ThunderStorm Documentation

Release 0.1

David Trémouilles

July 02, 2014

Contents

Contents:

CHAPTER 1

How to use Thunderstorm Library interactively

1.1 An example

```
from thunderstorm.interact import new_storm

mystorm = new_storm("tmp_storm.oef")
mystorm.import_SERMA("../TestData01012012/SERMA/101_s1V_A.csv", "serma data")
mystorm.overlay_raw_tlp((0,))
```

CHAPTER 2

ThunderStorm modules user guide

2.1 thunder

thunder module compiles utils to import and manipulate Transmission Line Pulse (TLP) measurement data

2.1.1 tlp

tlp data

class thunder.tlp.**Droplet**(*h5group*)
A Droplet is basically one TLP measurement i.e. a set of TLP pulses, a TLP curve, leakages measurement etc... A Droplet is base on a hdf5 file group

class thunder.tlp.**H5IVTime**(*droplet=None*)
Contain the transient waveforms

class thunder.tlp.**H5RawTLPdata**(*droplet=None*)
All measurement data: device name, pulses, TLP curve, leakage ... from the h5File are made accessible throught this class

class thunder.tlp.**RawTLPdata**(*device_name*, *pulses*, *iv_leak*, *tlp_curve*, *leak_evol*, *file_path*, *tester_name=None*)
All measurement data: device name, pulses, TLP curve, leakage ... are packed in this class

class thunder.tlp.**TLPcurve**(*current*, *voltage*)
The data for a TLP curve

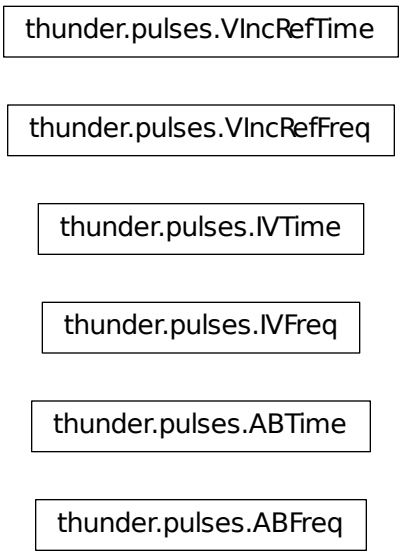
current
Return the current array of the TLP curve

data
Return a copy of the raw tlp curve data

voltage
Return the voltage array of the TLP curve

2.1.2 pulses

Define several classes to manipulate a set of TLP measurement



2.1.3 leakage evolution

Various way to calculate leakage evolution

`thunder.leak_evol_calculation.`**`point_evol`**(*iv_leak*, *evol_point*)
Return the voltage and current evolution at the point define in the given measure.

`thunder.leak_evol_calculation.`**`sum_var`**(*iv_leak*)
Return the relative evolution of the integral of the absolute value of the leakage for a given measurement.

2.1.4 import plug-ins

This package is the place for data import plugins. This __init__.py populate the ImportPlugin class with the plugin files. Only plugin files starting with "plug" followed by underscore and ending with ".py" are taken into account. import_plugs variable contains all the import plugins

2.2 lightning

lightning module compiles utils to view and make graphs out of Transmission Line Pulse (TLP) measurement data Simple typical TLP curves plot

class `lightning.simple_plots.`**`LeakageIVsFigure`**(*figure*, *ivs_data*, *title=''*)
Plot all leakge-iv data

class `lightning.simple_plots.`**`PulsesFigure`**(*figure*, *pulses*, *title=''*)
Plot all transient curve

class lightning.simple_plots.**TLPFigure**(*figure*, *tlp_curve_data*, *title=''*, *leakage_evol=None*)
A simple TLP figure

class lightning.simple_plots.**TLPOverlay**(*figure*, *title=''*)
A tool to visualize overlay of TLP I-V curves

class lightning.simple_plots.**TLPOverlayWithLeakEvol**(*figure*, *title=''*)
A tool to visualize overlay of TLP I-V curves

This module contain base utils to observ a TLP curve

class lightning.tlp_observer.**TLPPickFigure**(*figure*, *raw_data*, *title=''*)
Base class for tlp point picking

specific_key_press(*key_code*)
Method to handle key press event specific to the assiciated graph. Should be implemented by child class.

update()
Method to update the associated graph (leakages or pulses for example) associated with the TLP IV plot. Must be implemented by child class.

Tools to observe transient curves corresponding to TLP points

class lightning.pulse_observer.**TLPPulsePickFigure**(*figure*, *raw_data*, *title=''*)
TLP picking tool showing transient pulses

CHAPTER 3

Indices and tables

- *genindex*
- *modindex*
- *search*

l

t