or **Z** components of the (scattered) magnetic field data at each cell edge.

(Scattered) Magnetic Flux Density (B): magnitude, normal, or **X**, **Y** or **Z** components of the (scattered) B-field computed from the magnetic fields at each cell edge and the associated permeability for that cell edge.

Average Power Density (SAV): magnitudes of the average power density computed from the electric and magnetic fields at each cell edge.

Conduction Current Magnitude (Jc): conduction current at an electric field cell edge.

SAR (Specific Absorption Rate): computed for each complete cell containing a lossy dielectric with a non-zero material density.

Viewing Data Collected with Three Dimensional Far-zone Fields

The Results workspace window also displays the data collected within 3D far-zone fields.

Any 3D far-zone request generated using the post-processing engine (run separately from the calculation engine) is automatically added once the post-processing is complete. If steady-state far-zone data is enabled for the sensor, the 3-D results will include the E-field and axial ratio in the discrete-frequency domain. Gain or discrete-frequency radar cross section (RCS) will also be available when using a feed or external excitation, respectively.

Note
Note that the polarization (Theta/Phi, Ludwig-2, etc.) is selected through the *Setup* tab of the *Field Editing Toolbar*.

Viewing 3-D Field Displays

EMPro displays 3D field data in the Geometry window. The following subsections discuss how to configure and analyze the display, using the *Scale Bar*, the *Field Reader Tool*, rescaling, and the field editing toolbar.

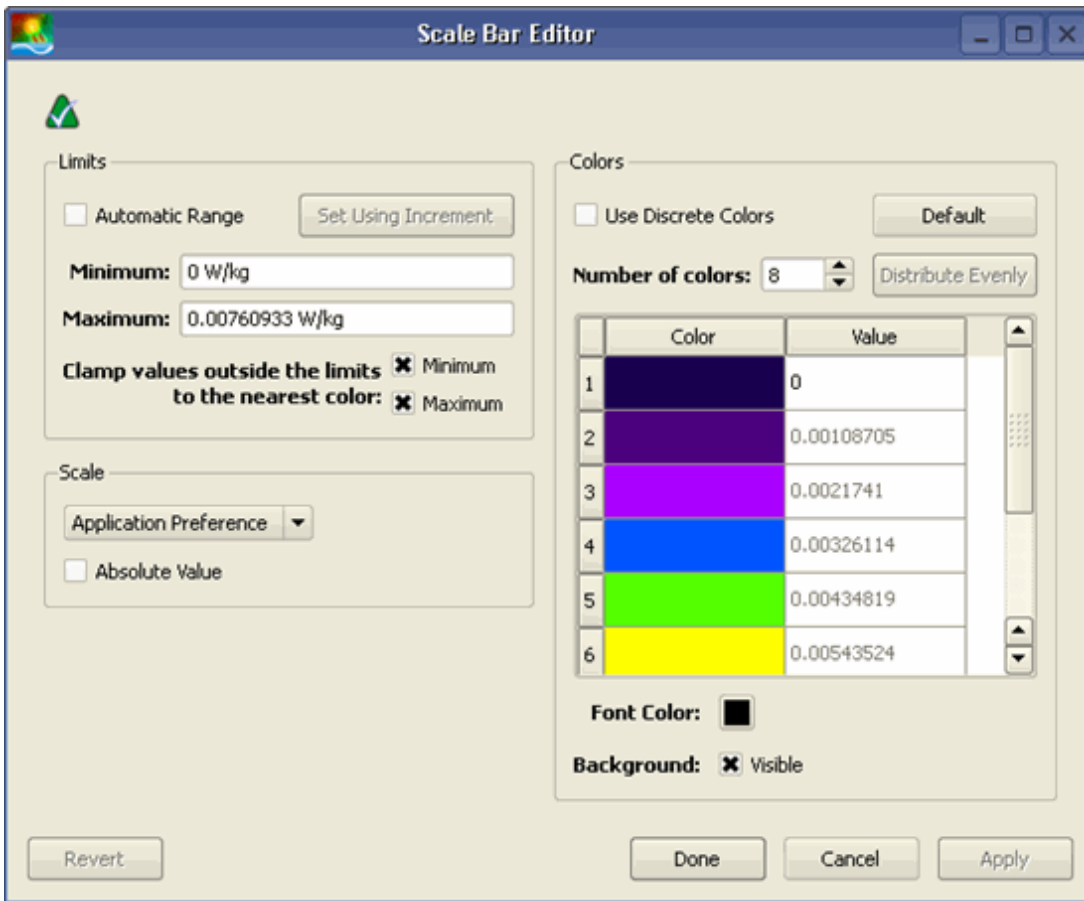
Using the Scale Bar

The *Scale Bar*, located at the top of the *Geometry* workspace window, "paints" the view with a range of colors which correspond to the range of values displayed. By default, the *Scale Bar* color palette is shown in continuous mode within a default range of values. You can adjust these properties by right-clicking on the scale bar and selecting *Discrete Mode*, to change the palette to discrete colors, or *Automatic Range*, to change the range of values to that which is actually present.

There is also a *Properties* option under the right-click menu. Clicking this will bring up the Scale Bar Editor. Using this editor, you can manually set the *Scale Bar* limits, units and colors to your preference. Take note of several settings that may not be intuitive for the first-time user:

- Under the *Limits* section, it is possible to "**clamp**" values outside the defined scale bar limits to the nearest color. This makes it possible to view outliers that otherwise would not be colored with the *Scale Bar*.
- Under the *Scale* section, when *Relative dB* is selected, you can define its reference value or use the value automatically selected by EMPro.
- Checking *Use Discrete Colors* makes it easier to view contours in the 3-D results.
- *Font Color* and *Background Color* change the colors of the *Scale Bar* itself, not the coloration of the results.

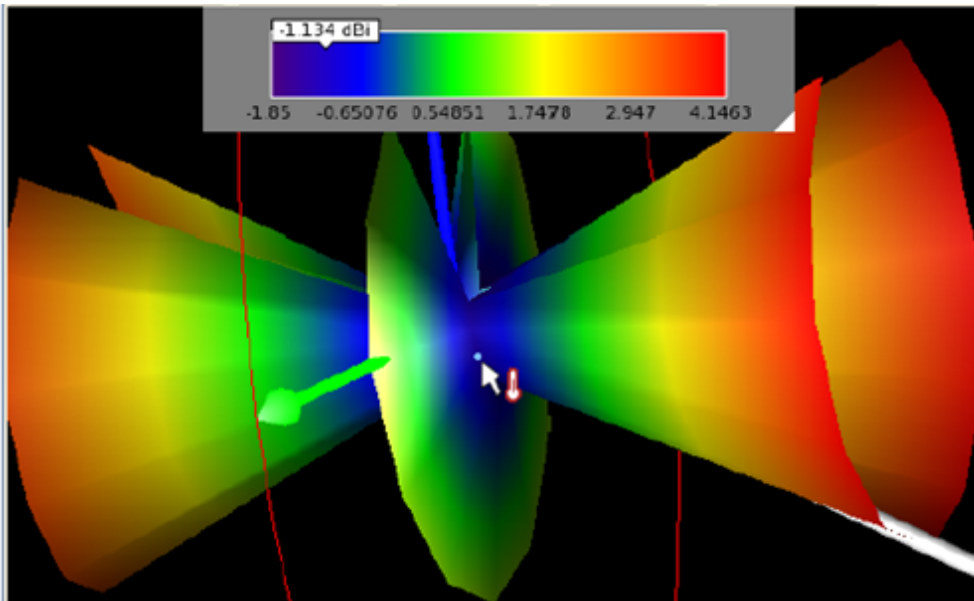
The following figure displays the Scale Bar Editor dialog box:



Using the Field Reader Tool

The *Field Reader Tool* is located in the toolbar to the right of the *Geometry* workspace simulation space. After selecting this icon, wheel the mouse over the geometry object to identify its field values. A marker in the Scale Bar will display the nearest known field value to the location of the mouse. This location is represented by a small dot on the screen.

The Field Reader tool



Rescaling

In the *Results* workspace window, filter the results by the output object System. Double-clicking on results such as *Dissipated Power* or *System Efficiency* will open the *System Sensor Output* dialog box. Changing any value in this table will rescale the other values shown.

Note

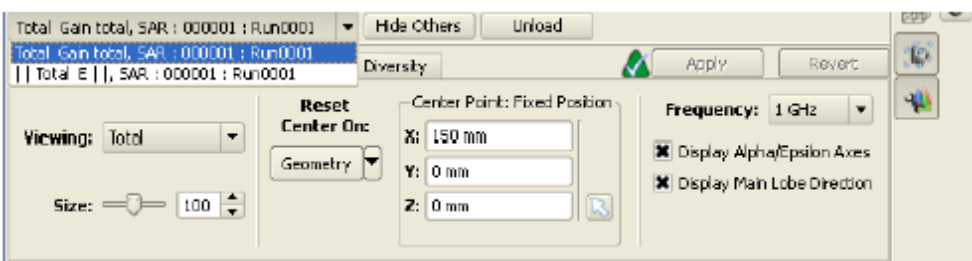
The *Show Scaled Values* box must be selected to enable editing in the *System Sensor Output* dialog box.

- Scaling only affects results which are in the discrete frequencies domain.
- When you change the scaling by editing values in the *System Sensor Output* dialog, you only scale the results for that particular calculation engine run at that particular DFT frequency. Any other tables of data and plots associated with this run and frequency will also be affected.

Using the Field Editing Toolbar

Located at the bottom of the *Geometry* workspace window, the field editing toolbar is used to configure the properties of the view. In the upper-left corner of this toolbar, a drop-down list will display any view that you have opened from the *Results* window. You can use the *Hide Others* and *Unload* buttons to single out certain view(s) if necessary. The following figure shows a drop-down list of such results in the field editing toolbar.

The results drop-down list of the field editing toolbar



The tabs and configuration options below will change depending on the active view. A comprehensive list of the available options are described below.

Using the Setup Tab

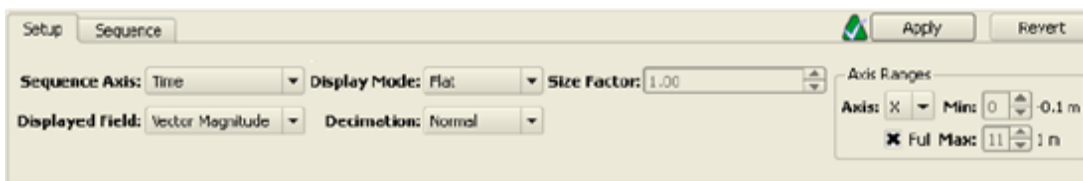
There are two main configurations for the *Setup tab*, depending on the type of sensor you are evaluating.

- Keep in mind that not all fields listed will be available for every view.

Surface, Solid SAR Sensor Configuration

- **Sequence Axis** - Select the axis along which to view the sequence (from the *Sequence tab*).
- **Display Mode** - The available display modes are *Vector Field*, *Point Cloud*, *Flat*, and *3D*.
- **Size Factor** - Used to scale the size for point clouds and vector display modes
- **Displayed Field** - Lists the available components of a vector field that can be shown in a plot.
- **Decimation** - Used to sample a subset of points for solid sensors
- **Surface Resolution** - Used to sample a subset of points for Surface sensors. This control is only available if the Surface sensor sampling method is configured as *Field Interpolation*.
- **Complex Part** - Used to single out real/imaginary/magnitude values for single-frequency data
- **Axis Ranges** - Set the independent axis during the sequence
 - **Axis** - defines the axis
 - **Full** - runs along the full range of the axis
 - **Min And Max** - subsets non-sequence axes
- **Enable Scaling** - enable the scaling factor

The Setup tab for a solid sensor configuration



Far-Zone Sensor Configuration

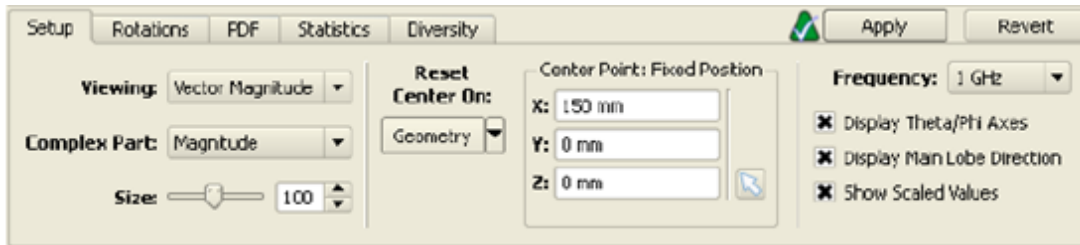
- In the first section, set up your viewing preferences.
 - **Viewing** - choose among the available result types
 - **Complex Part** - view the *Magnitude*, *Real*, or *Imaginary* part of the results.
 - **Size** - enter the size of the maximum radius.
- In the second section, choose the orientation of the *Center* point.
 - Enter the (**X**, **Y**, **Z**) coordinates of the center, or
 - Choose the center using the *Pick* tool.
 - The *Reset Center On* drop-down enables you to center the plot at the center of

the

geometry or at the location of any of the ports in your active project.

- Select the following viewing options:
 - *Display Axes Theta/Phi*
 - *Display Main Lobe Direction* (the direction of maximum gain)
 - *Show Scaled Values*

The Setup tab for a far-zone sensor configuration

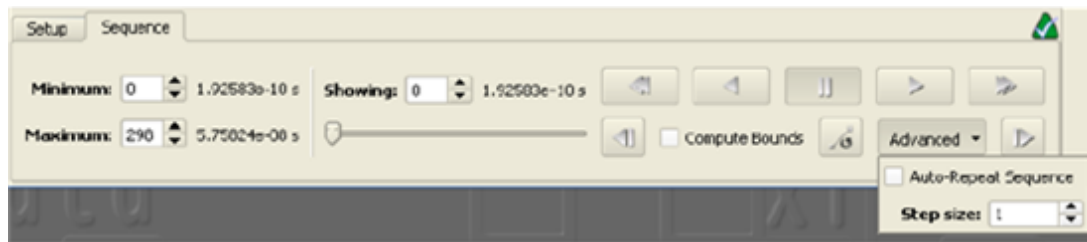


Using the Sequence Tab

This tab is available when viewing an SAR, Solid, or Volume sensor. Configure the settings here to "play through" a simulation.

- **Minimum/Maximum** - values entered here set limits on the range of the sequence axis.
- **Showing** - shows the currently plotted value of the sequence axis.
- The scroll bar at the bottom of the window can be used to view any point along the sequence.
- The simulation buttons enable you to move through the sequence:
 - The *forward* and *back* buttons play the sequence forwards and backwards, respectively.
 - The *pause* button pauses a playing sequence.
 - The *fast forward* and *rewind* buttons fast forward and rewind the sequence, respectively.
 - The *jump forward* and *jump backward* buttons jump to the end or beginning of the sequence, respectively.
- **Compute Bounds** - when this box is checked, EMPro will compute the bounds during the sequence.
- - performs the **Compute Bounds** calculation
- The **Advanced** tab
 - Checking the **Auto-Repeat Sequence** box will automatically replay the sequence continuously.
 - The **Step Size** determines the number of indices to change for the simulation buttons.

The Sequence tab of the field editing toolbar



Using the Rotations Tab

The Rotations tab, shown in Figure 13.20, offers several operations to adjust the orientation of the far-zone results by setting the Up Vector of the view. Changing the Up Vector (which is the Z-axis by default), will set the reference point for the spherical coordinate systems (e.g., theta/phi, alpha/epsilon, etc.) This affects field plots of single polarization components, partial power efficiency and power computations, mean effective gain computations, and antenna diversity. This is useful in cases where the geometry is aligned with an axis of the computation grid, but the real-world position of the geometry is not axis-aligned.

- The *Up Vector Presets* list enables you to select a pre-defined *Up Vector* in the **X**-, **Y**-, or **Z** -direction.
- The *Pick Up Vector* tool enables you to click and drag within the simulation space to create your own *Up Vector*.
- The *Undo* button will undo an operation.
- The *Redo* button will redo an operation.
- The *Reset* button will reset the *Up Vector*.

The Rotations tab of the field editing toolbar



Using the PDF Tab

The PDF tab controls the settings used in the computation of mean effective gain and antenna diversity far zone results. The following illustration shows the PDF tab.

- **XPD** is cross-polarization discrimination of the incident multipath field. It can be expressed by the ratio of time-averaged vertical power to time-averaged horizontal power in the fading environment. Users can input a linear value or a decibel value.
- **PDF**, or Probability Density Function, is used to model communication channels. This function "**weights**" different directions in the far zone sphere such that certain far zone direction are taken into account more than others in the mean effective gain and antenna diversity calculations. It has a maximum value of one and a minimum value of zero. EMPro provides the following choices for PDF's:

- **Uniform Value:** in this case, PDF is simply one.
- **Gaussian:** this is most commonly used as a model of channels. *Theta Max* and *Phi Max* are the angles of the maximum incoming field. *Sigma Phi* and *Sigma Theta* are the standard deviations for the ϕ and θ directions, respectively.
- **User-Defined:** this enables you to import a PDF from a text file. The file must contain a grid of theta/phi angles and the corresponding probability density at each angle. For example, to define the PDF at ten theta angles with twenty phi angles, the text file data would look something like the following:

```
theta0radians phi0radians pdfAtTheta0Phi0
theta0radians phi1radians pdfAtTheta0Phi1
...
theta0radians phi19radians pdfAtTheta0Phi19
theta1radians phi0radians pdfAtTheta1Phi0
...
theta9radians phi19radians pdfAtTheta9Phi19
```

The PDF tab in Gaussian mode



Using the Statistics Tab

There are two main configurations for the Statistics tab, depending on the type of sensor you are evaluating.

Note
Keep in mind that not all fields listed will be available for every view.

Far-Zone Sensor Statistics

The next figure shows the *Statistics* tab for a far-zone sensor configuration.

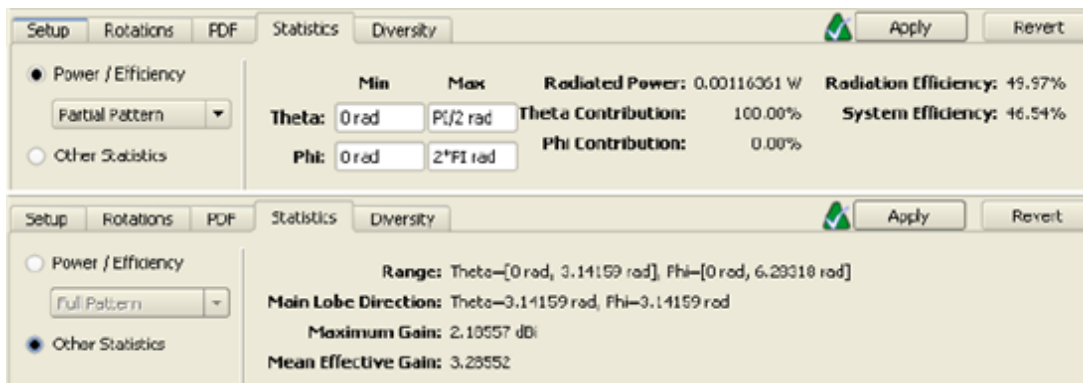
If you select Power/Efficiency, power and efficiency computed for the far zone data (with rotations applied) is displayed. The following options determine the set of angles that are considered for this data:

- Full Pattern - specifies the use of a full far zone sphere.
- Upper Hemisphere - specifies the use of $\theta = [0, 90^\circ]$ and $\phi = [0, 360^\circ]$.
- Open Sky - specifies the use of $\theta = [0, 80^\circ]$ and $\phi = [0, 360^\circ]$.
- Partial Pattern - enables you to specify an arbitrary solid angle. These angles are defined relative to the coordinate system defined under *Rotations*.

Each pattern will display its statistics to the right. In the case of *Partial Pattern*, the user can redefine the min and max values of *Phi* and *Theta* by typing them in to the corresponding boxes. The statistics will update accordingly.

The *Other Statistics* option presents a list of additional statistics.

The Statistics tab for a Far-Zone sensor, showing Power/Efficiency and Other



statistics

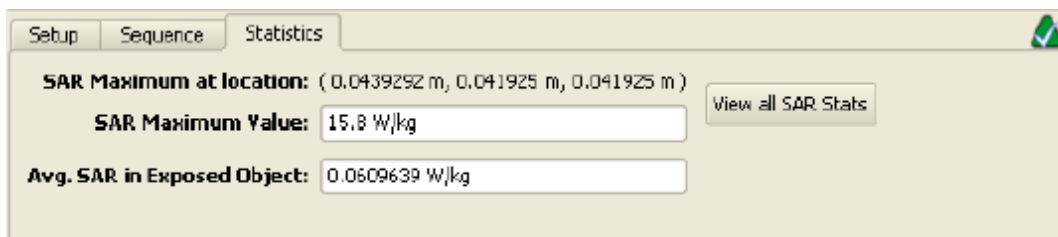
SAR Sensor Statistics

The next illustration shows the *Statistics* tab for an SAR sensor configuration. This tab is available when looking at SAR data for both the *SAR Sensor* and the *SAR Averaging Sensor*. The following figure displays the dialog that appears when the *View All SAR Stats* button is pressed.

- The *SAR Maximum Value* displays a max SAR value dependent on the properties of the SAR Sensor, (i.e., the maximum 1-gram Averaged SAR value, the maximum 10-gram Averaged SAR value, or the maximum raw SAR value). A value entered here affects the power scaling for this run and frequency, in the same manner as adjusting the *Power Scaling Factor* in the *System Sensor Output* dialog.
- *SAR Maximum at Location* displays the global coordinates of the location where the SAR maximum value is found.
- The *Avg. SAR in Exposed Object* field is available when looking at raw SAR sensor data only. It displays the whole body average SAR value. An adjustment entered here affects the power scaling in the same way as adjusting the *SAR Maximum Value*.
- The *View Slice With Max Value* button takes you to the planar slice of SAR data which contains the maximum value.
- The *View All SAR Stats* button opens a separate window displaying the statistics for the Raw SAR, 1g Averaged SAR, and 10g Averaged SAR together in a table. Through this window, the scaling factor can be adjusted, which affects all SAR values in this window, including data displayed in the SAR Sensor Statistics tab. The scaling factor and SAR statistics values will also automatically update when the user manually changes (rescales) any other statistics available through the *Results* workspace window, such as power and efficiency values.

Note
For more information on rescaling values in the *System Sensor Output* dialog, refer to [Rescaling](#).

The Statistics tab for an SAR sensor



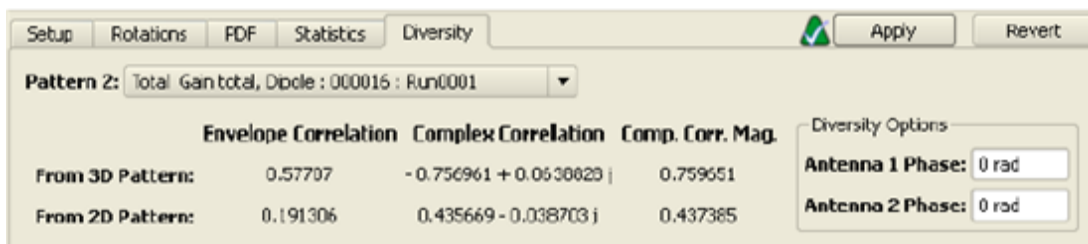
Using the Diversity Tab

The *Diversity* tab is used to compute antenna diversity metrics between two far zone patterns. To perform the diversity computations, you must load the data for both far zone patterns in the *Results* workspace. Then go to the diversity tab for one of the two patterns. The following figure shows the *Diversity* tab.

- The *Pattern 2* drop-down list enables you to select the far zone result which your current far zone result is compared against. The only choices listed here are results which have been loaded into the *Results* workspace window.
- The statistics next to *From 3D Pattern* show the diversity computation for the full far zone sphere. The statistics next to *From 2D Pattern* show the diversity computation at $\theta = 90^\circ$.
- Under *Diversity Options*, you can set the *Antenna 1 Phase* and *Antenna 2 Phase* controls to specify the phase difference between the two far zone patterns.

Note
Note that the *XPD* and *PDF* settings under the *PDF* tab impact this computation, as well as the settings under the *Rotations* tab.

The Diversity tab of the Field editing toolbar



Using the Hearing Aid Compatibility Tab

This tab, as the name suggests, is only available when viewing (HAC) sensors.

- **Frequency** - Select the frequency of interest for the hearing aid.
- **Display Mode** - The available display modes are *Point Cloud* and *Flat*.
- **Show Scaled Values** - When checked, the displayed results are scaled by the scaling factor entered by the user on the system sensor output table.
- *Surface Resolution* - This option appears whenever there is a HAC or interpolated

surface sensor. It controls the resolution at which field data is shown for the sensor. The higher the resolution setting, the more data points are shown on screen (and the closer they are together).

- *M-Rating* - Displays the M-rating of the sensor, or the suitability of the hearing aid used with a wireless device. The rating range is 1-4, with 4 being the best rating.
- *Threshold* - This is the maximum allowed field value given the current M-rating.
- *Band* - Displays the band type. Under the *Edit* button, you can configure the properties of the band, or define a pre-set list of properties using the *Manage Presets* button. An editor dialog will appear where you can add a new preset or import presets from a text file. Each line in the file must follow this format:
 - Lines in the file starting with **#** or **!** are comments and are ignored.
 - Otherwise, the line must contain six semicolon-delimited items in this order: name; waveform modulation factor; E-field probe modulation factor; H-field probe modulation factor; articulation weighting factor, in dB (must be either 0 or -5); and a boolean value which determines whether RMS conversion should be used (1 to enable, 0 to disable).

An example of this format is as follows:

```
! This is a comment
Band 1 Name;0.85;0.9;0.8;-5; 0
Band 2 Name; 1.0; 0.92; 0.33; 0; 1
```

- The 3x3 Grid Region - This shows the maximum field values in each of the 9 squares of the HAC grid. The double-line on the top-level of the readout helps you orient in reference to the white grid on the displayed field (it also has a double-line in the top-left corner). Three of the cells in the 3x3 grid will have a dotted red border around them. These are the three cells chosen to be excluded from the HAC rating computation as allowed by the IEEE HAC standard. Also, any cells with red text are cells which exceed the field value threshold amount for the current M-rating.

Post-Processing

Post-processing gives the user the option of running additional computations after the calculation engine run is finished to save certain results to disk. Two types of post-processing are available.

Far Zone Post-Processing

Far-zone post-processing is enabled by checking the *Save Data for Post-simulation Far Zone Steady-state Processing* box under the *Frequencies of Interest: Data Storage* tab of the *Simulations* workspace window. When this configuration is set, there will be an entry for *Raw Steady-State Far Zone Data* in the *Results* workspace window. Double-clicking this entry will enable you define a new far zone geometry at which the steady-state far zone pattern will be computed. Edit the geometry exactly as you would edit a far zone sensor. When you click **Done** in the far zone sensor editor, the computation of the far zone pattern will begin. Once complete, you can find a new result in the *Results* workspace window which corresponds to the newly defined far zone pattern.

SAR Post-Processing

Right-clicking on any Raw SAR Sensor entry in the *Results* workspace window will show an option in the context menu to *Postprocess Results*. This enables you to perform SAR averaging. Edit the geometry and averaging parameters exactly as you would in the *SAR Averaging Sensor* editor. Once you click **Done** in the editor, you will be given the option to automatically run the post-processor. If you say yes, the post-processing operation is queued as if you had created a new simulation. Otherwise, you will have to do the post-processing run manually. Once the SAR averaging is complete, new SAR average data will appear in the *Results* workspace window.

Viewing FDTD Default Output

You can view the results of an FDTD simulation in the Results workspace window. The results that are available in this window depend on the characteristics of the project such as discrete sources, sensors, external excitations, and other project criteria specified in the Simulation workspace window. The results are displayed in the following ways:

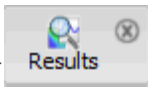
- Numerical values: Some results are displayed in the form of numerical values for single-frequency simulations with a near-zone source.
- Plots: You can create various types of plots to view results based on time, frequency, or angle.
- Colored field displays: Some results are available as Broadband results, which are collected by Surface and Solid sensors. These results are viewed as individual **Field Snapshots** or **Field Sequences**. Three-dimensional far-zone fields may also be available for view depending on simulation criteria.

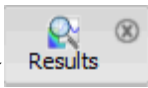
This section provides information about how to:

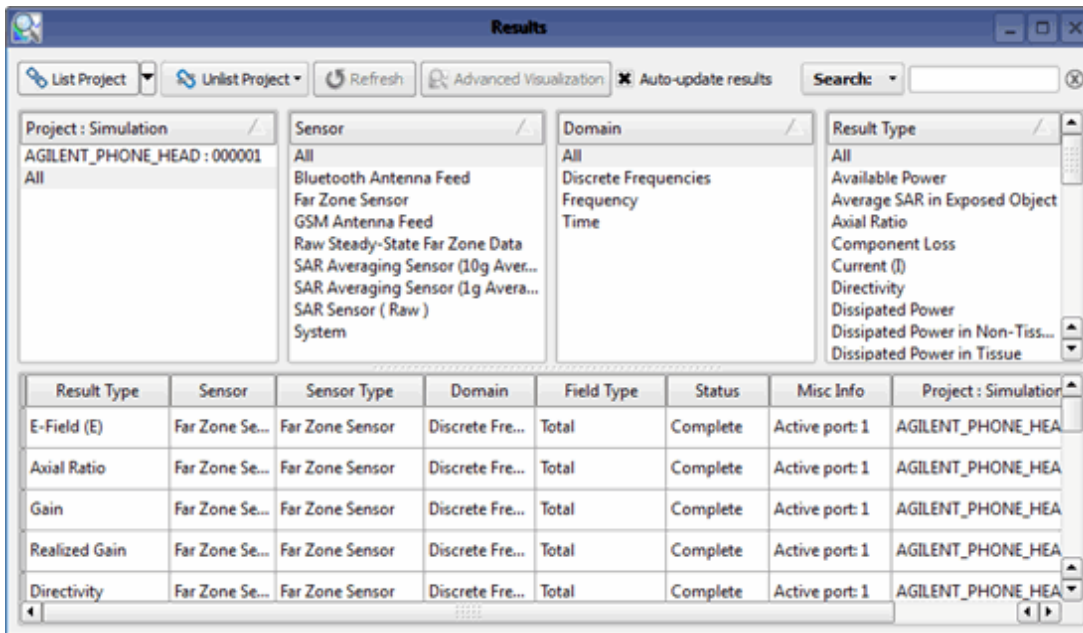
- Read and filter calculation results.
- Use tools to analyze 2-D (plotted) and 3-D (field display) results.

Using the Results Workspace Window

Once a simulation is queued for calculation, the project is listed in the Results window. This window stores all the data collected by the sensors during a calculation. To view results:



1. Click **Results** () to open the workspace.
2. Select a project from the **List Project** drop-down list. The results are displayed, as shown in the following figure:



3. Click **Refresh** to update the list of results in the workspace window.



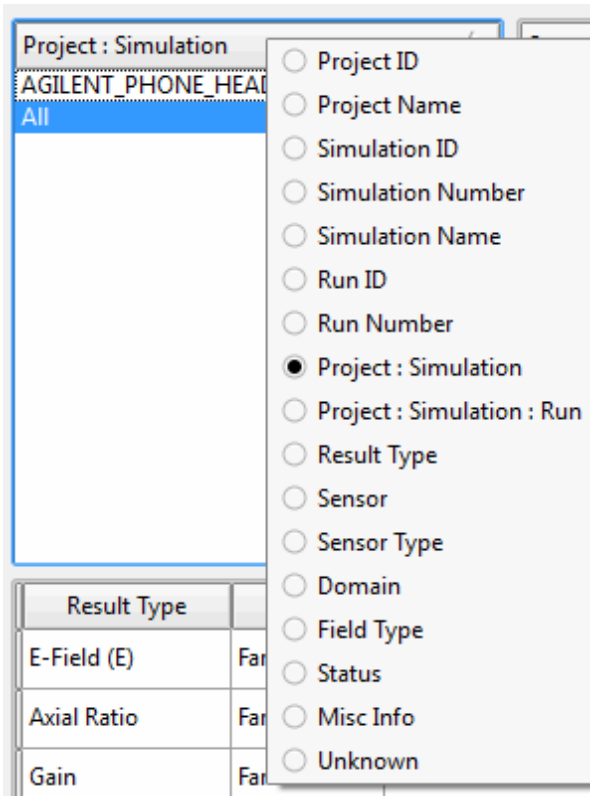
Note

To automatically update results, select the **Auto-Update Results** check box. In addition, this option automatically updates plotted data when a specific plot window is opened as the calculation progresses.

You can also load the results of any other saved project by clicking **List Project**. This enables you to view and compare the results of multiple projects without loading them individually. Similarly, you can close the results of any project by selecting the project and clicking **Unlist Project**.

Filtering Data Results

You can filter the data in the Results window by searching or categorizing the required data. To find data, type the field name or results data in the **Search** text box. You also customize the four columns at the top of the Results window, which filter data according to the specified categories. To select the required heading, right-click any existing column heading and select one of the available categories, as shown in the following figure:



The following table describes each category:

Data Filter	Description
Project Id	Displays the EMPro project ID, which references the location of the loaded project in the file directory.
Project Name	Displays the project name indicated by the user.
Simulation Id	Displays the simulation ID that refers to the simulation's location in the file directory.
Simulation Number	Displays the simulation number that is automatically generated based on how many simulations have been created in a specific project.
Simulation Name	Displays the simulation name specified in the Simulations workspace window.
Run Id	Displays the run ID which references the location of the run in the file directory. Multiple runs are created in the case that a simulation collects data for more than one variable or location (i.e., during parameter sweeps, multiple ports, etc.).
Run Number	Displays the run number.
Result Type	Filters results by type. Any of the following results may be viewed depending on the project, simulation criteria, and the type of data that was requested: <ul style="list-style-type: none"> Electric Field (E) Magnetic Field (H) Conduction Current (Jc) Magnetic Flux Density (B) Poynting Vector (S) Voltage (V) Current (I) Impedance Available Power Input Power Instantaneous Power

FDTD Simulation

	<p>Feed Loss Reflection Coefficient S-Parameters SAR (Specific Absorption Rate) Maximum SAR Value Average SAR in Exposed Object Average Power Net Input Power Net Feed Loss Net Available Power System Efficiency Radiation Efficiency Dissipated Power Dissipated Power in Tissue Dissipated Power in Non-Tissue Dissipated Power per Material Radiated Power Dissipated Power per Electric Material Component Dissipated Power per Magnetic Material Component Dimension (time, position, etc.) Axial Ratio Radar Cross Section Gain Voltage Standing Wave Ratio (VSWR)</p>
OUTPUT OBJECT	Filters results according to its <i>Output Object</i> , which refers to a specific sensor by name.
DATA TYPE	<p>Filters results according to the type of sensor that collected the data. The sensor is referenced by its general type, rather than by its user-defined name (Refer to <i>Output Objects</i> for filtering data by sensor name). The following is a comprehensive list of the sensors that may be listed within this filter:</p> <p>Point Sensor - Retrieves data from any <i>Point</i> sensor in the project. Surface Sensor - Retrieves data from any <i>Surface</i> sensor in the project. Solid Sensor - Retrieves data from any <i>Solid</i> sensor in the project. Far Zone Sensor - Retrieves data from any FAR ZONE sensor in the project.</p> <p>Raw SAR Sensor - Retrieves raw SAR data. Averaged SAR Sensor - Retrieves averaged SAR data. Circuit Component - Retrieves data from a <i>Circuit Component</i> HAC Sensor - Retrieves hearing aid compatibility data from HAC sensors. System - Retrieves ambient result data (not associated with a sensor object). External Excitation - Retrieves data on the <i>External Excitation</i> waveform. Raw Steady-State Far-Zone Data - Contains information which can be used to generate new steady-state far zone patterns after a simulation is run.</p>
Project:Simulation Run	Displays the Project, Simulation, and Run name in one column.
Domain	Filters data according to Time, Frequency or Discrete Frequencies domains.
Field Type	Filters data according to total-field or scattered-field.
Status\	Displays whether a result is complete or still being calculated while the simulation is running. This status can be refreshed manually by pressing the <i>Refresh</i> button or automatically by selecting <i>Auto-Update Results</i> .
Misc	Displays query-specific information. For example, for the circuit component voltage when collecting S-Parameters, it could contain the active port. For S-parameters, it could

contain the **S** number or simply the active port. For System numbers, it could contain the number (such as efficiency) or net input power.

Viewing Numerical Results

When a single-frequency calculation has been performed with a near-zone source (voltage or current), parameters such as input impedance, S-parameters, VSWR and the reflection coefficient are displayed as numerical values rather than as line plots since the data only is relevant for the input frequency. Other numerical values are collected by means of a system sensor, which is automatically present in every EMPro project.

Numerical Data Collected by EMPro

Results Displayed as Numerical Data

Sensor Type	Time Dependence	Result Type
Port Sensor	Single-Frequency	Available Power
		Current
		Impedance
		Input Power
		Reflection Coefficient
		S-Parameters
		Voltage
		VSWR
System Sensor	Single-Frequency	Dissipated Power
		Dissipated Power in Non-Tissue
		Dissipated Power in Tissue
		Dissipated Power Per Electric Material Component
		Dissipated Power Per Magnetic Material Component
		Dissipated Power Per Material
		Net Available Power
		Net Feed Loss
		Net Input Power
		Radiated Power
		Radiation Efficiency
		System Efficiency

Impedance

All input impedances are calculated by the ratio of the complex V over complex I for the FDTD mesh edge at the port location. The sign convention is positive for power flow into the antenna or other structure. Thus for a port which delivers power to the antenna, the impedance will have a positive real part, and for a port which absorbs power from the antenna, the impedance will have a negative real part.

When only one active port is present, the feed point impedance at that port is the self-impedance at that port. If more than one port is active, the port impedance values listed represent the ratio of the complex voltage and current at each port including the effects of

the sources at all active ports. Thus these impedance values are not the self-impedances at each port, but rather terms of an impedance matrix.

S-Parameter Calculations

When an S-parameter calculation is made with the steady-state far-zone transformation selected, then both the steady-state antenna data and the S-parameter data will be displayed. This may be useful, for example, when making calculations for a microstrip antenna when both S_{11} and input impedance are of interest.

Net Available Power

The available power is calculated as the total power delivered by the active port or ports into a matched load.

Radiated Power

The radiated power is computed as the difference between the total net input power delivered by the active ports and the dissipative losses from conductive materials and resistive loads in inactive ports.

System Efficiency

The system efficiency is calculated as the ratio of radiated power to available power. Thus, it includes radiation efficiency and mismatch efficiency.

Power Scaling Factor

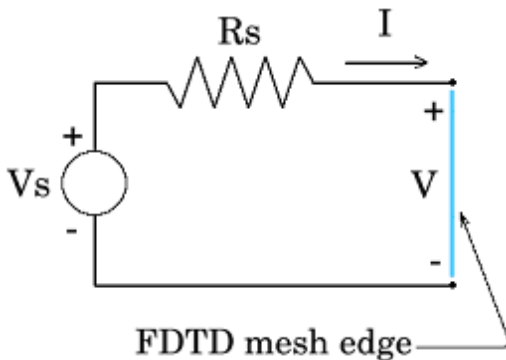
When viewing single-frequency results collected by the system sensor, the user can specify a scaling factor of the available power to the calculation and determine the scaled result. The absolute input power can be adjusted as desired or an overall scaling can be applied. For example, if the user wanted to know what the output power would have been with an input of one mW, they can enter this value in the available power (will be subject to mismatch loss) or in the net power (after mismatch loss) to see the overall effect. Similarly, any of the output results may be scaled to determine the effect on input and output. Clicking reset will return the values to the un-scaled state.

Input power, Input Impedance, and Loss

When making calculations that include the input voltage, current and/or power in the calculation formulas, such as antenna gain or input impedance, the input voltage, current and/or power will be that provided at the terminals of the mesh edge. Referring to the following figure, the impedance at the port would be the (complex) mesh edge voltage V divided by the (complex) mesh edge current I . The complex values would be obtained from an FFT for a broadband calculation or from two samples of the voltage and current (electric and magnetic fields at the port mesh edge) for a sine wave excitation. If the port is delivering power, then the impedance at that port will have a positive real part. If the port is absorbing power, then the impedance at that port will have a negative real part and the input power will be negative. This is determined by the direction of current flow

and voltage polarity for the FDTD mesh edge.

Feed schematic, including FDTD mesh edge voltage, V , and current, I



Active ports (those with active voltage or current sources) are treated differently than inactive ports for some calculations. For an antenna calculation, the input power to the antenna is the algebraic sum of all powers delivered by active ports. Power absorbed by active ports will not reduce the antenna efficiency; however, power absorbed by inactive ports will reduce antenna efficiency.

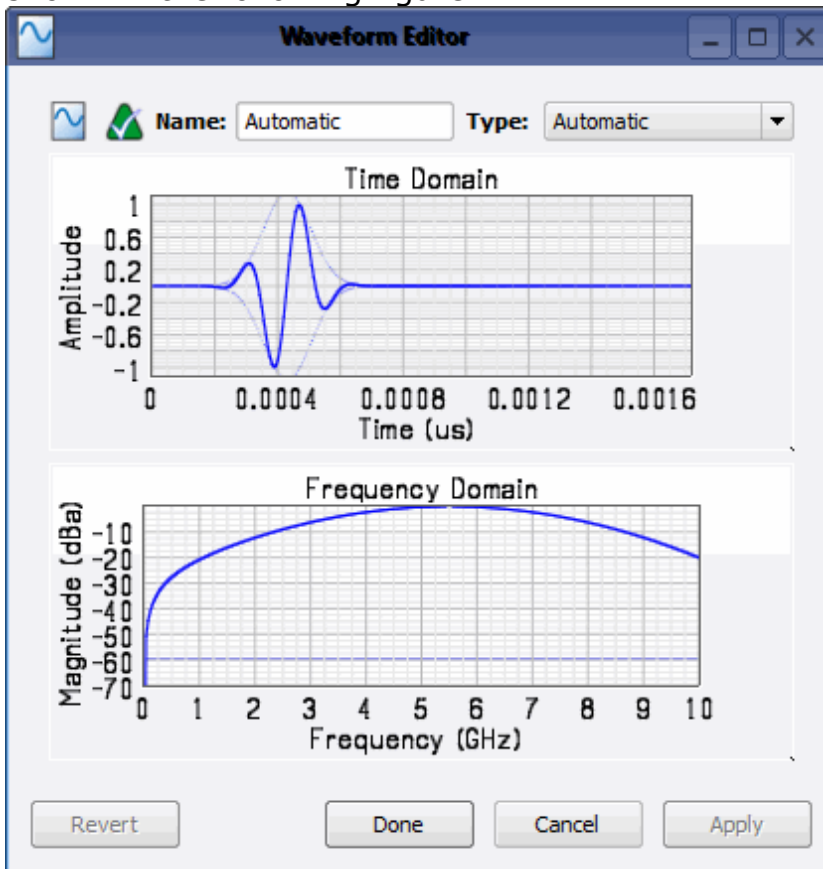
To clarify this, consider two different situations. An antenna composed entirely of perfect conductor includes 2 ports, the first containing a 1-V source and a resistor, the second a passive port containing only the resistor. Assuming that some current flows in the passive resistor, the antenna efficiency will be less than 100%. If we repeat the same calculation, but with a 0.00001-V source added to the formerly passive port, and with both sources set active, the antenna efficiency will now be calculated as 100% even though the 0.00001-V source will have negligible effect on the antenna currents and radiation. This discrimination is done so that active ports may utilize lumped circuit elements to match to an antenna without changing the antenna efficiency, impedance, and gain, while passive elements may be added to an antenna with their effects included in the antenna efficiency and gain results.

Similarly, for input impedance calculations, the source resistance, capacitance and inductance values will not be included in the input impedance. For example, the input impedance of an antenna as calculated using EMPro should not change regardless of any changes in the active port components (source/R/L/C). This is as it should be, since the antenna impedance is a function of the antenna geometry/materials and not of how the antenna is fed.

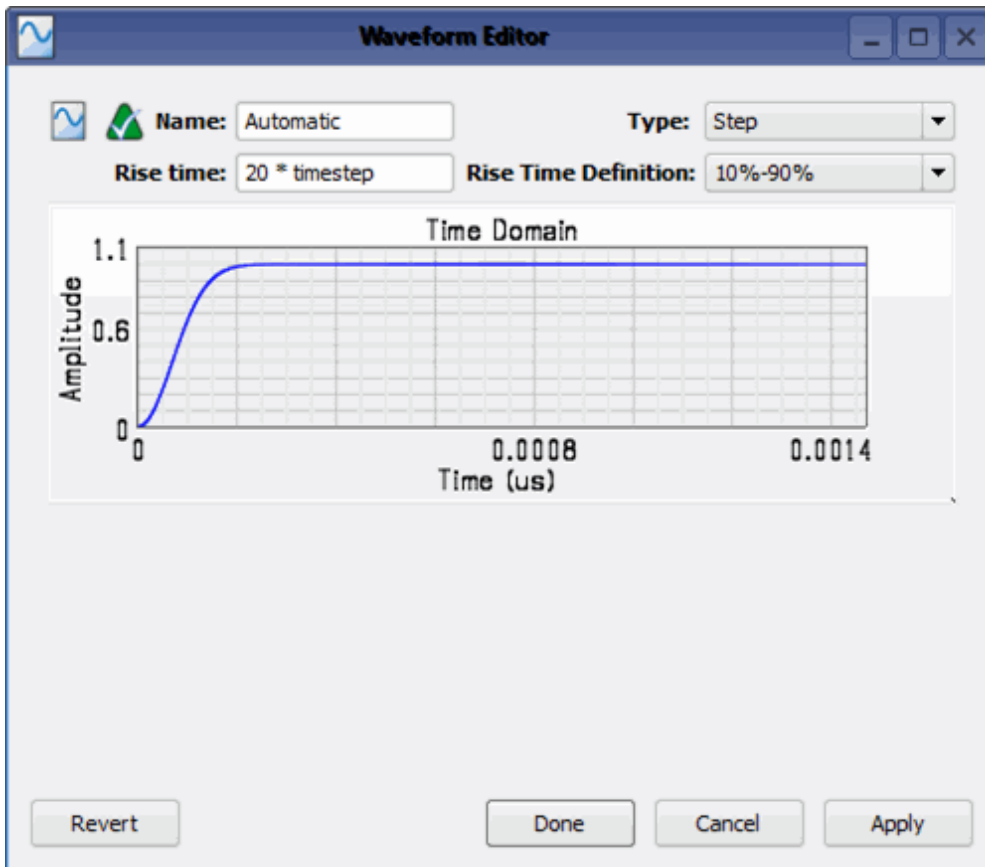
Adding TDR Waveforms

To add a step waveform for TDR simulations:

1. Right-click **Waveforms** in the **Definitions** list and select **Create New Waveform** or double-click any existing waveform. The Waveform Editor dialog box is displayed, as shown in the following figure:



2. Type a name for the waveform in the **Name** text box.
3. Select **Step** from the **Type** drop-down list.



4. Type a value in the **Rise Time** text box. This value specifies the rise time of incident TDR pulse. The resolution of a TDR measurement system is directly related to the rise/fall time of the incident pulse.
5. Select a value from the **Rise Time Definition** drop-down list.
6. Click **Done**.

Viewing TDR Waveform Results

You can view the following types of data plots for TDR waveforms:

- Current (I)
- Impedance
- Instantaneous Impedance (TDR)
- Instantaneous Power

The Instantaneous Impedance (TDR) result is specific to the waveform that is used to analyze Instantaneous impedance. The following figure displays the TDR waveform results:

FDTD Simulation

Results

List Project Unlist Project Refresh Advanced Visualization Auto-update results Search

Project : Simulation	Sensor	Domain	Result Type
All	All	All	All
SATA_Connector : 000001	Component	Frequency	Current (I)
SATA_Connector : 000002	System	Time	Impedance
TDR_SATA_Connector : 000002			Input Power
			Instantaneous Impedance (TDR)
			Instantaneous Power
			Net Input Power
			Reflection Coefficient
			S-Parameters
			Voltage (V)
			VSWR

Result Type	Sensor	Sensor Type	Domain	Field Type	Status	Misc Info	Project : Simulation
Instantaneous Impedance (TDR)	Component	Circuit Component	Time	N/A	Partial	Active port: 1	TDR_SATA_Connector : 000002

GPU Acceleration for FDTD Simulations

You can accelerate FDTD simulations by using GPU hardware. The following types of GPU acceleration solutions are supported:

- CUDA based acceleration
- Acceleware based GPU acceleration

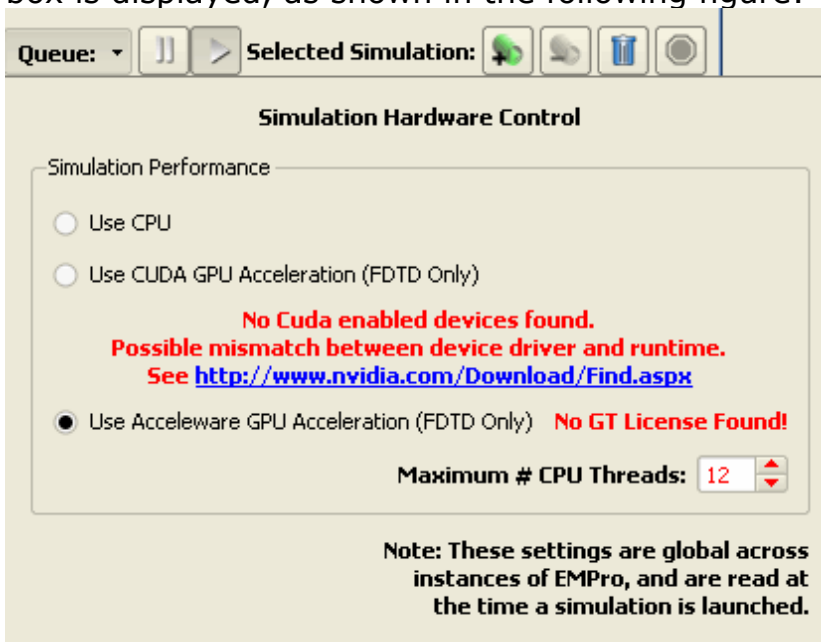
For using these acceleration solutions, you need to install GPU hardware and drivers.

Note
GPU acceleration is also supported by FDTD simulations that are using the *conformal meshing* (fddd) feature.


Selecting GPU Acceleration

To enable GPU acceleration:

1. Click **Simulations** to open the Simulations workspace.
2. Select **Queue > Use GPU Acceleration**. The Simulation Hardware Control dialog box is displayed, as shown in the following figure:



3. Select one of the following GPU accelerator:
 - Use CUDA GPU Acceleration (FDTD only)
 - Use Acceleware GPU Acceleration (FDTD only)

To define the acceleration options, click  option and choose **CUDA Acceleration** or **Acceleware Acceleration**, as appropriate.

Note
Only free space, perfect conductor and normal materials are supported for GPU acceleration. In case a non-supported material is used in the design, the complete simulation will be run on CPU and a warning is given in the simulation output window.

The following features are not supported on GPU:

- Static voltage points
- Periodic boundaries with scan angle
- Scattered field plane wave with Gaussian beam excitations

These simulations will run on CPU with a warning, which is displayed in the simulation output window.

Supported Cards

The table below lists the cards supported by EMPro for GPU acceleration.

Operating System	Acceleware for Windows	Acceleware for Linux	CUDA for Windows	CUDA for Linux
Quadro 4500X2	No	No	No	No
Quadro 5600	Yes	Yes	Yes	Yes
Quadro Plex IV	Yes	Yes	Yes	Yes
Tesla S870	No	Yes	Yes	Yes
Quadro 5800	Yes	Yes	Yes	Yes
Tesla C1060	No	Yes	Yes	Yes
Quadro Plex D2	Yes	Yes	Yes	Yes
Tesla S1070	No	Yes	Yes	Yes
Tesla C2050	Yes	Yes	Yes	Yes
Tesla C2070	Yes	Yes	Yes	Yes

Supported Drivers

The table below lists the driver supported by EMPro for GPU acceleration.

Operating System	Acceleware for Windows	Acceleware for Linux	CUDA for Windows	CUDA for Linux
Quadro 5600	160.02	100.14.19	★	★
Quadro Plex IV	160.02	100.14.20	★	★
Tesla S870	X	100.14.21	★	★
Quadro 5800	181.20	180.22	★	★
Tesla C1060	X	180.22	★	★
Quadro Plex D2	181.20	180.22	★	★
Tesla S1070	X	180.22	★	★

★ : Use latest available drivers.

The drivers and installation instructions are available on the [nVidia website](#) .

Example: How to Simulate using FDTD

You can use the FDTD solver to simulate various types of components such as Low Pass Filter, Microstrip Patch Antenna, Monopole Antenna on a Conducting Box, and SAR Calculations.

This section provides information about the following topics:

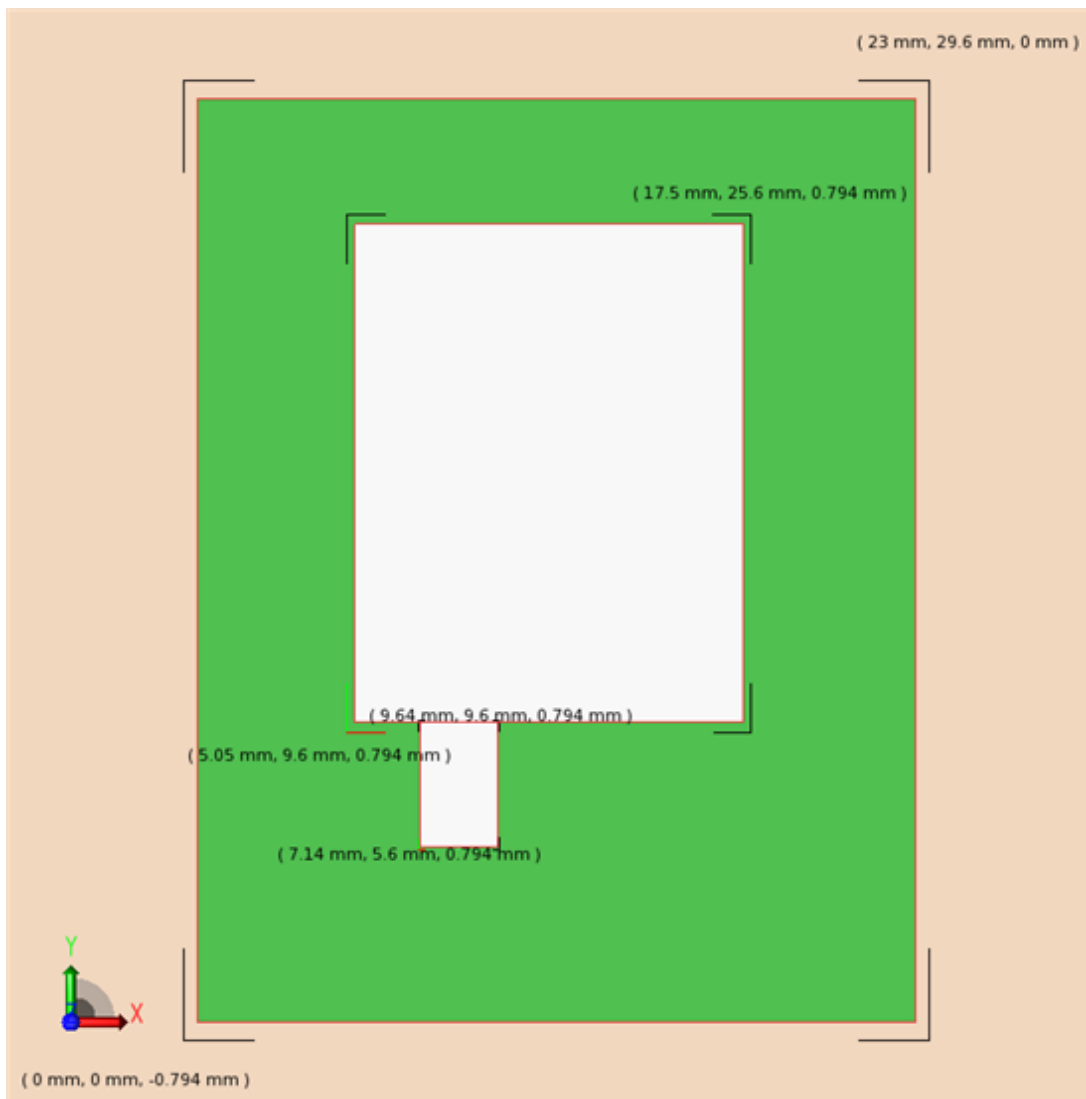
- *Creating a Low Pass Filter Simulation (fDTD)*
- *Creating a Microstrip Patch Antenna Simulation (fDTD)*
- *Creating a Monopole Antenna on a Conducting Box Simulation (fDTD)*
- *Creating a Simple SAR Calculation Simulation (fDTD)*
- *Validating SAR Calculations (fDTD)*

Creating a Microstrip Patch Antenna Simulation

In this example, a Microstrip Patch Antenna design is created by performing the following steps:

1. Set project preferences and display units to initialize an FDTD project.
2. Create the shape and set the material properties of your patch antenna.
3. Add a feed to the antenna and simulate its effects.
4. View the plotted results of your simulation.
5. Run an additional simulation with a parameter sweep.

This Microstrip Patch Antenna example is based on a paper by Sheen et al. The patch antenna from Figure 3 of this paper is constructed and the S-parameters compared with the measured and computed return loss of Figure 5 of the paper. The substrate thickness is 0.794 mm with a relative permittivity $\epsilon_r = 2.2$, as shown in the following figure:



Getting Started

You need to set the following display units for the Patch Antenna example:

1. Select **Edit > Project Properties Editor** to open the **Project Properties Editor** window.
2. Select the **Display Units** tab, as shown in the following figure:
3. Select **SI Metric** in the **Unit Set** drop-down list.
4. Select the **Show all units** check box, and adjust the following settings:
 - Change Capacitance to **millifarads (mF)**.
 - Change Current to **milliamperes (mA)**.
 - Change Frequency to **gigahertz (GHz)**.
 - Change Length to **millimeters (mm)**.
 - Change Power to **milliwatts (mW)**.
 - Change Electrical Potential to **millivolts (mV)**.

Note
The value of Unit Set changes to **Custom** after these settings are adjusted.

5. Click **Done**.

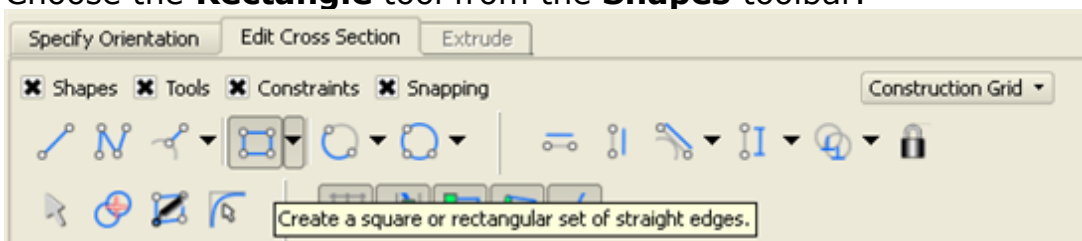
Creating the Patch Antenna Geometry

In this example, a patch antenna geometry is created from of two components: a rectangular substrate and microstrip patch.

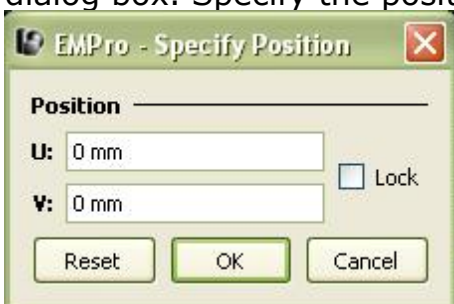
Modeling the Substrate

You need to first create a rectangular substrate named Substrate. This object stretches from (0, 0, 0) to (23, 29.6, 0) and have a .794mm extrusion in the +Z direction.

1. Right-click **Parts** and choose **Create New > Extrude**.
2. Name the substrate by typing **Substrate** in the Name box in the upper-right corner of the window.
3. Choose the **Rectangle** tool from the **Shapes** toolbar.



4. The Creation dialog box allows exact entry of coordinates. Right-click in the geometry editing space and press the **Tab** key in the geometry space to activate the Creation dialog box. Specify the position of the first point.



to **activate** the window

5. Press the **Tab** key to display the Creation dialog box for the second point. Enter

(23mm, 29.6mm) and press **OK** to complete the rectangle.

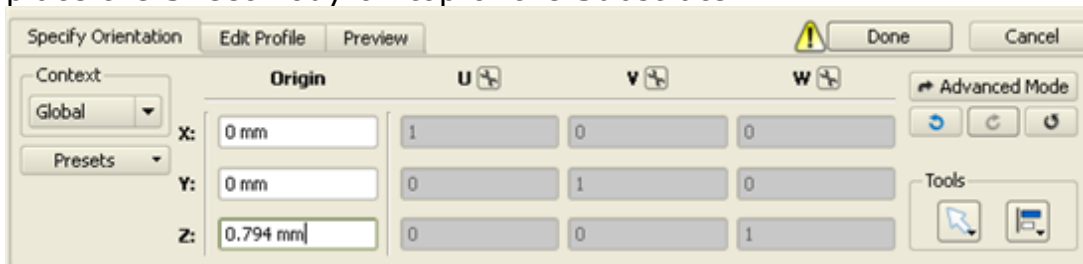


6. Navigate to the **Extrude** tab to extrude the rectangular region. Enter a distance of 0.794mm.
7. Click **Done** to finish the Substrate geometry.

Modeling the Microstrip Patch

The microstrip patch is created with a Sheet Body object that rests on top of the Substrate. This shape comprises two rectangles. The patch stretches from (5.05, 9.6, 0.794) to (17.5, 25.6, 0.794). The stub stretches from (7.2, 5.6, 0.794) to (9.6, 9.6, 0.794).

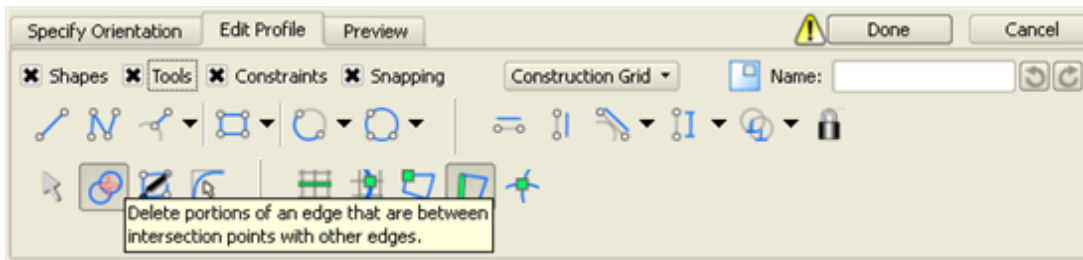
1. Right-click **Parts** and choose **Create New Sheet Body**.
2. Navigate to the **Specify Orientation** tab. Set the origin to (0, 0, 0.794;mm) to place the Sheet Body on top of the Substrate.



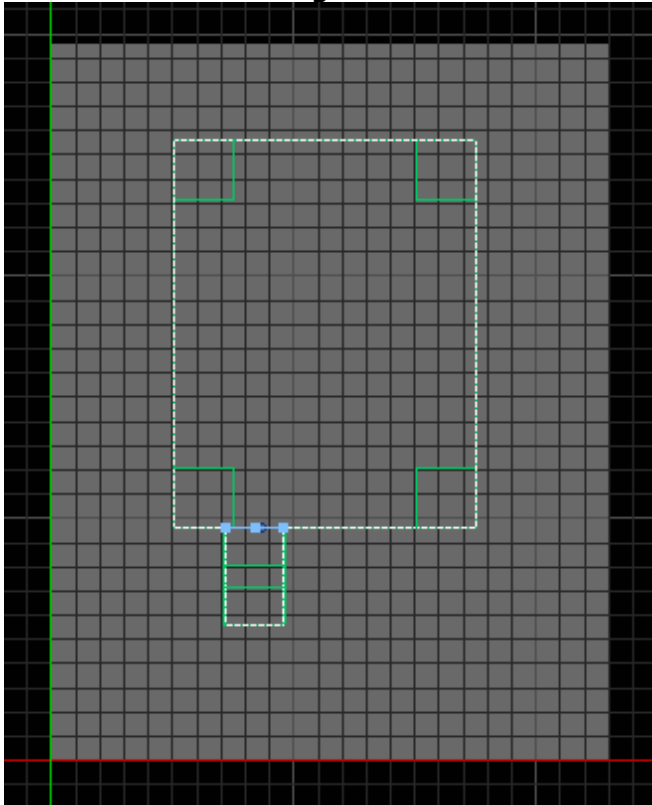
3. Click the **Edit Profile** tab.
4. Type **Microstrip** in the **Name** text box.

To draw the microstrip and its stub individually and then combine them into a single polygon:

1. Select the **Rectangle** tool. Use the *Creation* dialog box to enter the corners of the microstrip rectangle:
 - **Endpoint 1:** (5.05mm, 9.6mm)
 - **Endpoint 2:** (17.5mm, 25.6mm)
 - Now use the *Creation* dialog box to enter the corners of the stub rectangle:
 - **Endpoint 1:** (7.2mm, 5.6mm)
 - **Endpoint 2:** (9.6mm, 9.6mm)
2. Select the **Trim Curves** tool.



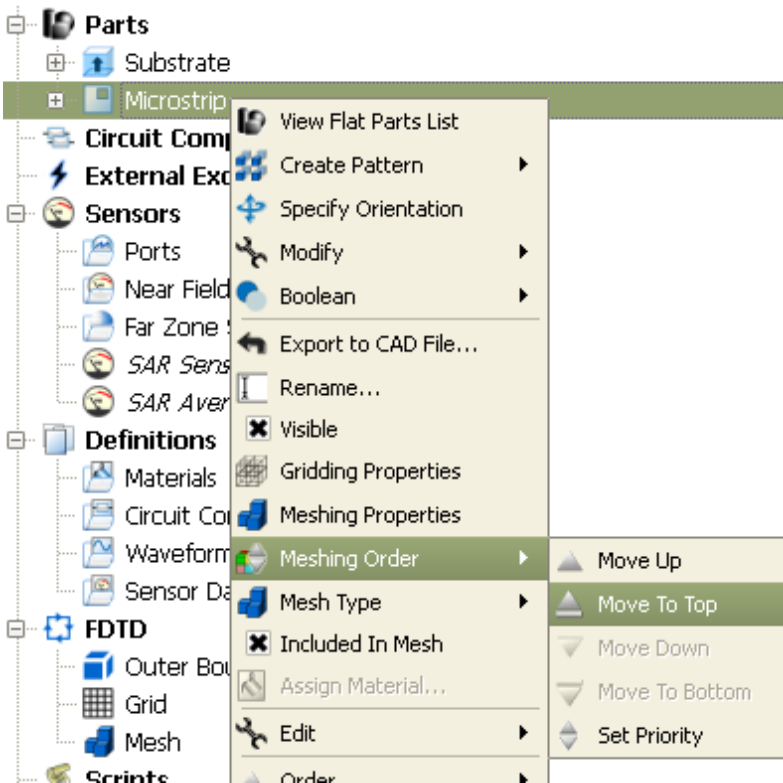
3. Remove the line segment between the microstrip and the stub by clicking on it.



4. Click **Done** to finish the Microstrip geometry.

Meshing Priority

Ensure that the meshing priority of the Microstrip is greater than the Substrate for an accurate calculation. Right-click Microstrip in the Project tree. Under **Meshing Order**, select **Move to Top** if it is an available option.

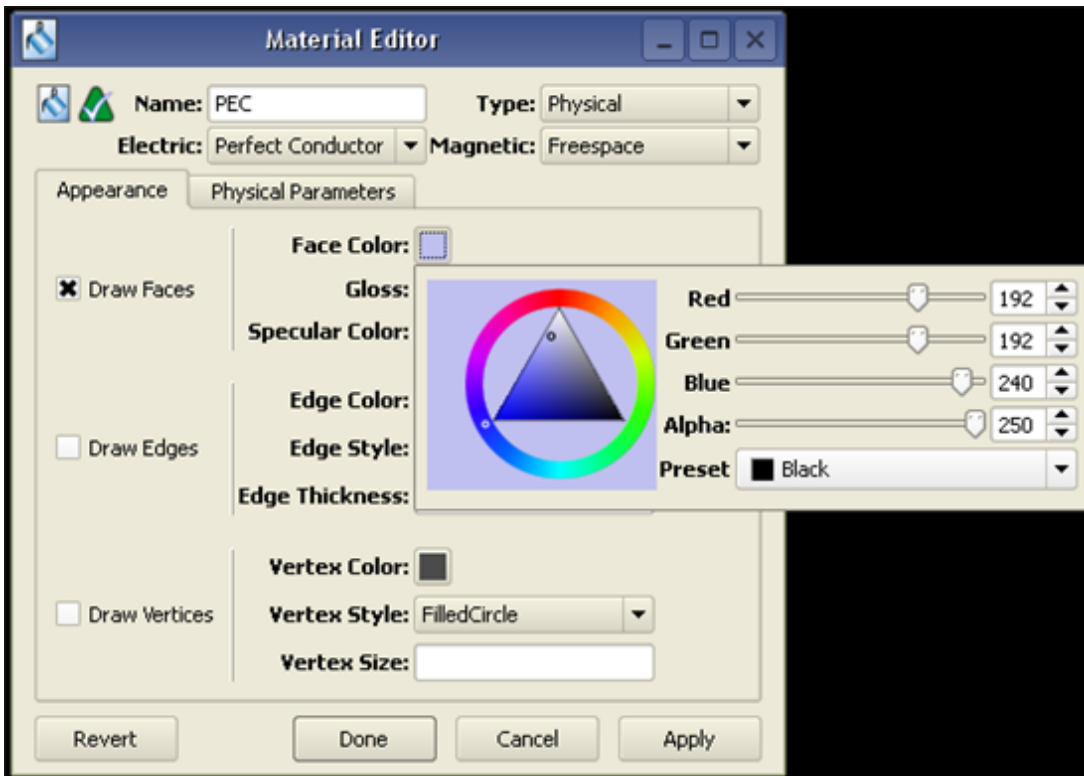


Creating Materials

Define material, PEC

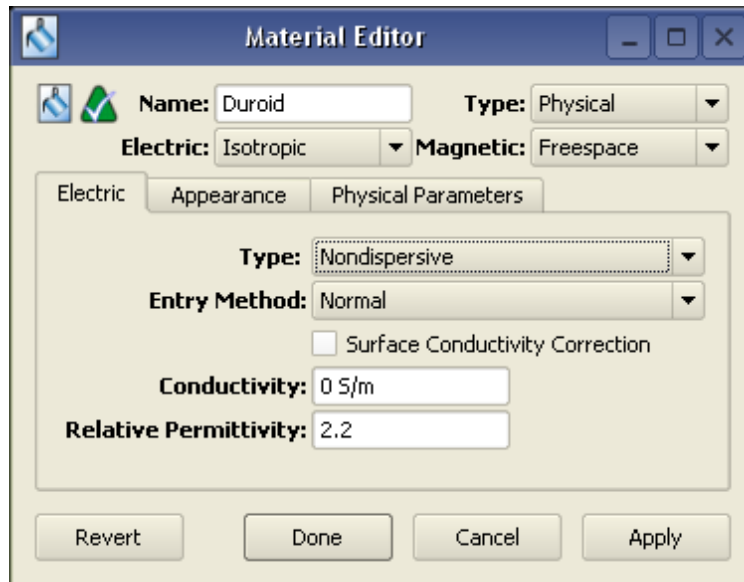
1. Create a perfect electric conductor material by right-clicking the **Materials** in the **Definitions** list and choose **New Material Definition**.
2. Double-click the new material to edit its properties.
3. Set the perfect electric conductor material properties as follows:
 - **Name:** PEC
 - **Electric:** Perfect Conductor
 - **Magnetic:** Freespace

If required, navigate to the Appearance tab to set the PEC material display color.



Define material, Duroid

1. Right-click the **Definitions:Materials** branch of the Project Tree. Choose **New Material Definition** from the context menu.
2. Double-click the new material to edit its properties. Set the duroid material properties as follows:
 - Name: Duroid
 - Electric: Isotropic
 - Magnetic: Freespace
 - Under the Electric tab:
 - Type: Nondispersive
 - Entry Method: Normal
 - Conductivity: 0 S/m
 - Relative Permittivity: 2.2

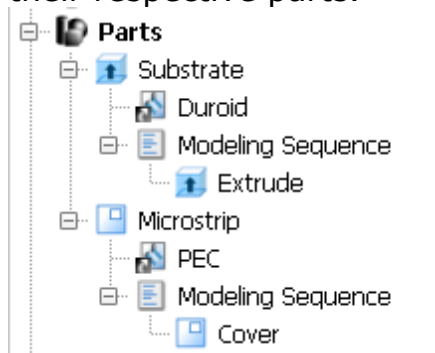


3. If desired, navigate to the Appearance tab to set the Duroid material's display color.

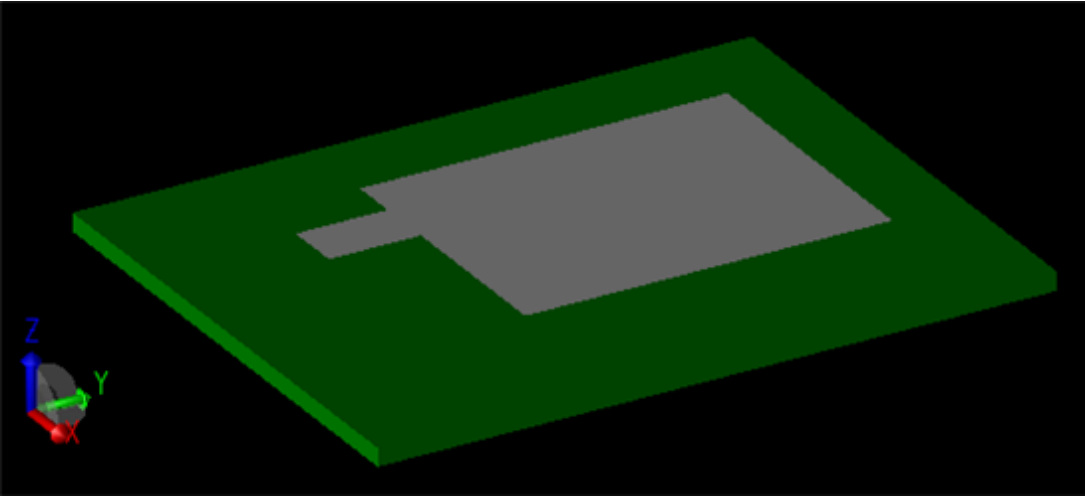
Assigning Materials

1. Click and drag the PEC material object located in the Project Tree and drop it on top of the Microstrip objects in the **Parts** branch of the tree.
2. Assign the **Duroid** material to the **Substrate** object using the same procedure.

The following image shows the Project Tree after material objects have been dropped on their respective parts.



This image shows the microstrip patch antenna geometry with materials applied and colors set for each.



Creating the Grid

Now, define characteristics of the cells in preparation to perform an accurate calculation.

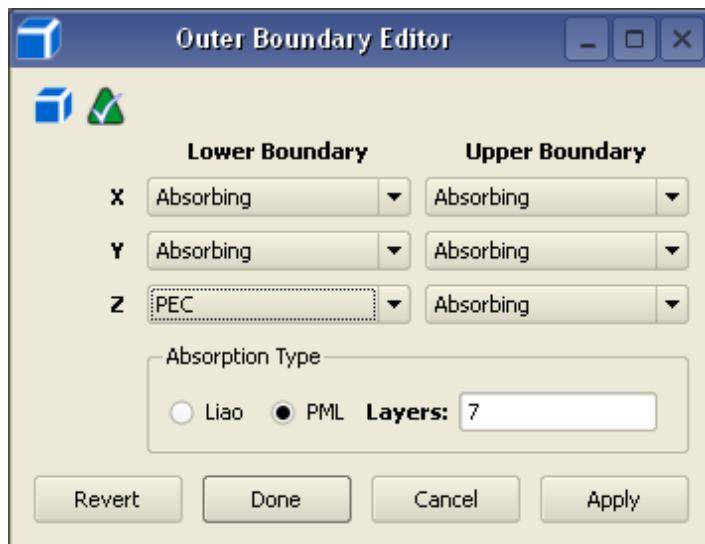
1. Double-click the **FDTD:Grid** branch of the Project Tree to open the **Grid Tools** dialog box.
2. Set the Size properties of the grid as follows:
 - Base Cell Sizes: Target 0.6 mm, Merge 0.8, Ratio boxes selected
 - Free Space Padding: 20 in all directions except Lower Z, which is 0

Size	Fixed Points	Grid Regions	Limits	Info
<input checked="" type="radio"/> Specify Padding <input type="radio"/> Specify Bounds				
Base Cell Sizes				
		Target	Merge	
X:		0.6 mm	0.8	<input checked="" type="checkbox"/> Ratio
Y:		0.6 mm	0.8	<input checked="" type="checkbox"/> Ratio
Z:		0.6 mm	0.8	<input checked="" type="checkbox"/> Ratio
Free Space Padding (base cells)				
		Lower	Upper	
X:		20	20	
Y:		20	20	
Z:		0	20	
Size Options ▾				

3. Click **Done** to apply the grid settings.

Defining the Outer Boundary

1. Double-click the **FDTD:Outer Boundary** branch of the Project Tree to open the Outer Boundary Editor.
2. Set the outer boundary properties as follows:
 - Boundary: Absorbing for all boundaries except Lower Boundary Z, which should be PEC
 - Absorption Type: PML
 - Layers: 7



3. Click **Done** to apply the outer boundary settings.

Adding a Feed

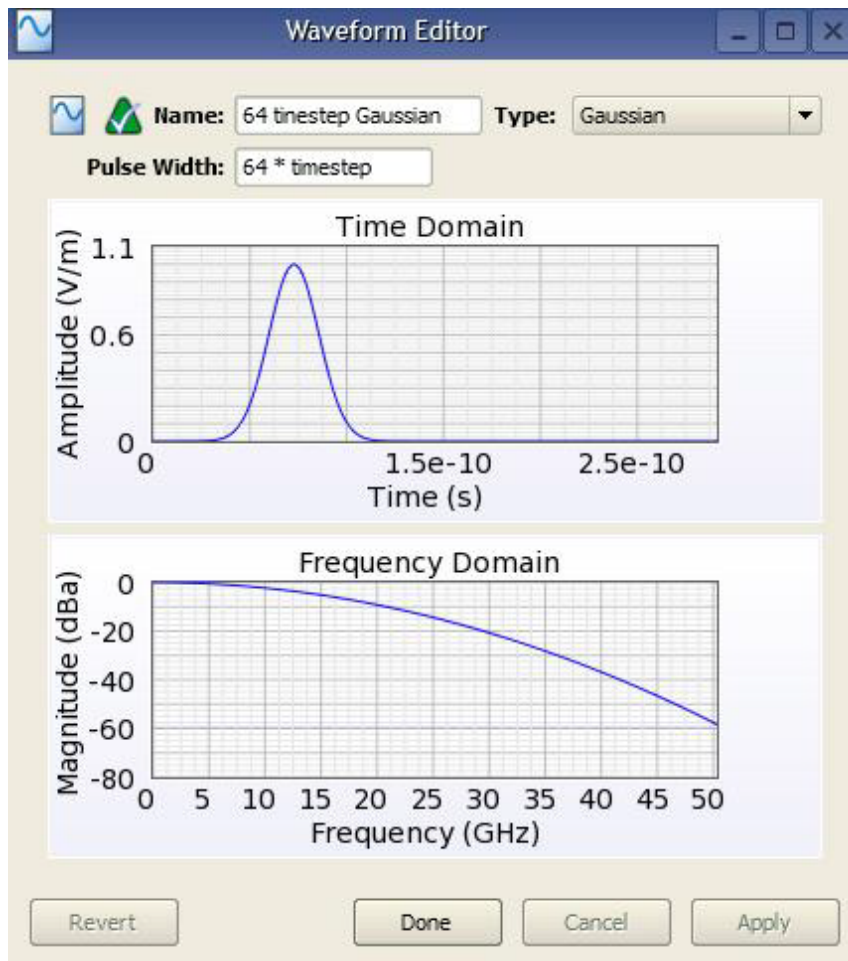
You can now add a Feed to the patch antenna geometry. It consists of a voltage source and series 50Ω resistor connected between the base of the stub portion of the Microstrip and the ground plane. Then, apply a 64-timestep Gaussian waveform to the circuit through this feed.

1. Right-click the **Circuit Components** branch in the Project Tree. Choose **New Circuit Component with > New Feed Definition** from the context menu.
2. Define the endpoints of the feed.
 - Endpoint 1: X: 8.15 mm, Y: 5.6 mm, Z: 0.794 mm
 - Endpoint 2: X: 8.15 mm, Y: 5.6 mm, Z: 0 mm
3. Navigate to the Properties tab, and name the component Feed.
4. Click **Done** to add the Feed.

Editing the Waveform

An associated waveform was automatically created for the feed definition.

1. Navigate to the **Definitions:Waveforms** branch of the Project Tree. Double-click the **Broadband Pulse waveform** to edit its properties.
2. Set the properties of the waveform as follows:
 - Name: 64 timestep Gaussian
 - Type: Gaussian
 - Pulse Width: 64 timestep

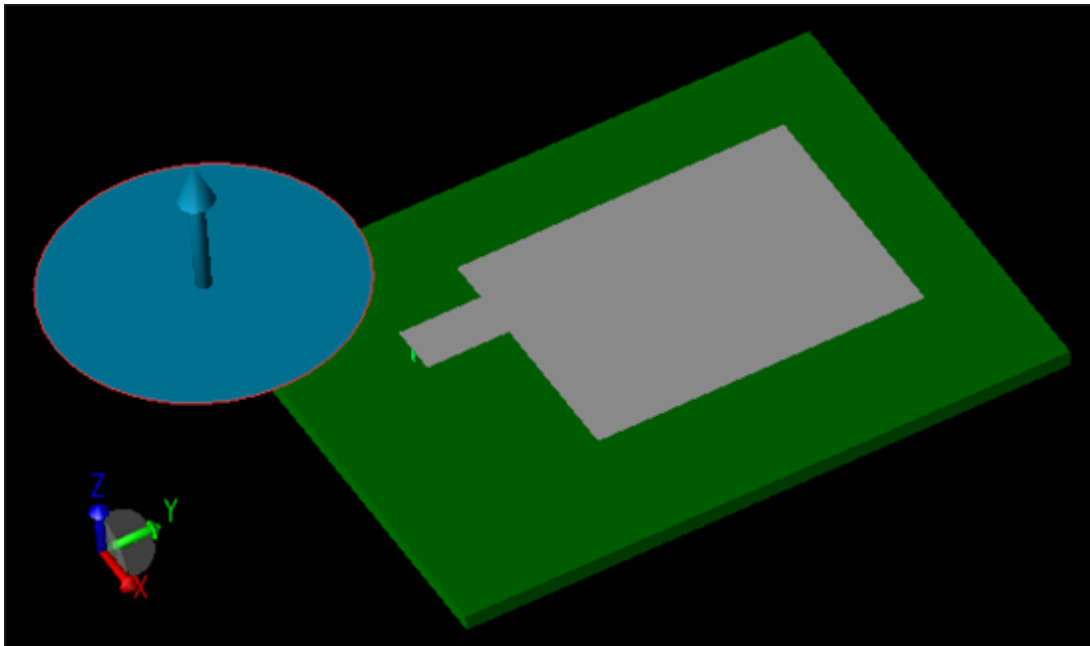


3. Click **Done** to apply the changes.

Requesting Output Data

Add a **Planar Sensor** at the surface of the Microstrip plate to retrieve electric field sampling data.

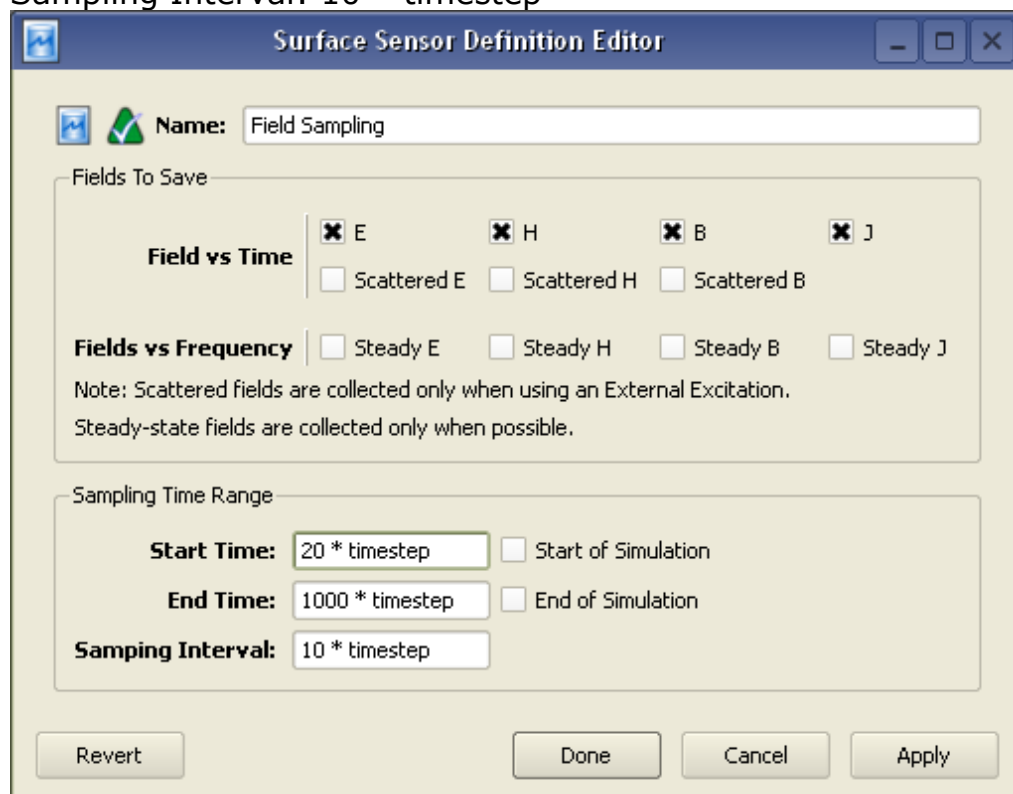
1. Right-click the **Sensors:Near Field Sensors** branch of the Project Tree. Select **New Planar Sensor** from the context menu.
2. Place the Point at (0, 0, 0.794 mm) to position the sensor on top of the Substrate. Set the Normal to (0, 0, 1).



3. Click **Done** to add the planar sensor.

This sensor requires a data definition.

1. Right-click the **Definitions:Sensor Data Definitions** branch of the Project Tree. Choose **New Surface Sensor Definition** from the context menu.
2. Set the properties of the surface sensor definition as follows:
 - Name: Field Sampling
 - Field vs. Time: E, H, B, and J
 - Start Time: 20 * timestep
 - End Time: 1000 * timestep
 - Sampling Interval: 10 * timestep




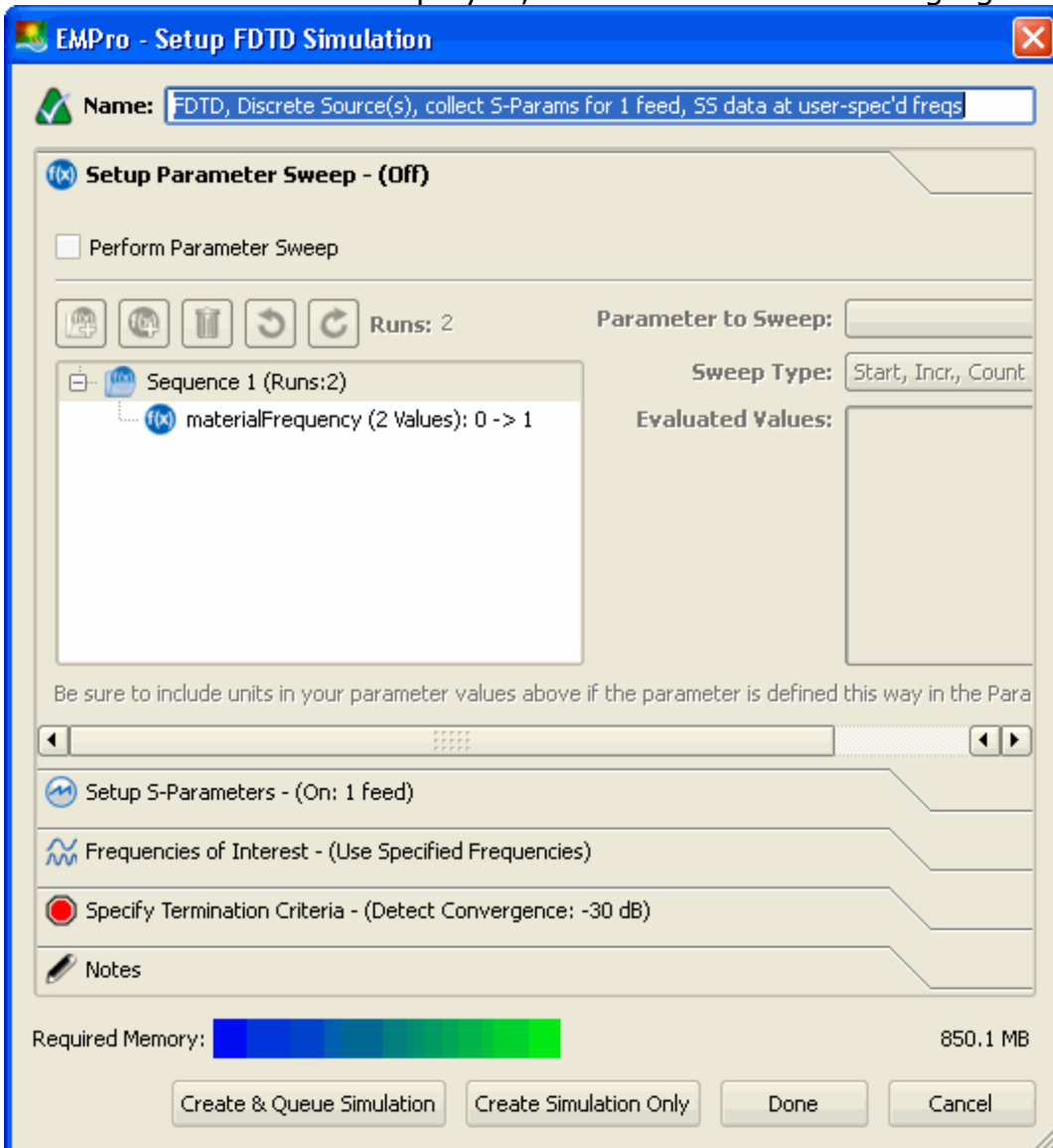
3. Click **Done** to finish editing the surface sensor definition.

Now, assign the new definition to the surface sensor. Click and drag the **Field Sampling** definition located in the Project Tree and drop it on top of the **Surface Sensor** in the **Sensors:Near Field Sensors** branch.

Running a Simulation

If you have not already saved your project, do so by selecting **File>Save Project**. After the project is saved, a new simulation can be created to send to the calculation engine.

1. Select **FDTD** from the drop-down list on the Simulations toolbar, as shown in the following figure:
2. Click **Setup** () to create and edit the setup of a new simulation. The **Setup FDTD Simulation** window is displayed, as shown in the following figure:



3. The default settings are sufficient for this example. Click **Create & Queue Simulation** to close the dialog and run the new simulation.

Viewing the Results

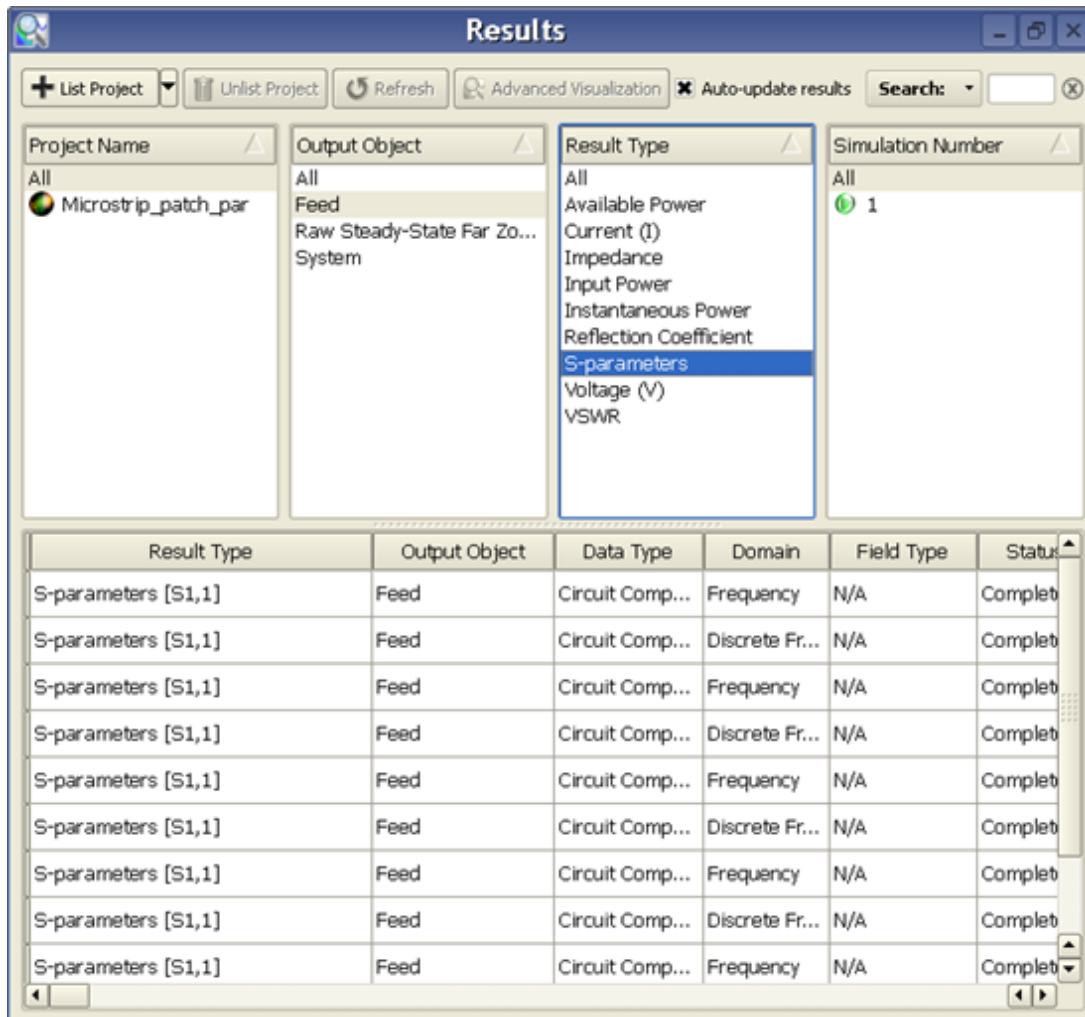
The Output tab of the Simulations workspace window displays the progress of the simulation. After the Status column shows that the simulation has completed, you can view its results in the Results workspace window.

S-Parameter Results from the Port Sensor

First, view the S-parameter results retrieved with the port sensor placed at the location of the Feed.

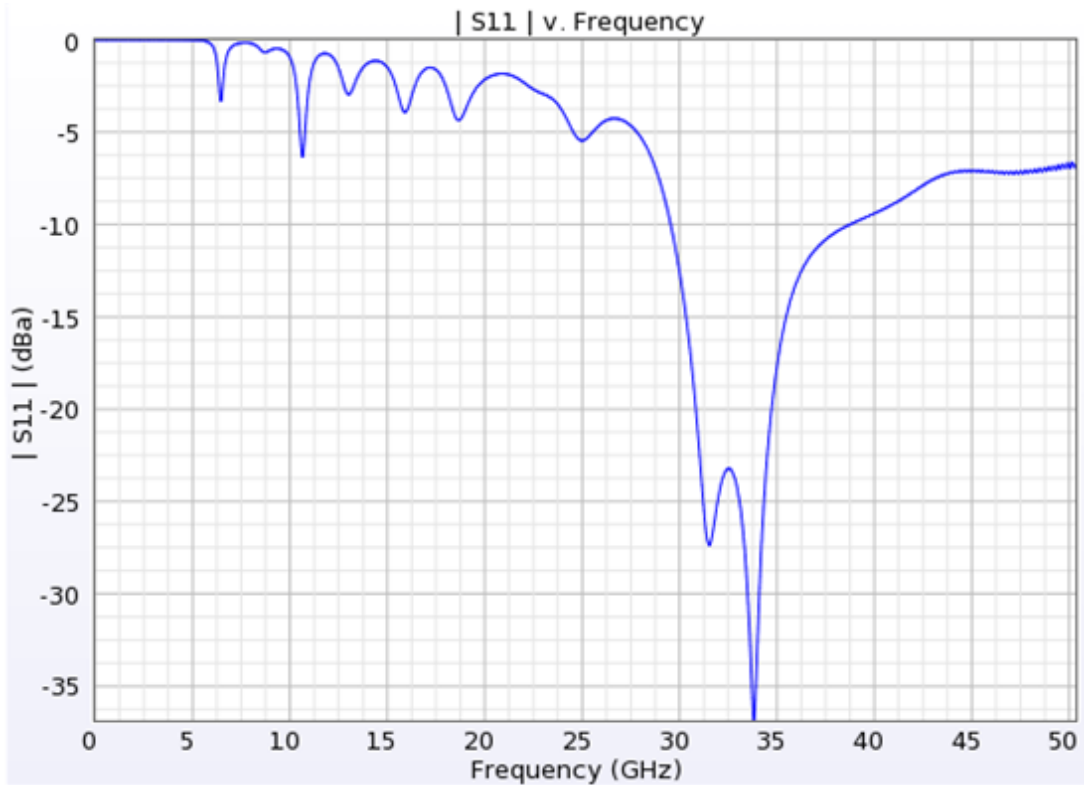
- To filter the list accordingly, select the following options in the columns in the top pane of the Results window (You may need to change your column headings first).
 - Output Object: Feed
 - Data Type: Circuit Component
 - Result Type: S-Parameters

Viewing S-Parameters in the Results window



- Double-click the result with a Domain value of Frequency to view transient S-parameter results. The following plot appears:

Viewing S-Parameters v. Frequency plot

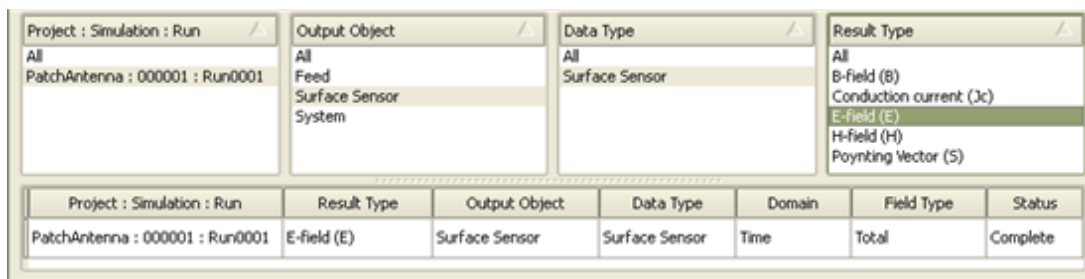


E-Field Results from the Surface Sensor

You can now view the results retrieved from the Surface Sensor.

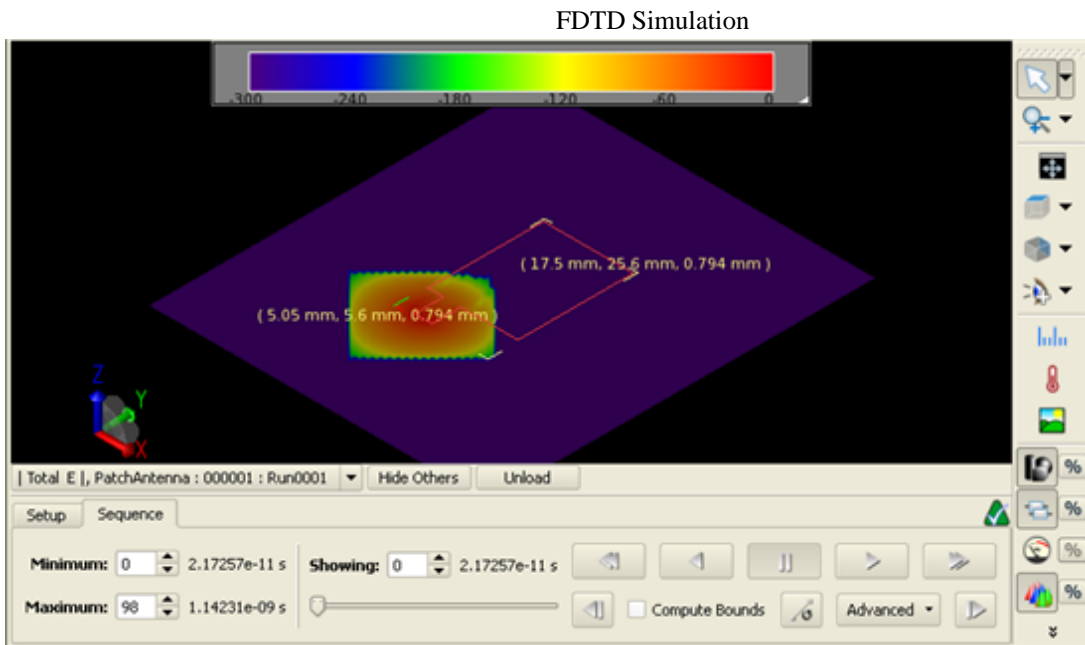
- To filter the E-field results, select the following options:
 - Output Object: Planar Sensor
 - Data Type: Surface Sensor
 - Result Type: E-Field (E)

Viewing E-Field in the Results window



- Double-click on the E-Field (E) result in the filtered list. The Results workspace window appears to view the electric field time sequence.

Viewing E-Field output in the Geometry window



3. Navigate to the **Sequence** tab to view the results. You can play back the results as an animation or step through them with the Showing control. If you wish, change the Minimum and Maximum settings to only display a certain range of the sequence.
4. Click the **Unload** button when you are finished viewing the E-field results.

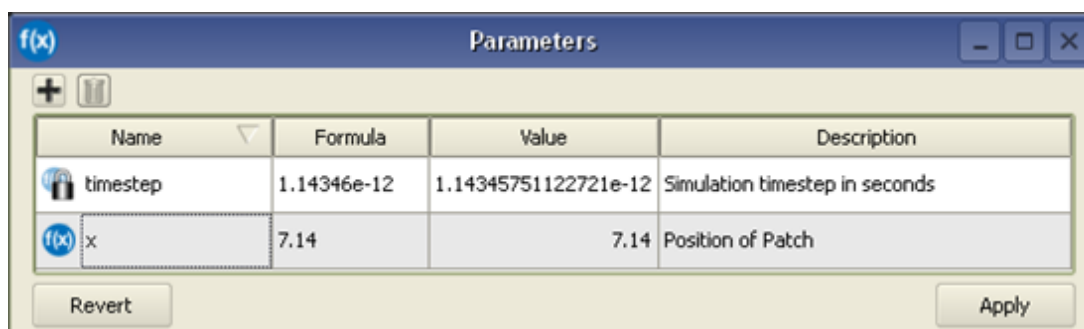
Adding a Parameter Sweep

You can parameterize your project by defining variables within the Parameters workspace window so that you can reference them in any editor or dialog window. Additionally, it incorporates the ability to perform a Parameter Sweep so that a calculation increments the value of a variable in order to perform a calculation at every iteration.

For this patch antenna example, define a parameter called **x** that controls the position of the Feed and antenna Stub. Later, set up a parameter sweep so that the calculation engine retrieves values for several incremented locations of the feed.

1. Open the Parameters workspace window. Click **Add** to add a new parameter.
 - Name: x
 - Formula: 7.14
 - Description: Position of Patch

Defining a global parameter

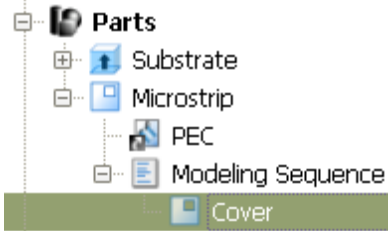


2. Press **Apply** to add the parameter to the project.

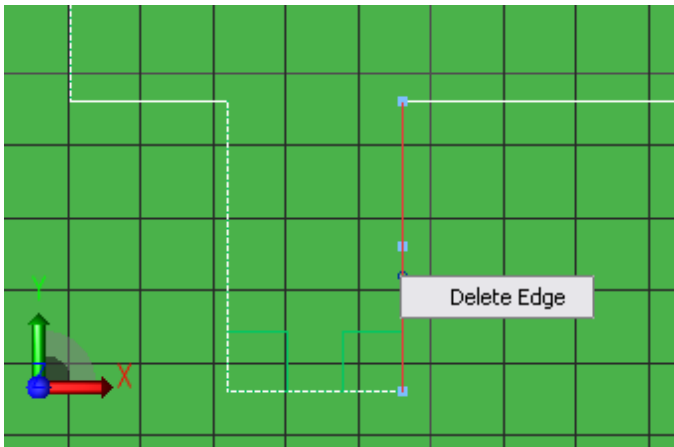
Parameterizing the Geometry

For brevity, we originally created the antenna Microstrip in one piece. To parameterize the Stub location, redraw it as two separate sheet bodies.

1. In the Project Tree, navigate to the **Microstrip:Modeling Sequence** branch and double-click the **Cover** object.



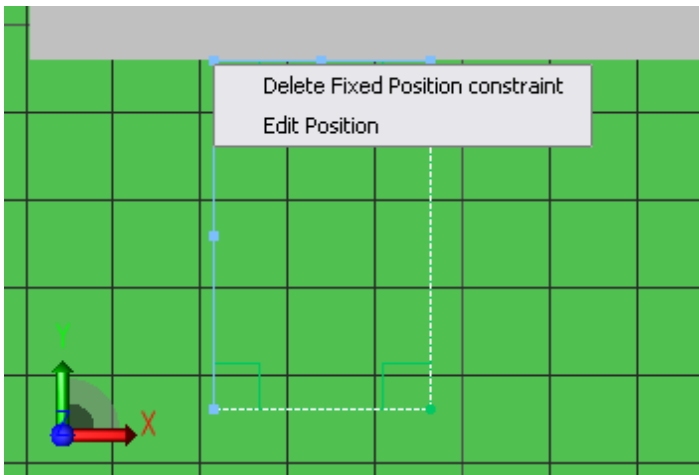
2. Select the **Select/Manipulate** tool at the top left of the Geometry workspace window. Right-click on an edge of the stub extension, and select Delete Edge to remove it.



3. Repeat this process to remove the other two stub edges.
4. Left-click the left endpoint in the bottom edge of the sketch. Drag it to connect with the neighboring endpoint and close the gap.
5. Click **Done** to apply your changes.

Now, add a Stub sheet body with a parameterized location.

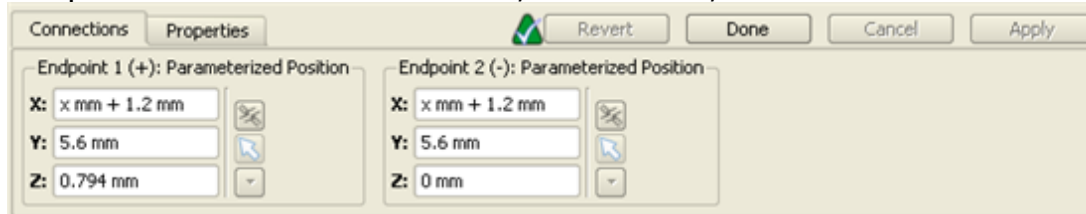
1. Right-click on the Parts branch in the Project Tree. Choose **Create New Sheet Body** from the context menu.
2. Navigate to the **Specify Orientation** tab. Set the origin to (0, 0, 0.794 mm) to place the **Sheet Body** on top of the Substrate.
3. Navigate to the **Edit Profile** tab. Type **Stub** into the Name box.
4. Select the **Rectangle** tool. Use the creation dialog to enter the corners of the stub rectangle:
 - Endpoint 1: (7.2 mm, 5.6 mm)
 - Endpoint 2: (9.6 mm, 9.6 mm)
 Because of the sketcher automatic constraint behavior, you must parameterize all four corners of the stub so it can move whenever the value of **x** changes.
5. To change the position of a vertex, choose the **Select/Manipulate** tool again and right-click the vertex. Choose **Edit Position** from the menu.



6. Set the stub rectangle corner positions as follows (be sure to add mm where necessary):
 - Upper left: (x mm, 9.6 mm)
 - Upper right: (x mm + 2.5 mm, 9.6 mm)
 - Lower left: (x mm, 5.6 mm)
 - Lower right: (x mm + 2.5 mm, 5.6 mm)
7. Click **Done** to finish the Stub geometry.
8. Assign the PEC material to the Stub by drag and dropping it onto the object.

Parameterizing the Feed

1. Locate the **Circuit Components** branch of the Project Tree, and double-click the **Feed** object to edit its position.
2. Set the endpoints of the feed as follows:
 - Endpoint 1: X: x mm + 1.2 mm, Y: 5.6 mm, Z: 0.794 mm
 - Endpoint 2: X: x mm + 1.2 mm, Y: 5.6 mm, Z: 0 mm



3. Click **Done** to finish editing the Feed.

Running a Simulation with Parameter Sweep

You can save the project by selecting **File> Save Project**.

1. Open the **Simulations** workspace window. Click **New FDTD Simulation** in the upper-left to set up a new simulation. Most of the default settings are sufficient. For this simulation, define a parameter sweep so that the calculation engine collects 5 sets of results, each based on an incremented value of our global parameter, **x**.
2. Navigate to the **Setup Parameter Sweep** tab. Select the **Perform Parameter Sweep** check box. Specify the following values:
 - Parameter to Sweep: x
 - Sweep Type: Start, Incr, Count
 - Count: 5
 - Starting Value: 6.9

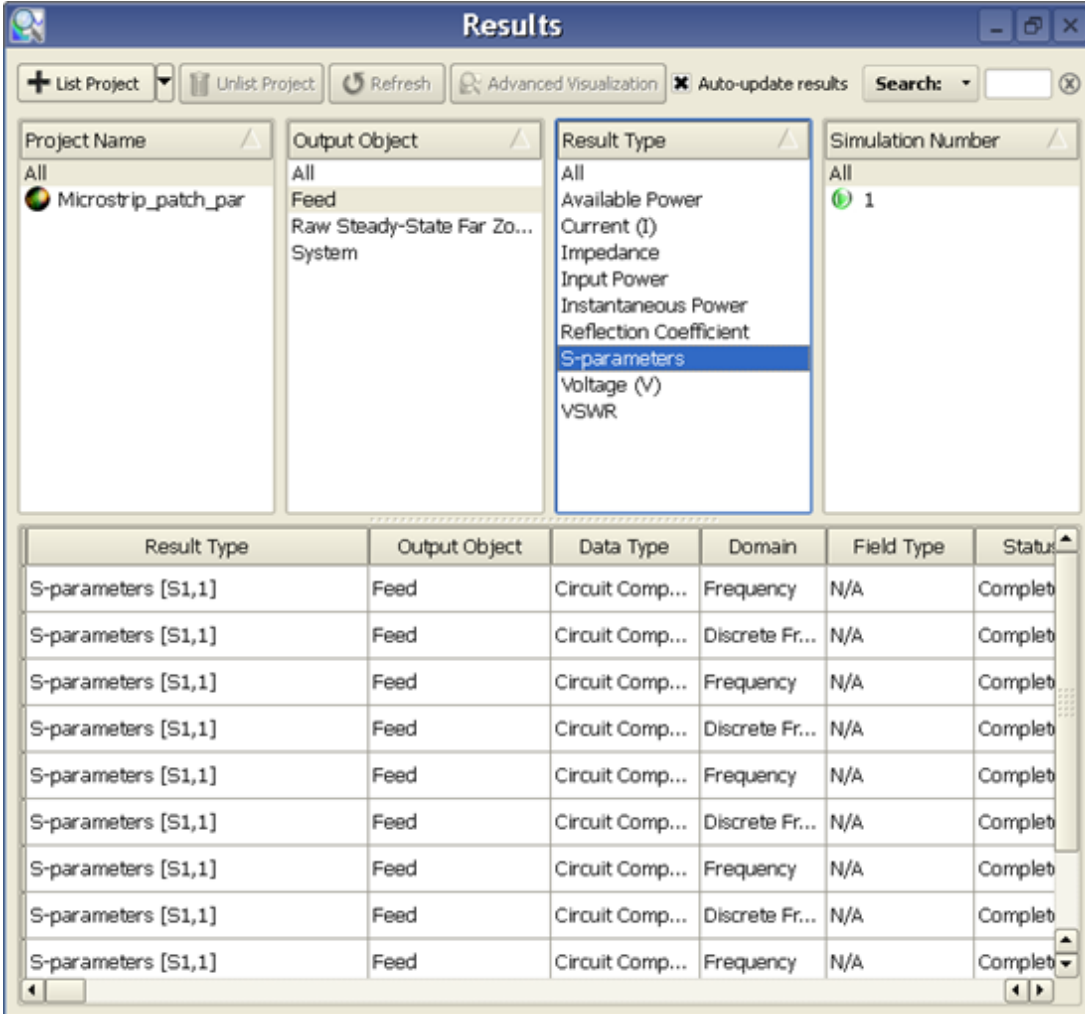
- Increment: 0.1

These settings generates a range of values from 6.9 to 7.3.

3. Click **Create & Queue Simulation** to run this simulation.

Viewing Results of the Parameter Sweep

After the Status column in the Simulations workspace window shows that the simulation has completed, you can view its results in the Results window. Under the **Project:Simulation:Run** column, notice that within the simulation, a new run is created for each parameter value.



Creating a Low Pass Filter Simulation

In this example, a low pass filter simulation is created by performing the following tasks:

- Set the material properties of a low pass filter and create the geometry through scripts.
- Define the properties of the low pass filter environment.
- Add a feed and a load to the filter and simulate their effects.
- Retrieve port sensor and planar surface sensor data after running the calculation.

Getting Started

To create a Low Pass Filter geometry, create a new project:

1. Select **Edit > Project Properties Editor** to open the **Project Properties Editor** window.
2. Select the **Display Units** tab.
3. Select **SI Metric** in the **Unit Set** drop-down list.
4. Change **Length** to **millimeters (mm)**. This changes the value of Unit Set to **Custom**.
5. Click **Done**.

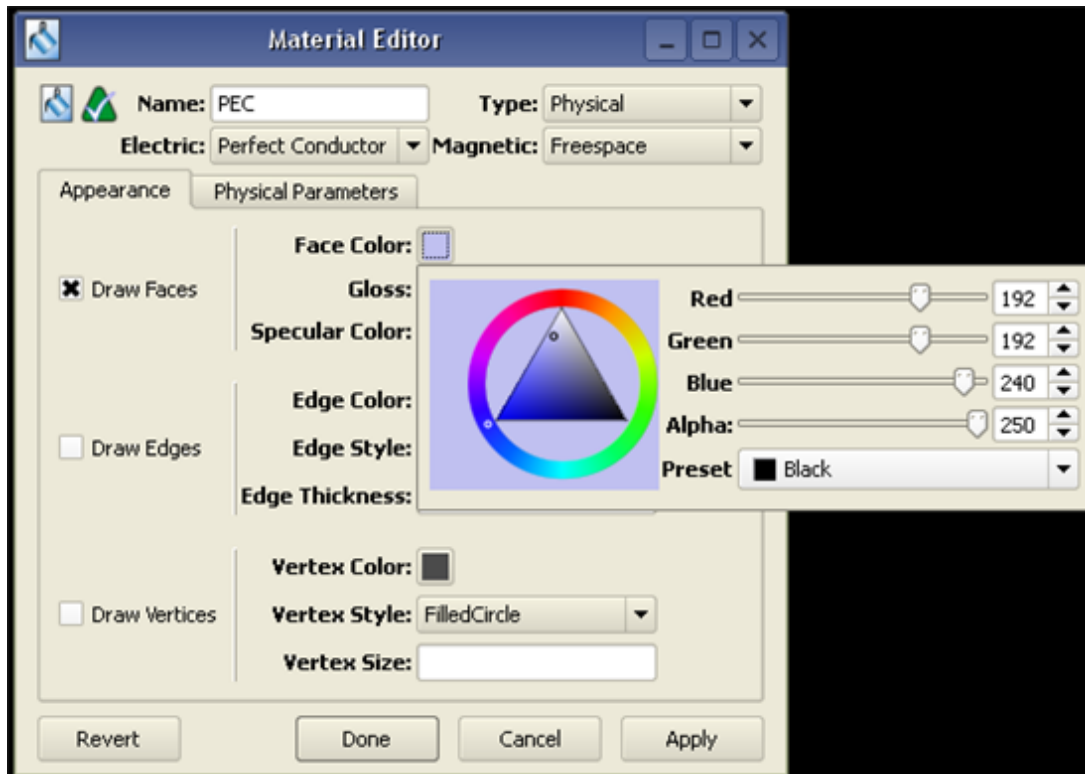
Creating Materials

For this example, you need to create material definitions before creating the geometry so that the script that is executed to build the Substrate block can access the material definitions. The low pass filter will consist of Perfect Electric Conductor and Substrate.

Defining Material (PEC)

1. Right-click **Definitions:Materials** branch of the project tree and select **New Material Definition**.
2. Double-click the new material to edit its properties. Specify the following properties for the electric conductor material:
 - **Name:** PEC
 - **Electric:** Perfect Conductor
 - **Magnetic:** Freespace
 Depending on your requirements, you can also set the display color of the PEC material in the **Appearance** tab, as shown in the following figure.

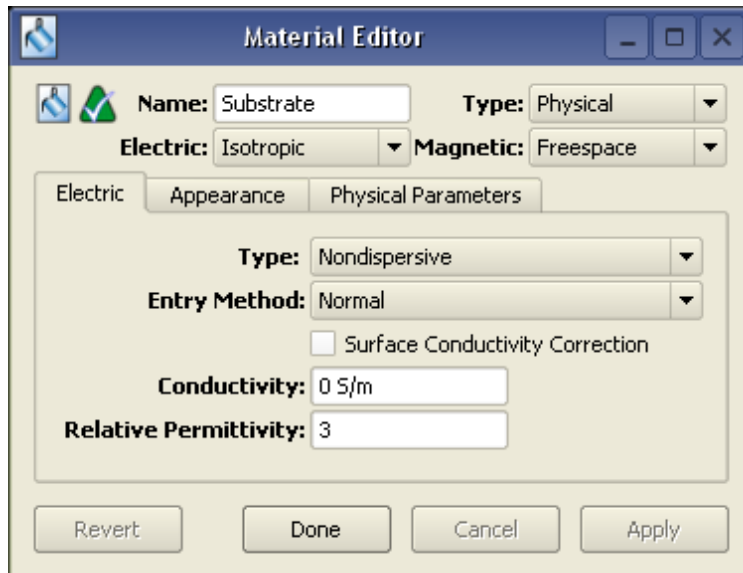
[Editing the color of the PEC material](#)



Defining Material, Substrate

1. Right-click the **Definitions:Materials** branch of the Project Tree and select **New Material Definition**.
2. Double-click the new material to edit its properties. Set the substrate material properties as follows:
 - **Name:** Substrate
 - **Electric:** Isotropic
 - **Magnetic:** Freespace
 - Under the Electric tab:
 - **Type:** Nondispersive
 - **Entry Method:** Normal
 - **Conductivity:** 0 S/m
 - **Relative Permittivity:** 3

Defining the properties of the Substrate material



3. In the **Appearance** tab, assign a new color to this material to distinguish it from the first material, PEC.
4. Click **Done** to add the new material, Substrate.

Creating the Low Pass Filter Geometry

Modeling the Substrate

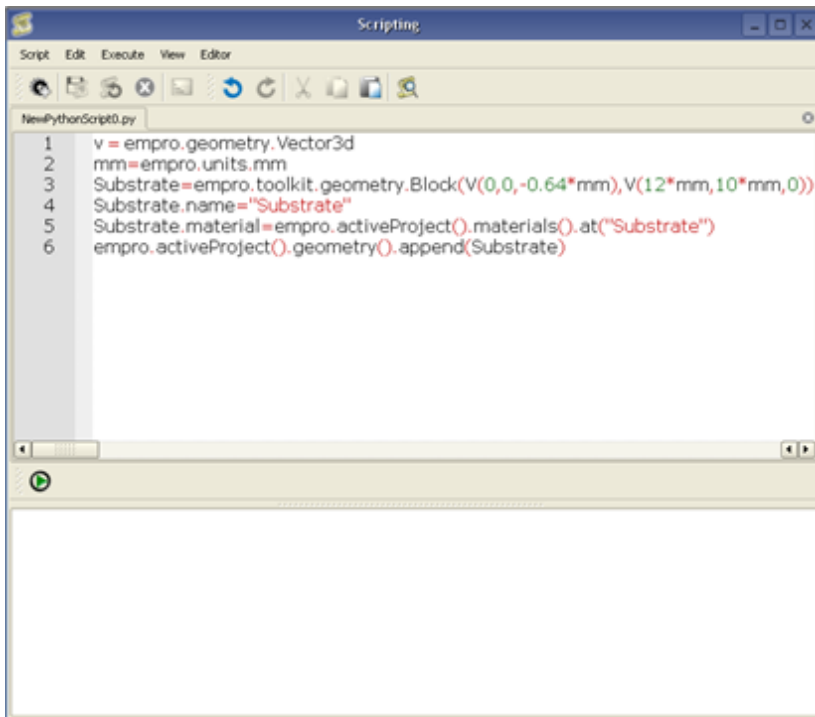
The Substrate block geometry for the low pass filter is a simple rectangular block. For this example, you will use a script to prompt an interface where we can create a rectangular block with an applied material.

1. Right-click the **Scripts** branch of the Project Tree and select **New Python Script**.
2. This automatically adds a **New Python Script** object to the branch. Right-click the object, select **Rename**, and type **Rectangular_Block**.
3. Copy the following script into the *Scripting* workspace window.

Note

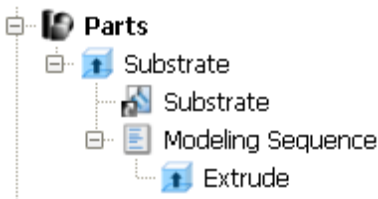
This script can also be found in the `python_scripts/demo/miscellaneous` folder of your installation directory, depending on the location where you have installed on your computer.

[The substrate script in the Scripting workspace window](#)



4. Choose **Script > Commit** to commit your new script to the project.
5. Click **Execute Python Script** to run the script which will add the Substrate block to the project.
6. Drag-and-drop the object material **Substrate** under Materials on the Substrate block to assign a material to this object.

Substrate block in the Project Tree



Modeling the Strip Line

The Strip Line will be modeled as a polygon Sheet Body. To simplify this operation, you will use another script to add the Strip Line to the project.

1. Right-click the **Scripts** branch of the Project Tree and select **New Python Script**.
2. Right-click the **New Python Script** object, select **Rename**, and type **Strip_Line**.
3. Copy the following script in the *Scripting* workspace window.

```
V = empro.geometry.Vector3d
E = empro.core.Expression
SCALED_UNIT = E("1 mm")
V1= V(0, 4.77, 0) * SCALED_UNIT; V2=V(0, 5.23, 0)*SCALED_UNIT; V3=V(2, 5.23,
0)*SCALED_UNIT; V4=V(2, 6.9, 0)*SCALED_UNIT;
V5=V(4.7, 6.9, 0) * SCALED_UNIT; V6=V(4.7, 5.23, 0) * SCALED_UNIT; V7=V(7.3, 5.23, 0) *
SCALED_UNIT; V8= V(7.3, 6.9, 0) * SCALED_UNIT;
V9=V(10, 6.9, 0) * SCALED_UNIT; V10=V(10, 5.23, 0) * SCALED_UNIT; V11=V(12, 5.23, 0) *
SCALED_UNIT; V12=V(12, 4.77, 0) * SCALED_UNIT;
V13=V(10, 4.77,0) *SCALED_UNIT; V14=V(10, 3.1,0) *SCALED_UNIT; V15=V(7.3, 3.1,0)
```

FDTD Simulation

```
*SCALED_UNIT;V16=V(7.3, 4.77,0) *SCALED_UNIT;  
V17=V(4.7, 4.77,0) *SCALED_UNIT; V18=V(4.7, 3.1,0) *SCALED_UNIT; V19=V(2, 3.1,0)  
*SCALED_UNIT; V20=V(2, 4.77,0) *SCALED_UNIT;  
vertices=[  
V1,V2,V3,V4,V5,V6,V7,V8,V9,V10,V11,  
V12,V13,V14,V15,V16,V17,V18,V19,V20  
]  
StripLine=empro.toolkit.geometry.PolyPlate(vertices,"Strip Line")  
empro.activeProject().geometry().append(StripLine)
```

Note

This script can also be found in the python_scripts/demo/miscellaneous folder of your installation directory, depending on the location where you have installed on your computer.

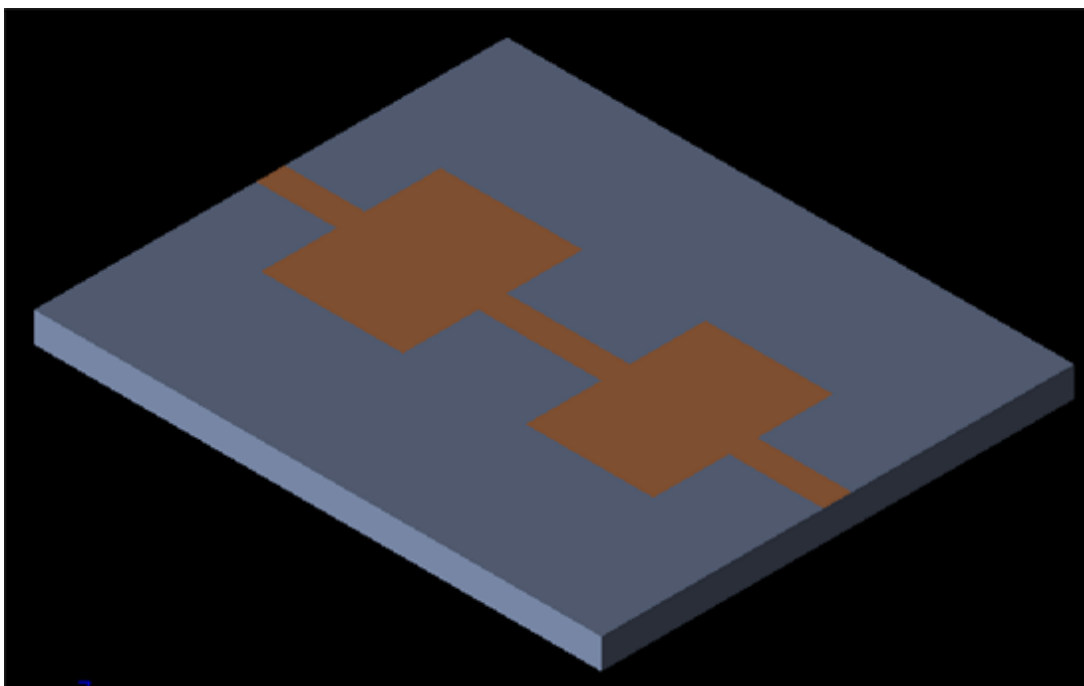
4. Choose **Script > Commit** to commit your new script to the project.
5. Click **Execute Python Script** to run the script and create the Strip Line object.

Strip Line geometry created from script



6. Drag-and-drop the material object PEC onto the object Strip Line to assign a material to this object.
The completed low pass filter geometry will appear in the *Geometry* workspace window.

Low pass filter geometry

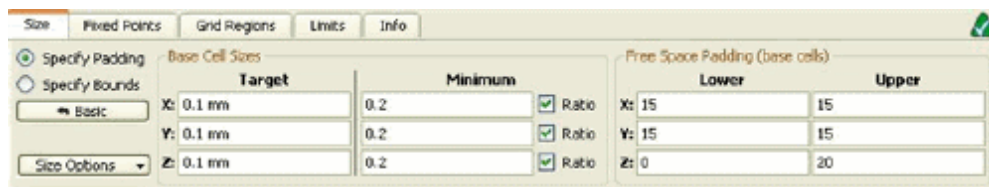


Creating the Grid

Now you will define characteristics of the cells in preparation to perform an accurate calculation.

1. Double-click the **FDTD:Grid** branch of the Project Tree to open **Grid Tools**.
2. Set the **Size** properties of the grid as given below:
 - **Base Cell Sizes:** Target 0.1 mm, Merge 0.2, Ratio boxes checked
 - **Free Space Padding:**
 - 15 for Lower X, Lower Y, Upper X, Upper Y
 - 0 for Lower Z
 - 20 for Upper Z

Defining cell size and free space padding within Grid Tools

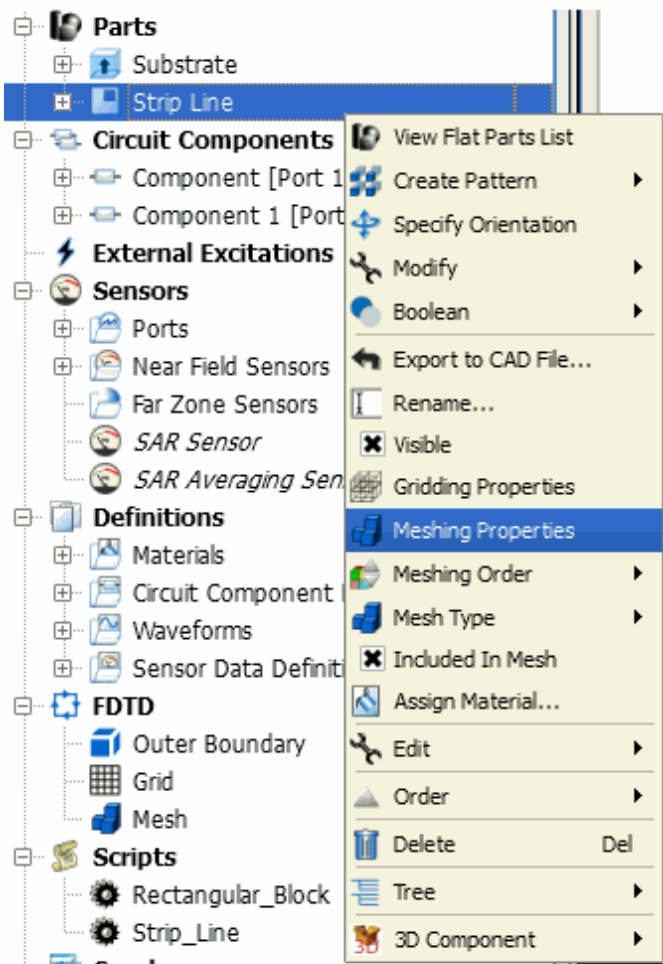


3. Click **Done** to apply the grid settings.

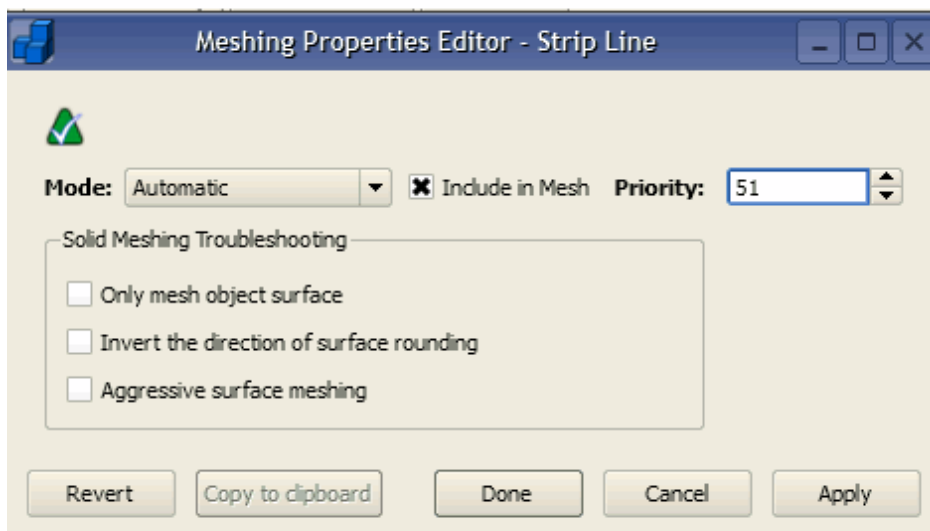
Setting Mesh Priority

For the overlapped geometry regions, you need to define the mesh with one material. Mesh priority also need to be defined because in overlap section the mesh material is decided by mesh priority. In this structure region, stripline is collocated with top plane of substrate as both are at the same z height. In stripline location, you want mesh of the material of this object, that is, PEC, hence mesh priority of stripline should be more that substrate.

1. Right-click **Stripline** and choose **Meshing Properties**.



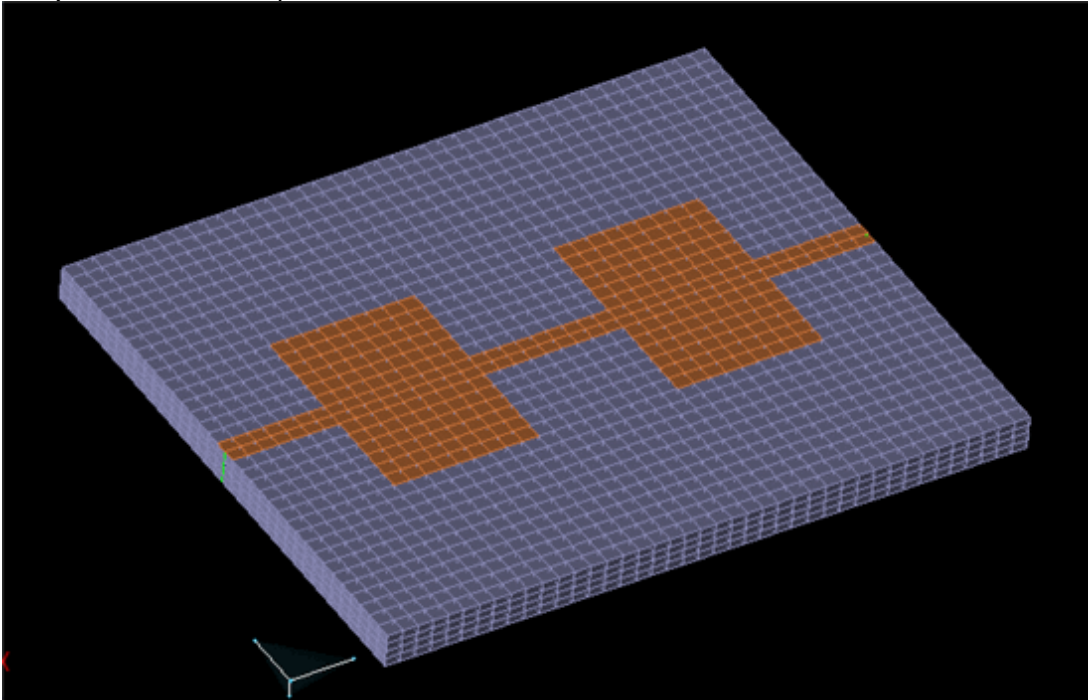
2. Set the **Mesh priority** to **51**. This sets the mesh priority of Stripline higher than the Substrate. By default, the mesh priority of all objects is set to 50.



3. Click **Toggle Mesh Control** on right side of GUI to view the 3D mesh.



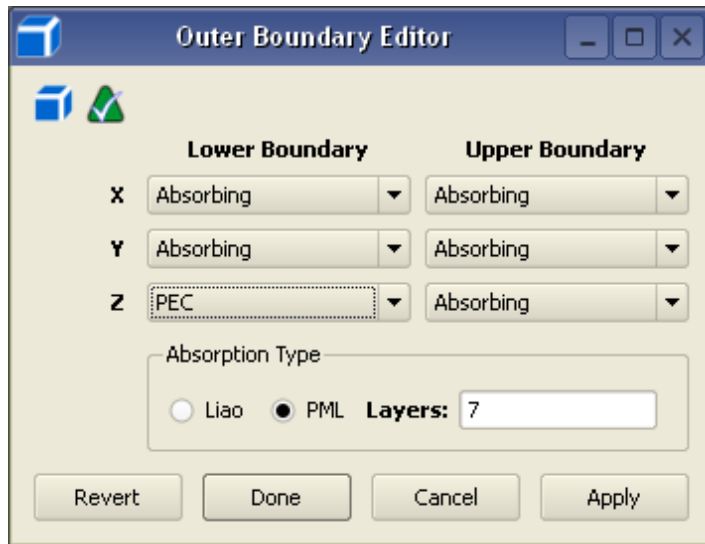
4. Mesh is displayed on the structure. Choose **3D mesh**. The color of mesh on the stripline is brown, which indicates that it is PEC material mesh.



Defining the Outer Boundary

1. Double-click the **Simulation Domain :Boundary Conditions** branch of the Project Tree to open the *Boundary Condition Editor*.
2. Set the outer boundary properties as follows:
 - **Boundary:** Absorbing for all boundaries except Lower Boundary Z, which should be PEC.
 - **Absorption Type:** PML
 - **Layers:** 7

Defining the outer boundary for the low pass filter

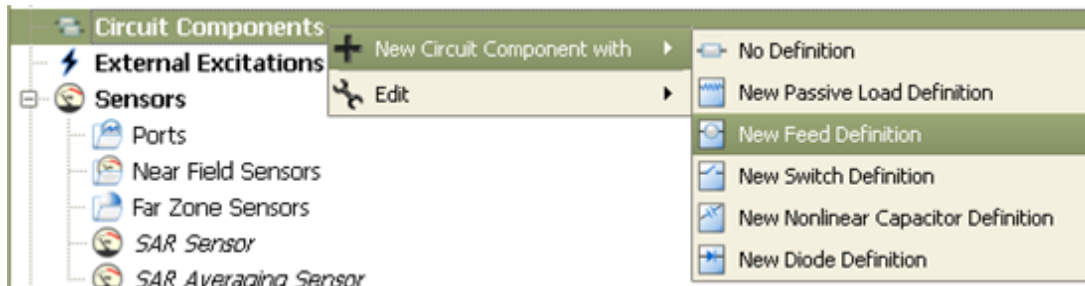


3. Click **Done** to apply the outer boundary settings.

Adding a Feed

1. Right-click the **Circuit Components/Ports** branch of the Project Tree. Choose **New Circuit Component with > New Feed Definition** from the context menu.

Adding a feed to the project



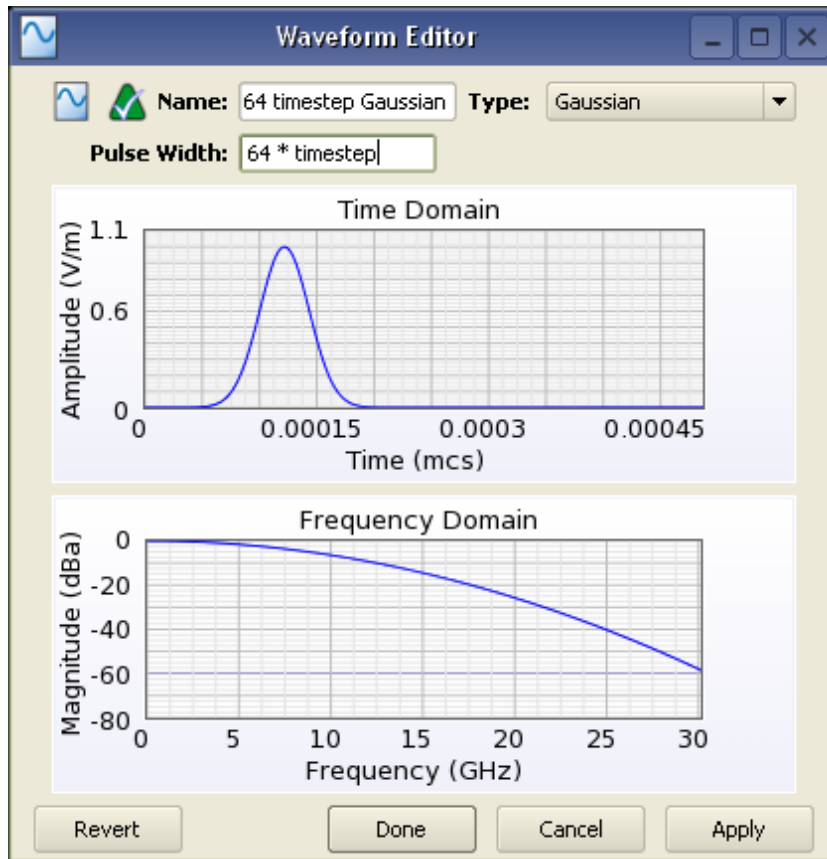
2. Define the endpoints of the feed.
 - Endpoint 1: X: 0 mm, Y: 5 mm, Z: 0 mm
 - Endpoint 2: X: 0 mm, Y: 5 mm, Z: -0.64 mm
3. Click **Done** to add the Feed.

Editing the Waveform

An associated waveform was automatically created for the feed definition.

1. Navigate to the **Definitions:Waveforms** branch of the Project Tree.
2. Double-click the **Broadband Pulse** waveform to edit its properties.
3. Set the properties of the waveform as follows:
 - **Name:** 64 timestep Gaussian
 - **Type:** Gaussian
 - **Pulse Width:** 64 * timestep

Editing the Gaussian waveform



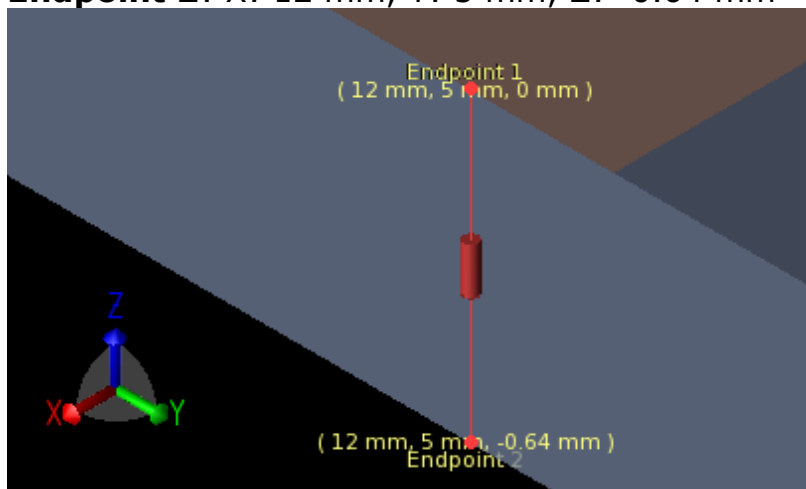
4. Click **Done** to apply the changes.

Adding a Load

The **Load** will be placed at the opposite end of the low pass filter geometry as the Feed.

Note
Before adding a load in FDTD, you should disable FEM.

1. Right-click on the **Circuit Components** branch of the Project Tree, and select **New Circuit Component with > New Passive Load** Definition.
2. Define the endpoints of the passive load.
 - **Endpoint 1:** X: 12 mm, Y: 5 mm, Z: 0 mm
 - **Endpoint 2:** X: 12 mm, Y: 5 mm, Z: -0.64 mm



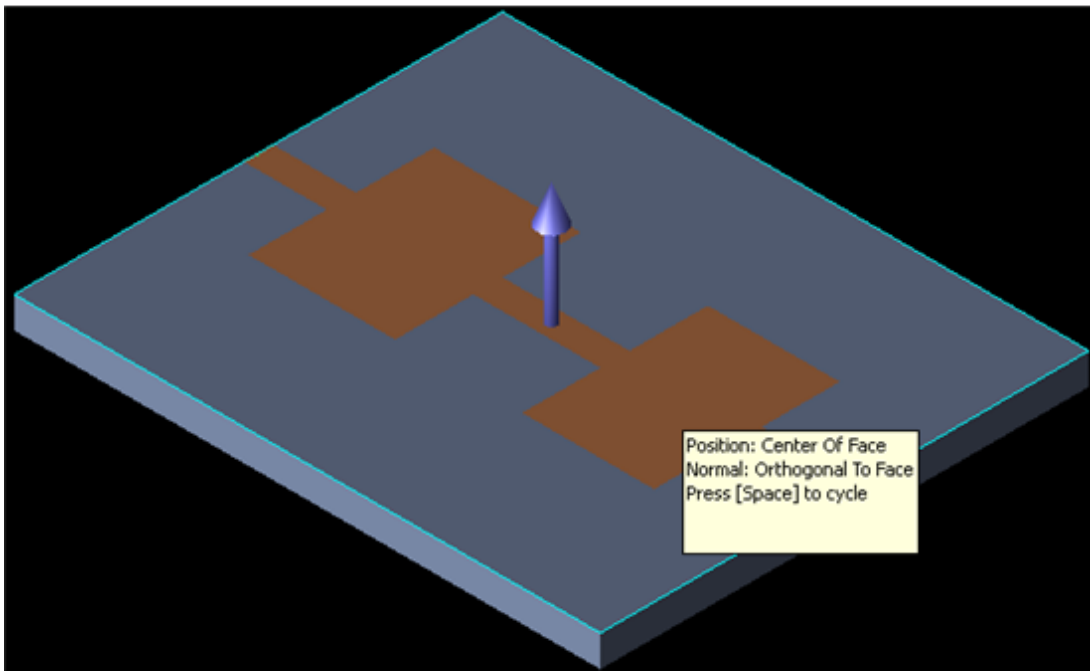
3. Click **Done** to add the Load to the project.

Requesting Output Data

This project already contains one port sensor named **Feed** that will request results. We also wish to collect field samplings at discrete intervals of time throughout the calculation. To retrieve this data, add a **Planar Sensor** at the surface of the Strip Line.

1. Right-click the **Sensors:Near Field Sensors** branch of the Project Tree. Select **New Planar Sensor** from the context menu.
2. Use the **Select** tool (at the top of the View Tools menu) to place the sensor in the middle of the **Strip Line**. Mouse over the Strip Line and press c to center the **Select** tool on the face. This will also set the sensor normal orthogonal to the face.

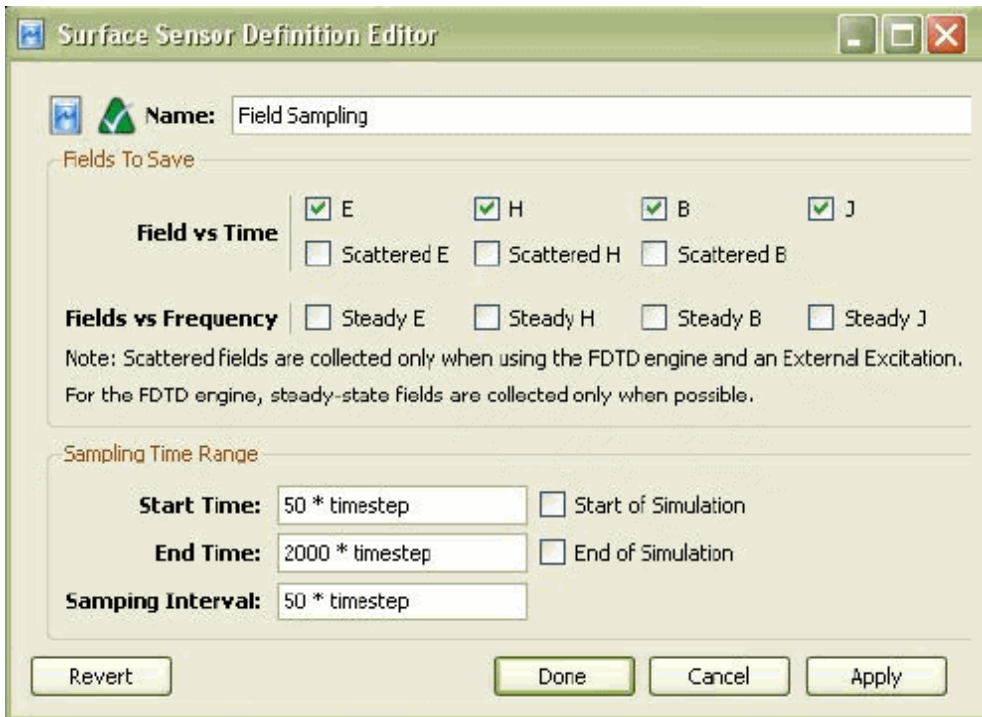
Centering the planar sensor on the Strip Line



3. Click **Done** to add the Planar Sensor to the project.

This sensor requires a data definition.


1. Right-click the **Definitions:Sensor Data Definitions** branch of the Project Tree. Choose **New Surface Sensor Definition** from the context menu.
2. Set the properties of the surface sensor definition as follows:
 - **Name:** Field Sampling
 - **Field vs. Time:** E, H, B, and J
 - **Start Time:** 50 * timestep
 - **End Time:** 2000 * timestep
 - **Sampling Interval:** 50 * timestep
3. Add the sensor definition, as shown in the following figure:

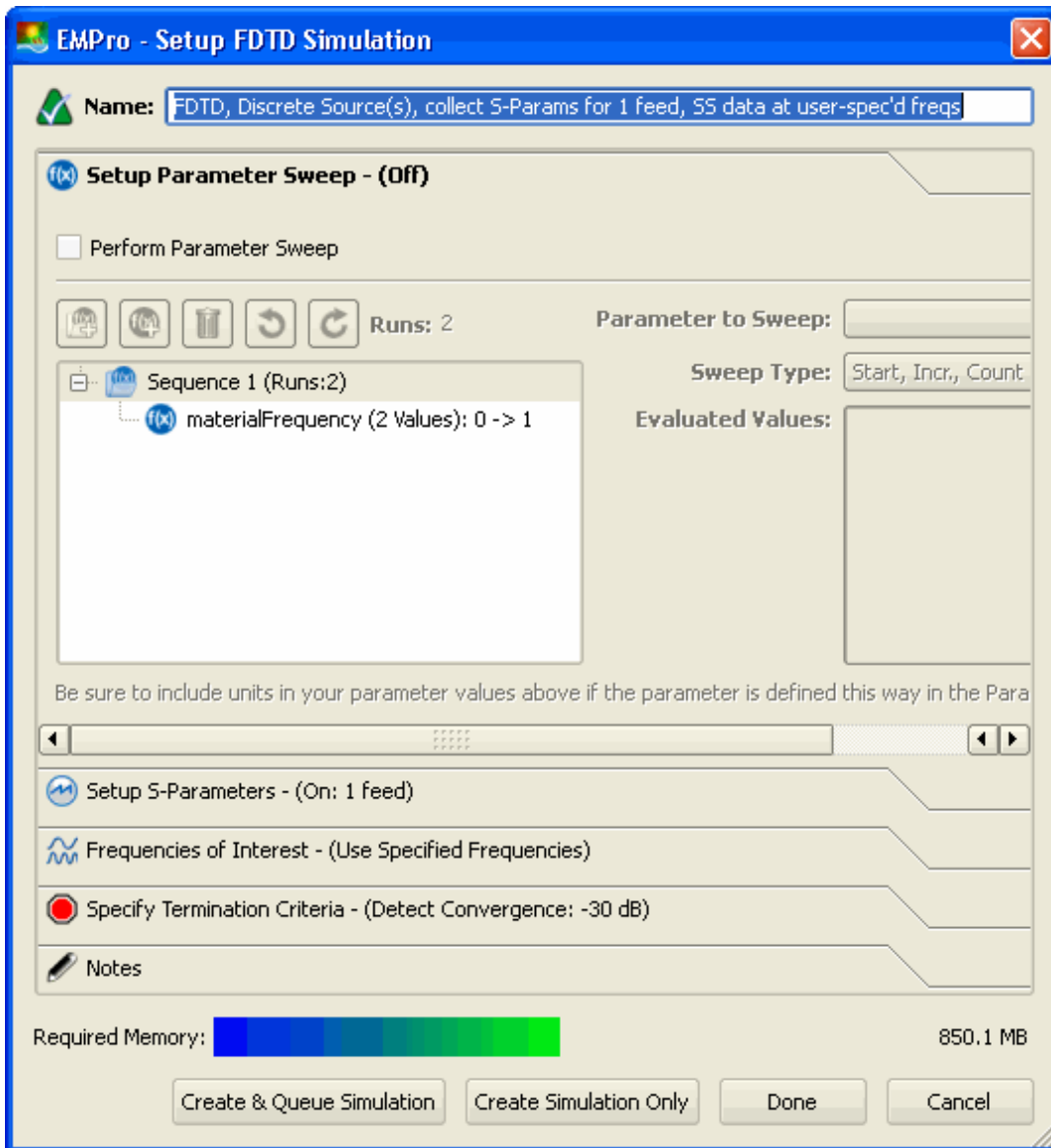


4. Click **Done** to finish editing the surface sensor definition.
5. Click and drag the **Field Sampling** definition located in the Project tree and drop it on top of the Surface Sensor in the **Sensors:Near Field Sensors** branch.

Running the Calculation

If you have not already saved your project, do so by selecting **File > Save Project**. After the project is saved, a new simulation can be created to send to the calculation engine.

1. Click **Simulations** to open the Simulations workspace.
2. Select **FDTD** from drop-down list on the Simulations toolbar, as shown in the following figure:
3. Click **Setup** () to create and edit the setup of a new simulation. The **Setup FDTD Simulation** window is displayed, as shown in the following figure:



4. Click the **Specify Termination Criteria** tab. Enter the following termination criteria:
 - **Maximum Simulation Time:** 10000 * timestep
 - **Detect Convergence:** Selected
 - **Threshold:** -40 dB
5. Select **Create and Queue Simulation** to run the new simulation.

Viewing the Results

First, you will view the results retrieved with the port sensor placed at the location of the Feed.

Ensuring convergence has been reached

Although automatic convergence has been set, it is good practice to view the waveforms in the model to ensure that the energy has completely dissipated, providing complete convergence.

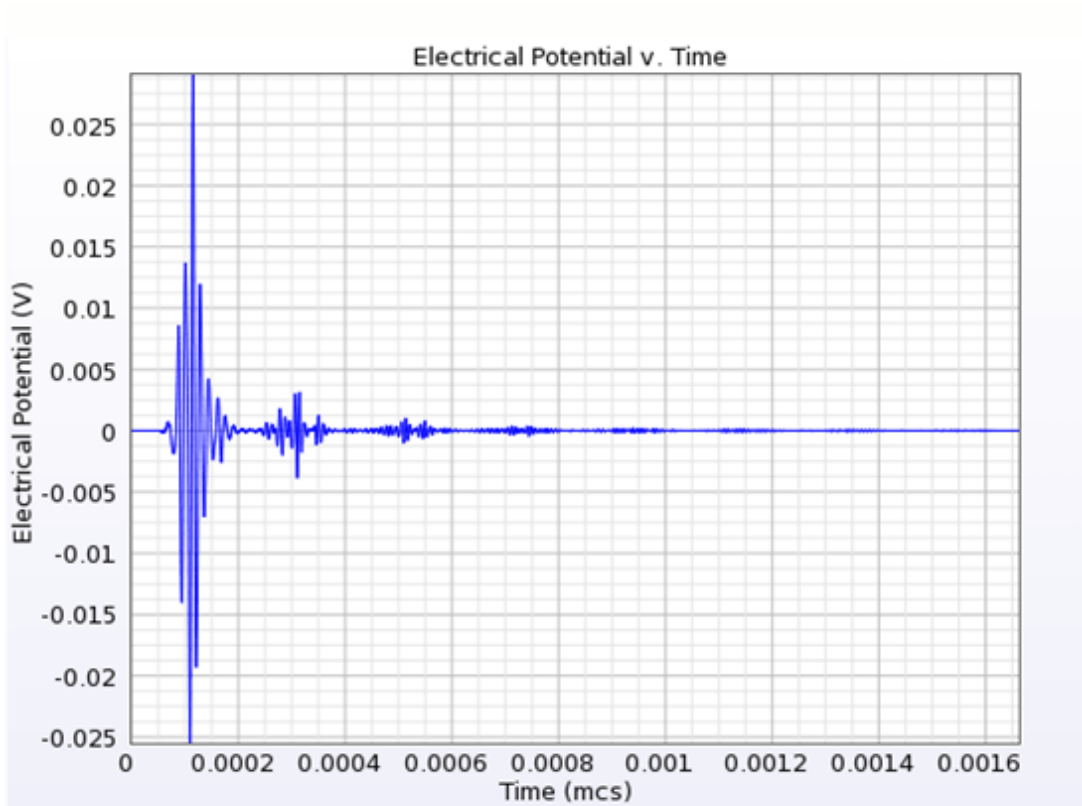
1. To filter the list accordingly, select the following options in the columns in the top pane of the *Results* window. (You may need to change your column headings first).

- **Data Type:** Circuit Component
- **Domain:** Time
- **Result Type:** Voltage (V)

This will filter all time-dependent voltage data collected by the Feed circuit component.

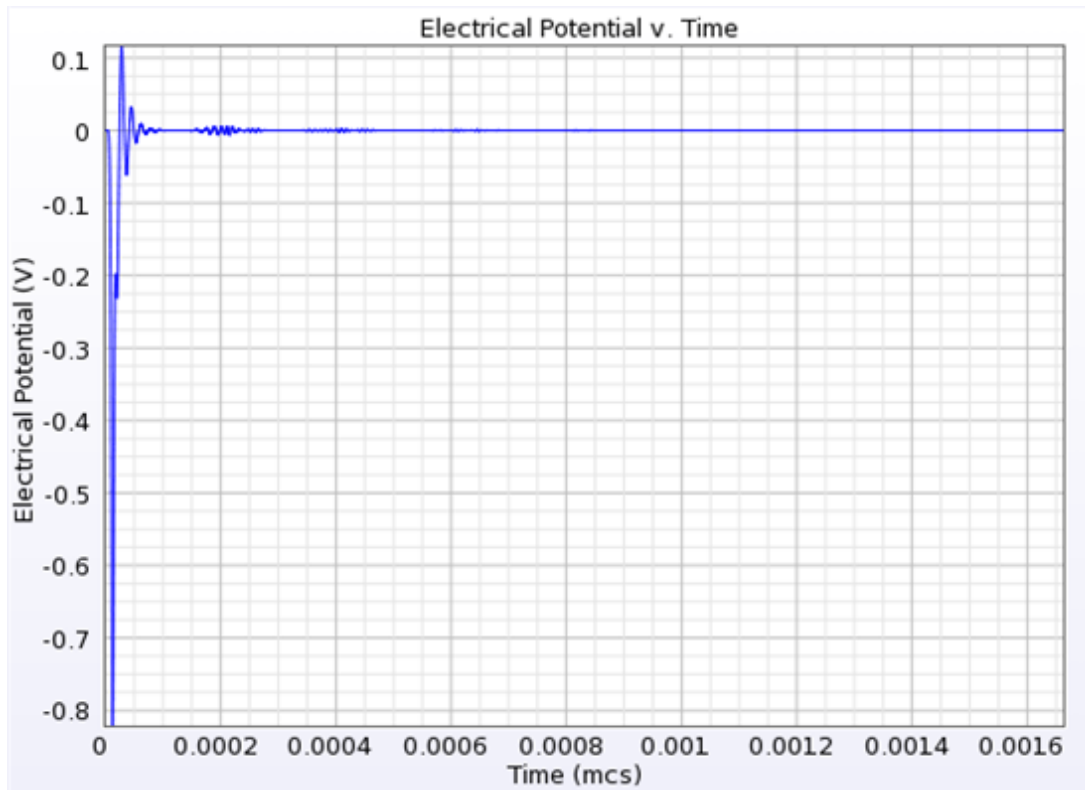
2. Double-click the **Load** result to view a 2-D plot to ensure convergence has been met.

Viewing results plot to ensure convergence at the Load



3. Repeat to view the results at the Feed.

Viewing results plot to ensure convergence at the Feed



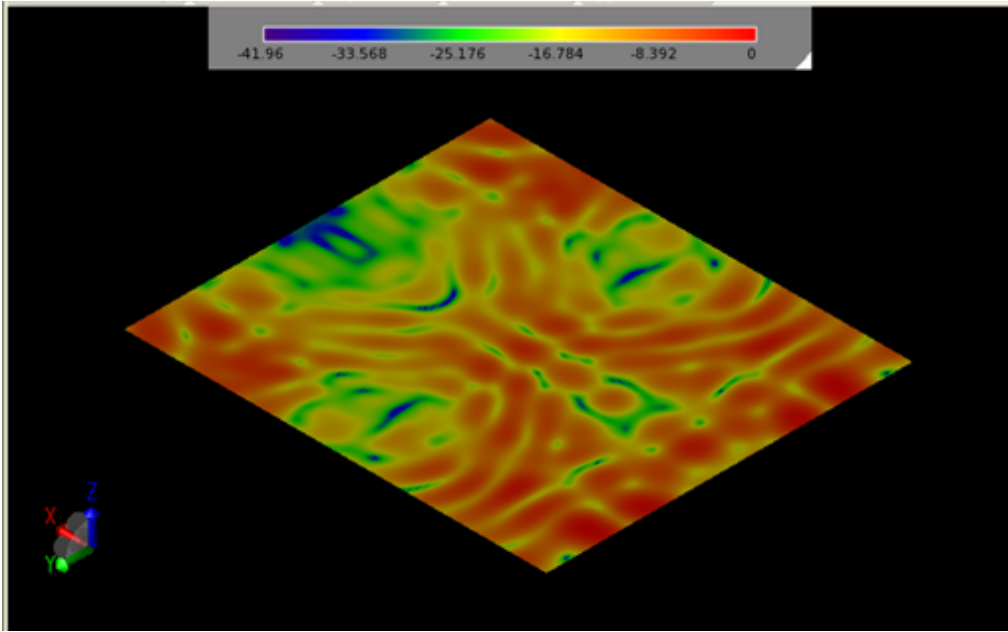
E-Field Results from the Surface Sensor

Now, you will view the field sequences collected by the surface sensor that was placed at the surface of the Strip Line.

1. To filter the E-field results, select:
 - **Data Type:** Surface Sensor
 - **Result Type:** E-Field (E)
2. Double-click the result to open the interface and view the 3-D field sequence.
3. Navigate to the Sequence tab to view the results. You can play back the results as an animation or step through them with the Showing control. If you wish, change the Minimum and Maximum settings to only display a certain range of the sequence.

Viewing E-field results for the surface sensor at the last frame of the sequence

FDTD Simulation



Creating a Monopole Antenna on a Conducting Box Simulation

In this example, a Monopole Antenna on a conducting box is simulated by performing the following tasks:

- Build a monopole antenna using solid modeling techniques.
- Define the properties of the antenna environment.
- Add a feed to the antenna and simulate its effects.
- Add a surface sensor to the box and view the calculated surface current.
- Retrieve far zone results after running the calculation.

In this example, a wire monopole is connected to a conducting box and fed at the junction. The radiation pattern is calculated at a frequency of 1.47GHz.

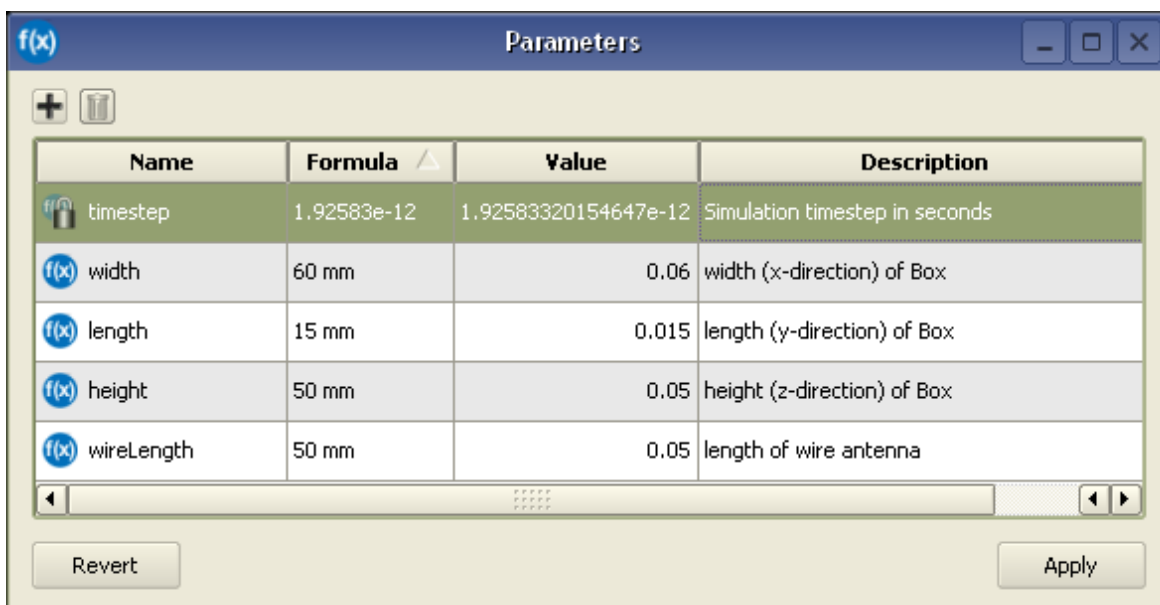
Getting Started

To create a Monopole Antenna on a conducting box, create a *new project* (global).

Parameterizing the Project

In this example, you will parameterize the dimensions of the monopole antenna geometry so that any value can be easily changed in the **Parameters** browser window.

1. Open the **Parameters** workspace window. Click **Add** to add a new parameter.
 - **Name:** width
 - **Formula:** 60 mm
 - **Description:** width (x direction) of Box
2. Add parameters named length, height, and Length in the same manner.
3. Click **Apply** to add the parameters to the project.



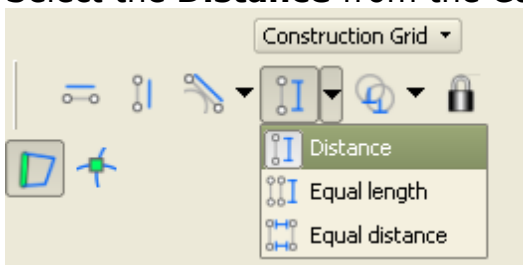
Creating the Monopole Antenna Geometry

The Monopole Antenna geometry is created with a simple Box and a Monopole antenna. The dimensions of the Box and antenna offset location is defined with parameters.

Modeling the Box

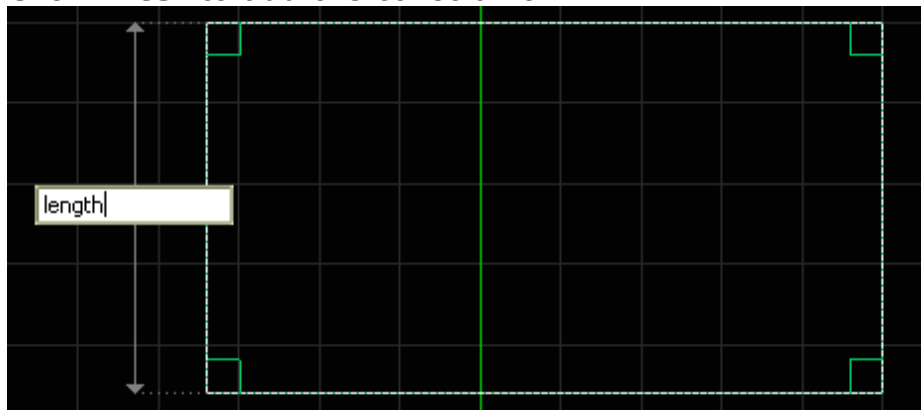
First, you need to create the rectangular substrate named Box. This object will use the parameters length, width, and height for its dimensions with an extrusion in the $+Z$ direction.

1. Right-click the **Parts** list and **Create New > Extrude**.
2. Type **Box** into the Name text box.
3. Choose the **Rectangle** tool from the **Shapes** toolbar, and draw a rectangle in the sketching plane. (Dimensions are not important).
4. Select the **Distance** from the Constraints toolbar.

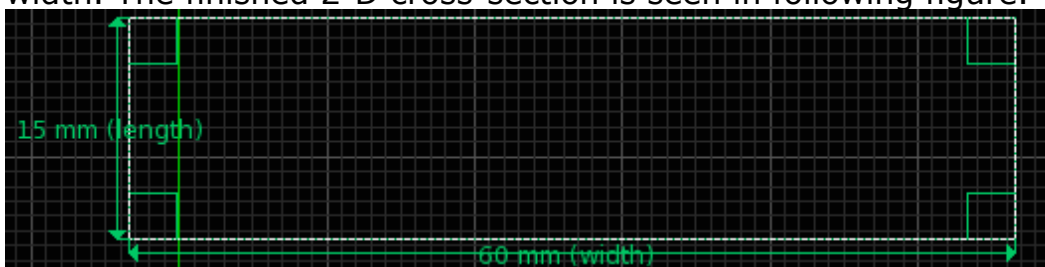


#* Click on the left vertical side of the rectangle (it will turn blue), then move the mouse slightly to its left and click again.

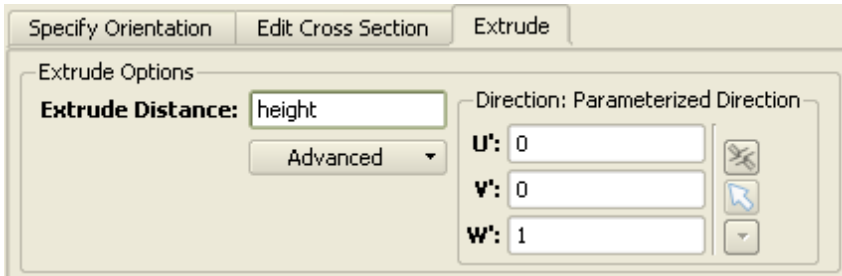
- Type **length** in the dialog box, to set its value equal to the length parameter.
- Click **Enter** to add the constraint.



5. Add a constraint to the bottom horizontal side of the rectangle, defining its value as width. The finished 2-D cross-section is seen in following figure.



6. Navigate to the **Extrude** tab to extrude the rectangular region. Enter **height** as the distance, in the $+Z$ direction.



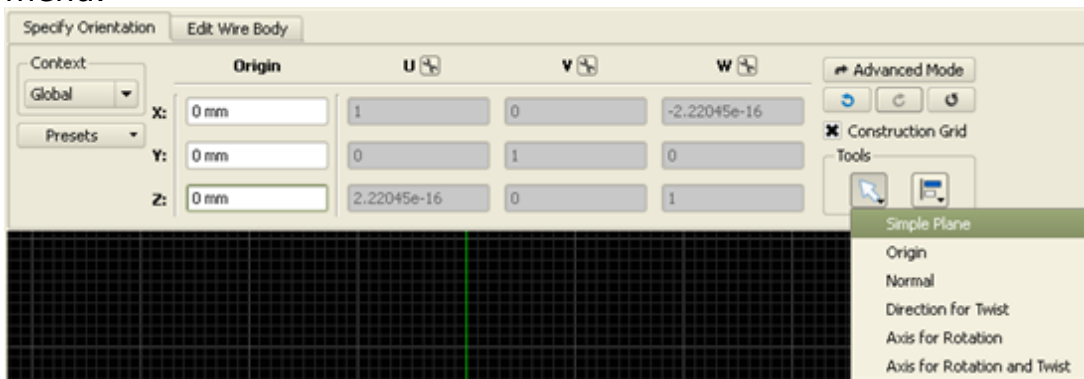
7. Click **Done** to complete the Box geometry.

Now the dimensions of the Extrude Box are completely parameterized and adjustable from the Parameters workspace window.

Modeling the Monopole

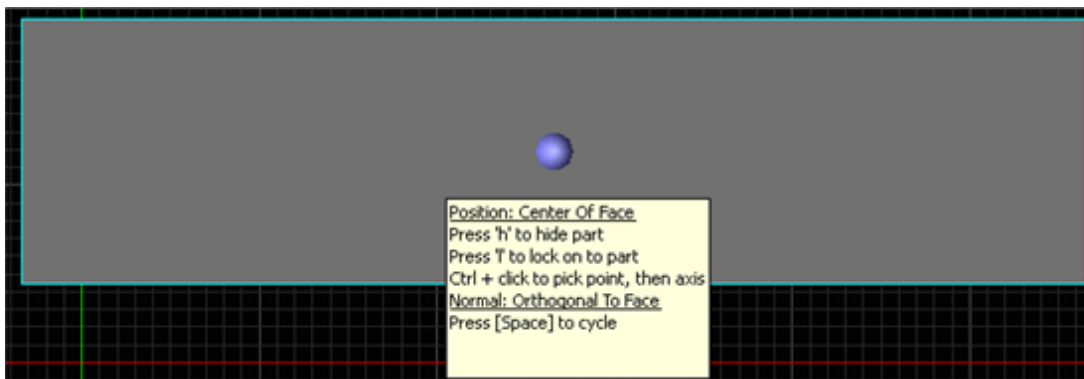
The Monopole will be created with a **Wire Body** object that is locked to the top center of the Extrude Box. Its length will be defined by the parameter Length.

1. Right-click the **Parts** branch of the Project Tree. Choose **Create New > Wire Body** from the context menu.
2. In the **View Tools** toolbar, select the **Top (-Z)** orientation.
3. Navigate to the **Specify Orientation** tab and select Simple Plane from the **Pick** menu.

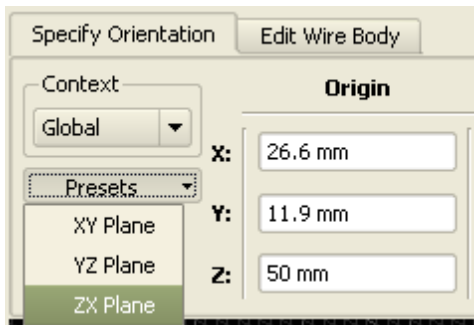


- Place the mouse over the face of the box and click c to center the Origin. Click on this location to set the values.

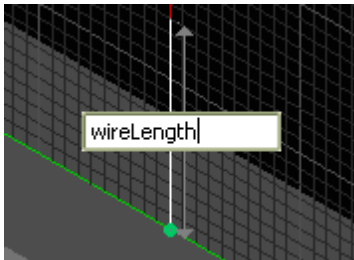
Note
If the Origin does not move to the center as expected, right-click in the geometry space to activate the window.



4. Redefine the orientation of the sketching plane by selecting the **ZX** Plane under the **Presets** drop-down list.



5. Navigate to the **Edit Wire Body** tab. Type **Monopole** into the Name box.
6. In the **View Tools** toolbar, select the **Front/Right/Top** orientation.
7. Select the **Straight Edge** tool.
 - Click on the origin (where the green and red axes intersect) to place the first point of the wire antenna.
 - Click the second point anywhere along the axis directed normal to the plane of the box.
8. Select the **Select/Manipulate** tool at the top left of the **Geometry** workspace window. Right-click on the end of the wire at the origin and select Lock Position.
9. Select the Distance constraint tool to constrain the length of the wire as wireLength.

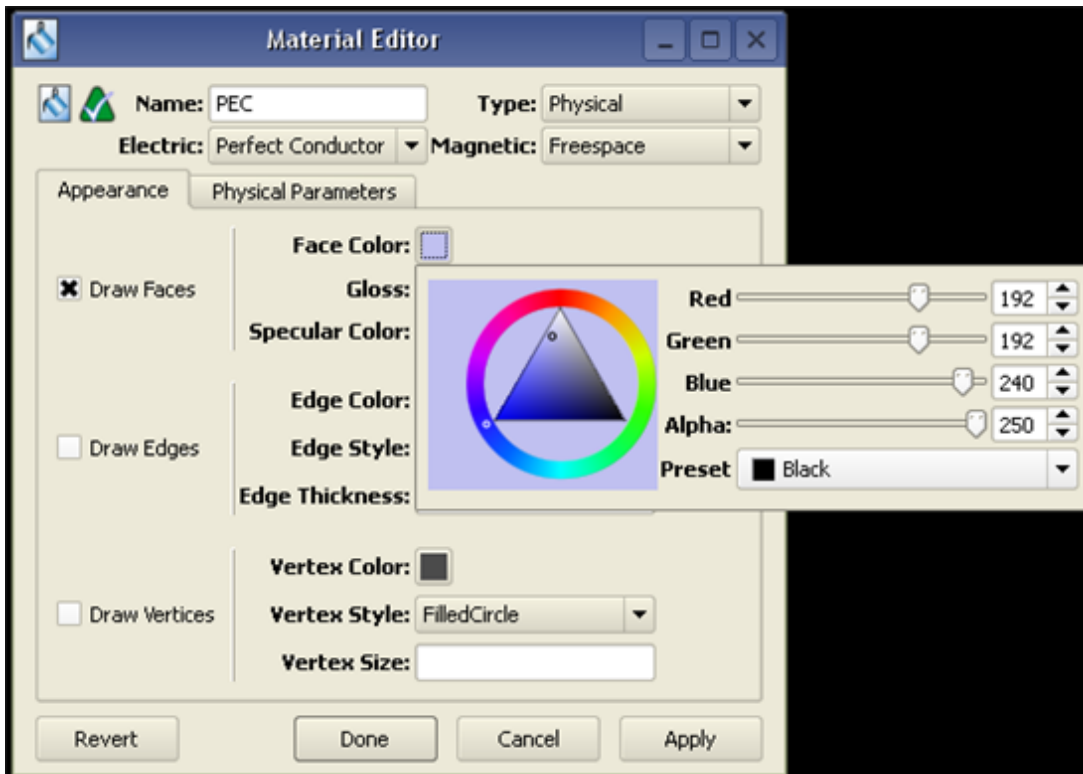


10. Click **Done** to finish the Monopole geometry.

Creating Materials

Define Material, PEC

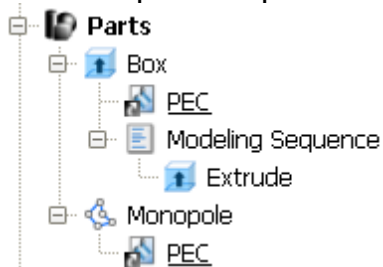
1. Right-click on the **Definitions:Materials** branch of the Project Tree and select **New Material Definition** from the context menu.
2. Set the perfect electric conductor material properties as follows:
 - **Name:** PEC
 - **Electric:** Perfect Conductor
 - **Magnetic:** Freespace
3. If desired, navigate to the **Appearance** tab to set the PEC material display color.



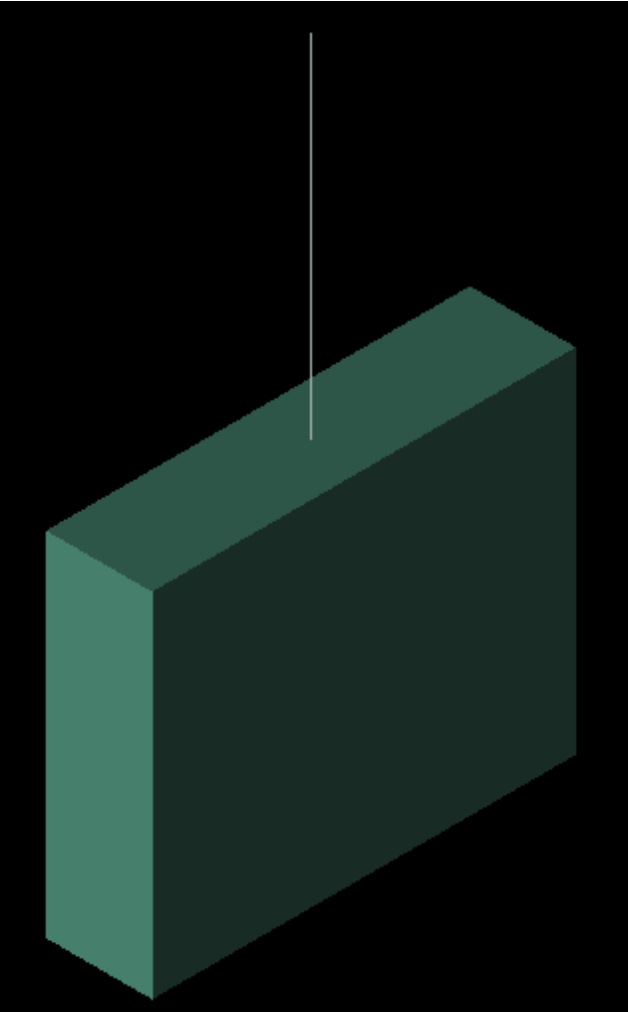
Assigning Materials

Click-and-drag the PEC material object located in the Project Tree and drop it on top of the Monopole and Extrude Box objects.

The following image shows the Project Tree after material objects have been dropped on their respective parts.



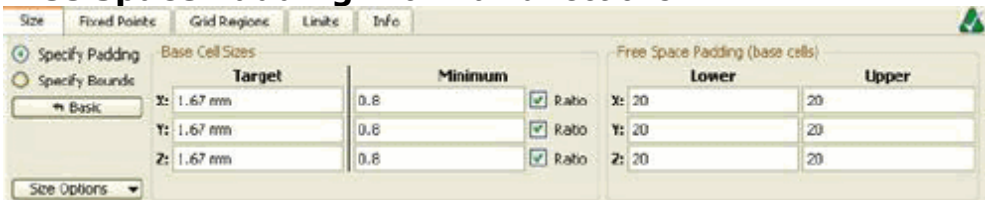
This image shows the monopole box geometry with materials applied and colors set for each.



Creating the Grid

Now, you can define the characteristics of the cells in preparation to perform an accurate calculation.

1. Double-click the **FDTD:Grid** branch to open the **Grid Tools** dialog box.
2. Set the **Size** properties of the grid as follows:
 - **Base Cell Sizes:** Target 1.67 mm, Merge 0.8, Ratio boxes checked
 - **Free Space Padding:** 20 in all directions.



3. Click **Done** to apply the grid settings.

Adding fixed points to the geometry

1. In the **Parts** branch of the Project Tree, right-click on the Extrude Box object and select **Gridding** Properties to open the **Gridding Properties Editor**.
 - Select the **Use Automatic Fixed Points** check box.
2. Click **Done** to close the editor.

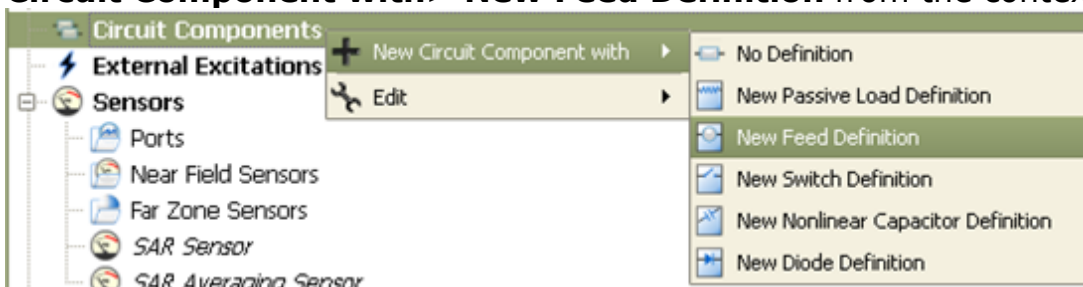
Creating a Mesh

In the **FDTD** branch of the Project tree, double-click the **Mesh** icon. This opens the mesh view and automatically create the mesh.

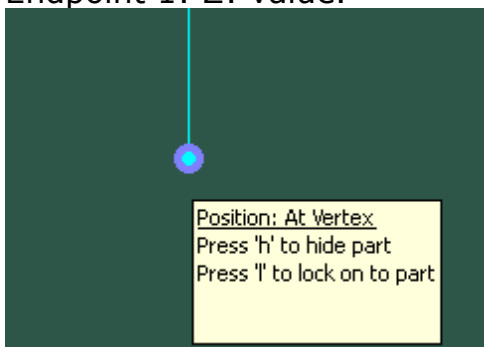
Adding a Feed

You will now add a **Feed** to the monopole geometry at the base of the **Monopole** antenna. The feed will consist of a voltage source and series 50Ω resistor connected at the base of the Monopole. Then, apply a **Ramped Sinusoid** waveform to the circuit through this feed.

1. Right-click on the **Circuit Components** branch in the Project Tree. Choose **New Circuit Component with > New Feed Definition** from the context menu.



2. Define the endpoints of the feed.
 - Endpoint 1: Zoom in to the area where the Monopole meets the Box. Using the **Pick** tool, click the point at the base of the wire. Then type in height for the Endpoint 1: Z: value.

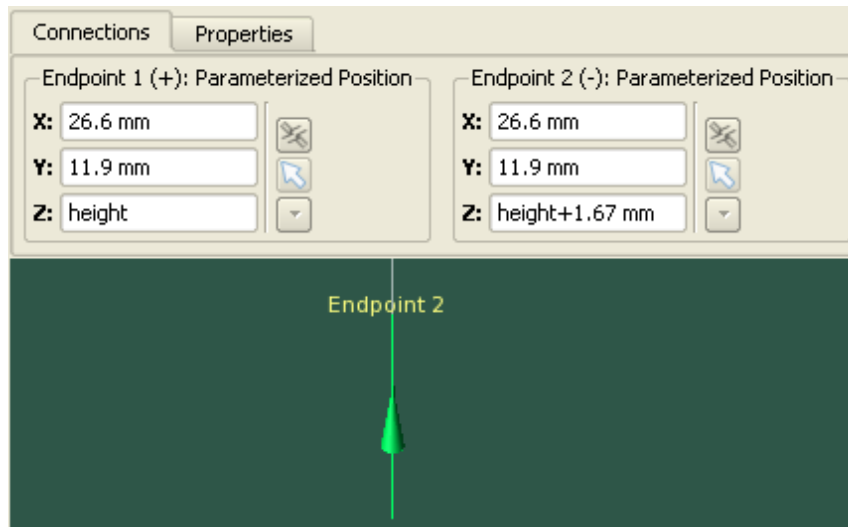


- Endpoint 2: Select the **Pick** tool under Endpoint 2, and click a higher location along the wire. Edit Endpoint 2: Z: to be height+1.67 mm. The X: and Y: locations should be the same as the values for Endpoint 1. Your X: and Y: locations may differ from the figure since it is an arbitrary rectangular sketch in the XY plane.



Note

The **Feed** is the same length as a grid cell, as defined in [Creating the Grid](#).

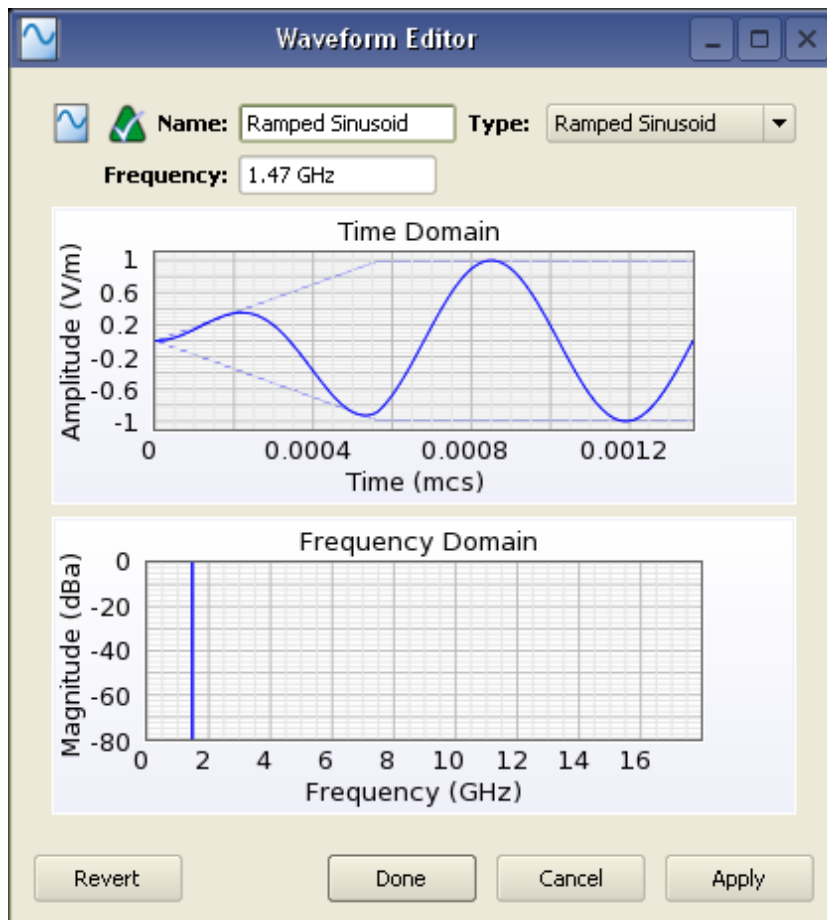


3. Navigate to the **Properties** tab, and enter the following:
 - Name: Feed
 - Component Definition: 50 ohm Voltage Source
 - Polarity: Positive
 - Select the box labeled **This component is a port.**
4. Click **Done** to add the Feed.

Editing the Waveform

An associated waveform was automatically created for the feed definition.

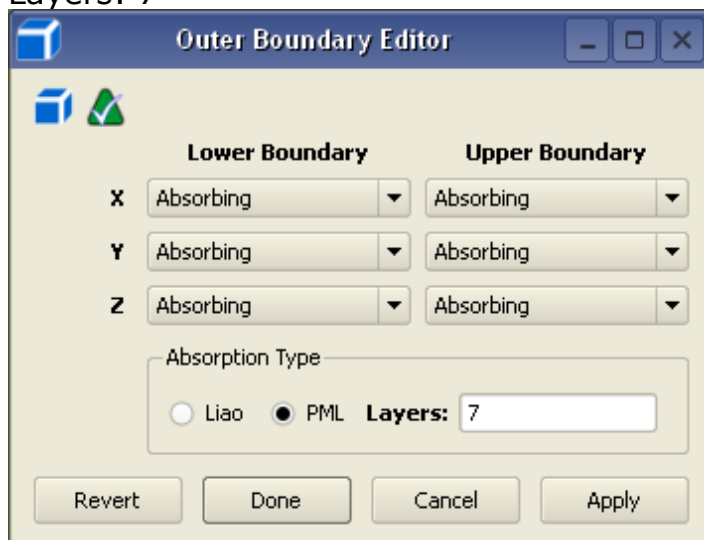
1. Navigate to the **Definitions:Waveforms** branch of the Project Tree. Double-click on the **Broadband Pulse** waveform to edit its properties.
2. Set the properties of the waveform as follows:
 - Name: Ramped Sinusoid
 - Type: Ramped Sinusoid
 - Frequency: 1.47 GHz



3. Click **Done** to apply the changes.

Defining the Outer Boundary

1. Double-click the **Simulation Domain: Boundary Conditions** branch of the Project Tree to open the Boundary Condition Editor.
2. Set the outer boundary properties as follows:
 - Boundary: Absorbing for all boundaries
 - Absorption Type: PML
 - Layers: 7



3. Click **Done** to apply the outer boundary settings.

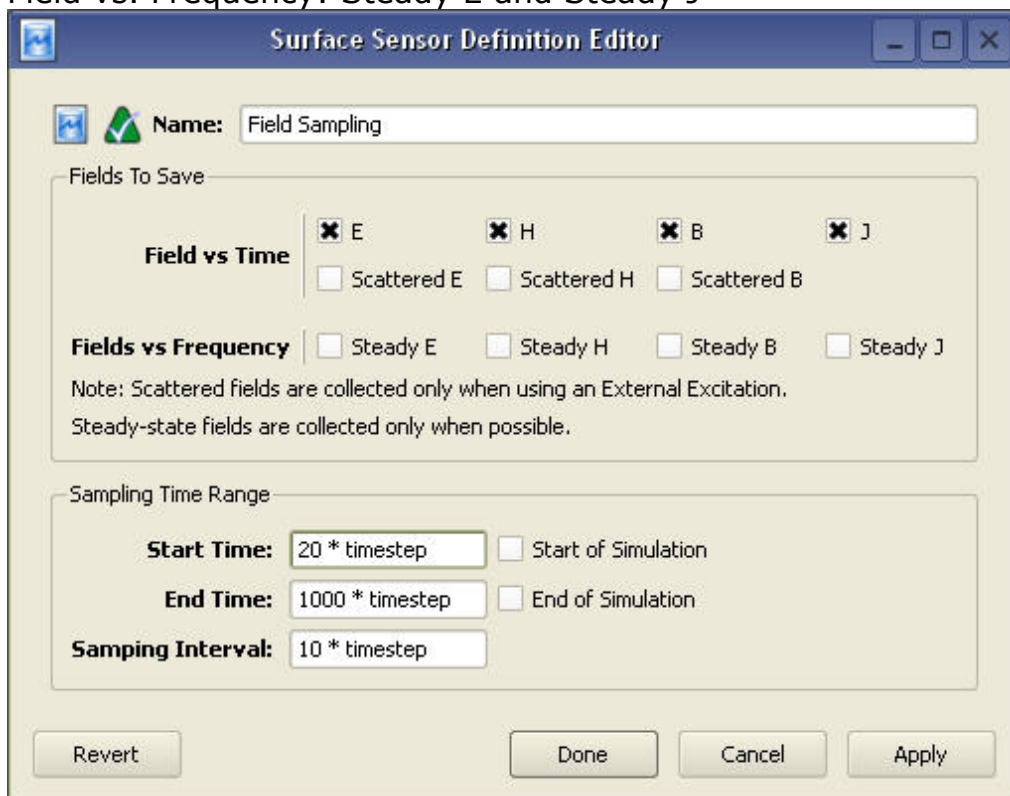
Requesting Output Data

The project already contains one port sensor named **Feed** that will request results. You may also want to collect field samplings at discrete frequencies throughout the calculation. To retrieve this data, add a **Surface Sensor**.

Adding a Surface Sensor Definition

First, create the **Surface Sensor Definition**. Right-click on the **Definitions:Sensor Data Definitions** branch of the Project Tree. Choose **New Surface Sensor Definition** from the context menu.

1. Set the properties of the surface sensor definition as follows:
 - Name: Field Sampling
 - Field vs. Frequency: Steady E and Steady J



2. Click **Done** to finish editing the Field Sampling definition.

Adding a Surface Sensor

1. Right-click the **Sensors:Near Field Sensors** branch of the Project Tree. Select **New Sensor on Model Surface** from the context menu.
2. From the Select Model tab, use the **Select** tool (at the top of the **View** Tools menu) and double-click on the box. You will know that the box is selected when it changes color.
 - Under the Properties tab, enter the following:
 - Name: Surface Sensor
 - Sensor Definition: Field Sampling
 - Sampling Method: Snapped to E-Grid

3. Click **Done** to finish editing the Surface Sensor.

Running the Calculation

If you have not already saved your project, do so by choosing **File>Save Project**. After saving the project, you can create a new simulation to send to the calculation engine.

1. Open the **Simulations** workspace window. Click **New FDTD Simulation** to set up a new simulation.
2. Under **Frequencies of Interest**, select the **Collect Steady-State Data** check box.
 - Under the **Frequencies** tab, select **Use Waveform Frequency**.
3. Select **Create and Queue Simulation** to close the dialog and run the new simulation.

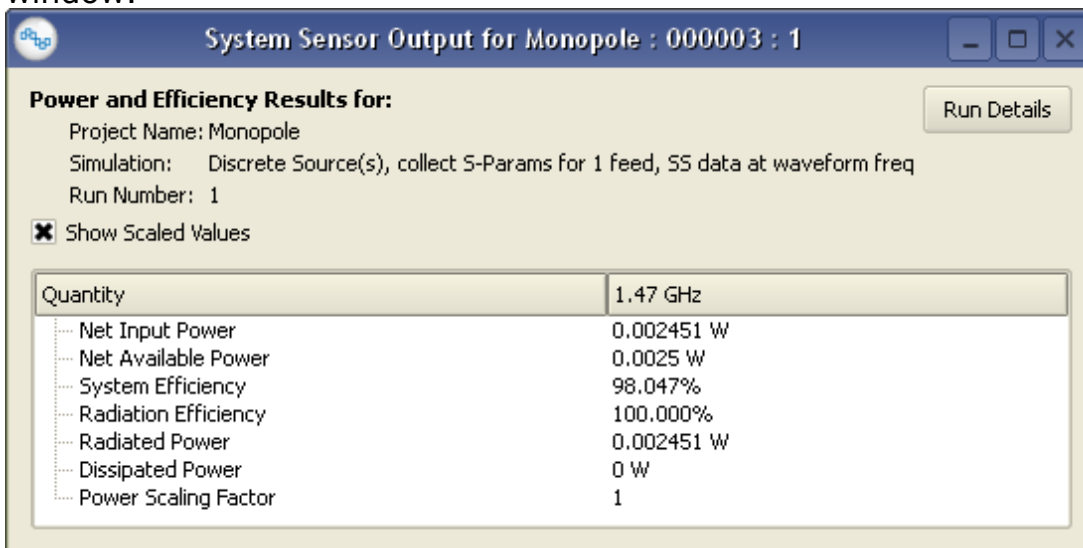
Viewing the Results

The **Output** tab of the **Simulations** workspace window displays the progress of the simulation. Once the **Status** column shows that the simulation has completed, you can view its results in the **Results** workspace window.

System Efficiency Results

First, you will view the System results.

1. To filter the list accordingly, select the following options in the columns in the top pane of the **Results** window. (You may need to change your column headings first.)
 - Output Object: System
 - Result Type: System Efficiency
2. Double-click the result. A list of power and efficiency results will appear in a dialog window.



The screenshot shows a dialog window titled "System Sensor Output for Monopole : 000003 : 1". It contains the following information:

Power and Efficiency Results for:

- Project Name: Monopole
- Simulation: Discrete Source(s), collect S-Params for 1 feed, SS data at waveform freq
- Run Number: 1
- Show Scaled Values

There is a "Run Details" button in the top right corner. Below the text is a table with two columns: "Quantity" and "1.47 GHz".

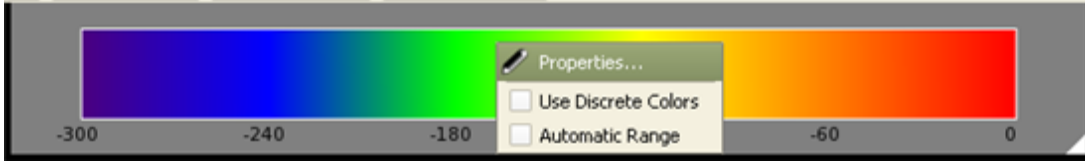
Quantity	1.47 GHz
Net Input Power	0.002451 W
Net Available Power	0.0025 W
System Efficiency	98.047%
Radiation Efficiency	100.000%
Radiated Power	0.002451 W
Dissipated Power	0 W
Power Scaling Factor	1

You can close the window when you are finished viewing the results.

Surface Current Results from the Surface Sensor

Now you can view the surface current results retrieved with the **Surface Sensor** placed on the Extrude Box.

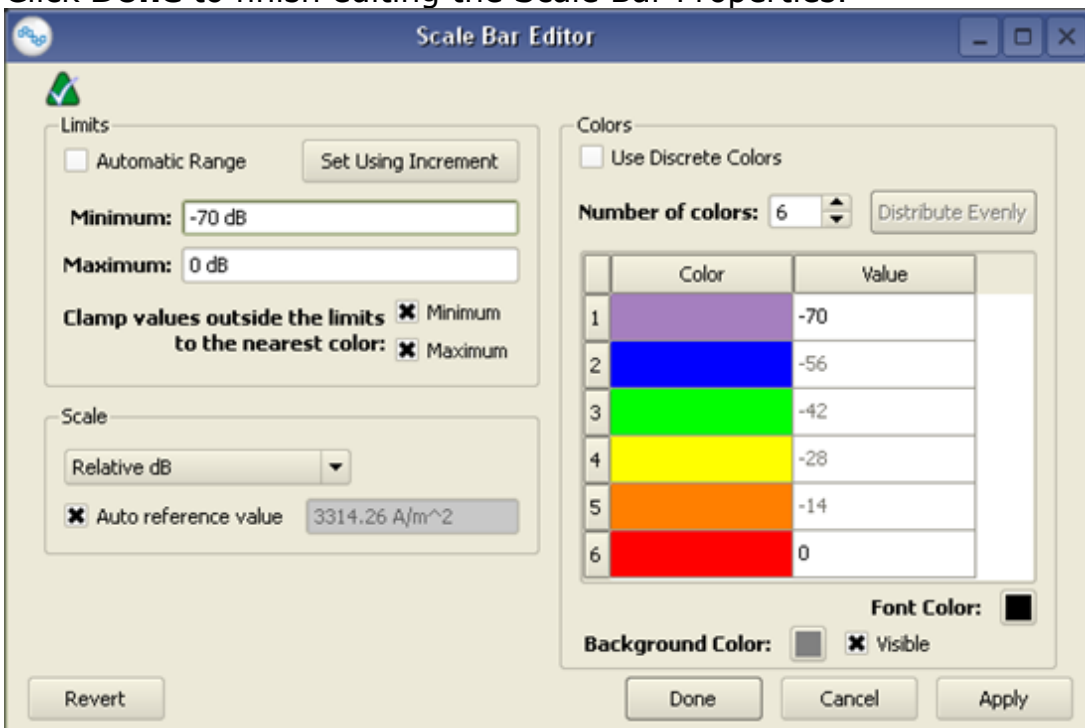
1. Select the following options in the **Results** window:
 - Output Object: Surface Sensor
 - Result Type: Conduction Current (Jc)
2. Double-click on the result. The plot will appear in the **Geometry** workspace window.
3. Right-click on the **Scale Bar** at the top of the screen, and select **Properties**.



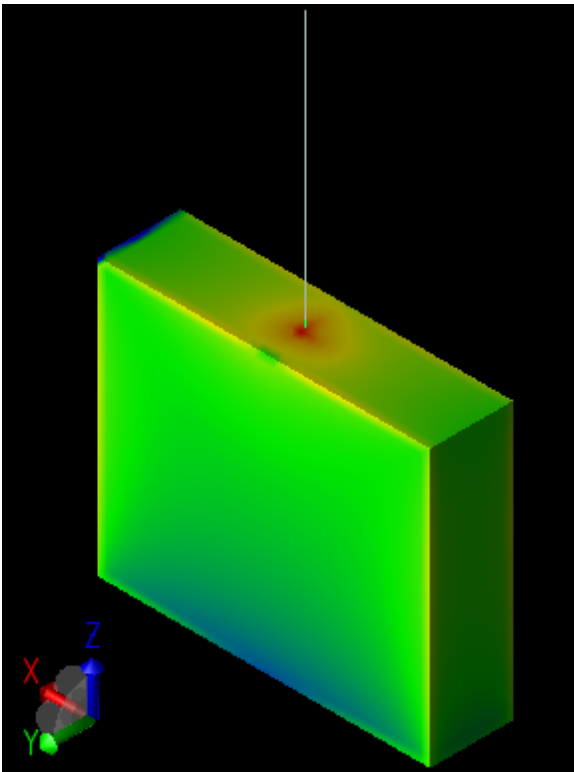
#* Under the

Limits Section, remove selection from the **Automatic Range** box.

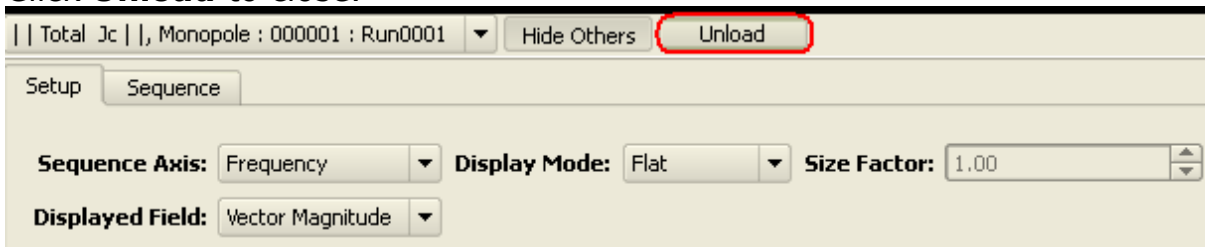
- Set the Minimum to -70 dB.
4. Click **Done** to finish editing the Scale Bar Properties.



5. You can see the surface current data on the surface of the box, as shown in the following figure:



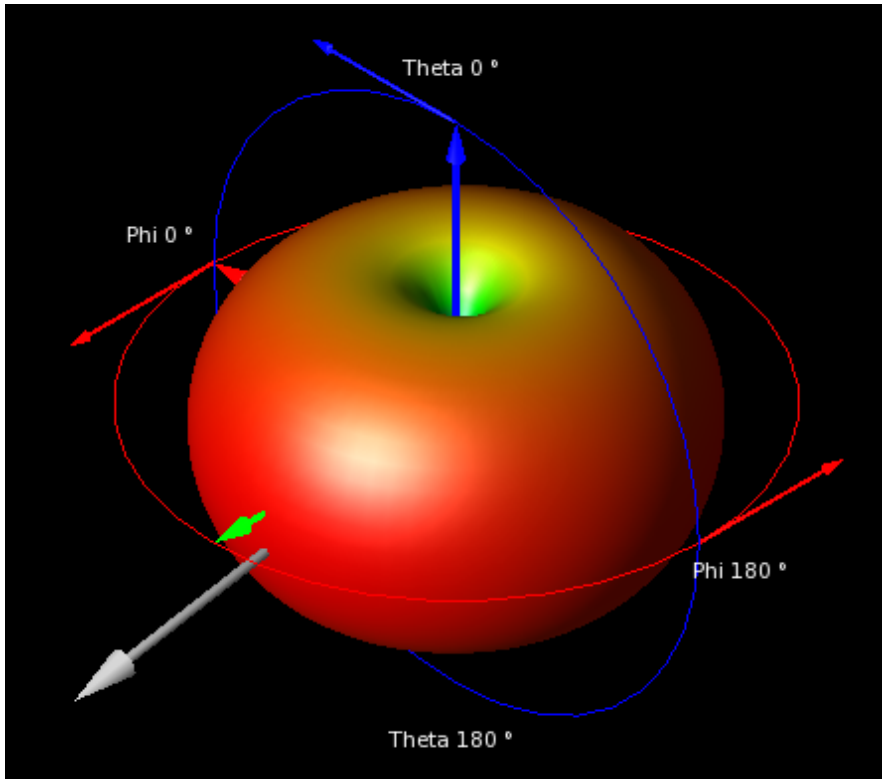
6. Click **Unload** to close.



Far Zone Post Processing

1. To begin the far zone post-processing, select the following:
 - Output Object: Raw Steady-State Far Zone Data
 - Result Type: E-Field (E)
2. Right-click the E-Field (E) result in the filtered list, and select **Post-Process Results**. The **Results** workspace window will appear to set the properties of the Far Zone sensor.
3. The default definition is sufficient for this calculation. Click Done to begin the steady state far zone data transform.
4. Select the following options in the Results window:
 - Output Object: Post Processed
 - Result Type: Gain
5. Double-click the result. The plot will appear in the Geometry workspace window.
6. Right-click on the Scale Bar at the top of the screen, and select Properties.
 - Under the Limits section, clear the Automatic Range check box. Set the Minimum to -70 dB and the Maximum to 0 dB.
 - Under the Scale section, select Relative dB and check the the Auto reference value box.
 - Click **Done** to finish editing the Scale Bar Properties.
The following figure displays the Far zone data:

FDTD Simulation



Creating a Simple SAR Calculation Simulation

In this example, a simple SAR calculation is simulated by performing the following tasks:

- Model a tissue with a dipole.
- Define the properties of the environment.
- Add a feed to the dipole and simulate its effects.
- Add a point sensor and measure E-field at the center of the tissue.
- Add an SAR sensor and retrieve SAR data.

Getting Started

To create a SAR project:

1. Select **Edit > Project Properties Editor** to open the **Project Properties Editor** window.
2. Click the **Display Units** tab.
3. Select **SI Metric** in the **Unit Set** drop-down list.
4. Change **Length** to **millimeters (mm)**. This changes the value of Unit Set to **Custom**.
5. Click **Done**.

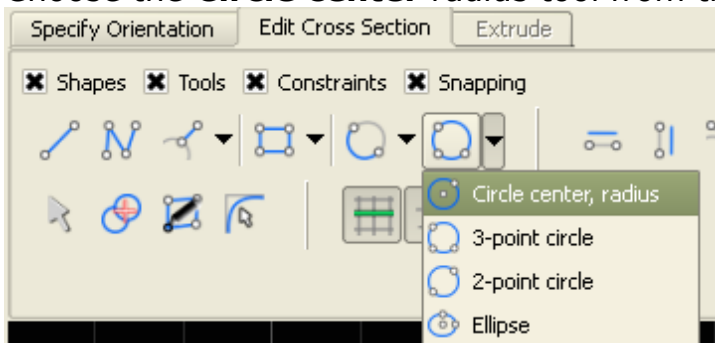
Creating the SAR Geometry

The geometry for this simple SAR calculation consists of a cylinder and a dipole antenna. The cylinder, named **Tissue**, is modeled with a simple **Extrusion**. The dipole antenna, named **Dipole**, is modeled with a **Wire Body**.

Modeling the Tissue

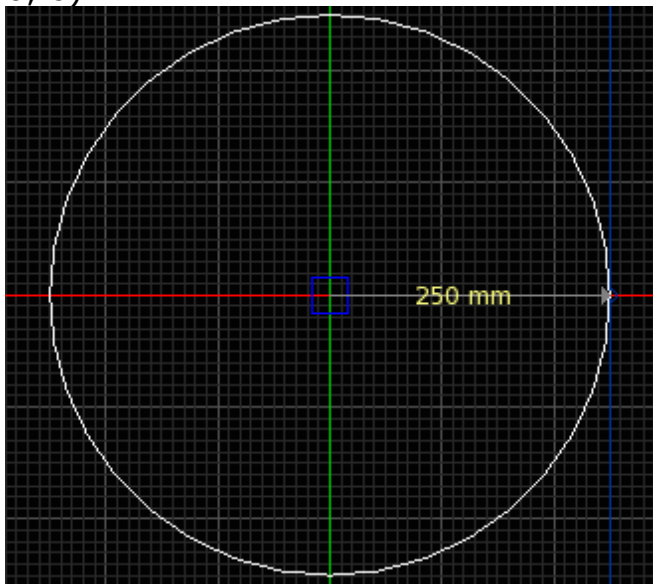
Create Tissue with a cylindrical Extrusion with a radius of 250mm in the +Z direction:

1. Right-click **Parts** and choose **Create New>Extrude**.
2. Enter **Tissue** as the Name.
3. Click the **Construction Grid** button.
 - Set the minor grid Line spacing to 10 mm.
 - Set Mouse spacing to increments of 0.1 mm.
 - Click **OK**.
4. Choose the **Circle center** radius tool from the **Shapes** toolbar.

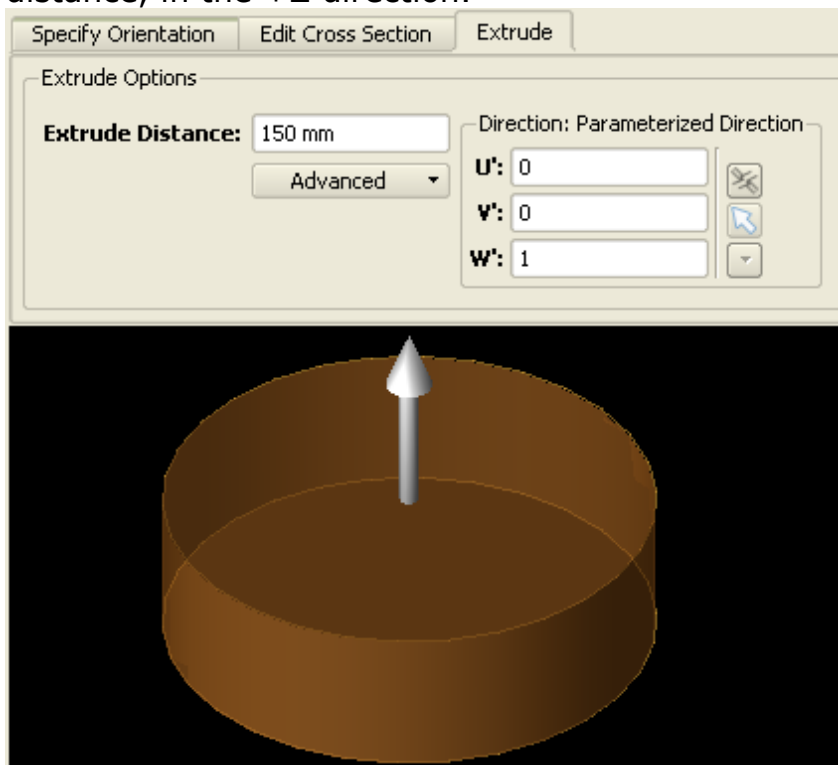


5. Draw a circle by clicking the point (0, 0, 0) and then clicking on the point (250 mm,

0, 0).



- Click the **Extrude** tab to extrude the cylindrical region. Enter 150 mm as the distance, in the +Z direction.



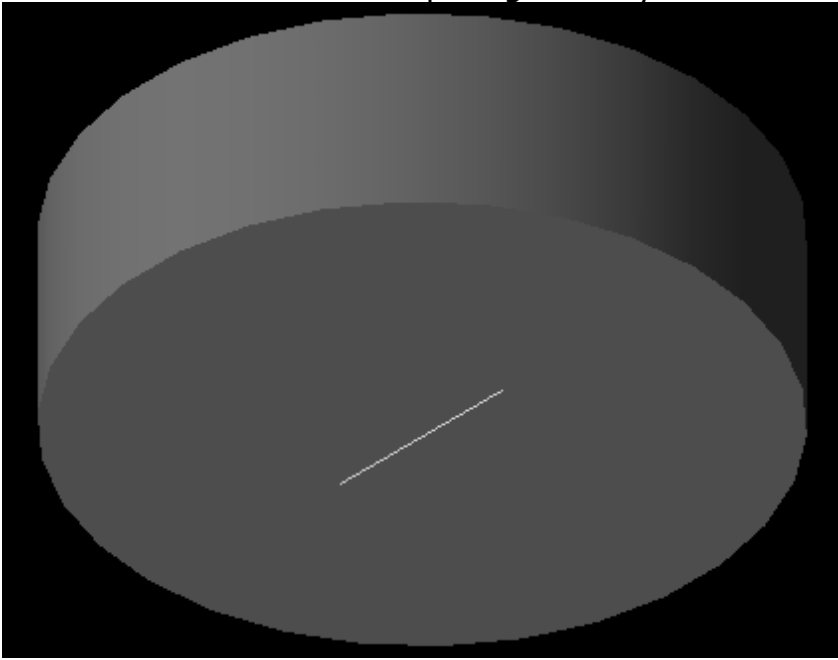
- Click **Done** to complete the Tissue geometry.

Modeling the Dipole

The Dipole is created with a 100mm Wire Body object that is centrally located just under the Tissue cylinder.

- Right-click **Parts** and choose **Create New > Wire Body**.
- Click the **Specify Orientation** tab.
- Set the origin to (0, 0, -10 mm).
- Click the **Edit Wire Body** tab.
In the Name box, type **Dipole**.

- 5.
6. Select the **Straight Edge** tool.
7. In the **View Tools** toolbar, select the Left (+X) orientation.
 - Press the **Tab** key to display the creation dialog for the first point. Enter (-50 mm, 0 mm) and click **OK**.
 - Press the **Tab** key to display the creation dialog for the second point. Enter (U : 50, V : 0, Length :100 mm) and click **OK** to complete the Dipole.
8. Click **Done** to finish the Dipole geometry.

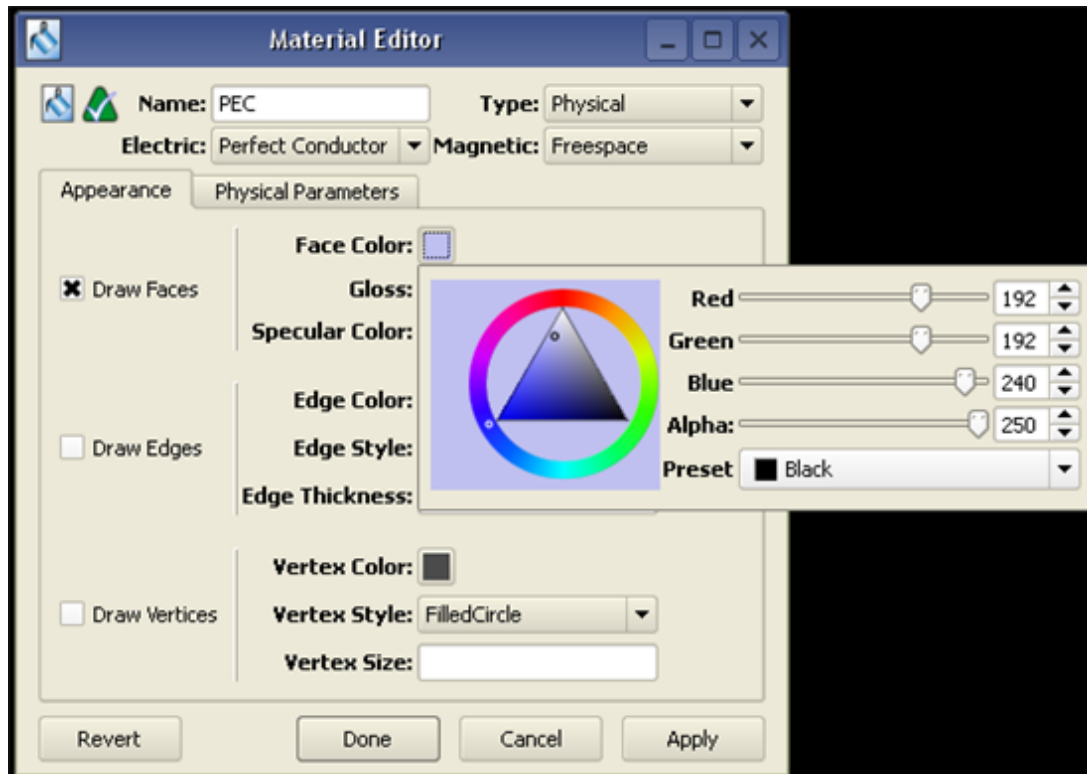


Creating Materials

Define Material, PEC

To define the PEC material:

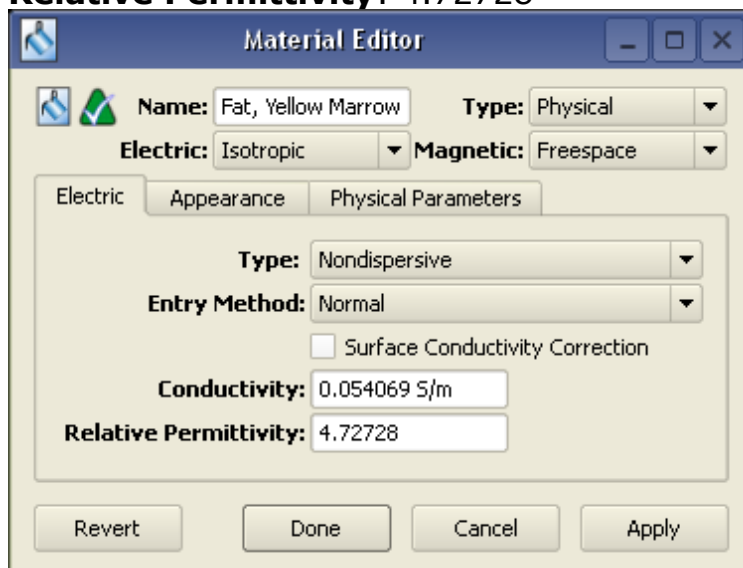
1. Right-click **Materials** in the Definitions list and choose **New Material Definition**.
2. Set the perfect electric conductor material properties as follows:
 - **Name:** PEC
 - **Electric:** Perfect Conductor
 - **Magnetic:** Freespace



3. If required, navigate to the **Appearance** tab to set the PEC material display color.

Defining Material as Fat, Yellow Marrow

1. Right-click **Materials** in the Definitions list and choose **New Material Definition**.
2. Set the material properties as follows:
 - **Name:** Fat, Yellow Marrow
 - **Electric:** Isotropic
 - **Magnetic:** Freespace
 - Under the Electric tab:
 - **Type:** Nondispersive
 - **Entry Method:** Normal
 - Surface Conductivity Correction
 - **Conductivity:** 0.054069 S/m
 - **Relative Permittivity:** 4.72728



- Under the **Physical Parameters** tab, enter 943 kg/m³ as the Density.

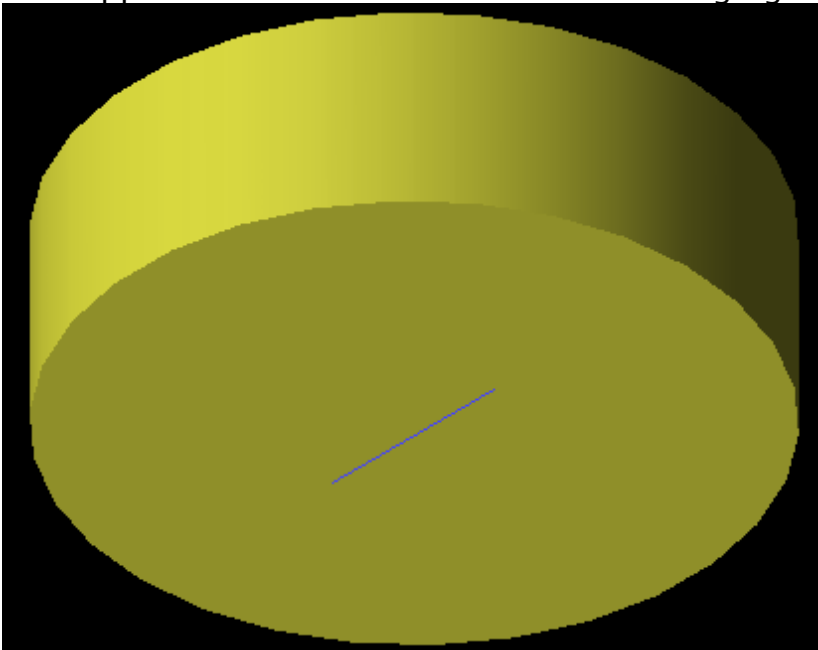
Note

If the FEM option is enabled in EMPro, you will not be able to enter the value of Density. You need to disable FEM to enter the density value.

- Navigate to the **Appearance** tab and assign the Fat, Yellow Marrow material a new color to distinguish it from PEC.
- Click **Done** to add the new material Fat, Yellow Marrow.

Assigning Materials

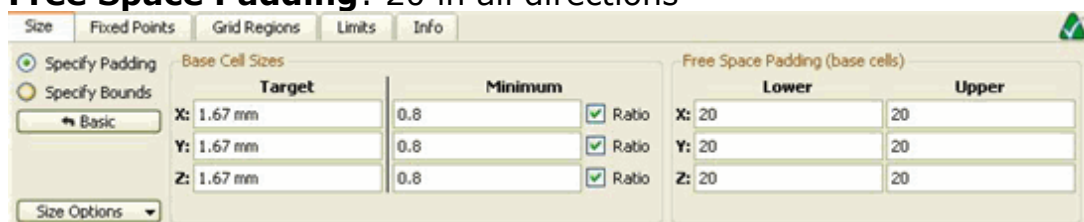
- Click and drag the PEC material object located in the Project Tree and drop it on top of the Dipole object in the Parts branch of the tree.
- Assign the Fat, Yellow Marrow material to the Tissue object. The finished geometry with applied materials is seen in the following figure.



Creating the Grid

Now, you can define the characteristics of the cells in preparation to perform an accurate calculation.

- Double-click **Grid** in the FDTD list to open the **Grid Tools** dialog box.
- Navigate to the **Size** tab:
 - Click the **Advanced** button (if necessary) to access Base Cell Sizes and Free Space Padding. Set the Size properties of the grid as follows:
 - Base Cell Sizes:** Target 1.67mm, Minimum 0.5, Ratio boxes checked
 - Free Space Padding:** 20 in all directions



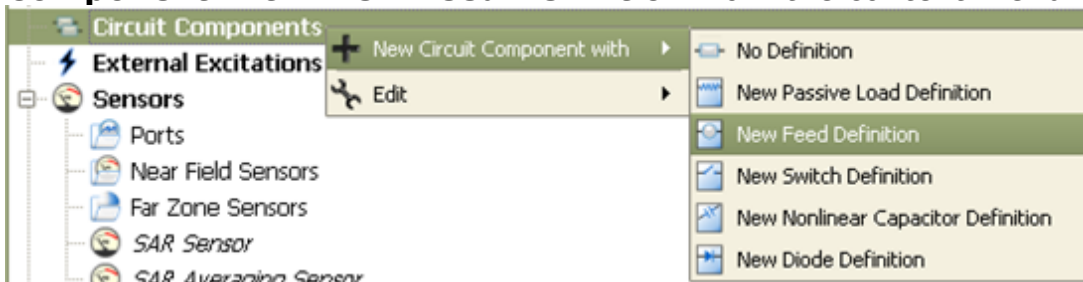
3. Click **Done** to apply the grid settings.

Creating a Mesh

In the **FDTD** branch of the Project Tree, double-click **Mesh**. This opens the mesh view and automatically create the mesh.

Adding a Feed to the Dipole Wire

1. Right-click the **Circuit Components** branch of the Project Tree. Choose **New Circuit Component with>New Feed Definition** from the context menu.

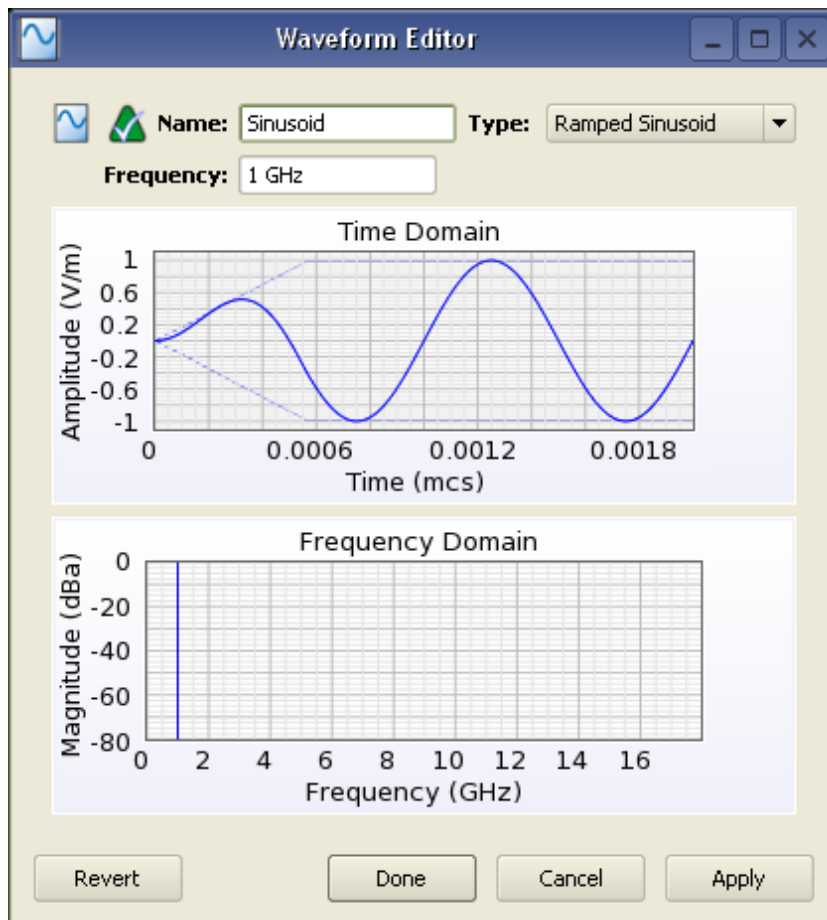


2. Define the endpoints of the feed.
 - Endpoint 1: X: -0.5 mm, Y: 0 mm, Z: -10 mm
 - Endpoint 2: X: 0.5 mm, Y: 0 mm, Z: -10 mm
3. Navigate to the **Properties** tab, and enter the following:
 - **Name:** Feed
 - **Component Definition:** 50 ohm Voltage Source
 - **Direction:** Auto
 - **Polarity:** Positive
 - Select the checkbox labeled **This component is a port.**
4. Click **Done** to add the Feed.

Editing the Waveform

An associated waveform was automatically created for the feed definition.

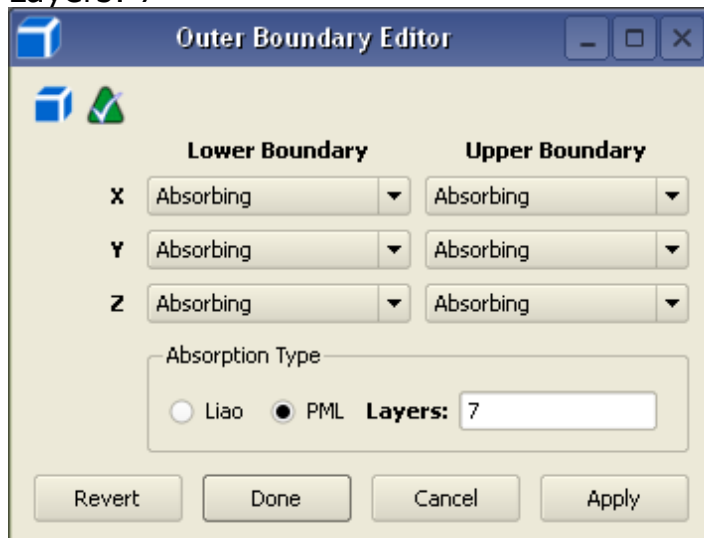
1. Navigate to the **Definitions:Waveforms** branch of the Project Tree. Double-click the Broadband Pulse waveform to edit its properties. Set the properties of the waveform as follows:
 - Name: Sinusoid
 - Type: Ramped Sinusoid
 - Frequency: 1 GHz



2. Click **Done** to apply the changes.

Defining the Outer Boundary

1. Double-click the **FDTD :Outer Boundary** branch of the Project Tree to open the Outer Boundary Editor.
2. Set the outer boundary properties as follows:
 - Boundary: Absorbing for all boundaries
 - Absorption Type: PML
 - Layers: 7



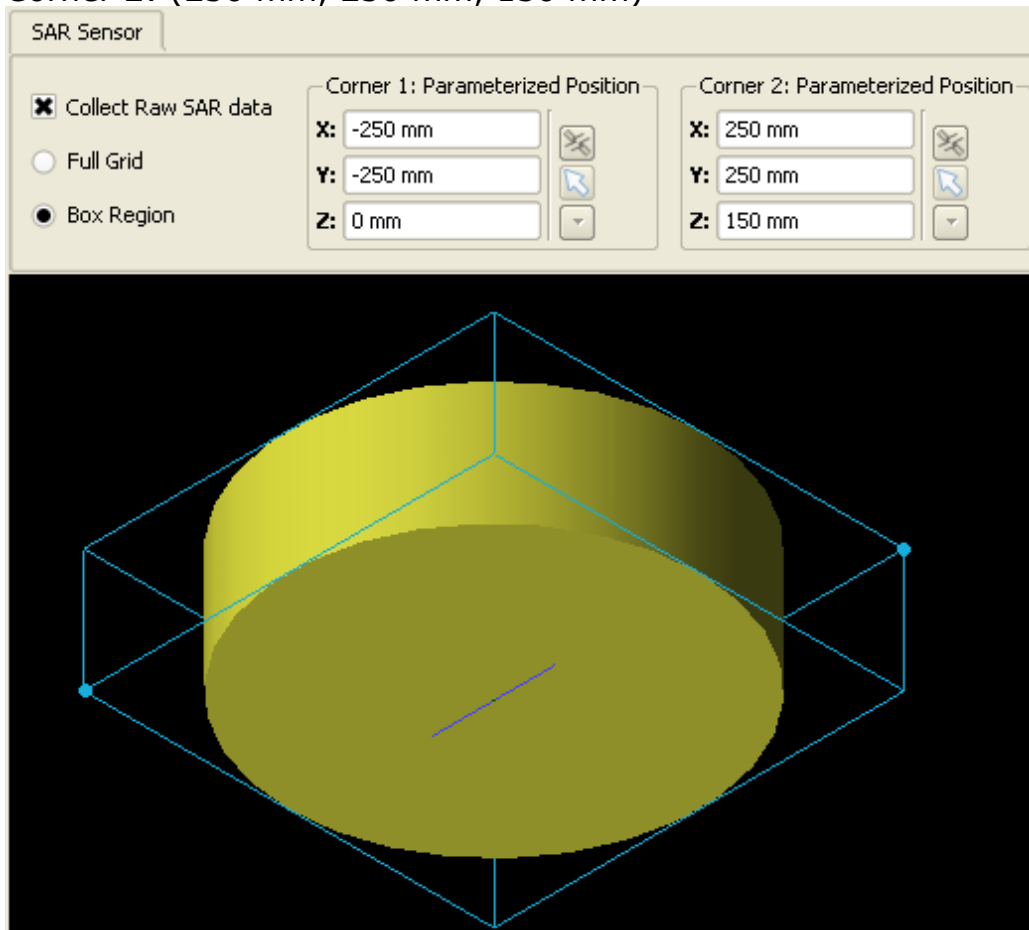
3. Click **Done** to apply the outer boundary settings.

Requesting Output Data

The project already contains one port sensor named Feed that will request results. You may also want to collect SAR results by adding an SAR Sensor.

Adding an SAR Sensor

1. Right-click the **Sensors:SAR Sensors** branch of the Project Tree. Select **Properties** from the context menu.
2. Select the **Collect Raw SAR Data** box.
3. Select **Box Region**, and enter the following coordinates:
 - Corner 1: (-250 mm, -250 mm, 0)
 - Corner 2: (250 mm, 250 mm, 150 mm)



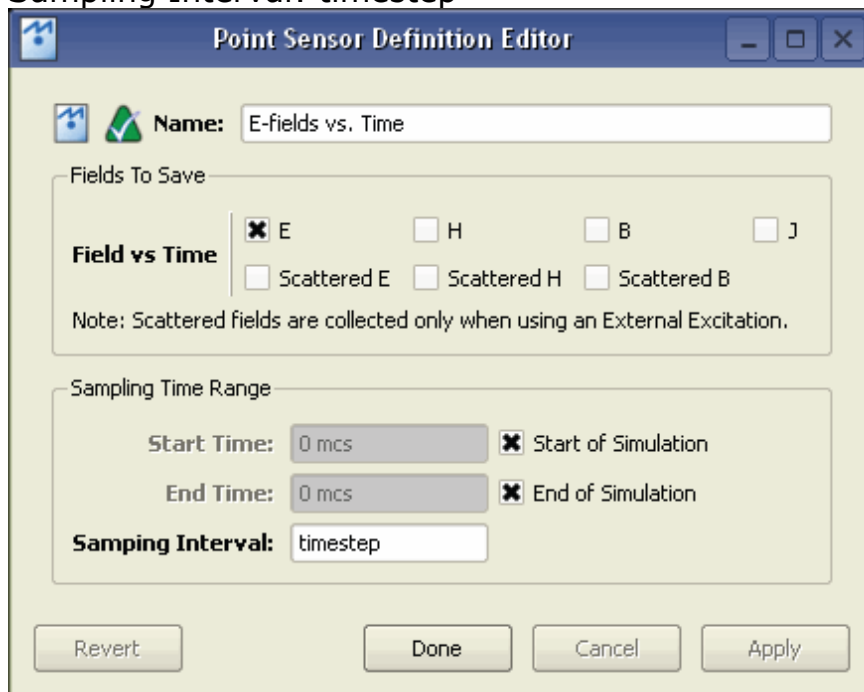
4. Click **Done**.

Adding a Point Sensor Definition

A Point Sensor may be saved inside the Tissue object to monitor the convergence of the fields during the calculation. First, you will create its definition.

1. Right-click the **Definitions:Sensor Data Definitions** branch of the Project Tree. Choose **New Point Sensor Definition** from the context menu.
2. Set the properties of the surface sensor definition as follows:
 - Name: E-field vs. Time
 - Field vs. Time: E

- Sampling Interval: timestep



3. Click **Done**.

Adding a Point Sensor

1. Right-click the **Sensors:Near Field** Sensors branch of the Project Tree. Select **New Point Sensor** from the context menu.
 - Enter its Location as (0, 0, 75 mm).
 - Under the Properties tab, enter the following:
 - Name: E-field at Tissue Center
 - Sensor Definition: E-field vs. Time
 - Sampling Method: Snapped to E-Grid
2. Click **Done** to finish editing the E-field at Tissue Center Sensor.

Running the Calculation

If you have not already saved your project, do so by selecting **File>Save Project**. After saving the project, you can create a new simulation to send to the calculation engine.

1. Open the **Simulations** workspace window. Click the **Create Simulation** button in the upper-left to set up a new simulation.
2. Type a descriptive Name for the simulation, such as **Tissue Cylinder exposed to 1 GHz Dipole, SAR Saved**.
3. Under **Frequencies of Interest**, select the Collect Steady-State Data checkbox.
4. Under the Frequencies tab, select **Use Waveform Frequency**.
Most of the default settings are sufficient. Navigate to the **Specify Termination Criteria** tab. Set up the termination criteria as follows:
 - Maximum Simulation Time: 10000 * timestep
 - Detect Convergence: Checked
 - Threshold: -20 dB
5. Select **Create and Queue Simulation** to close the dialog box and run the new

simulation.

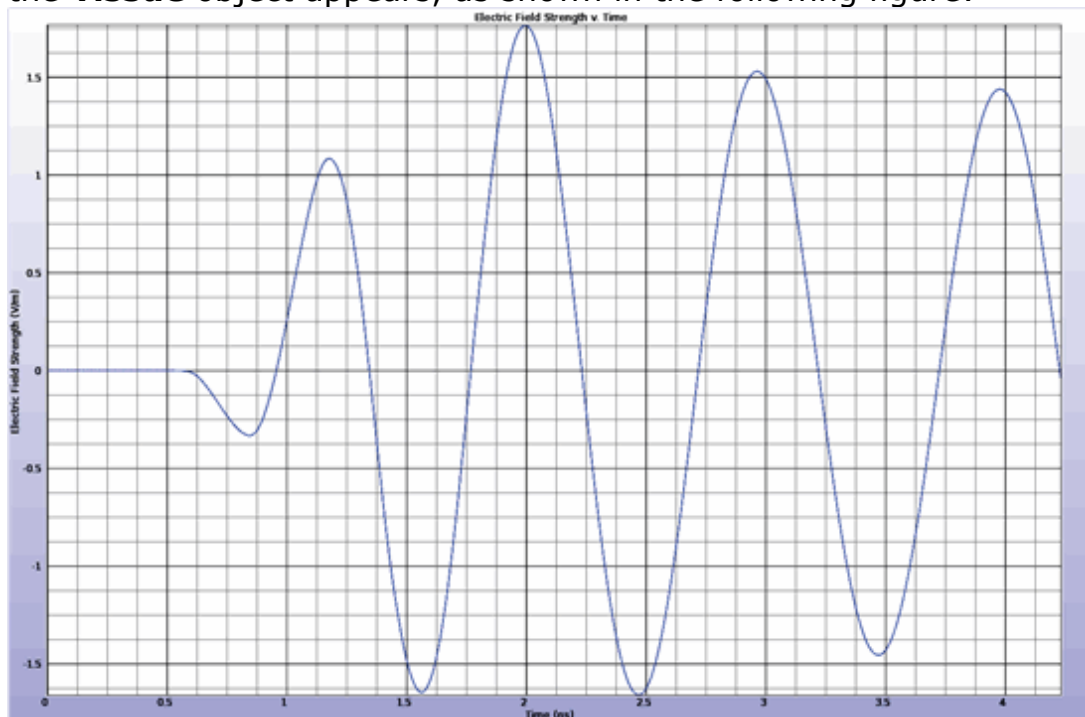
Viewing the Results

The **Output** tab of the Simulations workspace window displays the progress of the simulation. After the Status column shows that the simulation is complete, you can view its results in the **Results** workspace window.

E-field Results

You can view the E-field results retrieved from the center of the Tissue.

- To filter the list accordingly, select the following options in the columns in the top pane of the **Results** window. You may need to change your column headings first:
 - **Sensor:** E-field at Tissue Center
 - **Result Type:** E-field (E)
- Right-click the result and select **Create Line Graph**.
- Select **X** as the Component, and click **View**. The plot of the E-field at the center of the **Tissue** object appears, as shown in the following figure:

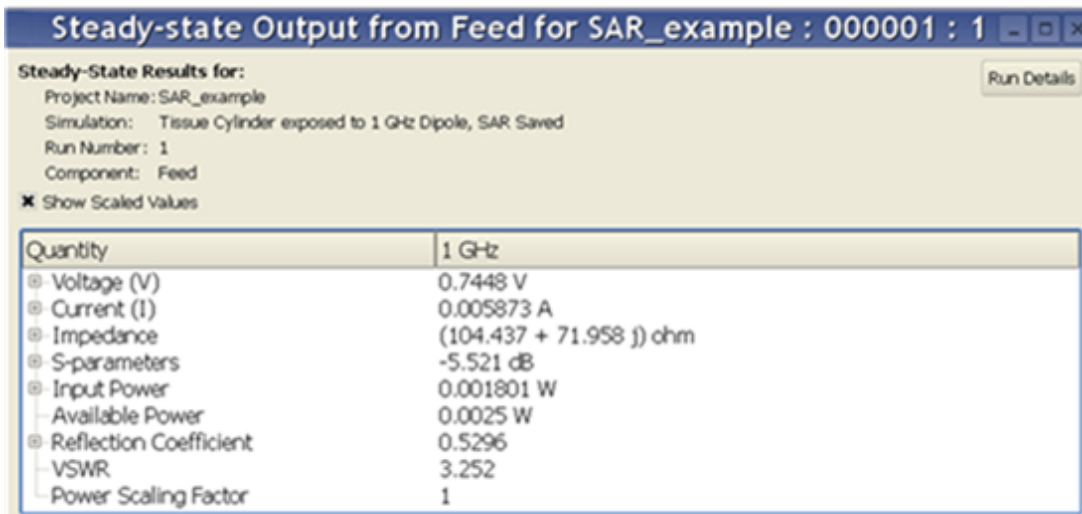


You can view the data before the simulation is complete. The plot is updated automatically as more data is computed.

System Efficiency Results

Now, you can view data from the point.

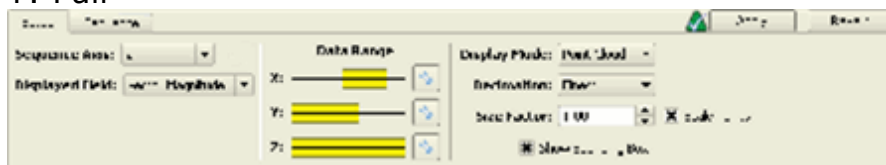
- To view the Feed results, select the following:
 - **Sensor:** Feed
 - **Result Type:** S-Parameters
- Double-click the results under the **Discrete** domain. The following results will appear showing the impedance at the feed, the input power delivered, and the return loss.



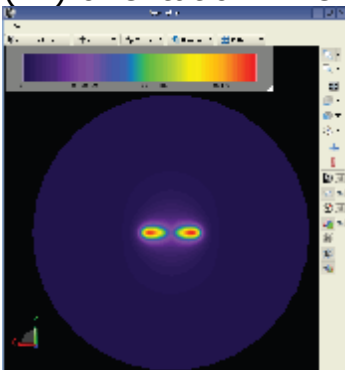
SAR Sensor Data

You can load the SAR data into the field viewer.

- To view the SAR sensor data, select the following:
 - **Sensor:** SAR Sensor (Raw)
 - **Result Type:** SAR (Specific Absorption Rate)
- Double-click the result in the filtered list. The plot will appear in the Geometry workspace window.
- Under the **Setup** tab adjust the following settings:
 - Sequence Axis: Z
 - Decimation: Finest
 - Under Axis Ranges:
 - Frequency: 1 GHz
 - X: Full
 - Y: Full



- Click **Apply** to finish editing the SAR sensor setup.
- Toggle the Parts Visibility to turn off the display of the geometry, and select the Top (-Z) orientation. The resulting image should appear.



6. Click the **Sequence** tab and then click **Play** to view a movie of the SAR slices. To increase the speed of the movie, change the **Decimation** on the **Setup** tab to **Normal**.
7. To review the SAR statistics of the peak and average SAR, click the **Statistics** tab.

Validating SAR Calculations

In this example, SAR calculations are validated by performing the following tasks:

- Model a tissue-simulating liquid with a dipole.
- Define the properties of the environment.
- Add a feed to the dipole and simulate its effects.
- Add a point sensor and measure E-field at the center of the liquid.
- Add SAR sensors and retrieve raw and averaged SAR data.

Getting Started

To validate a SAR calculation, create a new project:

1. Select **Edit > Project Properties Editor** to open the **Project Properties Editor** window.
2. Select the **Display Units** tab.
3. Select **SI Metric** in the **Unit Set** drop-down list.
4. Change **Length** to **millimeters (mm)**. This changes the value of Unit Set to **Custom**.
5. Click **Done**.

Creating the Geometry

The geometry for this example consists of a Flat Phantom, Phantom Shell, and a dipole made of two cylinders.

Modeling the Flat Phantom

First, you will create the rectangular extrusion named Flat Phantom which represents the tissue simulating liquid used for SAR measurements. You will perform the simulation at 835MHz, so the phantom dimensions will be 220x150mm with an extrusion in the +Z direction of 150mm.

1. Right-click the Parts branch of the Project Tree. Choose **Create New > Extrude** from the context menu.
2. Name the object by typing **Flat Phantom** in the Name text box.
3. Choose **View > Standard Views > Top(-Z)** orientation.
4. Choose the **Rectangle** tool from the Shapes toolbar.
5. Click the mouse on the origin of the coordinate system.
6. Press **Tab** to display the creation dialog for the second point. Enter **225mm,150mm** and click **OK** to complete the rectangle.
7. Navigate to the **Extrude** tab to extrude the rectangular region. Enter the **Extrude Distance** of 150mm.
8. Click **Done** to finish the Flat Phantom geometry.

Modeling the Phantom Shell

Next, you will create the rectangular extrusion named Phantom Shell. This shell is a plastic

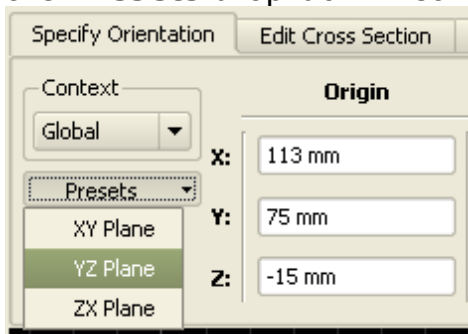
vessel that will hold the simulating liquid. For this simulation, you need to add only the bottom of the vessel that separates the liquid from the dipole source. This shell size will match the phantom size in X and Y, and have a thickness of 2mm.

1. Right-click the **Parts** branch and choose **Create New > Extrude** from the context menu.
2. Under the Specify Orientation tab, define the origin at (0, 0, 2mm).
3. Under the **Edit Cross Section** tab, type **Phantom Shell** in the Name text box.
4. Choose **View > Standard Views > Bottom(+Z)** orientation.
5. Choose the **Rectangle** tool from the Shapes toolbar.
6. Trace the new cross-section over the existing cross-section (of the flat phantom) since they are of equal width and length.
7. Navigate to the **Extrude** tab to extrude the rectangular region a distance of -2mm.
8. Click **Done** to finish the Phantom Shell geometry.

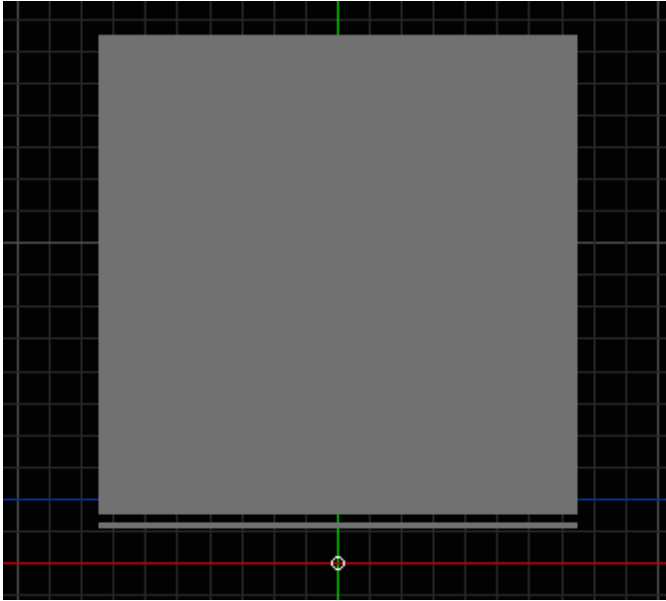
Modeling the Dipole

Now you will create the dipole geometry, which comprises two cylindrical extrusions. Typically the dipole will have a balun structure as well, but we will omit that for simplicity in this example. The dipole will have a radius of 1.8mm and a length of 161mm.

1. Right-click the **Parts** branch and choose **Create New > Extrude** from the context menu.
2. Under the **Specify Orientation** tab, define the origin at (113mm, 75mm, -15mm).
 - Redefine the orientation of the sketching plane by selecting the **YZ** Plane under the **Presets** drop-down list.



3. Under the **Edit Cross Section** tab, type **Cylinder1** in the Name box.
4. Choose **View > Standard Views > Left(+X)** orientation.
5. Choose the **Circle center** radius tool from the Shapes toolbar.
 - Click the mouse on the origin of the coordinate system.
 - Press **Tab** to display the creation dialog for the radius. Enter 1.8mm and click **OK**.

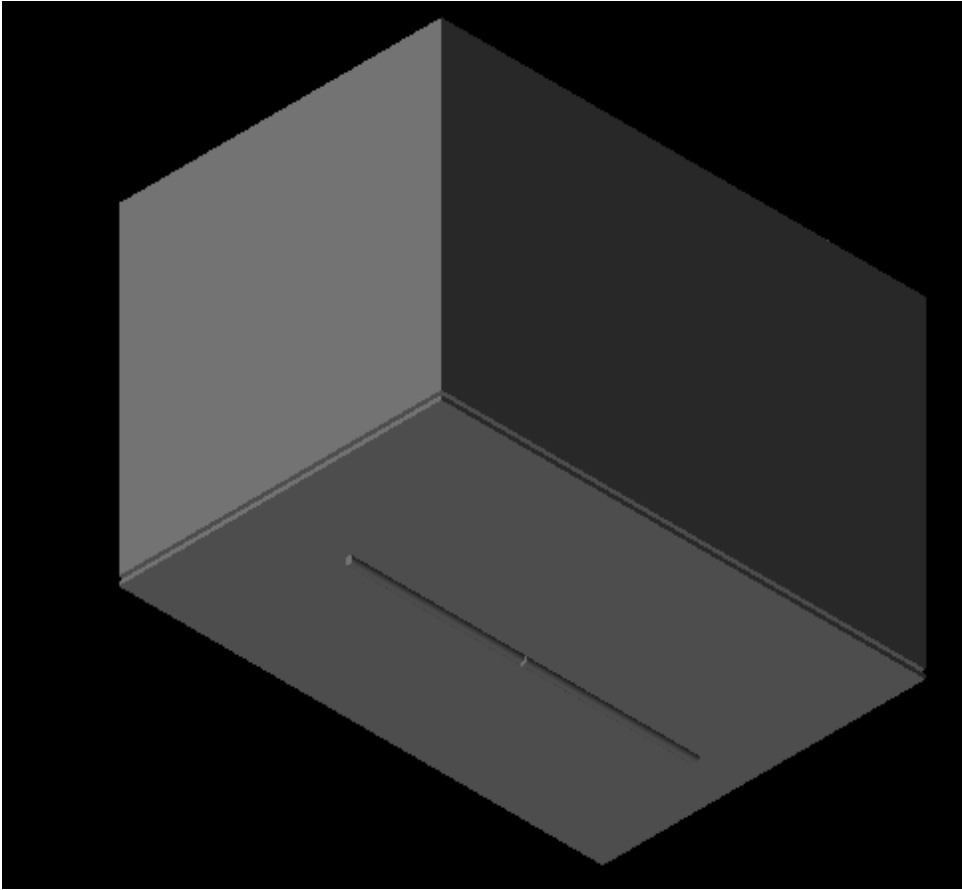


6. Navigate to the **Extrude** tab to extrude the cylinder. Enter a distance of 80mm.
7. Click **Done** to finish the Cylinder1 geometry.

Create the second extrusion

Now you will create the second part of the dipole, Cylinder2.

1. Right-click the **Parts** branch and choose **Create New>Extrude**.
 2. Under the Specify Orientation tab, define the origin at (32mm, 75mm, -15mm).
 - Redefine the orientation of the sketching plane by selecting the YZ Plane under the Presets drop-down.
 3. Under the **Edit Cross Section** tab, type **Cylinder2** in the **Name** text box.
 4. In the **View Tools** toolbar, select the **Right** (-X) orientation.
 5. Choose the **Circle center, Radius** tool from the **Shapes** toolbar.
 - Click the mouse on the origin of the coordinate system.
 - Click **Tab** to display the creation dialog for the radius. Enter 1.8mm and click **OK**.
 6. Navigate to the **Extrude** tab to extrude the cylinder. Enter a distance of 80mm.
 7. Click **Done** to finish the Cylinder2 geometry.
- The following figure displays a view of the finished geometry before materials are added.

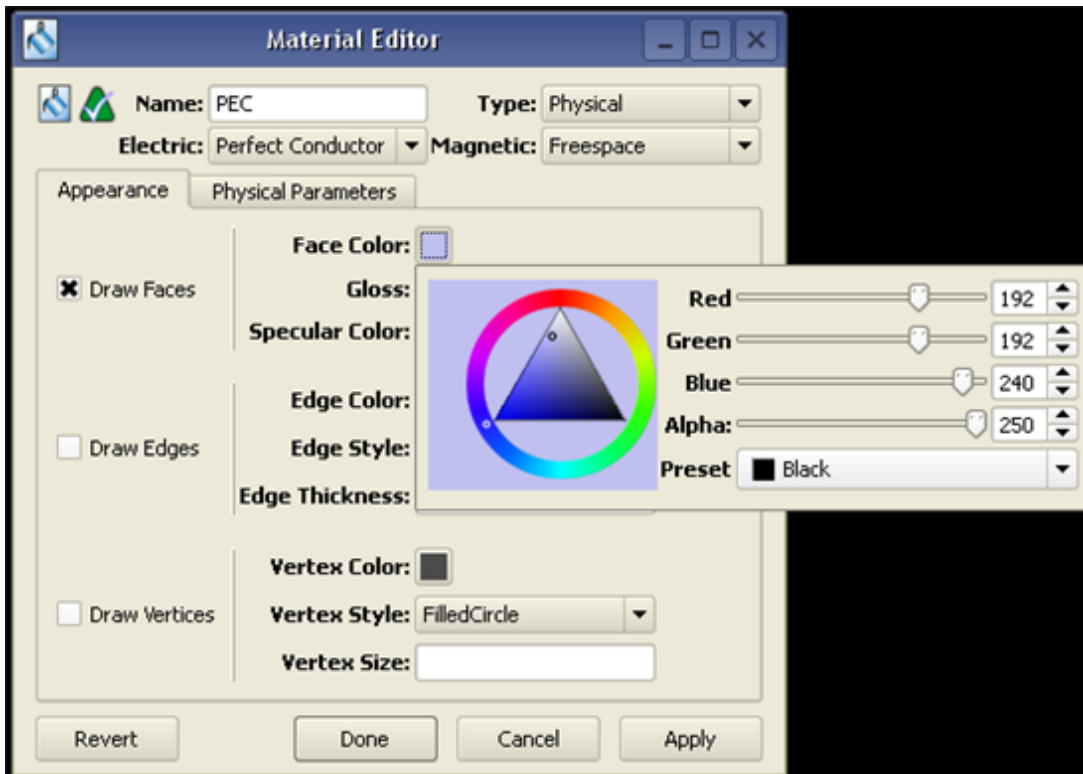


Creating Materials

After creating four new objects, you will assign materials to them. Cylinder1 and Cylinder2 will be perfect electric conductors, PEC. The Flat Phantom and Phantom Shell objects will be isotropic materials named Phantom Liquid and Phantom Shell, respectively.

Define material, PEC

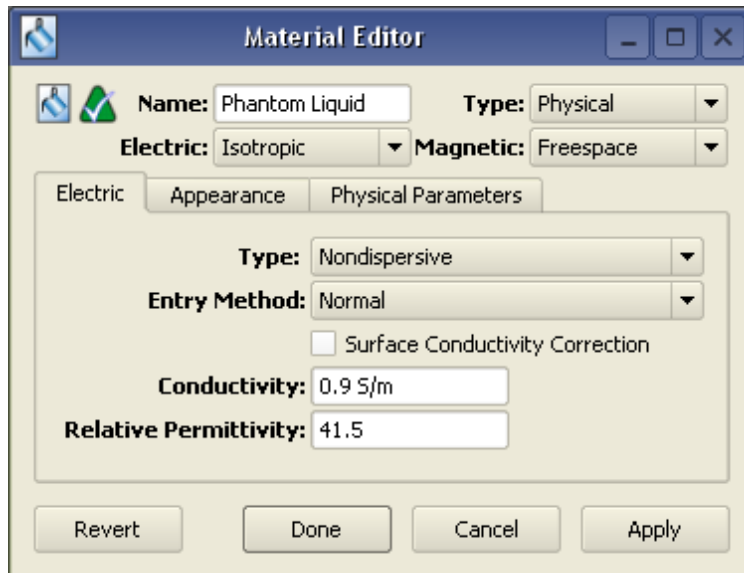
1. Right-click the **Definitions:Materials** branch of the Project Tree. Choose **New Material Definition** from the context menu.
2. Set the perfect electric conductor material properties as follows:
 - Name: PEC
 - Electric: Perfect Conductor
 - Magnetic: Freespace
3. If desired, navigate to the Appearance tab to set the display color of the PEC material.



Define Material, Phantom Liquid

1. Right-click the **Definitions:Materials** branch of the Project Tree and select **New Material Definition**.
2. Set the material properties as follows:
 - Name: Phantom Liquid
 - Electric: Isotropic
 - Magnetic: Freespace
 - Under the Electric tab:
 - Type: Nondispersive
 - Entry Method: Normal
 - Conductivity: 0.9 S/m
 - Relative Permittivity: 41.5

Editing the color of the Phantom Liquid material



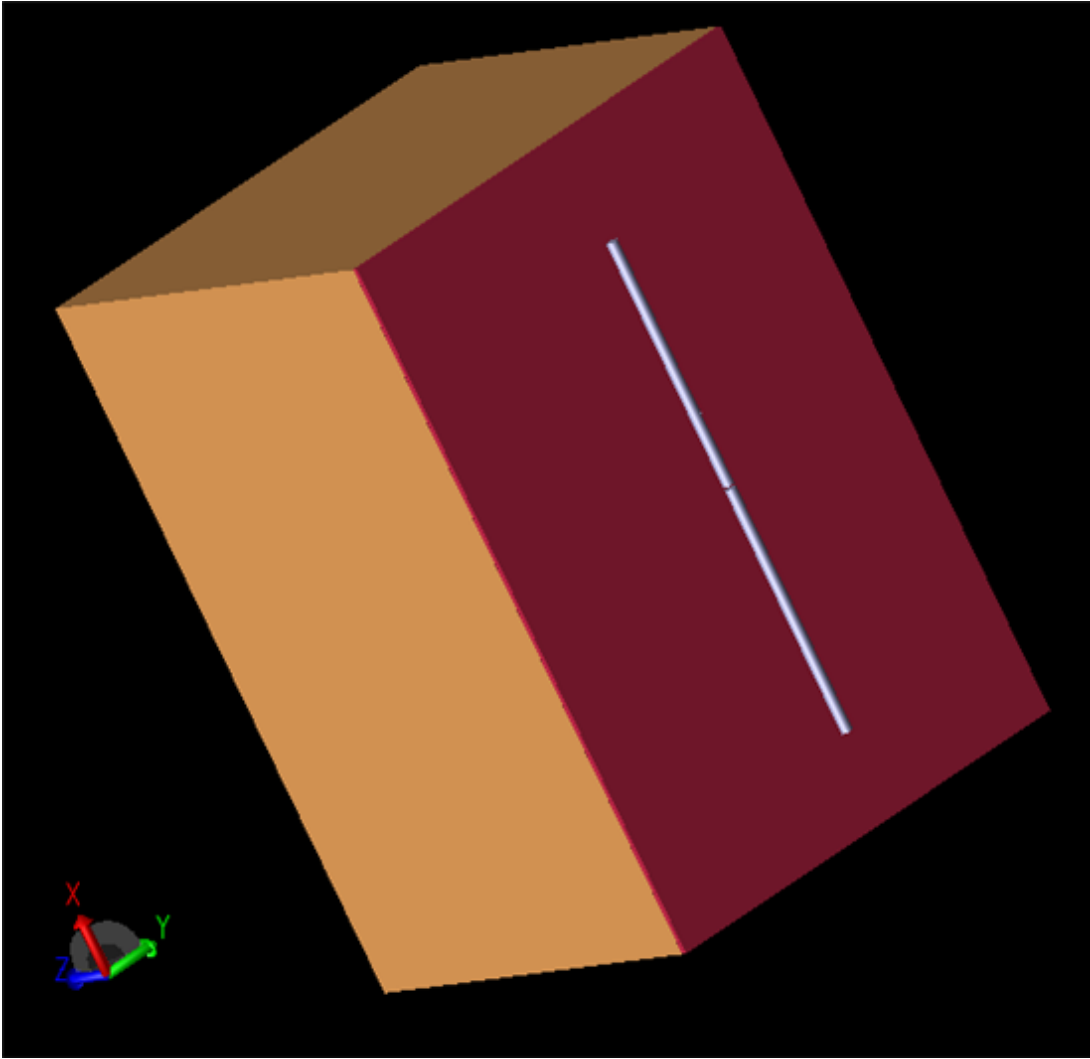
3. Under the **Physical Parameters** tab, enter 1000 kg/m³ as the Density.
4. Navigate to the **Appearance** tab and assign the **Phantom Liquid** material a new color to distinguish it from PEC.
5. Click **Done** to add the new material, Phantom Liquid.

Define material, Phantom Shell

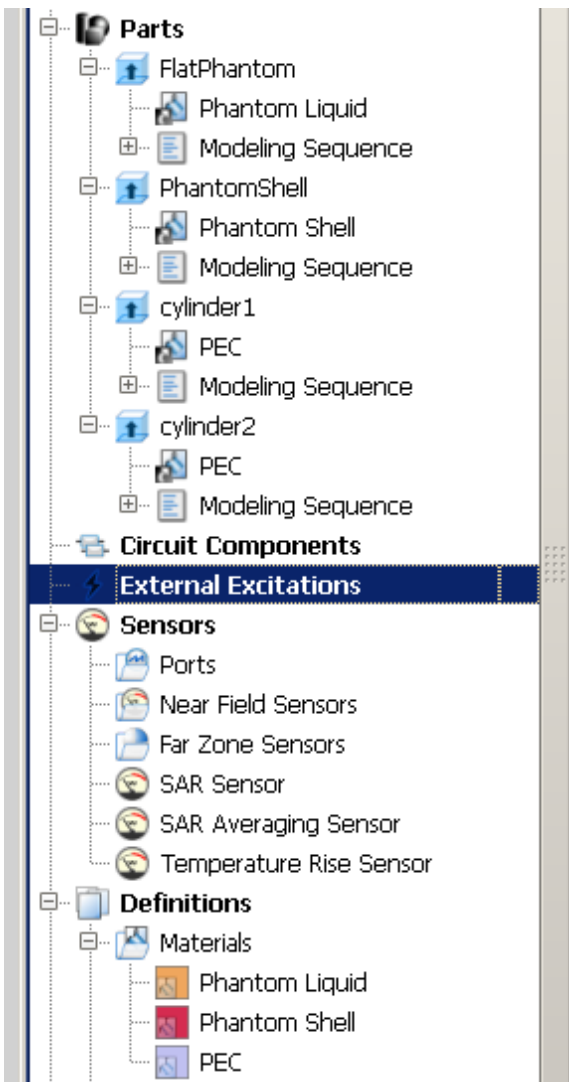
1. Right-click the **Definitions:Materials** branch of the Project Tree and select **New Material Definition**.
2. Set the material properties as follows:
 - Name: Phantom Shell
 - Electric: Isotropic
 - Magnetic: Freespace
 - Under the Electric tab:
 - Type: Nondispersive
 - Entry Method: Normal
 - Conductivity: 0 S/m
 - Relative Permittivity: 3.7
3. Navigate to the Appearance tab and assign the **Phantom Shell** material a new color to distinguish it from PEC.
4. Click **Done** to add the new material, Phantom Shell.

Assigning Materials

1. Click and drag the PEC material object located in the Project Tree and drop it on top of Cylinder1 and Cylinder2.
2. Assign the **Phantom Liquid** material to the **Flat Phantom** object.
3. Assign the **Phantom Shell** material to the **Phantom Shell** object.
The finished geometry with applied materials is seen in the following figure.



The Project Tree



Creating the Grid

Now, you will define characteristics of the cells in preparation to perform an accurate calculation.

Define cell size and padding

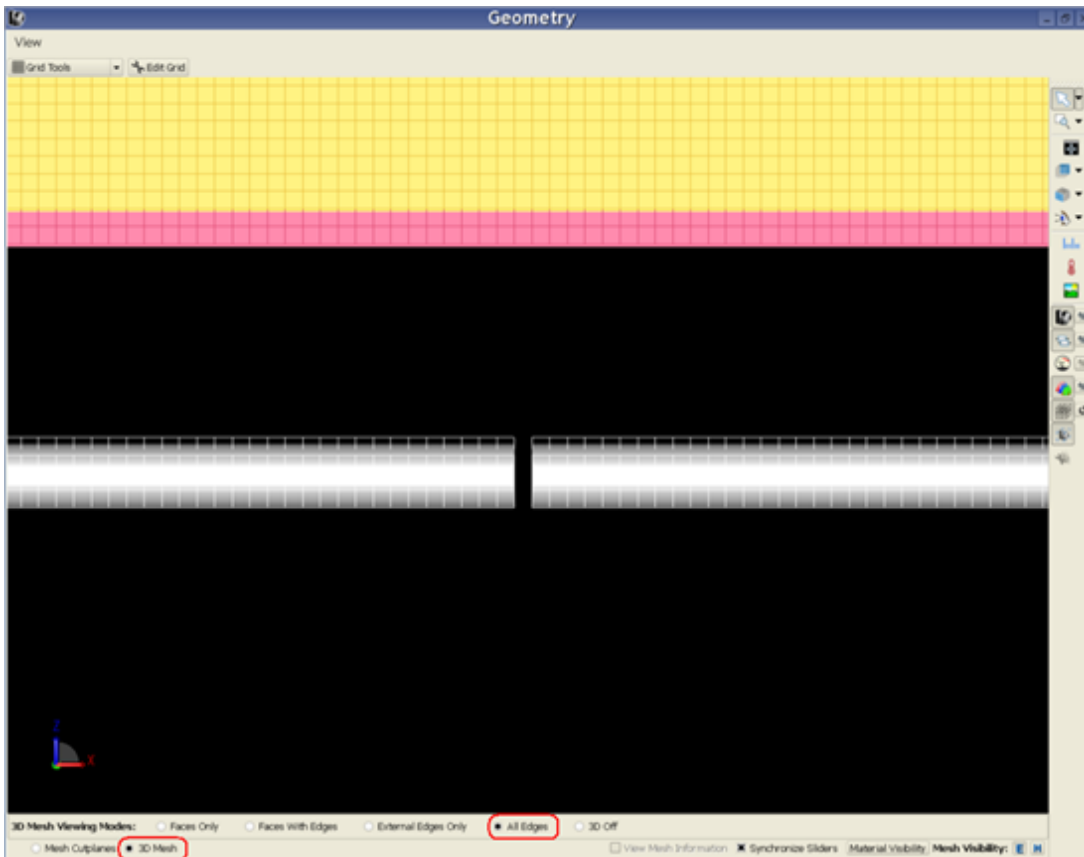
1. Open the **Geometry** browser window, select **Grid Tools** and click **Edit Grid**.
2. Navigate to the **Size** tab.
 - Define Base Cell Sizes as Target 1mm and Merge 0.8 in all directions, with the Ratio boxes selected.
 - Free Space Padding: 10 in all directions except Lower Z, which will be 20.



3. Click **Done** to apply the grid settings.

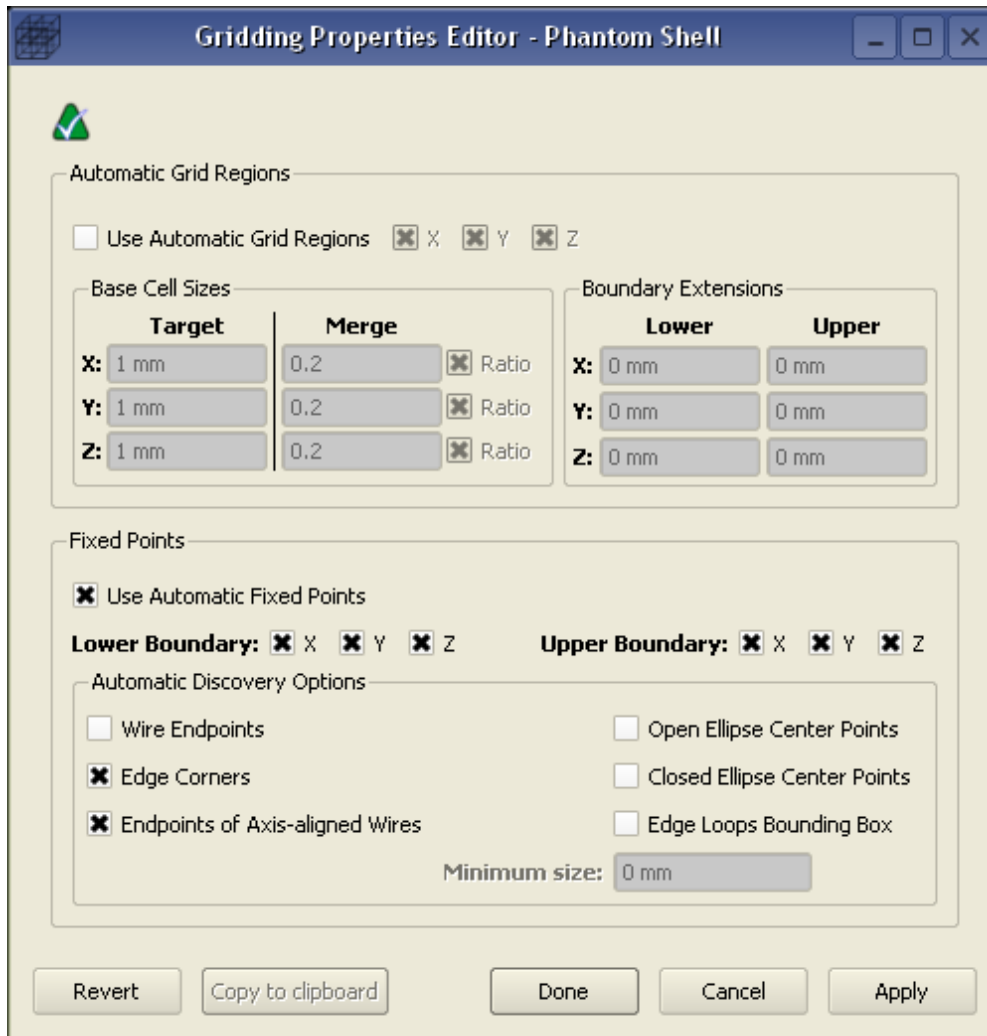
Creating a Mesh

In the FDTD branch of the Project Tree, double-click the **Mesh** icon. This displays the mesh view and automatically create the mesh. If you switch to the 3D Mesh view of **All Edges**, note that the grid does not align with the CAD view of the geometry objects This is because the cell size does not overlap the geometry dimensions exactly.



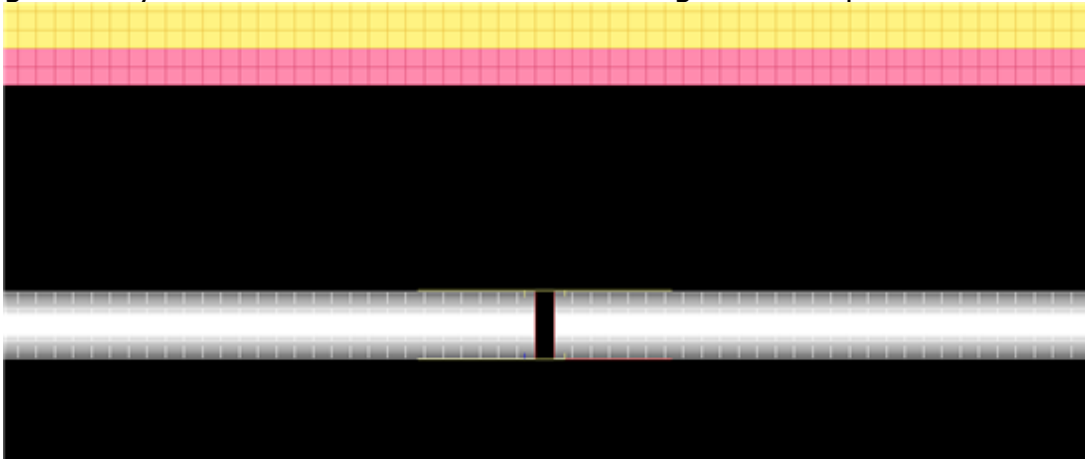
To align the mesh, you can turn on the fixed points for several of the geometry objects. This will adjust the mesh so that the grid lines overlap the edges of the CAD geometry objects.

1. From the **Parts** branch, right-click the **Phantom Shell** object.
2. Select **Gridding Properties** from the menu.
3. In the **Gridding Properties Editor** dialog box, select **Use Automatic Fixed Points**.



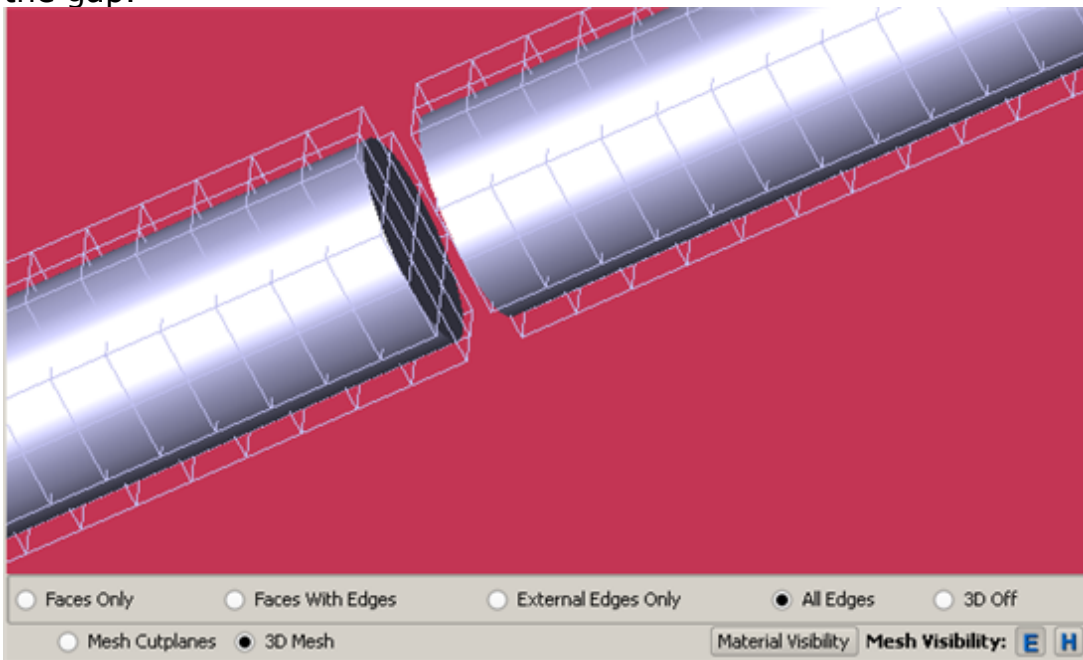
4. Click **Apply** to apply the fixed points extraction to this geometry object.
5. Click **Copy** to clipboard to save these settings.
6. Click **Done**.

Now, you will turn on Fixed Points for the cylinders. Select both Cylinder1 and Cylinder2 from the Parts branch. Right-click and select **Edit> Paste** to copy the clipboard contents to these two objects. This will turn on fixed points for the dipole as well. The resulting geometry view should now show that the grid overlaps well with the CAD objects.

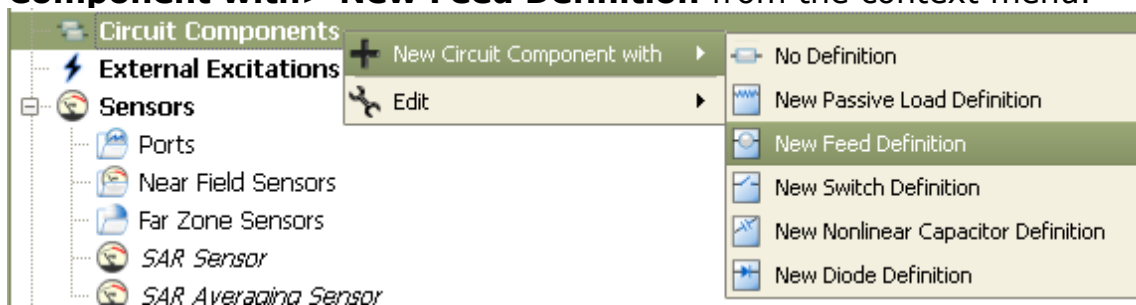


Adding a Feed

Now, you will add a Feed to the geometry. We want to place the feed in the gap between the two cylinders made of PEC materials. The following figure displays a 3D Mesh View of the gap.



1. Right-click the **Circuit Components** branch of the Project Tree. Choose **New Circuit Component with > New Feed Definition** from the context menu.



2. Define the endpoints of the feed.
 - Endpoint 1: X: 113 mm, Y: 75 mm, Z: -15 mm
 - Endpoint 2: X: 112 mm, Y: 75 mm, Z: -15 mm
3. Navigate to the Properties tab, and enter the following:
 - Name: Feed
 - Component Definition: 50 ohm Voltage Source
 - Direction: Auto
 - Polarity: Positive
 - Select the **This component is a port** checkbox.
4. Click **Done** to add the Feed.

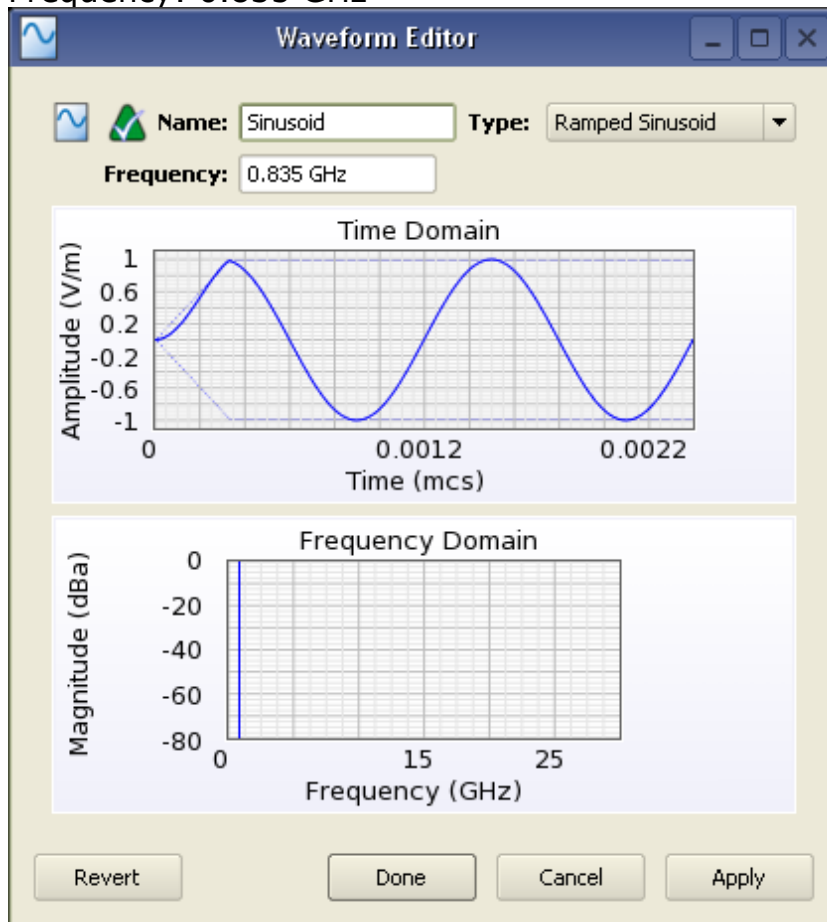
Editing the Waveform

An associated waveform was automatically created for the feed definition.

1. Navigate to the **Definitions:Waveforms** branch of the Project Tree. Double-click the Broadband Pulse waveform to edit its properties. Set the properties of the waveform as follows:

2.

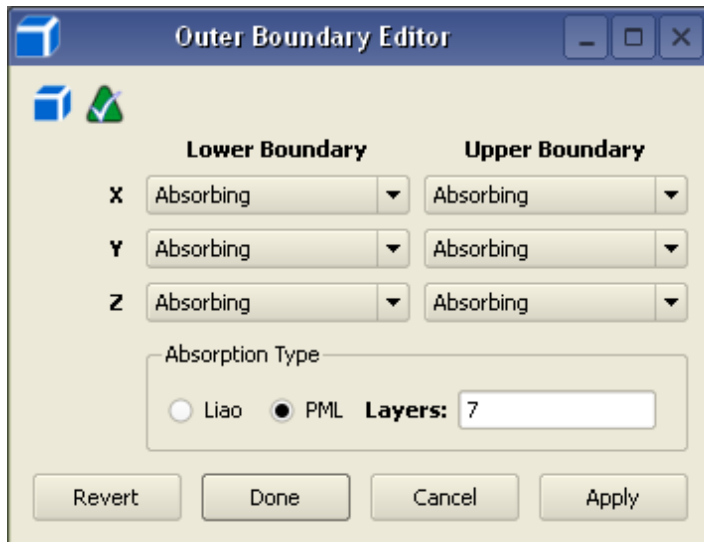
- Name: Sinusoid
- Type: Ramped Sinusoid
- Frequency: 0.835 GHz



3. Click **Done** to apply the changes.

Defining the Outer Boundary

1. Double-click the **FDTD:Outer Boundary** branch of the Project Tree to open the Outer Boundary Editor.
2. Set the outer boundary properties as follows:
 - Boundary: Absorbing for all boundaries
 - Absorption Type: PML
 - Layers: 7



3. Click **Done** to apply the outer boundary settings.

Requesting Output Data

Recall that the project already contains one port sensor named Feed that will request results. You can collect SAR results by adding an SAR Sensor.

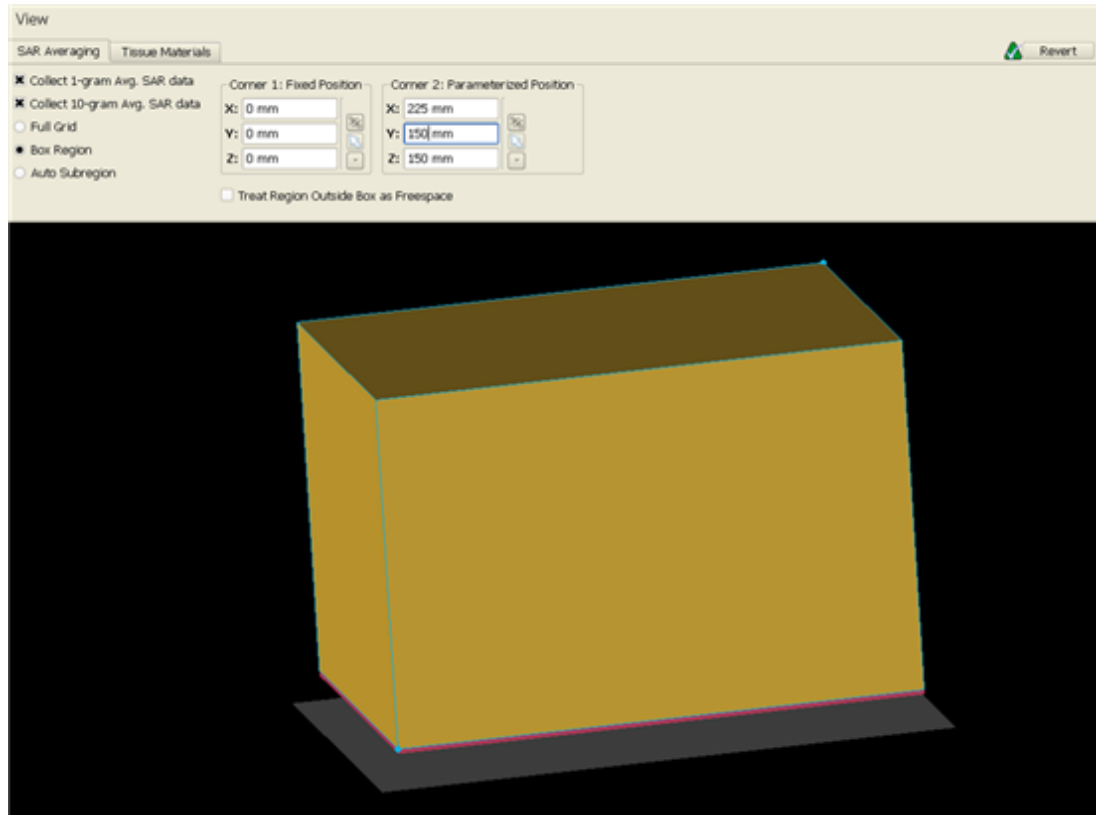
Adding an SAR Sensor

1. Right-click the **Sensors:SAR Sensors** branch of the Project Tree. Select **Properties** from the context menu.
 - Select the **Collect Raw SAR Data** checkbox.
 - Select the **Full Grid** box. It requires that the data be saved over the full grid if Averaged SAR values will be computed.
2. Click **Done** to finish editing the SAR Sensor.

To collect averaged SAR data, you must define a sensor.

1. Right-click the **Sensors:SAR Averaging Sensor** branch of the Project Tree. Select Properties from the context menu.
 - Check the Collect 1-gram Avg. SAR data and Collect 10-gram Avg. SAR data boxes.
 - Select the Box Region box, and enter the following coordinates:
 - Corner 1: (0 mm, 0 mm, 0 mm)
 - Corner 2: (225 mm, 150 mm, 150 mm)

FDTD Simulation

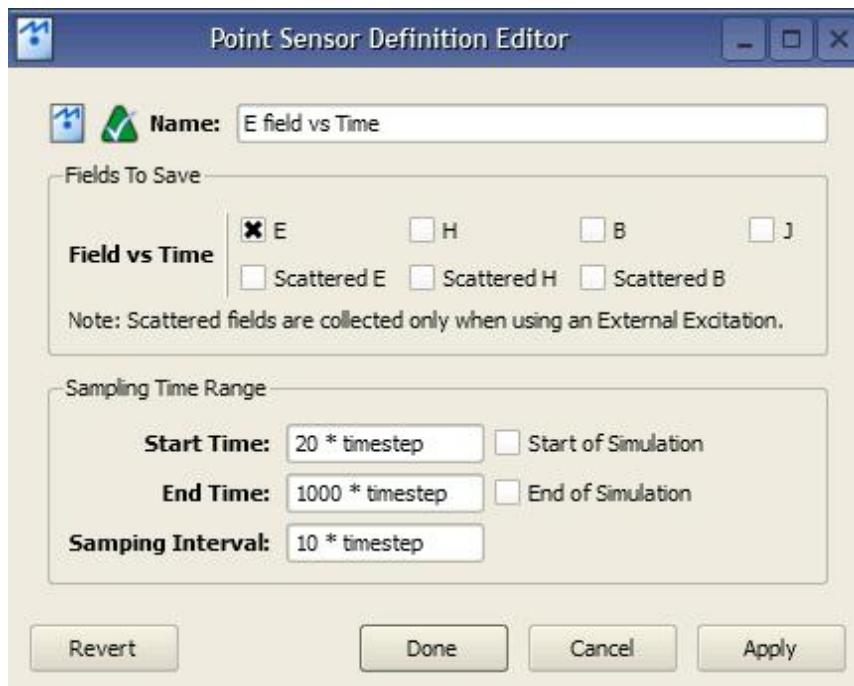


2. Click **Done** to finish editing the SAR Averaging Sensor.

Adding a Point Sensor Definition

A Point Sensor may be saved inside the **Flat Phantom** object to monitor the convergence of the fields during the calculation. First, you will create its definition.

1. Right-click the **Definitions:Sensor Data Definitions** branch of the Project Tree. Choose **New Point Sensor Definition** from the context menu.
2. Set the properties of the surface sensor definition as follows:
 - Name: E-field vs. Time
 - Field vs. Time: E
 - Sampling Interval: timestep



3. Click **Done** to finish editing the definition.

Adding a Point Sensor

1. Right-click the **Sensors:Near Field Sensors** branch of the Project Tree. Select **New Point Sensor** from the context menu.
 - Enter its Location as (112.5 mm, 75 mm, 15 mm).
 - Under the Properties tab, enter the following:
 - Name: E-field
 - Sensor Definition: E-field vs. Time
 - Sampling Method: Snapped to E-Grid
2. Click Done to finish editing the E-field Sensor.

Running the Calculation

If you have not already saved your project, do so by selecting **File>Save Project**. After the project is saved, a new simulation can be created to send to the calculation engine.

1. Open the **Simulations** workspace window. Click the **New Simulation** button in the upper-left to set up a new simulation.
2. Type a descriptive name for the simulation, such as Flat Phantom at 835MHz.
3. Most of the default settings are sufficient. Navigate to the Specify Termination Criteria tab. Set up the termination criteria as follows:
 - Maximum Simulation Time: 10000 * timestep
 - Detect Convergence: Checked
 - Threshold: -30 dB
4. Click **Create and Queue Simulation**.

Viewing the Results

The Output tab of the Simulations workspace window displays the progress of the

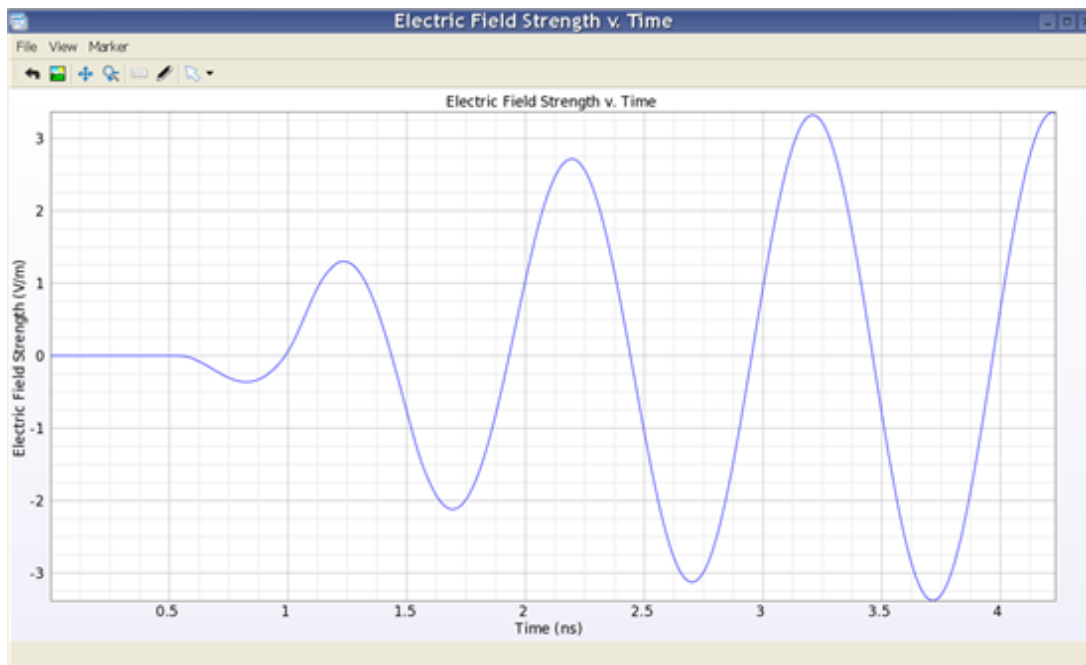
simulation. After the Status column shows that the simulation has completed, you can view the results in the Results workspace window.

E-field Results

Now, you can view the E-field results retrieved from the center of the Tissue.

- To filter the list accordingly, select the following options in the columns in the top pane of the Results window. (You may need to change your column headings first.)
 - Output Object: E-field
 - Result Type: E-field (E)
- Right-click the result and select Create Line Graph.
 - Select X as the Component, and click **View**. The plot of the E-field at the center of the Flat Phantom object will appear.

Note
It is possible to view the data before the simulation is complete. The plot will update automatically as more data is computed.



The resulting plot indicates that the fields inside the phantom are at steady-state as a smooth sine wave is visible. This confirms our convergence condition of -30 dB that was set during the simulation setup.

- You may close the window when you are finished viewing the results.

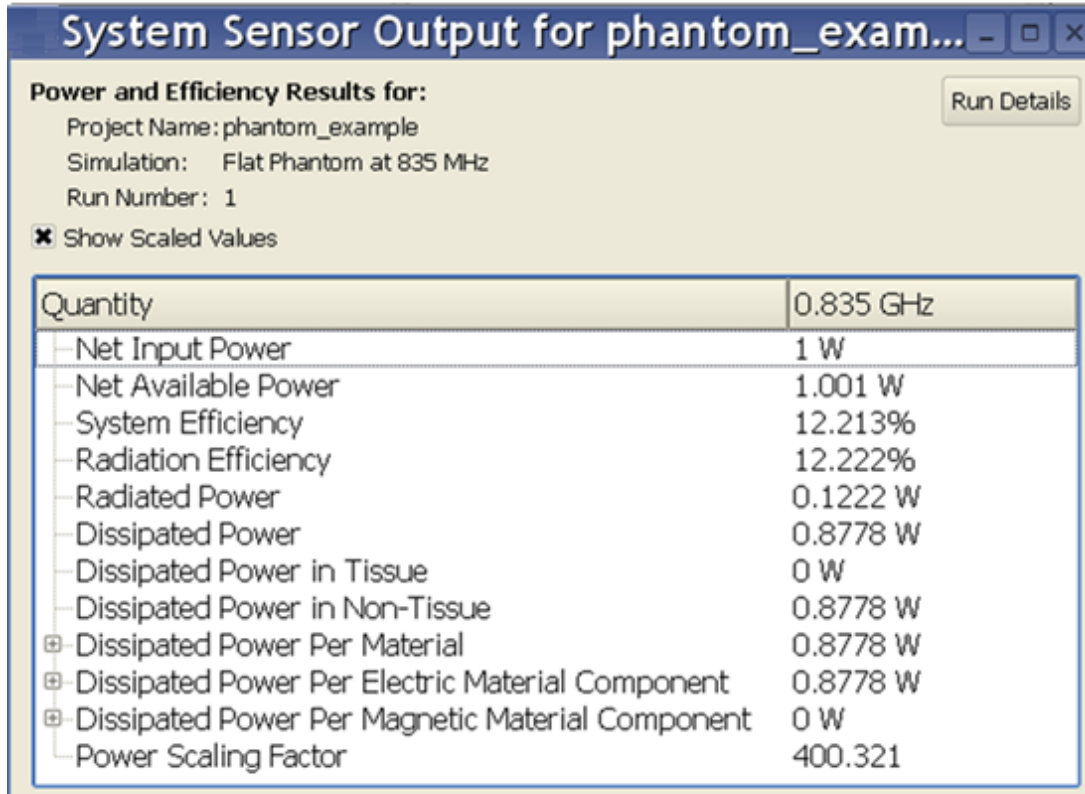
System Efficiency Results

Now you can view data from the point.

- To view the system efficiency results, select the following:
 - Output Object: System
 - Result Type: Net Input Power
- Double-click on the result. The powers in the simulation are displayed. As you can see, the power delivered to the antenna is relatively small, just under 2.5mW. For many SAR analyzes, the power is adjusted to a value such as 1W to normalize all

results. You can do this by clicking on the System Sensor Output window.

3. Click the power value to the right of Net Input Power (0.002498 W).
4. Type a value of 1W and click **Enter**. The powers should now scale to the 1W input. This will also scale the SAR value.

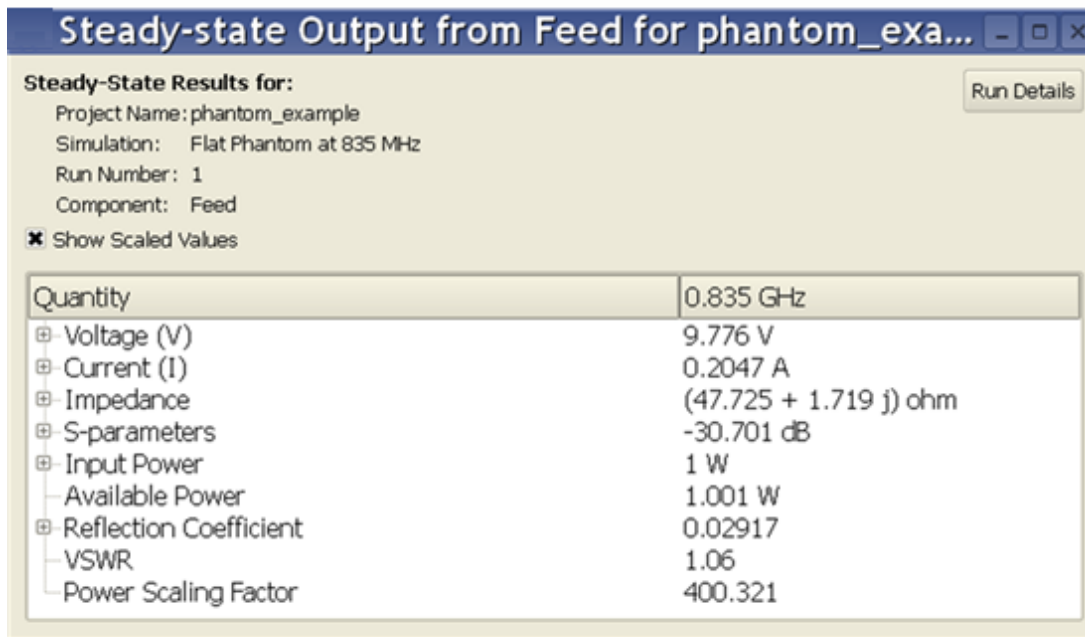


Quantity	0.835 GHz
Net Input Power	1 W
Net Available Power	1.001 W
System Efficiency	12.213%
Radiation Efficiency	12.222%
Radiated Power	0.1222 W
Dissipated Power	0.8778 W
Dissipated Power in Tissue	0 W
Dissipated Power in Non-Tissue	0.8778 W
Dissipated Power Per Material	0.8778 W
Dissipated Power Per Electric Material Component	0.8778 W
Dissipated Power Per Magnetic Material Component	0 W
Power Scaling Factor	400.321

5. You can close the window when you are finished viewing the results.

To view the Feed results:

1. In the Results workspace window, select:
 - Output Object: Feed
 - Result Type: S-Parameters
2. Double-click the result under the Discrete domain. The following results will appear showing the impedance at the feed, the input power delivered, and the return loss.



Steady-State Results for:

Project Name: phantom_example
Simulation: Flat Phantom at 835 MHz
Run Number: 1
Component: Feed

Show Scaled Values

Quantity	0.835 GHz
⊕ Voltage (V)	9.776 V
⊕ Current (I)	0.2047 A
⊕ Impedance	(47.725 + 1.719 j) ohm
⊕ S-parameters	-30.701 dB
⊕ Input Power	1 W
Available Power	1.001 W
⊕ Reflection Coefficient	0.02917
VSWR	1.06
Power Scaling Factor	400.321

You can see from the table that our return loss is less than -30 dB, so you have a good match at the selected frequency.

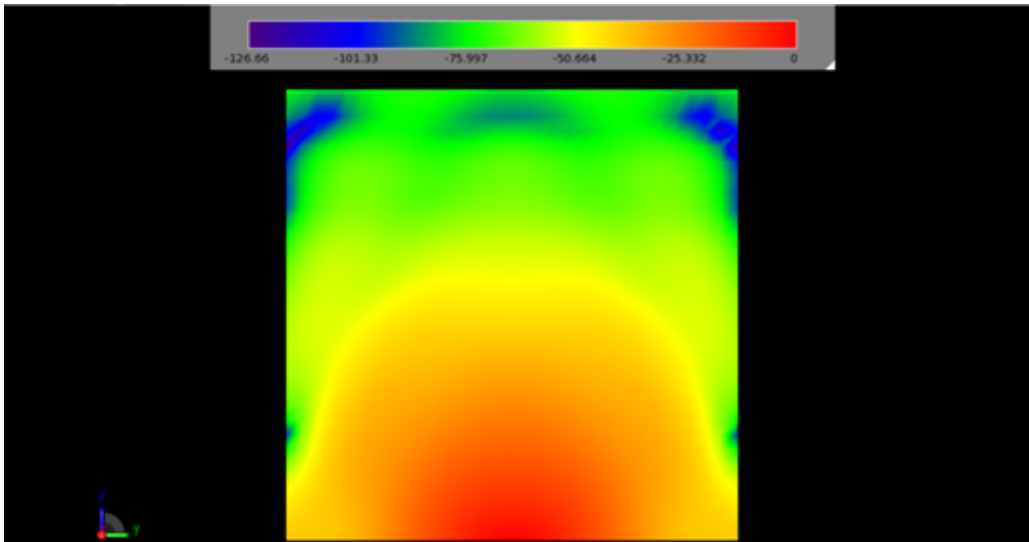
3. You can close the window when you are finished viewing the results.

SAR Sensor Data

Now you will load the SAR data into the field viewer.

1. To view the SAR sensor data, select the following:
 - Output Object: SAR Sensor (Raw)
 - Result Type: SAR (Specific Absorption Rate)
2. Double-click the result in the filtered list. The plot will appear in the **Geometry** workspace window.
3. Under the **Setup** tab, adjust the following settings:
 - Sequence Axis: X
 - Display Mode: Flat
 - Decimation: Normal
 - Under Axis Ranges:
 - Y: Full
 - Z: Full
4. Toggle the **Parts Visibility** to turn off the display of the geometry, and select the **Left (+X)** orientation. The resulting image should appear.
5. Under the Sequence tab, define Showing: 112. The following SAR image appears.

FDTD Simulation

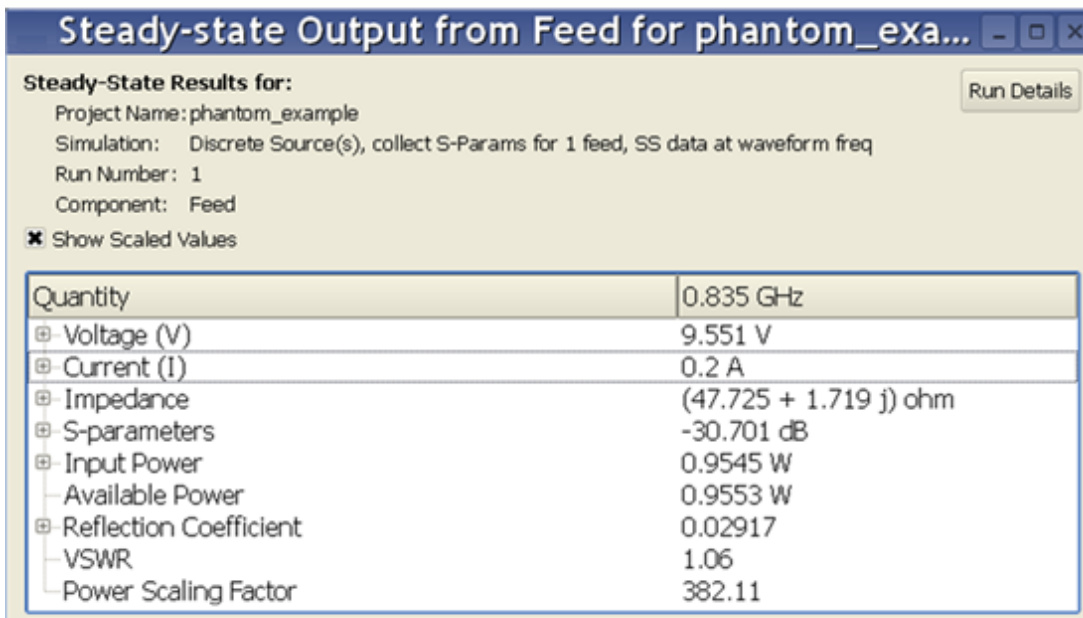


- Under the **Statistics** tab, choose **View all SAR Stats**. A summary table of the SAR values appears, as shown in the following figure:

SAR Statistics for phantom_example : 000002 : 1	
SAR Statistics for:	
Project Name: phantom_example	
Simulation: Discrete Source(s), collect S-Params for 1 feed, SS data at waveform freq	
Run Number: 1	
<input checked="" type="checkbox"/> Show Scaled Values	
Quantity	0.835 GHz
[-] SAR Sensor (Raw)	
[-] Maximum SAR Value	13.975 W/kg
[-] Max Value Location	(112.5 mm, 74.55 mm, 0.5 mm)
[-] Avg SAR in Exposed Object	0.1688 W/kg
[-] SAR Averaging Sensor (10g Average)	
[-] Maximum SAR Value	5.875 W/kg
[-] Max Value Location	(112.5 mm, 74.55 mm, 11.5 mm)
[-] SAR Averaging Sensor (1g Average)	
[-] Maximum SAR Value	9.148 W/kg
[-] Max Value Location	(112.5 mm, 74.55 mm, 5.5 mm)
[-] Power Scaling Factor	400.321

For some situations, the SAR results should be normalized to the feed point current rather than the forward power.

- To make this adjustment, return to the **Results** workspace window and select:
 - Output Object: Feed
 - Result Type: Current
- Double-click the result under the Discrete domain.
- On this screen, the current value may be edited by clicking on the numerical value. Type in a value of 200mA and all results, including the SAR values, will be adjusted to a feed point current of 200mA.



4. You may close the window when you are finished viewing the results. It may be of interest to plot the SAR as a function of distance along a line extending above the feed point.
5. In the Results workspace window, select the SAR data.
6. Right-click on the result and select **Create Line Graph**.
 - Independent Axis: Z
 - X: 112
 - Y: 75 mm
7. Click **View** to see a line plot of the SAR as a function of distance from the feed point in the center of the phantom, as shown in the following figure:

