GuiCfg – Central place for application specific UI-expressions

Graphical control elements use different labels for MOKE and VSM version. These label strings are defined in GuiCfg.

|  |  |
| --- | --- |
|  | classdef GuiCfg  %GUICFG  %  % Application specific "string table" and layout configuration for VSM/MOKE setup  % Please modify in order to match your specific hardware setup  %  % VSM Setup  %%{  properties (Constant)  % Plot window  plotw\_showquadr = 'Show quadrature'  plotw\_lockin = 'Lock-in sensitivity: '  plotw\_pickup = 'Pickup inphase'  plotw\_quadr = 'Pickup quadrature'  % Plot text  plot\_label\_Pickup\_Y = 'Pickup Signal [V]'  … |

These lines show(1) or hide(0) additional options in the main configuration window:

cfgmain\_show\_ask\_lockin = 1 % whether option 'ask if lockin sensitivity is set

up correctly' is shown

cfgmain\_show\_ask\_saveconfig\_before\_meas = 1 % whether option 'ask if config

shall be saved before

measurement starts' is shown

Hardware emulation, Implementation of Matlab’s legacy interface, DAQ hardware

The Definition of the data acquisition hardware takes place at PkgAnalogIO.AnalogIO.

Uncommenting PkgAnalogIO.Fake\_AnalogInAndOut activates Hardware Emulation mode for running the program without attached hardware.

Uncommenting the second line is intended for normal usage. AnalogIn\_mcc & AnalogOut\_mcc both implement the Matlab’s “legacy interface” for data acquisition require the license for the Data Acquisition Toolbox. These classes are used for all supported cards (mcc,ni,..) that support the legacy DAQ interface from Mathwork’s Matlab.

|  |  |
| --- | --- |
|  | %classdef AnalogIO < PkgAnalogIO.Fake\_AnalogInAndOut  classdef AnalogIO < PkgAnalogIO.AnalogIn\_mcc & PkgAnalogIO.AnalogOut\_mcc |

Check <http://de.mathworks.com/help/daq/choose-the-right-interface.html> and <http://de.mathworks.com/hardware-support/data-acquistion-software.html> for supported Hardware.

Tracking (most) errors using ErrorHandler

Most IO and other Exceptions are saved in error\_log.mat and hence can be reviewed using static functions of the ErrorHandler class.

View list of last 40 registered Exceptions:

>> ErrorHandler.getList

ans =

Nr Time stamp runstate info - description

1: 30-Mar-2015 16:31:10 running magnetizationLoop - Reference to non-existent field 'customval\_adc'. …

2: 30-Mar-2015 16:34:44 stopped viewFile - Error using MeasurementFile (line 51) | We really need a …

3: 30-Mar-2015 16:40:20 stopped viewFile - Error using MeasurementFile/loadMagnetizationCurve …

4: 02-Apr-2015 17:26:30 stopped adc\_constr - Error using analoginput (line 142) | The specified device ID does …

Detailed Information can be seen by:

>> ErrorHandler.report(ErrorNr)

PkgHardwExt

The classes TempController and CustomController inherit from HWController (the base class) and either

NetIO (Matlab acts as a client and establishes a TCP connection to another Program or Network device) or

ArduinoIO (Matlab communicates over the serial port, e.g. with an Arduino hardware connected to an USB port)

The functions beforeAllLoops, beforeLoop, … are called by the measurement main function Measurement.magnetizationLoop().

For instance: beforeAllLoops is called before the measurement starts while beforeLoop is called for each single loop. This is the place for code which communicates with the additional Hardware and waits until the temperature or something else is stable enough to continue the measurement.

The current measurement value (temperature, …) should be stored in controller\_value.

Config & Important constants

Most application specific properties are covered by the Config class. The Config class untilizes the singleton design pattern, which ensures there is only one instance of the Config class in memory. In this Program this feature is also exploited to store global variables, which can be found in the “Transient” properties section (Transient = will not be saved).

Some constants are spread and can be found in the class desfinitions were they are used:

PkgAdvGUI.SweepTable (Format for clipboard and measurement file):

|  |  |
| --- | --- |
|  | properties (Constant)  % how .data is written in to vsmdat files  dataformat\_header = 'from field [Oe]\t\t to field [Oe]\t\t in steps [Oe]\t\t active [0/1]\r\n'  dataformat\_descriptor = '%14.5e \t %14.5e \t %14.5e \t %14.5e\r\n'  dataformat\_descriptor\_read = '%f %f %f %f %f\r\n'  use\_unofficial\_java = 1 % restore scrollbar position after editing table, etc ..;  % this might cause problems on some  % systems  End |

MagnetizationCurve (Format for storing measured loops in measurement file):

|  |  |
| --- | --- |
|  | properties (Constant)  % how .data is written in to vsmdat files#  dataformat\_header = GuiCfg.file\_dataformat\_header; % moved to GuiCfg; VSM/MOKE edition  dataformat\_descriptor = '%3d %14.5e %14.5e %14.5e %14.5e %14.5e %14.5e %14.5e %14.5e %14.5e\r\n'  dataformat\_descriptor\_read = '%d %f %f %f %f %f %f %f %f %f\r\n'  end |

PkgAdvGUI

Please find usage and code examples inside class comments (Pressing F1 in Matlab or viewing Doxygen files)

An overview can be found in ClassDocu.pptx