100 Elwood Davis Road ♦ North Syracuse, NY 13212 ♦ USA

SonnetLab Toolbox Overview

Specialists in High-Frequency Electromagnetic Software
(315) 453-3096 Fax: (315) 451-1694 http://www.sonnetsoftware.com

SonnetLab Overview

The SonnetLab toolbox for Matlab (from here on called SonnetLab) is a Matlab toolbox that provides integration between Sonnet Software's design tools and MathWork's Matlab scripting environment. SonnetLab provides users with the ability to incorporate Sonnet Software's award winning high precision simulation tools in a powerful environment that simplifies design automation.

SonnetLab provides an invaluable tool for Sonnet users. SonnetLab can open existing Sonnet Project files and build new Sonnet projects from within Matlab. SonnetLab provides methods to simulate circuits using Sonnet Software's high precision simulation tool ***em***.

Users of SonnetLab are able to make new Sonnet Projects using the Matlab command:

```
>> VariableName=SonnetProject();
```

The new Sonnet project generated in Matlab uses the same settings as a new geometry project created using Sonnet's project editor. Users can open existing Sonnet project files with SonnetLab using the following command:

```
>> VariableName=SonnetProject('ProjectFile.son');
```

The above command opens an existing Sonnet project file located in the Matlab current working directory, parses all the data elements and stores them all in an object referenced by the Matlab variable VariableName. SonnetLab supports both Sonnet geometry and netlist projects.

The following command writes the Sonnet project to the hard drive:

```
>> VariableName.saveAs('ProjectFileBackup.son');
```

This command saves the Sonnet project represented by `VariableName` as a Sonnet compatible project file. If `VariableName` is a geometry or netlist project then `ProjectFileBackup.son` will be a Sonnet geometry project or Sonnet netlist project respectively.

SonnetLab includes many methods that help users modify their designs with relative ease. The following command adds a new metal polygon to the project on layer zero with (X,Y) coordinate pairs of (0,0), (0,2), (2,2), (0,0).

```
>> VariableName.addMetalPolygonEasy(0, [0,0,2,2], [0,2,2,0])
```

SonnetLab provides functions to modify simulation parameters and to simulate Sonnet project files. The user can add an ABS frequency sweep to a Sonnet project with the following command:

```
>> VariableName.addAbsFrequencySweep(5,10);
```

The above command adds an ABS frequency sweep to the project from 5 GHz to 10 GHz, as well as selecting the ABS frequency sweep for simulation. Users simulate their Sonnet Project using

the following command:

```
>> VariableName.simulate();
```

The above command simulates the Sonnet project using Sonnet's simulation engine, ***em***.

Brief method overview:

Below is a brief overview of some of the common functions used with SonnetLab. The method reference document contains a complete listing of all available functions along with a more in depth descriptions of usages along with examples.

Core Functions:

Function Definition	Description
`SonnetProject()`	Build a new Sonnet Project
`SonnetProject(Filename)`	Open an existing Sonnet Project
`simulate()`	Calls Sonnet ***em*** to simulate the Sonnet Project File
`openInSonnet()`	Opens the project file in the Sonnet project editor for viewing/editing. Any changes saved when editing are re-imported into the Matlab version of the project

Analysis Frequency Functions

Function Definition	Description
`changeSelectedFrequencySweep(StringForSelectedFrequencySweep)`	Change project's selected frequency sweep
`addSweepFrequencySweep(StartFrequency,EndFrequency,theStepFrequency)`	Add a linear frequency sweep
`addAbsFrequencySweep(StartFrequency,EndFrequency)`	Add an ABS frequency sweep
`addAbsEntryFrequencySweep(StartFrequency,EndFrequency)`	Add an ABS frequency sweep to a frequency combination set
`addAbsFmaxFrequencySweep(StartFrequency,EndFrequency,theMaximum)`	Add an ABS frequency maximum sweep to a frequency combination set
`addAbsFminFrequencySweep(StartFrequency,EndFrequency,theMinimum)`	Add an ABS frequency minimum sweep to a frequency combination set
`addDcFrequencySweep(Mode,Frequency)`	Add a DC frequency sweep to a frequency combination set
`addEsweepFrequencySweep(StartFrequency,EndFrequency,AnalysisFrequencies)`	Add an exponential frequency sweep to a frequency combination set
`addLsweepFrequencySweep(StartFrequency,EndFrequency,AnalysisFrequencies)`	Add a linear frequency sweep to a frequency combination set

Function Definition	Description
addSimpleFrequencySweep(StartFrequency,EndFrequency,StepFrequency)	Add a linear frequency sweep
addStepFrequencySweep(StepFrequency)	Add a step frequency sweep to a frequency combination set

Polygon Functions

Function Definition	Description
movePolygon(Polygon, X, Y)	Moves a polygon such that its center is at location (X,Y)
snapPolygonsToGrid(Axis)	Snaps polygons to the grid
deletePolygonUsingId(Id)	Delete a polygon
addViaPolygonEasy(MetalizationLevel,Level,X,Y)	Adds a via polygon to the polygon array
addMetalPolygonEasy(MetalizationLevel,X,Y)	Adds a metal polygon to the polygon array
addDielectricBrickEasy(MetalizationLevel,X,Y)	Adds a dielectric brick polygon to the polygon array
addPortToPolygon(Polygon,VertexNumber)	Adds a standard port to a polygon in the project
drawCircuit()	Plots a 3D view of the circuit

Netlist Functions

Function Definition	Description
addResistorElement(NodeNumber1,NodeNumber2,Resistance,NetworkNumber)	Adds a resistor element to a network in a Sonnet netlist project
addInductorElement(NodeNumber1,NodeNumber2,InductanceValue,NetworkNumber)	Adds an inductor element to a network in a Sonnet netlist project
addCapacitorElement(NodeNumber1,NodeNumber2,CapacitanceValue,NetworkNumber)	Adds a capacitor element to a network in a Sonnet netlist project
addTransmissionLineElement(NodeNumber1,NodeNumber2,ImpedanceValue,Length,Frequency,NetworkNumber)	Adds a transmission line element to a network in a Sonnet netlist project
addPhysicalTransmissionLineElement(NodeNumber1,NodeNumber2,ImpedanceValue,Length,Frequency,theEeffValue,Attentuation,NetworkNumber,theGroundNode)	Adds a physical transmission line element to a network in a Sonnet netlist project
addDataResponseFileElement(Filename,PortNodeNumbers,NetworkNumber,GroundReference)	Adds a data response file element to a network in a Sonnet netlist project

Function Definition	Description
addProjectFileElement(Filename,PortNodeNumbers,UseSweepFromSubproject,NetworkNumber)	Adds a project file element to a network in a Sonnet netlist project
addNetworkElement(Name,PortNodeNumbers,Impedence)	Adds another network to a Sonnet netlist project

Contact

Your feedback is important to us. If you have any questions or comments about SonnetLab, please contact Sonnet Support by email at support@sonnetsoftware.com.

Please make sure you are using the most up to date version of SonnetLab before submitting a bug report. When submitting a bug report please include the Sonnet project file that generated the error (Sonnet project files have the extension .son) and the output from the command "SonnetMatlabVersion". The more information that that we receive the faster it will be for us to resolve the issue and contact you back.