eSim code documentation

v1.1.2

Author:

Fahim Khan

November 4, 2016

Abstract

eSim (previously known as Oscad / FreeEDA) is an open source EDA tool for circuit design, simulation, analysis and PCB design. It is an integrated tool built using open source software such as KiCad (http://www.kicad-pcb.org) and Ngspice (http://ngspice.sourceforge.net/). eSim is released under GPL.

eSim offers similar capabilities and ease of use as any equivalent proprietary software for schematic creation, simulation and PCB design, without having to pay a huge amount of money to procure licenses. Hence it can be an affordable alternative to educational institutions and SMEs. It can serve as an alternative to commercially available/ licensed software tools like OrCAD, Xpedition and HSPICE..

1

1 Prerequisite

eSim has following dependency.

1. Software Package

   - Python 2.7
   - PyQt4
   - Matplotlib
   - numpy

2. Open source tool

   - Python 2.7
   - PyQt4
   - Matplotlib
   - numpy

2 eSim installation

eSim is available for Ubuntu and Windows(7,8,10).

2.1 Ubuntu

1. Download eSim installer for Linux from http://esim.fossee.in/downloads to a local directory and unpack it. You can also unpack the installer through the terminal. Open the terminal and navigate to the directory where this INSTALL file is located. Use the following command to unpack:

   ```
   unzip eSim-1.1.2.zip
   ```

2. To install eSim and other dependecies run the following command.

   ```
   ../install-linux.sh --install
   ```

3. Once the eSim is installed, you can open it from the terminal.

   ```
   esim
   ```

   or you can double click on eSim icon created on the Desktop after installation.

2.2 Windows

1. Download eSim windows installer from website http://esim.fossee.in/downloads

2. Open eSim-Windows-Installer folder, right click on Setup file and select run as administrator. Click Yes and Next to complete the installation.

3. eSim icon will be created on deskop. You can double click on the eSim icon created on the Desktop after installation.

3 eSim Modules

eSim code is divied into modules based on the functionality. All modules are under 'src' folder of eSim.

To run eSim application from command line. Open command line inteprreter and go to the location under src/frontend

Now run the below command to open eSim window.

```
python Application.py
```

3.1 frontEnd

This package contain main gui modules of eSim. All main window compoenet such as Project explorer,Dock area,workspace are present in this package.

3.1.1 Application.py

This file contain main function. The main function initiate the Application class as well as instantiate workspace module where user can define the workspace location where all the project will be stored.

- Class Application : This class is repponsible for initiating all the neccessary class. Also it creates the gui for tool bar and menu bar. The function details are as follows.

  - initToolBar(): This initiate the tool bar.
  - closeEvent(event): This function close the event. Event can be any process or widget.
  - close_project(): It close the project and clear all the variable which hold project information.
  - new_project(): It creates new project in eSim and add the information in project explorer. If project is already open then it overwrite project related variable with new project information.
  - open_project(): It open the project and add it to project explorer.
  - help_project(): It open the eSim user manual.

- open_ngspice(): It execute the ngspice netlist of current project. And also open the Python plotting window inside dock area. If the netlist is not present then it will throw an exception.
- open_subcircuit(): It open sub circuit widget inside the dock area.
- open_nghdl(): It opens the nghdl widget where user can upload the nghdl model. Nghdl has to be installed before using it in eSim. You can install it independently as well as along with eSim installation.
- open_modelEditor(): It opens model editor widget in the dock area.
- open_OMedit(): This function call ngspice to OM edit converter and then launch OM edit provided OM edit is installed in the machine. OMEdit and OMOptim is part of Open modelica tool which needs to be installed separately.
- open_OMoptim(): This function open the OM Optim (optimization tool of Open Modellica)

- Class MainView : This class set the main view of window. It add note area, split the widget and complete the layout management for frontend. Also it initialze the DockArea and Project Explorer.

3.1.2 Workspace.py

This module create workspace window where user can browse to his/her desired workspace location. The default workspace location is in home directory.

- Class Workspace : This create the workspace window which appear on the top of main window.
  - initWorkspace(): This create gui of Workspace widget.
  - defaultWorkspace(): This function select default workspace location.
  - returnWhetherClickedOrNot(): It checked if it is clicked or not.
  - createWorkspace(): If press ok then it creats proper workspace location
  - browseLocation(): This function return the path/location which user has selected.

3.1.3　DockArea.py

This module is responisble to create widget inside dock area.

- Class DockArea : Responsible for instantiating the dock widget.

  - createTestEditor(): This function is for testing new widget inside dock area.

  - plottingEditor(): This function creates the python plotting widget inside dock area.

  - ngspiceEditor(): This function open the ngspice editor but not in dock area.

  - modelEditor(): This function create the model editor widget and instatiate the Model editor class.

  - kicadToNgspiceEditor(): This function create the kicad to ngspice converter widget.

  - subcircuiteditor(): This function create the widget for sub circuir.

  - usermanual(): This function create the widget for user manual.

  - modelicaEditor() : This function create the widget for Modellica Editor from where user can open OMEdit and OMOptim.

  - closeDock(): This function is called when any dock widget is closed.

3.2 browser

This package is responsible for openning user manual in browser.

3.2.1 UserManual.py

This module open the user manual in web browser using python webbrowser package.

3.2.2 Welcome.py

This module creates the eSim welcome page.