
labscript-utils

Release 3.3.0

labscript suite contributors

Apr 18, 2023

DOCUMENTATION

1	The <i>labconfig.ini</i> file	3
1.1	The Default <i>labconfig.ini</i>	3
2	API Reference	5
2.1	<i>labscrip</i> _utils	5
2.2	<i>labscrip</i> _profile	387
3	<i>labscrip</i> suite components	389
	Python Module Index	391
	Index	393

labscript-utils contains code shared between multiple programs in the labscript suite. This documentation is primarily for developers who might want to use this common code in their own custom features.

THE *LABCONFIG.INI* FILE

The `labconfig.ini` file is a global configuration file for your **labscript-suite** installation. It contains configurable settings that govern how the individual components of the suite operate. The name of this file must be the host computer's system name. So if my system's name was `heisenberg`, the `labconfig` file name would be `heisenberg.ini`. This file should be located in the `labscript-suite` directory in the user space, in the `labconfig` subdirectory.

When installing the **labscript-suite** for the first time, running the `labscript-profile-create` command will automatically generate the `labscript-suite` user space directory in the correct place and generate a `labconfig.ini` file for use on your system. By editing the `ini` file named after your system, you can update the configuration settings of your **labscript-suite** installation.

1.1 The Default *labconfig.ini*

Below is a copy of the default lab configuration if you were to install the **labscript-suite** today.

Note: When updates are made to the suite that add or change keys available in the `labconfig`, your local file will **NOT** be automatically updated to include them. Instead, if keys are missing from your local profile, default behavior will be assumed. To implement the added functionality, you will need to manually add/change the keys in your local `labconfig`.

```
[DEFAULT]
apparatus_name = example_apparatus
shared_drive = C:
experiment_shot_storage = %(shared_drive)s\Experiments\%(apparatus_name)s
userlib= %(labscript_suite)s\userlib
pythonlib = %(userlib)s\pythonlib
labscriptlib = %(userlib)s\labscriptlib\%(apparatus_name)s
analysislib = %(userlib)s\analysislib\%(apparatus_name)s
app_saved_configs = %(labscript_suite)s\app_saved_configs\%(apparatus_name)s
user_devices = user_devices

[paths]
connection_table_h5 = %(experiment_shot_storage)s\connection_table.h5
connection_table_py = %(labscriptlib)s\connection_table.py

[servers]
zlock = localhost
runmanager = localhost

[ports]
```

(continues on next page)

```
BLACS = 42517
lyse = 42519
runviewer = 42521
runmanager = 42523
zlock = 7339
zlog = 7340
zprocess_remote = 7341

[timeouts]
communication_timeout = 60

[programs]
text_editor = %%PROGRAMFILES%%\Sublime Text 3\sublime_text.exe
text_editor_arguments = {file}
hdf5_viewer = %%LOCALAPPDATA%%\HDF_Group\HDFView\3.1.0\hdfview.bat
hdf5_viewer_arguments = {file}

[labscript]
save_hg_info = False
save_git_info = False

[BLACS/plugins]
connection_table = True
connection_table.hashable_types = ['.py', '.txt', '.ini', '.json']
connection_table.polling_interval = 1.0

delete_repeated_shots = False
general = True
memory = False
progress_bar = False
theme = True

[lyse]
autoload_config_file = %(app_saved_configs)s\lyse\lyse.ini
integer_indexing = False

[runmanager]
autoload_config_file = %(app_saved_configs)s\runmanager\runmanager.ini
output_folder_format = %%Y\%%m\%%d\{sequence_index:04d}
filename_prefix_format = %%Y-%%m-%%d_{sequence_index:04d}_{script_basename}

[security]
shared_secret = %(labscript_suite)s\labconfig\zpsecret-b810f83f.key
```

API REFERENCE

labscript_utils

labscript_profile

2.1 labscript_utils

Sub-Modules

labscript_utils.camera_server

labscript_utils.connections

labscript_utils.device_registry

labscript_utils.dict_diff

labscript_utils.double_import_denier

labscript_utils.excepthook

labscript_utils.filewatcher

labscript_utils.h5_lock

labscript_utils.impprof

labscript_utils.labconfig

labscript_utils.ls_zprocess

labscript_utils.memprof

labscript_utils.modulewatcher

continues on next page

Table 2.2 – continued from previous page

<i>labscript_utils.properties</i>	
<i>labscript_utils.qtwidgets</i>	
<i>labscript_utils.remote</i>	Script to run a zprocess.remote server configured according to LabConfig.
<i>labscript_utils.settings</i>	
<i>labscript_utils.setup_logging</i>	
<i>labscript_utils.shared_drive</i>	
<i>labscript_utils.shot_utils</i>	
<i>labscript_utils.splash</i>	
<i>labscript_utils.testing_utils</i>	
<i>labscript_utils.tracelog</i>	
<i>labscript_utils.unitconversions</i>	
<i>labscript_utils.versions</i>	
<i>labscript_utils.zlock</i>	Script to run a zlock server configured according to Lab-Config.
<i>labscript_utils.zlog</i>	Script to run a zlog server configured according to Lab-Config.

2.1.1 **labscript_utils.camera_server**

Classes

<i>CameraServer</i> (port)	
<i>TubingenCameraServer</i> (port)	Minimalistic camera server.

labscript_utils.camera_server.CameraServer

```
class labscript_utils.camera_server.CameraServer(port)
    Bases: zprocess.clientserver.ZMQServer
    __init__(port)
```

Methods

<code>__init__(port)</code>	
<code>abort()</code>	To be overridden by subclasses.
<code>handler(request_data)</code>	To be overridden by subclasses.
<code>mainloop()</code>	
<code>setup_auth(context)</code>	Deprecated.
<code>shutdown()</code>	
<code>shutdown_on_interrupt()</code>	
<code>timeout()</code>	A function to call every <code>self.timeout_interval</code> seconds in the same thread as the handler.
<code>transition_to_buffered(h5_filepath)</code>	To be overridden by subclasses.
<code>transition_to_static(h5_filepath)</code>	To be overridden by subclasses.

abort()

To be overridden by subclasses. Return cameras and any other state to one in which `transition_to_buffered()` can be called again. `abort()` will be called if there was an exception in either `transition_to_buffered()` or `transition_to_static()`, and so should ideally be written to return things to a sensible state even if those methods did not complete. Like any cleanup function, `abort()` should proceed to further cleanups even if earlier cleanups fail. As such it should make liberal use of `try: except: blocks`, so that an exception in performing one cleanup operation does not stop it from proceeding to subsequent cleanup operations

handler(request_data)

To be overridden by subclasses. This is an example implementation

transition_to_buffered(h5_filepath)

To be overridden by subclasses. Do any preparatory processing before a shot, eg setting exposure times, readying cameras to receive triggers etc.

transition_to_static(h5_filepath)

To be overridden by subclasses. Do any post processing after a shot, eg computing optical depth, fits, displaying images, saving images and results to the h5 file, returning cameras to an idle state.

labscript_utils.camera_server.TubingenCameraServer

class `labscript_utils.camera_server.TubingenCameraServer(port)`

Bases: `labscript_utils.camera_server.CameraServer`

Minimalistic camera server. Transition to buffered and abort are not implemented, because we don't need to do anything in those cases. This camera server simply writes to the h5 file the images, which have been saved to disk during each shot by an external program.

`__init__(port)`

Methods

<code>__init__(port)</code>	
<code>abort()</code>	To be overridden by subclasses.
<code>handler(request_data)</code>	To be overridden by subclasses.
<code>mainloop()</code>	
<code>setup_auth(context)</code>	Deprecated.
<code>shutdown()</code>	
<code>shutdown_on_interrupt()</code>	
<code>timeout()</code>	A function to call every <code>self.timeout_interval</code> seconds in the same thread as the handler.
<code>transition_to_buffered(h5_filepath)</code>	To be overridden by subclasses.
<code>transition_to_static(h5_filepath)</code>	Read FITS images from file saved by an external program, and save them to the h5 file

`abort()`

To be overridden by subclasses. Return cameras and any other state to one in which `transition_to_buffered()` can be called again. `abort()` will be called if there was an exception in either `transition_to_buffered()` or `transition_to_static()`, and so should ideally be written to return things to a sensible state even if those methods did not complete. Like any cleanup function, `abort()` should proceed to further cleanups even if earlier cleanups fail. As such it should make liberal use of `try: except: blocks`, so that an exception in performing one cleanup operation does not stop it from proceeding to subsequent cleanup operations

`transition_to_buffered(h5_filepath)`

To be overridden by subclasses. Do any preparatory processing before a shot, eg setting exposure times, readying cameras to receive triggers etc.

`transition_to_static(h5_filepath)`

Read FITS images from file saved by an external program, and save them to the h5 file

2.1.2 labscript_utils.connections

Classes

<code>Connection(raw_row)</code>	A class to represent a row in the connection table, present the contents as instance attributes after deserialising their contents, and providing default values for backward compatibility with older HDF5 files.
<code>ConnectionTable(h5file[, logging_prefix, ...])</code>	

labscript_utils.connections.Connection**class** labscript_utils.connections.**Connection**(*raw_row*)Bases: `object`

A class to represent a row in the connection table, present the contents as instance attributes after deserialising their contents, and providing default values for backward compatibility with older HDF5 files. Contains links to Connection objects for child devices of each device

`__init__(raw_row)`**Methods**

`__init__(raw_row)`

`compare_to(other_connection)`

`diff(other)`

`find_by_name(name)`

`find_child(parent_name, parent_port)`

`print_details(indent)`

Attributes

`properties`

`unit_conversion_params`

`compare_to(other_connection)``diff(other)``find_by_name(name)``find_child(parent_name, parent_port)``print_details(indent)``property properties``property unit_conversion_params`

labscript_utils.connections.ConnectionTable

class labscript_utils.connections.**ConnectionTable**(*h5file, logging_prefix=None, exceptions_in_thread=False*)

Bases: `object`

__init__(*h5file, logging_prefix=None, exceptions_in_thread=False*)

Object to represent a connection table. Set logging prefix if you desire logging. Log used will be <prefix>.ConnectionTable

Methods

<code>__init__(h5file[, logging_prefix, ...])</code>	Object to represent a connection table.
<code>assert_superset(other)</code>	
<code>compare_to(other)</code>	
<code>find_by_name(name)</code>	
<code>find_child(parent_name, parent_port)</code>	
<code>get_attached_devices()</code>	Finds out which devices in the connection table are connected to BLACS, based on whether their 'BLACS_connection' attribute is non-empty.
<code>print_details()</code>	
<code>remove_device(device_name)</code>	Removes a device from the ConnectionTable, but keeps it in the raw_table.

assert_superset(*other*)

compare_to(*other*)

find_by_name(*name*)

find_child(*parent_name, parent_port*)

get_attached_devices()

Finds out which devices in the connection table are connected to BLACS, based on whether their 'BLACS_connection' attribute is non-empty. Returns a dictionary of them in the form {device_instance_name: labscript_class_name}

print_details()

remove_device(*device_name*)

Removes a device from the ConnectionTable, but keeps it in the raw_table. This can help make comparisons of connection tables fail for tables with broken devices.

2.1.3 labscript_utils.device_registry

2.1.4 labscript_utils.dict_diff

Functions

<i>dict_diff</i> (dict1, dict2)	Return the difference between two dictionaries as a dictionary of key: [val1, val2] pairs.
---------------------------------	--

labscript_utils.dict_diff.dict_diff

`labscript_utils.dict_diff.dict_diff(dict1, dict2)`

Return the difference between two dictionaries as a dictionary of key: [val1, val2] pairs. Keys unique to either dictionary are included as key: [val1, '-'] or key: ['- ', val2].

2.1.5 labscript_utils.double_import_denier

Functions

disable()

enable()

labscript_utils.double_import_denier.disable

`labscript_utils.double_import_denier.disable()`

labscript_utils.double_import_denier.enable

`labscript_utils.double_import_denier.enable()`

Classes

<i>DoubleImportDenier</i> ()	A module finder that tracks what's been imported and disallows multiple imports of the same module under different names, raising an exception upon detecting that this has occurred
------------------------------	--

labscript_utils.double_import_denier.DoubleImportDenier

class labscript_utils.double_import_denier.**DoubleImportDenier**

Bases: `object`

A module finder that tracks what's been imported and disallows multiple imports of the same module under different names, raising an exception upon detecting that this has occurred

`__init__()`

Methods

`__init__()`

`find_spec(fullname[, path, target])`

find_spec(*fullname*, *path=None*, *target=None*)

2.1.6 labscript_utils.excepthook

Sub-Modules

`labscript_utils.excepthook.tk_exception`

labscript_utils.excepthook.tk_exception

Classes

`ErrorWindow([master])` Class to display the error in a textbox.

labscript_utils.excepthook.tk_exception.ErrorWindow

class labscript_utils.excepthook.tk_exception.**ErrorWindow**(*master=None*, ***kw*)

Bases: `tkinter.Text`

Class to display the error in a textbox. Parts copied from Tkinter's ScrolledText widget

`__init__(master=None, **kw)`

Construct a text widget with the parent MASTER.

STANDARD OPTIONS

background, borderwidth, cursor, exportselection, font, foreground, highlightbackground, highlightcolor, highlightthickness, insertbackground, insertborderwidth, insertofftime, insertontime, insertwidth, padx, pady, relief, selectbackground, selectborderwidth, selectforeground, setgrid, takefocus, xscrollcommand, yscrollcommand,

WIDGET-SPECIFIC OPTIONS

autoseparators, height, maxundo, spacing1, spacing2, spacing3, state, tabs, undo, width, wrap,

Methods

<code>__init__([master])</code>	Construct a text widget with the parent MASTER.
<code>after(ms[, func])</code>	Call function once after given time.
<code>after_cancel(id)</code>	Cancel scheduling of function identified with ID.
<code>after_idle(func, *args)</code>	Call FUNC once if the Tcl main loop has no event to process.
<code>anchor([anchor])</code>	The anchor value controls how to place the grid within the master when no row/column has any weight.
<code>bbox(index)</code>	Return a tuple of (x,y,width,height) which gives the bounding box of the visible part of the character at the given index.
<code>bell([displayof])</code>	Ring a display's bell.
<code>bind([sequence, func, add])</code>	Bind to this widget at event SEQUENCE a call to function FUNC.
<code>bind_all([sequence, func, add])</code>	Bind to all widgets at an event SEQUENCE a call to function FUNC.
<code>bind_class(className[, sequence, func, add])</code>	Bind to widgets with bindtag CLASSNAME at event SEQUENCE a call of function FUNC.
<code>bindtags([tagList])</code>	Set or get the list of bindtags for this widget.
<code>cget(key)</code>	Return the resource value for a KEY given as string.
<code>clipboard_append(string, **kw)</code>	Append STRING to the Tk clipboard.
<code>clipboard_clear(**kw)</code>	Clear the data in the Tk clipboard.
<code>clipboard_get(**kw)</code>	Retrieve data from the clipboard on window's display.
<code>columnconfigure(index[, cnf])</code>	Configure column INDEX of a grid.
<code>compare(index1, op, index2)</code>	Return whether between index INDEX1 and index INDEX2 the relation OP is satisfied.
<code>config([cnf])</code>	Configure resources of a widget.
<code>configure([cnf])</code>	Configure resources of a widget.
<code>copy([event])</code>	
<code>count(index1, index2, *args)</code>	Counts the number of relevant things between the two indices.
<code>debug([boolean])</code>	Turn on the internal consistency checks of the B-Tree inside the text widget according to BOOLEAN.
<code>delete(index1[, index2])</code>	Delete the characters between INDEX1 and INDEX2 (not included).
<code>deletecommand(name)</code>	Internal function.
<code>destroy()</code>	Destroy this and all descendants widgets.
<code>dlineinfo(index)</code>	Return tuple (x,y,width,height,baseline) giving the bounding box and baseline position of the visible part of the line containing the character at INDEX.
<code>dump(index1[, index2, command])</code>	Return the contents of the widget between index 1 and index2.
<code>edit(*args)</code>	Internal method
<code>edit_modified([arg])</code>	Get or Set the modified flag
<code>edit_redo()</code>	Redo the last undone edit

continues on next page

Table 2.16 – continued from previous page

<code>edit_reset()</code>	Clears the undo and redo stacks
<code>edit_separator()</code>	Inserts a separator (boundary) on the undo stack.
<code>edit_undo()</code>	Undoes the last edit action
<code>event_add(virtual, *sequences)</code>	Bind a virtual event VIRTUAL (of the form <<Name>>) to an event SEQUENCE such that the virtual event is triggered whenever SEQUENCE occurs.
<code>event_delete(virtual, *sequences)</code>	Unbind a virtual event VIRTUAL from SEQUENCE.
<code>event_generate(sequence, **kw)</code>	Generate an event SEQUENCE.
<code>event_info([virtual])</code>	Return a list of all virtual events or the information about the SEQUENCE bound to the virtual event VIRTUAL.
<code>focus()</code>	Direct input focus to this widget.
<code>focus_displayof()</code>	Return the widget which has currently the focus on the display where this widget is located.
<code>focus_force()</code>	Direct input focus to this widget even if the application does not have the focus.
<code>focus_get()</code>	Return the widget which has currently the focus in the application.
<code>focus_lastfor()</code>	Return the widget which would have the focus if top level for this widget gets the focus from the window manager.
<code>focus_set()</code>	Direct input focus to this widget.
<code>forget()</code>	Unmap this widget and do not use it for the packing order.
<code>get(index1[, index2])</code>	Return the text from INDEX1 to INDEX2 (not included).
<code>getboolean(s)</code>	Return a boolean value for Tcl boolean values true and false given as parameter.
<code>getdouble(s)</code>	
<code>getint(s)</code>	
<code>getvar([name])</code>	Return value of Tcl variable NAME.
<code>grab_current()</code>	Return widget which has currently the grab in this application or None.
<code>grab_release()</code>	Release grab for this widget if currently set.
<code>grab_set()</code>	Set grab for this widget.
<code>grab_set_global()</code>	Set global grab for this widget.
<code>grab_status()</code>	Return None, "local" or "global" if this widget has no, a local or a global grab.

continues on next page

Table 2.16 – continued from previous page

<code>grid([cnf])</code>	Position a widget in the parent widget in a grid. Use as options: <code>column=number</code> - use cell identified with given column (starting with 0) <code>columnspan=number</code> - this widget will span several columns <code>in=master</code> - use master to contain this widget <code>in_=master</code> - see 'in' option description <code>ipadx=amount</code> - add internal padding in x direction <code>ipady=amount</code> - add internal padding in y direction <code>padx=amount</code> - add padding in x direction <code>pady=amount</code> - add padding in y direction <code>row=number</code> - use cell identified with given row (starting with 0) <code>rowspan=number</code> - this widget will span several rows <code>sticky=NSEW</code> - if cell is larger on which sides will this widget stick to the cell boundary.
<code>grid_anchor([anchor])</code>	The anchor value controls how to place the grid within the master when no row/column has any weight.
<code>grid_bbox([column, row, col2, row2])</code>	Return a tuple of integer coordinates for the bounding box of this widget controlled by the geometry manager grid.
<code>grid_columnconfigure(index[, cnf])</code>	Configure column INDEX of a grid.
<code>grid_configure([cnf])</code>	Position a widget in the parent widget in a grid. Use as options: <code>column=number</code> - use cell identified with given column (starting with 0) <code>columnspan=number</code> - this widget will span several columns <code>in=master</code> - use master to contain this widget <code>in_=master</code> - see 'in' option description <code>ipadx=amount</code> - add internal padding in x direction <code>ipady=amount</code> - add internal padding in y direction <code>padx=amount</code> - add padding in x direction <code>pady=amount</code> - add padding in y direction <code>row=number</code> - use cell identified with given row (starting with 0) <code>rowspan=number</code> - this widget will span several rows <code>sticky=NSEW</code> - if cell is larger on which sides will this widget stick to the cell boundary.
<code>grid_forget()</code>	Unmap this widget.
<code>grid_info()</code>	Return information about the options for positioning this widget in a grid.
<code>grid_location(x, y)</code>	Return a tuple of column and row which identify the cell at which the pixel at position X and Y inside the master widget is located.
<code>grid_propagate([flag])</code>	Set or get the status for propagation of geometry information.
<code>grid_remove()</code>	Unmap this widget but remember the grid options.
<code>grid_rowconfigure(index[, cnf])</code>	Configure row INDEX of a grid.
<code>grid_size()</code>	Return a tuple of the number of column and rows in the grid.
<code>grid_slaves([row, column])</code>	Return a list of all slaves of this widget in its packing order.
<code>image_cget(index, option)</code>	Return the value of OPTION of an embedded image at INDEX.
<code>image_configure(index[, cnf])</code>	Configure an embedded image at INDEX.
<code>image_create(index[, cnf])</code>	Create an embedded image at INDEX.

continues on next page

Table 2.16 – continued from previous page

<code>image_names()</code>	Return all names of embedded images in this widget.
<code>image_types()</code>	Return a list of all available image types (e.g.
<code>index(index)</code>	Return the index in the form line.char for INDEX.
<code>info()</code>	Return information about the packing options for this widget.
<code>insert(index, chars, *args)</code>	Insert CHARS before the characters at INDEX.
<code>keys()</code>	Return a list of all resource names of this widget.
<code>lift([aboveThis])</code>	Raise this widget in the stacking order.
<code>location(x, y)</code>	Return a tuple of column and row which identify the cell at which the pixel at position X and Y inside the master widget is located.
<code>lower([belowThis])</code>	Lower this widget in the stacking order.
<code>mainloop([n])</code>	Call the mainloop of Tk.
<code>mark_gravity(markName[, direction])</code>	Change the gravity of a mark MARKNAME to DIRECTION (LEFT or RIGHT).
<code>mark_names()</code>	Return all mark names.
<code>mark_next(index)</code>	Return the name of the next mark after INDEX.
<code>mark_previous(index)</code>	Return the name of the previous mark before INDEX.
<code>mark_set(markName, index)</code>	Set mark MARKNAME before the character at INDEX.
<code>mark_unset(*markNames)</code>	Delete all marks in MARKNAMES.
<code>nametowidget(name)</code>	Return the Tkinter instance of a widget identified by its Tcl name NAME.
<code>ok_clicked(*ignore)</code>	
<code>option_add(pattern, value[, priority])</code>	Set a VALUE (second parameter) for an option PATTERN (first parameter).
<code>option_clear()</code>	Clear the option database.
<code>option_get(name, className)</code>	Return the value for an option NAME for this widget with CLASSNAME.
<code>option_readfile(fileName[, priority])</code>	Read file FILENAME into the option database.
<code>pack([cnf])</code>	Pack a widget in the parent widget. Use as options: after=widget - pack it after you have packed widget anchor=NSEW (or subset) - position widget according to given direction before=widget - pack it before you will pack widget expand=bool - expand widget if parent size grows fill=NONE or X or Y or BOTH - fill widget if widget grows in=master - use master to contain this widget in_=master - see 'in' option description ipadx=amount - add internal padding in x direction ipady=amount - add internal padding in y direction padx=amount - add padding in x direction pady=amount - add padding in y direction side=TOP or BOTTOM or LEFT or RIGHT - where to add this widget.

continues on next page

Table 2.16 – continued from previous page

<code>pack_configure([cnf])</code>	Pack a widget in the parent widget. Use as options: <code>after=widget</code> - pack it after you have packed widget <code>anchor=NSEW</code> (or subset) - position widget according to given direction <code>before=widget</code> - pack it before you will pack widget <code>expand=bool</code> - expand widget if parent size grows <code>fill=NONE</code> or <code>X</code> or <code>Y</code> or <code>BOTH</code> - fill widget if widget grows <code>in=master</code> - use master to contain this widget <code>in_=master</code> - see 'in' option description <code>ipadx=amount</code> - add internal padding in x direction <code>ipady=amount</code> - add internal padding in y direction <code>padx=amount</code> - add padding in x direction <code>pady=amount</code> - add padding in y direction <code>side=TOP</code> or <code>BOTTOM</code> or <code>LEFT</code> or <code>RIGHT</code> - where to add this widget.
<code>pack_forget()</code>	Unmap this widget and do not use it for the packing order.
<code>pack_info()</code>	Return information about the packing options for this widget.
<code>pack_propagate([flag])</code>	Set or get the status for propagation of geometry information.
<code>pack_slaves()</code>	Return a list of all slaves of this widget in its packing order.
<code>peer_create(newPathName[, cnf])</code>	Creates a peer text widget with the given <code>newPathName</code> , and any optional standard configuration options.
<code>peer_names()</code>	Returns a list of peers of this widget (this does not include the widget itself).
<code>place([cnf])</code>	Place a widget in the parent widget. Use as options: <code>in=master</code> - master relative to which the widget is placed <code>in_=master</code> - see 'in' option description <code>x=amount</code> - locate anchor of this widget at position x of master <code>y=amount</code> - locate anchor of this widget at position y of master <code>relx=amount</code> - locate anchor of this widget between 0.0 and 1.0 relative to width of master (1.0 is right edge) <code>rely=amount</code> - locate anchor of this widget between 0.0 and 1.0 relative to height of master (1.0 is bottom edge) <code>anchor=NSEW</code> (or subset) - position anchor according to given direction <code>width=amount</code> - width of this widget in pixel <code>height=amount</code> - height of this widget in pixel <code>relwidth=amount</code> - width of this widget between 0.0 and 1.0 relative to width of master (1.0 is the same width as the master) <code>relheight=amount</code> - height of this widget between 0.0 and 1.0 relative to height of master (1.0 is the same height as the master) <code>border-mode="inside"</code> or <code>"outside"</code> - whether to take border width of master widget into account.

continues on next page

Table 2.16 – continued from previous page

<code>place_configure([cnf])</code>	Place a widget in the parent widget. Use as options: <code>in=master</code> - master relative to which the widget is placed <code>in_=master</code> - see 'in' option description <code>x=amount</code> - locate anchor of this widget at position <code>x</code> of master <code>y=amount</code> - locate anchor of this widget at position <code>y</code> of master <code>relx=amount</code> - locate anchor of this widget between 0.0 and 1.0 relative to width of master (1.0 is right edge) <code>rely=amount</code> - locate anchor of this widget between 0.0 and 1.0 relative to height of master (1.0 is bottom edge) <code>anchor=NSEW</code> (or subset) - position anchor according to given direction <code>width=amount</code> - width of this widget in pixel <code>height=amount</code> - height of this widget in pixel <code>relwidth=amount</code> - width of this widget between 0.0 and 1.0 relative to width of master (1.0 is the same width as the master) <code>relheight=amount</code> - height of this widget between 0.0 and 1.0 relative to height of master (1.0 is the same height as the master) <code>border-mode="inside" or "outside"</code> - whether to take border width of master widget into account.
<code>place_forget()</code>	Unmap this widget.
<code>place_info()</code>	Return information about the placing options for this widget.
<code>place_slaves()</code>	Return a list of all slaves of this widget in its packing order.
<code>propagate([flag])</code>	Set or get the status for propagation of geometry information.
<code>quit()</code>	Quit the Tcl interpreter.
<code>register(func[, subst, needcleanup])</code>	Return a newly created Tcl function.
<code>replace(index1, index2, chars, *args)</code>	Replaces the range of characters between <code>index1</code> and <code>index2</code> with the given characters and tags specified by <code>args</code> .
<code>rowconfigure(index[, cnf])</code>	Configure row <code>INDEX</code> of a grid.
<code>scan_dragto(x, y)</code>	Adjust the view of the text to 10 times the difference between <code>X</code> and <code>Y</code> and the coordinates given in <code>scan_mark</code> .
<code>scan_mark(x, y)</code>	Remember the current <code>X</code> , <code>Y</code> coordinates.
<code>search(pattern, index[, stopindex, ...])</code>	Search <code>PATTERN</code> beginning from <code>INDEX</code> until <code>STOPINDEX</code> .
<code>see(index)</code>	Scroll such that the character at <code>INDEX</code> is visible.
<code>selection_clear(**kw)</code>	Clear the current <code>X</code> selection.
<code>selection_get(**kw)</code>	Return the contents of the current <code>X</code> selection.
<code>selection_handle(command, **kw)</code>	Specify a function <code>COMMAND</code> to call if the <code>X</code> selection owned by this widget is queried by another application.
<code>selection_own(**kw)</code>	Become owner of <code>X</code> selection.
<code>selection_own_get(**kw)</code>	Return owner of <code>X</code> selection.
<code>send(interp, cmd, *args)</code>	Send Tcl command <code>CMD</code> to different interpreter <code>INTERP</code> to be executed.
<code>setvar([name, value])</code>	Set Tcl variable <code>NAME</code> to <code>VALUE</code> .

continues on next page

Table 2.16 – continued from previous page

size()	Return a tuple of the number of column and rows in the grid.
slaves()	Return a list of all slaves of this widget in its packing order.
tag_add(tagName, index1, *args)	Add tag TAGNAME to all characters between INDEX1 and index2 in ARGS.
tag_bind(tagName, sequence, func[, add])	Bind to all characters with TAGNAME at event SEQUENCE a call to function FUNC.
tag_cget(tagName, option)	Return the value of OPTION for tag TAGNAME.
tag_config(tagName[, cnf])	Configure a tag TAGNAME.
tag_configure(tagName[, cnf])	Configure a tag TAGNAME.
tag_delete(*tagNames)	Delete all tags in TAGNAMES.
tag_lower(tagName[, belowThis])	Change the priority of tag TAGNAME such that it is lower than the priority of BELOWTHIS.
tag_names([index])	Return a list of all tag names.
tag_nextrange(tagName, index1[, index2])	Return a list of start and end index for the first sequence of characters between INDEX1 and INDEX2 which all have tag TAGNAME.
tag_prevrange(tagName, index1[, index2])	Return a list of start and end index for the first sequence of characters between INDEX1 and INDEX2 which all have tag TAGNAME.
tag_raise(tagName[, aboveThis])	Change the priority of tag TAGNAME such that it is higher than the priority of ABOVETHIS.
tag_ranges(tagName)	Return a list of ranges of text which have tag TAGNAME.
tag_remove(tagName, index1[, index2])	Remove tag TAGNAME from all characters between INDEX1 and INDEX2.
tag_unbind(tagName, sequence[, funcid])	Unbind for all characters with TAGNAME for event SEQUENCE the function identified with FUNCID.
tk_bisque()	Change the color scheme to light brown as used in Tk 3.6 and before.
tk_focusFollowsMouse()	The widget under mouse will get automatically focus.
tk_focusNext()	Return the next widget in the focus order which follows widget which has currently the focus.
tk_focusPrev()	Return previous widget in the focus order.
tk_setPalette(*args, **kw)	Set a new color scheme for all widget elements.
tk_strictMotif([boolean])	Set Tcl internal variable, whether the look and feel should adhere to Motif.
tkraise([aboveThis])	Raise this widget in the stacking order.
unbind(sequence[, funcid])	Unbind for this widget for event SEQUENCE the function identified with FUNCID.
unbind_all(sequence)	Unbind for all widgets for event SEQUENCE all functions.
unbind_class(className, sequence)	Unbind for all widgets with bindtag CLASSNAME for event SEQUENCE all functions.
update()	Enter event loop until all pending events have been processed by Tcl.
update_idletasks()	Enter event loop until all idle callbacks have been called.
wait_variable([name])	Wait until the variable is modified.
wait_visibility([window])	Wait until the visibility of a WIDGET changes (e.g.

continues on next page

Table 2.16 – continued from previous page

<code>wait_window([window])</code>	Wait until a WIDGET is destroyed.
<code>waitvar([name])</code>	Wait until the variable is modified.
<code>window_cget(index, option)</code>	Return the value of OPTION of an embedded window at INDEX.
<code>window_config(index[, cnf])</code>	Configure an embedded window at INDEX.
<code>window_configure(index[, cnf])</code>	Configure an embedded window at INDEX.
<code>window_create(index[, cnf])</code>	Create a window at INDEX.
<code>window_names()</code>	Return all names of embedded windows in this widget.
<code>winfo_atom(name[, displayof])</code>	Return integer which represents atom NAME.
<code>winfo_atomname(id[, displayof])</code>	Return name of atom with identifier ID.
<code>winfo_cells()</code>	Return number of cells in the colormap for this widget.
<code>winfo_children()</code>	Return a list of all widgets which are children of this widget.
<code>winfo_class()</code>	Return window class name of this widget.
<code>winfo_colormapfull()</code>	Return True if at the last color request the colormap was full.
<code>winfo_containing(rootX, rootY[, displayof])</code>	Return the widget which is at the root coordinates ROOTX, ROOTY.
<code>winfo_depth()</code>	Return the number of bits per pixel.
<code>winfo_exists()</code>	Return true if this widget exists.
<code>winfo_fpixels(number)</code>	Return the number of pixels for the given distance NUMBER (e.g.
<code>winfo_geometry()</code>	Return geometry string for this widget in the form "widthxheight+X+Y".
<code>winfo_height()</code>	Return height of this widget.
<code>winfo_id()</code>	Return identifier ID for this widget.
<code>winfo_interps([displayof])</code>	Return the name of all Tcl interpreters for this display.
<code>winfo_ismapped()</code>	Return true if this widget is mapped.
<code>winfo_manager()</code>	Return the window manager name for this widget.
<code>winfo_name()</code>	Return the name of this widget.
<code>winfo_parent()</code>	Return the name of the parent of this widget.
<code>winfo_pathname(id[, displayof])</code>	Return the pathname of the widget given by ID.
<code>winfo_pixels(number)</code>	Rounded integer value of <code>winfo_fpixels</code> .
<code>winfo_pointerx()</code>	Return the x coordinate of the pointer on the root window.
<code>winfo_pointerxy()</code>	Return a tuple of x and y coordinates of the pointer on the root window.
<code>winfo_pointery()</code>	Return the y coordinate of the pointer on the root window.
<code>winfo_reqheight()</code>	Return requested height of this widget.
<code>winfo_reqwidth()</code>	Return requested width of this widget.
<code>winfo_rgb(color)</code>	Return tuple of decimal values for red, green, blue for COLOR in this widget.
<code>winfo_rootx()</code>	Return x coordinate of upper left corner of this widget on the root window.
<code>winfo_rooty()</code>	Return y coordinate of upper left corner of this widget on the root window.
<code>winfo_screen()</code>	Return the screen name of this widget.

continues on next page

Table 2.16 – continued from previous page

<code>winfo_screencells()</code>	Return the number of the cells in the colormap of the screen of this widget.
<code>winfo_screendepth()</code>	Return the number of bits per pixel of the root window of the screen of this widget.
<code>winfo_screenheight()</code>	Return the number of pixels of the height of the screen of this widget in pixel.
<code>winfo_screenmmheight()</code>	Return the number of pixels of the height of the screen of this widget in mm.
<code>winfo_screenmmwidth()</code>	Return the number of pixels of the width of the screen of this widget in mm.
<code>winfo_screenvisual()</code>	Return one of the strings <code>directcolor</code> , <code>grayscale</code> , <code>pseudocolor</code> , <code>staticcolor</code> , <code>staticgray</code> , or <code>truecolor</code> for the default colormap of this screen.
<code>winfo_screenwidth()</code>	Return the number of pixels of the width of the screen of this widget in pixel.
<code>winfo_server()</code>	Return information of the X-Server of the screen of this widget in the form "XmajorRminor vendor vendorVersion".
<code>winfo_toplevel()</code>	Return the toplevel widget of this widget.
<code>winfo_viewable()</code>	Return true if the widget and all its higher ancestors are mapped.
<code>winfo_visual()</code>	Return one of the strings <code>directcolor</code> , <code>grayscale</code> , <code>pseudocolor</code> , <code>staticcolor</code> , <code>staticgray</code> , or <code>truecolor</code> for the colormap of this widget.
<code>winfo_visualid()</code>	Return the X identifier for the visual for this widget.
<code>winfo_visualsavailable([includeids])</code>	Return a list of all visuals available for the screen of this widget.
<code>winfo_vrootheight()</code>	Return the height of the virtual root window associated with this widget in pixels.
<code>winfo_vrootwidth()</code>	Return the width of the virtual root window associated with this widget in pixel.
<code>winfo_vrootx()</code>	Return the x offset of the virtual root relative to the root window of the screen of this widget.
<code>winfo_vrooty()</code>	Return the y offset of the virtual root relative to the root window of the screen of this widget.
<code>winfo_width()</code>	Return the width of this widget.
<code>winfo_x()</code>	Return the x coordinate of the upper left corner of this widget in the parent.
<code>winfo_y()</code>	Return the y coordinate of the upper left corner of this widget in the parent.
<code>xview(*args)</code>	Query and change the horizontal position of the view.
<code>xview_moveto(fraction)</code>	Adjusts the view in the window so that FRACTION of the total width of the canvas is off-screen to the left.
<code>xview_scroll(number, what)</code>	Shift the x-view according to NUMBER which is measured in "units" or "pages" (WHAT).
<code>yview(*args)</code>	Query and change the vertical position of the view.
<code>yview_moveto(fraction)</code>	Adjusts the view in the window so that FRACTION of the total height of the canvas is off-screen to the top.
<code>yview_pickplace(*what)</code>	Obsolete function, use <code>see</code> .

continues on next page

Table 2.16 – continued from previous page

<code>yview_scroll(number, what)</code>	Shift the y-view according to NUMBER which is measured in "units" or "pages" (WHAT).
---	--

`copy(event=None)``ok_clicked(*ignore)`

Functions

<code>install_thread_excepthook()</code>	Workaround for sys.excepthook thread bug (https://sourceforge.net/tracker/?func=detail&atid=105470&aid=1230540&group_id=5470).
<code>logwarning(message, category, filename, lineno)</code>	
<code>set_logger(logger)</code>	
<code>tkhandler(exceptclass, exception, exec_info)</code>	

labscript_utils.excepthook.install_thread_excepthook

`labscript_utils.excepthook.install_thread_excepthook()`

Workaround for sys.excepthook thread bug (https://sourceforge.net/tracker/?func=detail&atid=105470&aid=1230540&group_id=5470). Call once from `__main__` before creating any threads.

labscript_utils.excepthook.logwarning

`labscript_utils.excepthook.logwarning(message, category, filename, lineno, file=None, line=None)`

labscript_utils.excepthook.set_logger

`labscript_utils.excepthook.set_logger(logger)`

labscript_utils.excepthook.tkhandler

`labscript_utils.excepthook.tkhandler(exceptclass, exception, exec_info, reraise=True)`

Classes

`I()`

labscript_utils.excepthook.I**class** labscript_utils.excepthook.IBases: `object``__init__()`**Methods**`__init__()`**Attributes**`logger``logger = None`**2.1.7 labscript_utils.filewatcher****Classes**`FileWatcher`(callback[, files, folders, ...])**labscript_utils.filewatcher.FileWatcher****class** labscript_utils.filewatcher.**FileWatcher**(*callback, files=None, folders=None, clean_modified_info=None, hashable_types=None, interval=1, **kwargs*)Bases: `object``__init__(callback, files=None, folders=None, clean_modified_info=None, hashable_types=None, interval=1, **kwargs)`

Detect modification, deletion, creation, or restoration of specific files (and all files in specific folders).

Parameters

- **callback** (*function*) – elicited whenever file events are detected, requires at least (name, info) arguments. Event specific callback requires (name, info, event) arguments, where event is on of: ‘modified’, ‘deleted’ (or None), ‘created’, ‘restored’, ‘original’ The ‘original’ event corresponds to a state change that results in the original file info at instantiation.
- **files** (*list, optional*) – List of specific files to watch. A single file can be specified as a string (default None).
- **folders** (*list, optional*) – List of specific folders to watch. A single folder can be specified as a string (default None). If a file is created/deleted in/from any watched folder, it is added/ removed to/from the FileWatcher.files attribute.

- **clean_modified_info** (*dict*, *optional*) – File info to detect modification/restoration with respect to. If None (default), or for files not present in clean_modified_info, the initial modified info will be based on the first polling of files.
- **hashable_types** (*iterable*, *optional*) – File extensions for which MD5 checksum will be used to detect modification/restoration with (default None). Files of any other type will be watched using their modified time. Restoration cannot be detected for types not in hashable_types.
- **interval** (*float*, *optional*) – Polling interval in seconds (default 1).

Methods

<code>__init__(callback[, files, folders, ...])</code>	Detect modification, deletion, creation, or restoration of specific files (and all files in specific folders).
<code>add_file(path)</code>	
<code>add_files(files[, clean_modified_info])</code>	
<code>add_folder(folder)</code>	
<code>add_folders(folders[, clean_modified_info])</code>	
<code>check([trigger_callback])</code>	
<code>get_clean_modified_info()</code>	
<code>get_modified_info()</code>	
<code>get_modified_times()</code>	
<code>mainloop()</code>	
<code>stop()</code>	
<code>update_files([folders, trigger_callback, ...])</code>	Refresh the watchlist of files (FileWatcher.files) by checking the folders kwarg or Filewatcher.folders if this is not specified.

```

add_file(path)
add_files(files, clean_modified_info=None)
add_folder(folder)
add_folders(folders, clean_modified_info=None)
check(trigger_callback=True)
get_clean_modified_info()
get_modified_info()
get_modified_times()
mainloop()
    
```

stop()

update_files(*folders=None, trigger_callback=True, recursive=True*)

Refresh the watchlist of files (FileWatcher.files) by checking the folders kwarg or Filewatcher.folders if this is not specified.

2.1.8 labscript_utils.h5_lock

Functions

hack_locks_onto_h5py()

labscript_utils.h5_lock.hack_locks_onto_h5py

labscript_utils.h5_lock.hack_locks_onto_h5py()

Classes

File(name[, mode, driver, libver])

labscript_utils.h5_lock.File

class labscript_utils.h5_lock.**File**(*name, mode=None, driver=None, libver=None, **kwds*)

Bases: h5py._hl.files.File

__init__(*name, mode=None, driver=None, libver=None, **kwds*)

Create a new file object.

See the h5py user guide for a detailed explanation of the options.

name Name of the file on disk, or file-like object. Note: for files created with the ‘core’ driver, HDF5 still requires this be non-empty.

mode r Readonly, file must exist (default) r+ Read/write, file must exist w Create file, truncate if exists w- or x Create file, fail if exists a Read/write if exists, create otherwise

driver Name of the driver to use. Legal values are None (default, recommended), ‘core’, ‘sec2’, ‘direct’, ‘stdio’, ‘mpio’, ‘ros3’.

libver Library version bounds. Supported values: ‘earliest’, ‘v108’, ‘v110’, ‘v112’ and ‘latest’. The ‘v108’, ‘v110’ and ‘v112’ options can only be specified with the HDF5 1.10.2 library or later.

userblock_size Desired size of user block. Only allowed when creating a new file (mode w, w- or x).

swmr Open the file in SWMR read mode. Only used when mode = ‘r’.

rdcc_nbytes Total size of the dataset chunk cache in bytes. The default size is 1024**2 (1 MiB) per dataset. Applies to all datasets unless individually changed.

rdcc_w0 The chunk preemption policy for all datasets. This must be between 0 and 1 inclusive and indicates the weighting according to which chunks which have been fully read or written are penalized when determining which chunks to flush from cache. A value of 0 means fully read or written chunks

are treated no differently than other chunks (the preemption is strictly LRU) while a value of 1 means fully read or written chunks are always preempted before other chunks. If your application only reads or writes data once, this can be safely set to 1. Otherwise, this should be set lower depending on how often you re-read or re-write the same data. The default value is 0.75. Applies to all datasets unless individually changed.

rdcc_nslots The number of chunk slots in the raw data chunk cache for this file. Increasing this value reduces the number of cache collisions, but slightly increases the memory used. Due to the hashing strategy, this value should ideally be a prime number. As a rule of thumb, this value should be at least 10 times the number of chunks that can fit in rdcc_nbytes bytes. For maximum performance, this value should be set approximately 100 times that number of chunks. The default value is 521. Applies to all datasets unless individually changed.

track_order Track dataset/group/attribute creation order under root group if True. If None use global default `h5.get_config().track_order`.

fs_strategy The file space handling strategy to be used. Only allowed when creating a new file (mode w, w- or x). Defined as: “fsm” FSM, Aggregators, VFD “page” Paged FSM, VFD “aggregate” Aggregators, VFD “none” VFD If None use HDF5 defaults.

fs_page_size File space page size in bytes. Only used when `fs_strategy=“page”`. If None use the HDF5 default (4096 bytes).

fs_persist A boolean value to indicate whether free space should be persistent or not. Only allowed when creating a new file. The default value is False.

fs_threshold The smallest free-space section size that the free space manager will track. Only allowed when creating a new file. The default value is 1.

page_buf_size Page buffer size in bytes. Only allowed for HDF5 files created with `fs_strategy=“page”`. Must be a power of two value and greater or equal than the file space page size when creating the file. It is not used by default.

min_meta_keep Minimum percentage of metadata to keep in the page buffer before allowing pages containing metadata to be evicted. Applicable only if `page_buf_size` is set. Default value is zero.

min_raw_keep Minimum percentage of raw data to keep in the page buffer before allowing pages containing raw data to be evicted. Applicable only if `page_buf_size` is set. Default value is zero.

locking The file locking behavior. Defined as:

- False (or “false”) – Disable file locking
- True (or “true”) – Enable file locking
- “best-effort” – Enable file locking but ignore some errors
- None – Use HDF5 defaults

Warning: The `HDF5_USE_FILE_LOCKING` environment variable can override this parameter.

Only available with HDF5 $\geq 1.12.1$ or $1.10.x \geq 1.10.7$.

alignment_threshold Together with `alignment_interval`, this property ensures that any file object greater than or equal in size to the alignment threshold (in bytes) will be aligned on an address which is a multiple of alignment interval.

alignment_interval This property should be used in conjunction with `alignment_threshold`. See the description above. For more details, see https://portal.hdfgroup.org/display/HDF5/H5P_SET_ALIGNMENT

meta_block_size Set the current minimum size, in bytes, of new metadata block allocations. See https://portal.hdfgroup.org/display/HDF5/H5P_SET_META_BLOCK_SIZE

Additional keywords Passed on to the selected file driver.

Methods

<code>__init__(name[, mode, driver, libver])</code>	Create a new file object.
<code>build_virtual_dataset(name, shape, dtype[, ...])</code>	Assemble a virtual dataset in this group.
<code>clear()</code>	
<code>close()</code>	Close the file.
<code>copy(source, dest[, name, shallow, ...])</code>	Copy an object or group.
<code>create_dataset(name[, shape, dtype, data])</code>	Create a new HDF5 dataset
<code>create_dataset_like(name, other, **kwupdate)</code>	Create a dataset similar to <code>other</code> .
<code>create_group(name[, track_order])</code>	Create and return a new subgroup.
<code>create_virtual_dataset(name, layout[, fill-value])</code>	Create a new virtual dataset in this group.
<code>flush()</code>	Tell the HDF5 library to flush its buffers.
<code>get(name[, default, getclass, getlink])</code>	Retrieve an item or other information.
<code>items()</code>	Get a view object on member items
<code>keys()</code>	Get a view object on member names
<code>move(source, dest)</code>	Move a link to a new location in the file.
<code>pop(k[,d])</code>	If key is not found, <code>d</code> is returned if given, otherwise <code>KeyError</code> is raised.
<code>popitem()</code>	as a 2-tuple; but raise <code>KeyError</code> if <code>D</code> is empty.
<code>require_dataset(name, shape, dtype[, exact])</code>	Open a dataset, creating it if it doesn't exist.
<code>require_group(name)</code>	Return a group, creating it if it doesn't exist.
<code>setdefault(k[,d])</code>	
<code>update([E,]**F)</code>	If <code>E</code> present and has a <code>.keys()</code> method, does: for <code>k</code> in <code>E</code> : <code>D[k] = E[k]</code> If <code>E</code> present and lacks <code>.keys()</code> method, does: for <code>(k, v)</code> in <code>E</code> : <code>D[k] = v</code> In either case, this is followed by: for <code>k, v</code> in <code>F.items()</code> : <code>D[k] = v</code>
<code>values()</code>	Get a view object on member objects
<code>visit(func)</code>	Recursively visit all names in this group and sub-groups (HDF5 1.8).
<code>visititems(func)</code>	Recursively visit names and objects in this group (HDF5 1.8).

Attributes

<code>attrs</code>	Attributes attached to this object
<code>driver</code>	Low-level HDF5 file driver used to open file
<code>file</code>	Return a File instance associated with this object
<code>filename</code>	File name on disk
<code>id</code>	Low-level identifier appropriate for this object
<code>libver</code>	low, high)

continues on next page

Table 2.26 – continued from previous page

<code>meta_block_size</code>	Meta block size (in bytes)
<code>mode</code>	Python mode used to open file
<code>name</code>	Return the full name of this object.
<code>parent</code>	Return the parent group of this object.
<code>ref</code>	An (opaque) HDF5 reference to this object
<code>regionref</code>	Create a region reference (Datasets only).
<code>swwr_mode</code>	Controls single-writer multiple-reader mode
<code>userblock_size</code>	User block size (in bytes)

close()

Close the file. All open objects become invalid

2.1.9 labscript_utils.impprof

Functions

disable()

enable([threshold])

labscript_utils.impprof.disable

labscript_utils.impprof.**disable**()

labscript_utils.impprof.enable

labscript_utils.impprof.**enable**(*threshold=0.1*)

2.1.10 labscript_utils.labconfig

Functions

<i>load_appconfig</i> (filename)	Load an .ini file and return a dictionary of its contents.
----------------------------------	--

<i>save_appconfig</i> (filename, data)	Save a dictionary as an ini file.
--	-----------------------------------

labscript_utils.labconfig.load_appconfig

`labscript_utils.labconfig.load_appconfig(filename)`

Load an .ini file and return a dictionary of its contents. All values will be converted to Python objects with `ast.literal_eval()`. All keys will be lowercase regardless of the written contents on the .ini file.

labscript_utils.labconfig.save_appconfig

`labscript_utils.labconfig.save_appconfig(filename, data)`

Save a dictionary as an ini file. The keys of the dictionary comprise the section names, and the values must themselves be dictionaries for the names and values within each section. All section values will be converted to strings with `pprint.pformat()`.

Classes

<code>EnvInterpolation()</code>	Interpolation which expands environment variables in values, by post-filtering <code>BasicInterpolation.before_get()</code>
<code>LabConfig([config_path, required_params, ...])</code>	

labscript_utils.labconfig.EnvInterpolation

class `labscript_utils.labconfig.EnvInterpolation`

Bases: `configparser.BasicInterpolation`

Interpolation which expands environment variables in values, by post-filtering `BasicInterpolation.before_get()`

`__init__()`

Methods

`__init__()`

`before_get(*args)`

`before_read(parser, section, option, value)`

`before_set(parser, section, option, value)`

`before_write(parser, section, option, value)`

`before_get(*args)`

labscript_utils.labconfig.LabConfig

```
class labscript_utils.labconfig.LabConfig(config_path=PosixPath('/home/docs/labscript-
suite/labconfig/build-20156903-project-24586-labscript-
utils.ini'), required_params=None,
defaults=None)
```

Bases: `configparser.ConfigParser`

```
__init__(config_path=PosixPath('/home/docs/labscript-suite/labconfig/build-20156903-project-24586-
labscript-utils.ini'), required_params=None,
defaults=None)
```

Methods

<code><i>__init__</i>([config_path, required_params, ...])</code>	
<code>add_section(section)</code>	Create a new section in the configuration.
<code>clear()</code>	
<code>defaults()</code>	
<code>get(section, option, *[, raw, vars, fallback])</code>	Get an option value for a given section.
<code>getboolean(section, option, *[, raw, vars, ...])</code>	
<code>getfloat(section, option, *[, raw, vars, ...])</code>	
<code>getint(section, option, *[, raw, vars, fallback])</code>	
<code>has_option(section, option)</code>	Check for the existence of a given option in a given section. If the specified 'section' is None or an empty string, DEFAULT is assumed. If the specified 'section' does not exist, returns False.
<code>has_section(section)</code>	Indicate whether the named section is present in the configuration.
<code>items([section, raw, vars])</code>	Return a list of (name, value) tuples for each option in a section.
<code>keys()</code>	
<code>options(section)</code>	Return a list of option names for the given section name.
<code>optionxform(optionstr)</code>	
<code>pop(k[,d])</code>	If key is not found, d is returned if given, otherwise KeyError is raised.
<code>popitem()</code>	Remove a section from the parser and return it as a (section_name, section_proxy) tuple.
<code>read(filename[, encoding])</code>	Read and parse a filename or an iterable of filenames.
<code>read_dict(dictionary[, source])</code>	Read configuration from a dictionary.
<code>read_file(f[, source])</code>	Like read() but the argument must be a file-like object.
<code>read_string(string[, source])</code>	Read configuration from a given string.

continues on next page

Table 2.31 – continued from previous page

<code>readfp(fp[, filename])</code>	Deprecated, use <code>read_file</code> instead.
<code>remove_option(section, option)</code>	Remove an option.
<code>remove_section(section)</code>	Remove a file section.
<code>sections()</code>	Return a list of section names, excluding [DEFAULT]
<code>set(section, option[, value])</code>	Set an option.
<code>setdefault(k[,d])</code>	
<code>update([E,]**F)</code>	If E present and has a <code>.keys()</code> method, does: for k in E: D[k] = E[k] If E present and lacks <code>.keys()</code> method, does: for (k, v) in E: D[k] = v In either case, this is followed by: for k, v in F.items(): D[k] = v
<code>values()</code>	
<code>write(fp[, space_around_delimiters])</code>	Write an <code>.ini</code> -format representation of the configuration state.

Attributes

BOOLEAN_STATES

NONSPACECRE

OPTCRE

OPTCRE_NV

SECTCRE

converters

exception NoOptionError(*option, section*)

Bases: `configparser.Error`

A requested option was not found.

exception NoSectionError(*section*)

Bases: `configparser.Error`

Raised when no section matches a requested option.

2.1.11 labscript_utils.ls_zprocess

Functions

Event(*args, **kwargs)

Handler(*args, **kwargs)

continues on next page

Table 2.33 – continued from previous page

<code>Lock(*args, **kwargs)</code>	
<code>RemoteProcessClient(host[, port])</code>	
<code>connect_to_zlock_server()</code>	
<code>ensure_connected_to_zlog()</code>	Ensure we are connected to a zlog server.
<code>get_config()</code>	Get relevant options from LabConfig, substituting defaults where appropriate and return as a dict
<code>zmq_get(*args, **kwargs)</code>	
<code>zmq_get_multipart(*args, **kwargs)</code>	
<code>zmq_get_raw(*args, **kwargs)</code>	
<code>zmq_get_string(*args, **kwargs)</code>	
<code>zmq_push(*args, **kwargs)</code>	
<code>zmq_push_multipart(*args, **kwargs)</code>	
<code>zmq_push_raw(*args, **kwargs)</code>	
<code>zmq_push_string(*args, **kwargs)</code>	

labscript_utils.ls_zprocess.Event`labscript_utils.ls_zprocess.Event(*args, **kwargs)`**labscript_utils.ls_zprocess.Handler**`labscript_utils.ls_zprocess.Handler(*args, **kwargs)`**labscript_utils.ls_zprocess.Lock**`labscript_utils.ls_zprocess.Lock(*args, **kwargs)`**labscript_utils.ls_zprocess.RemoteProcessClient**`labscript_utils.ls_zprocess.RemoteProcessClient(host, port=None)`

labscript_utils.ls_zprocess.connect_to_zlock_server

```
labscript_utils.ls_zprocess.connect_to_zlock_server()
```

labscript_utils.ls_zprocess.ensure_connected_to_zlog

```
labscript_utils.ls_zprocess.ensure_connected_to_zlog()
```

Ensure we are connected to a zlog server. If one is not running and we are the top-level process, start a zlog server configured according to LabConfig.

labscript_utils.ls_zprocess.get_config

```
labscript_utils.ls_zprocess.get_config()
```

Get relevant options from LabConfig, substituting defaults where appropriate and return as a dict

labscript_utils.ls_zprocess.zmq_get

```
labscript_utils.ls_zprocess.zmq_get(*args, **kwargs)
```

labscript_utils.ls_zprocess.zmq_get_multipart

```
labscript_utils.ls_zprocess.zmq_get_multipart(*args, **kwargs)
```

labscript_utils.ls_zprocess.zmq_get_raw

```
labscript_utils.ls_zprocess.zmq_get_raw(*args, **kwargs)
```

labscript_utils.ls_zprocess.zmq_get_string

```
labscript_utils.ls_zprocess.zmq_get_string(*args, **kwargs)
```

labscript_utils.ls_zprocess.zmq_push

```
labscript_utils.ls_zprocess.zmq_push(*args, **kwargs)
```

labscript_utils.ls_zprocess.zmq_push_multipart

```
labscript_utils.ls_zprocess.zmq_push_multipart(*args, **kwargs)
```

labscript_utils.ls_zprocess.zmq_push_raw

labscript_utils.ls_zprocess.zmq_push_raw(*args, **kwargs)

labscript_utils.ls_zprocess.zmq_push_string

labscript_utils.ls_zprocess.zmq_push_string(*args, **kwargs)

Classes

<code>Context</code> ([io_threads, shared_secret])	Subclass of <code>zprocess.security.SecureContext</code> configured with settings from <code>labconfig</code> , substitutable for a <code>zmq.Context</code> .
<code>ProcessTree</code> ([shared_secret, allow_insecure, ...])	A singleton <code>zprocess.ProcessTree</code> configured with settings from <code>labconfig</code> for security, <code>zlock</code> and <code>zlog</code> .
<code>ZMQClient</code> ()	A singleton <code>zprocess.ZMQClient</code> configured with settings from <code>labconfig</code> for security.
<code>ZMQServer</code> ([port, dtype, pull_only, ...])	A <code>ZMQServer</code> configured with security settings from <code>labconfig</code>

labscript_utils.ls_zprocess.Context

class labscript_utils.ls_zprocess.Context(*io_threads=1, shared_secret=None*)

Bases: `zprocess.security.SecureContext`

Subclass of `zprocess.security.SecureContext` configured with settings from `labconfig`, substitutable for a `zmq.Context`. Can be instantiated to get a unique context, or call the `.instance()` classmethod to possibly get an already-existing one. Only use the latter if you do not intend to terminate the context.

`__init__`(*io_threads=1, shared_secret=None*)

Methods

<code>__init__</code> ([io_threads, shared_secret])	
<code>destroy</code> ([linger])	Close all sockets associated with this context and then terminate the context.
<code>get</code> (option)	Get the value of a context option.
<code>getsockopt</code> (opt)	get default socket options for new sockets created by this Context
<code>instance</code> ()	Returns a shared instance with the same shared secret, if there is one, otherwise creates it.
<code>set</code> (option, optval)	Set a context option.
<code>setsockopt</code> (opt, value)	set default socket options for new sockets created by this Context
<code>shadow</code> (address)	Shadow an existing libzmq context
<code>shadow_pyczmq</code> (ctx)	Shadow an existing pyczmq context
<code>socket</code> (*args, **kwargs)	Create a Socket associated with this Context.
<code>term</code> ()	Close or terminate the context.

Attributes

client_publickey	
client_secretkey	
closed	
secure	
server_publickey	
server_secretkey	
underlying	The address of the underlying libzmq context

classmethod instance()

Returns a shared instance with the same shared secret, if there is one, otherwise creates it. If an instance already exists, `io_threads` will be ignored, otherwise it will be used in the new instance. Takes into account subclasses such that a subclass calling this method will always get back an instance of its own class

socket(*args, **kwargs)

Create a Socket associated with this Context.

Parameters

- **socket_type** (*int*) – The socket type, which can be any of the 0MQ socket types: REQ, REP, PUB, SUB, PAIR, DEALER, ROUTER, PULL, PUSH, etc.
- **socket_class** (*zmq.Socket or a subclass*) – The socket class to instantiate, if different from the default for this Context. e.g. for creating an asyncio socket attached to a default Context or vice versa.

New in version 25.

- **kwargs** – will be passed to the `__init__` method of the socket class.

sockopts: Dict[*int*, *Any*]

labscript_utils.ls_zprocess.ProcessTree

```
class labscript_utils.ls_zprocess.ProcessTree(shared_secret=None, allow_insecure=False,
                                             zlock_host=None, zlock_port=7339, zlog_host=None,
                                             zlog_port=7340)
```

Bases: `zprocess.process_tree.ProcessTree`

A singleton `zprocess.ProcessTree` configured with settings from `labconfig` for security, `zlock` and `zlog`. Being a singleton is not enforced - the class can still be instantiated as normal - but calling the `.instance()` classmethod will give the singleton.

```
__init__(shared_secret=None, allow_insecure=False, zlock_host=None, zlock_port=7339,
         zlog_host=None, zlog_port=7340)
```

Methods

<code>__init__([shared_secret, allow_insecure, ...])</code>	
<code>check_broker()</code>	
<code>connect_to_parent()</code>	
<code>event(event_name[, role, external_broker])</code>	
<code>instance()</code>	
<code>lock(key[, read_only])</code>	Return a <code>zprocess.locking.Lock</code> for a resource identified by the given key.
<code>logging_handler(filepath[, name])</code>	Return a <code>zprocess.zlog.ZMQLoggingHandler</code> for the given filepath.
<code>remote_process_client(host[, port])</code>	Return a <code>RemoteProcessClient</code> configured with this <code>ProcessTree</code> 's security settings
<code>subprocess(path[, output_redirection_host, ...])</code>	Start a subprocess and set up communication with it.

classmethod `instance()`

labscript_utils.ls_zprocess.ZMQClient

class `labscript_utils.ls_zprocess.ZMQClient`

Bases: `zprocess.clientserver.ZMQClient`

A singleton `zprocess.ZMQClient` configured with settings from `labconfig` for security. Being a singleton is not enforced - the class can still be instantiated as normal - but calling the `.instance()` classmethod will give the singleton.

`__init__()`

Methods

<code>__init__()</code>	
<code>clear_interrupt()</code>	Clear our internal <code>Interruptor</code> object so that future <code>get*()/push*()</code> calls can proceed as normal.
<code>instance()</code>	
<code>interrupt([reason])</code>	Interrupt any current and future <code>get*()/push*()</code> calls, causing them to raise <code>Interrupted(reason)</code> until <code>clear_interrupt()</code> is called.

classmethod `instance()`

labscript_utils.ls_zprocess.ZMQServer

```
class labscript_utils.ls_zprocess.ZMQServer(port=None, dtype='pyobj', pull_only=False,
                                           bind_address='tcp://0.0.0.0', timeout_interval=None,
                                           **kwargs)
```

Bases: `zprocess.clientserver.ZMQServer`

A ZMQServer configured with security settings from labconfig

```
__init__(port=None, dtype='pyobj', pull_only=False, bind_address='tcp://0.0.0.0', timeout_interval=None,
         **kwargs)
```

Methods

<code><i>__init__</i></code> (<i>port, dtype, pull_only, ...</i>)	
<code>handler</code> (<i>request_data</i>)	To be overridden by subclasses.
<code>mainloop</code> ()	
<code>setup_auth</code> (<i>context</i>)	Deprecated.
<code>shutdown</code> ()	
<code>shutdown_on_interrupt</code> ()	
<code>timeout</code> ()	A function to call every <code>self.timeout_interval</code> seconds in the same thread as the handler.

2.1.12 labscript_utils.memprof**Functions**

<code><i>check</i></code> ()
<code><i>start</i></code> (<i>[filepath]</i>)

labscript_utils.memprof.check

```
labscript_utils.memprof.check()
```

labscript_utils.memprof.start

labscript_utils.memprof.start(*filepath='memprof.txt'*)

Classes

<i>MemoryProfiler</i> ()	Class to count number instances of each type in the interpreter in order to detect Python memory leaks
--------------------------	--

labscript_utils.memprof.MemoryProfiler

class labscript_utils.memprof.MemoryProfiler

Bases: `object`

Class to count number instances of each type in the interpreter in order to detect Python memory leaks

`__init__()`

Methods

`__init__()`

`check()`

`count_types()`

`start([filepath])`

`write_to_file(types)`

`check()`

`count_types()`

`start(filepath='memprof.txt')`

`write_to_file(types)`

2.1.13 labscript_utils.modulewatcher

Classes

<i>ModuleWatcher</i> ([debug])	A watcher that reloads modules that have been modified on disk
--------------------------------	--

labscript_utils.modulewatcher.ModuleWatcher

class labscript_utils.modulewatcher.**ModuleWatcher**(*debug=False*)

Bases: `object`

A watcher that reloads modules that have been modified on disk

Only reloads modules imported after instantiation. Does not reload C extensions.

Parameters `debug` (*bool*, *optional*) – When True, prints debugging information when reloading modules.

`__init__`(*debug=False*)

Methods

`__init__`([*debug*])

`check`()

`mainloop`()

`unload`()

`check`()

`mainloop`()

`unload`()

2.1.14 labscript_utils.properties

Functions

`deserialise`(value)

`get`(h5_file, device_name, location)

`get_attribute`(group, name)

Return the attribute of the given name from the given HDF5 group, deserialising it if it has been encoded as JSON

`get_attributes`(group)

Return attributes of a HDF5 group as a dict, deserialising any that have been encoded as JSON

`is_json`(value)

`serialise`(value)

`set_attributes`(group, attributes)

Add attributes to a HDF5 group, serialising them to JSON if they do not map to native HDF5 datatypes

`set_device_properties`(h5_file, device_name, ...)

labscript_utils.properties.deserialise

labscript_utils.properties.**deserialise**(*value*)

labscript_utils.properties.get

labscript_utils.properties.**get**(*h5_file, device_name, location*)

labscript_utils.properties.get_attribute

labscript_utils.properties.**get_attribute**(*group, name*)

Return the attribute of the given name from the given HDF5 group, deserialising it if it has been encoded as JSON

labscript_utils.properties.get_attributes

labscript_utils.properties.**get_attributes**(*group*)

Return attributes of a HDF5 group as a dict, deserialising any that have been encoded as JSON

labscript_utils.properties.is_json

labscript_utils.properties.**is_json**(*value*)

labscript_utils.properties.serialise

labscript_utils.properties.**serialise**(*value*)

labscript_utils.properties.set_attributes

labscript_utils.properties.**set_attributes**(*group, attributes*)

Add attributes to a HDF5 group, serialising them to JSON if they do not map to native HDF5 datatypes

labscript_utils.properties.set_device_properties

labscript_utils.properties.**set_device_properties**(*h5_file, device_name, properties*)

2.1.15 labscript_utils.qtwidgets

Sub-Modules

labscript_utils.qtwidgets.InputPlotWindow

labscript_utils.qtwidgets.analoginput

labscript_utils.qtwidgets.analogoutput

continues on next page

Table 2.46 – continued from previous page

labscript_utils.qtwidgets.ddsoutput

labscript_utils.qtwidgets.digitaloutput

labscript_utils.qtwidgets.dragdroptab

labscript_utils.qtwidgets.elide_label

labscript_utils.qtwidgets.enumoutput

labscript_utils.qtwidgets.fingertab

*labscript_utils.qtwidgets.
headerview_with_widgets*

labscript_utils.qtwidgets.imageoutput

labscript_utils.qtwidgets.outputbox

labscript_utils.qtwidgets.toolpalette

labscript_utils.qtwidgets.InputPlotWindow

Classes

*PlotWindow(*args, **kwargs)*

labscript_utils.qtwidgets.InputPlotWindow.PlotWindow

class labscript_utils.qtwidgets.InputPlotWindow.PlotWindow(*args, **kwargs)

Bases: zprocess.process_tree.Process

__init__(*args, **kwargs)

Methods

__init__(*args, **kwargs)

interrupt_startup([reason])

Called from the parent process.

run(connection_name, hardware_name, de-
vice_name)

The method that gets called in the subprocess.

setTopLevelWindow()*start*(*args, **kwargs)

Call in the parent process to start a subprocess.

terminate([wait_timeout])

Interrupt process startup if not already done, ensuring self.child exists or is None if startup was interrupted before the process was created.

continues on next page

Table 2.48 – continued from previous page

`update_plot(new_data)`

run(*connection_name*, *hardware_name*, *device_name*)

The method that gets called in the subprocess. To be overridden by subclasses

setTopLevelWindow()

update_plot(*new_data*)

labscript_utils.qtwidgets.analoginput

Classes

AnalogInput(*device_name*, *hardware_name*[, ...])

labscript_utils.qtwidgets.analoginput.AnalogInput

class labscript_utils.qtwidgets.analoginput.**AnalogInput**(*device_name*, *hardware_name*,
connection_name='-', *display_name*=None,
horizontal_alignment=False,
parent=None)

Bases: PyQt5.QtWidgets.QWidget

__init__(*device_name*, *hardware_name*, *connection_name*='-', *display_name*=None,
horizontal_alignment=False, *parent*=None)

Methods

`__init__`(*device_name*, *hardware_name*[, ...])

`acceptDrops`(self)

`accessibleDescription`(self)

`accessibleName`(self)

`actionEvent`(self, a0)

`actions`(self)

`activateWindow`(self)

`addAction`(self, action)

`addActions`(self, actions)

continues on next page

Table 2.50 – continued from previous page

<code>adjustSize(self)</code>
<code>autoFillBackground(self)</code>
<code>backgroundRole(self)</code>
<code>baseSize(self)</code>
<code>blockSignals(self, b)</code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>

continues on next page

Table 2.50 – continued from previous page

<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>

continues on next page

Table 2.50 – continued from previous page

<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code><i>get_AI()</i></code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>

continues on next page

Table 2.50 – continued from previous page

<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>

continues on next page

Table 2.50 – continued from previous page

<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>

continues on next page

Table 2.50 – continued from previous page

<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code><i>open_plot_window()</i></code>
<code>overrideWindowFlags(self, type)</code>

continues on next page

Table 2.50 – continued from previous page

<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	
<code>resize()</code>	
<code>resizeEvent(self, a0)</code>	

continues on next page

Table 2.50 – continued from previous page

<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>

continues on next page

Table 2.50 – continued from previous page

<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>

continues on next page

Table 2.50 – continued from previous page

<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>set_AI(AI[, notify_old_AI, notify_new_AI])</code>
<code>set_value(value)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>

continues on next page

Table 2.50 – continued from previous page

<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>

continues on next page

Table 2.50 – continued from previous page

<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>
<code>windowModality(self)</code>
<code>windowOpacity(self)</code>
<code>windowRole(self)</code>
<code>windowState(self)</code>
<code>windowTitle(self)</code>
<code>windowType(self)</code>
<code>x(self)</code>
<code>y(self)</code>

labscript_utils.qtwidgets.analogoutput

Classes

AnalogOutput(hardware_name[, ...])

NoStealFocusDoubleSpinBox(*args, **kwargs) A QDoubleSpinBox that doesn't steal focus as you scroll over it with a mouse wheel.

labscript_utils.qtwidgets.analogoutput.AnalogOutput

class labscript_utils.qtwidgets.analogoutput.**AnalogOutput**(*hardware_name*, *connection_name*='-',
display_name=None,
horizontal_alignment=False,
parent=None)

Bases: PyQt5.QtWidgets.QWidget

__init__(*hardware_name*, *connection_name*='-', *display_name*=None, *horizontal_alignment*=False,
parent=None)

Methods

__init__(hardware_name[, connection_name, ...])

acceptDrops(self)

accessibleDescription(self)

accessibleName(self)

actionEvent(self, a0)

actions(self)

activateWindow(self)

addAction(self, action)

addActions(self, actions)

adjustSize(self)

autoFillBackground(self)

backgroundRole(self)

baseSize(self)

continues on next page

Table 2.53 – continued from previous page

<code>blockSignals(self, b)</code>
<code><i>block_combobox_signals()</i></code>
<code><i>block_spinbox_signals()</i></code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code><i>connect_value_change(func)</i></code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>

continues on next page

Table 2.53 – continued from previous page

depth(self)

destroy(self[, destroyWindow, destroySubWindows])

devType(self)

devicePixelRatio(self)

devicePixelRatioF(self)

devicePixelRatioFScale()

disconnect(-> bool)

disconnectNotify(self, signal)

disconnect_value_change()

dragEnterEvent(self, a0)

dragLeaveEvent(self, a0)

dragMoveEvent(self, a0)

dropEvent(self, a0)

dumpObjectInfo(self)

dumpObjectTree(self)

dynamicPropertyNames(self)

effectiveWinId(self)

ensurePolished(self)

enterEvent(self, a0)

event(self, a0)

eventFilter(self, a0, a1)

find(a0)

findChild(-> QObject)

findChildren(...)

focusInEvent(self, a0)

continues on next page

Table 2.53 – continued from previous page

<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code><i>get_AO()</i></code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>

continues on next page

Table 2.53 – continued from previous page

<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>

continues on next page

Table 2.53 – continued from previous page

<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code><i>lock</i>([notify_a0])</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>

continues on next page

Table 2.53 – continued from previous page

<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>overrideWindowFlags(self, type)</code>

continues on next page

Table 2.53 – continued from previous page

<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	
<code>resize()</code>	
<code>resizeEvent(self, a0)</code>	

continues on next page

Table 2.53 – continued from previous page

<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>

continues on next page

Table 2.53 – continued from previous page

<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>

continues on next page

Table 2.53 – continued from previous page

<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code><i>set_AO</i>(AO[, notify_old_AO, notify_new_AO])</code>
<code><i>set_combobox_model</i>(model)</code>
<code><i>set_limits</i>(lower, upper)</code>
<code><i>set_num_decimals</i>(decimals)</code>
<code><i>set_selected_unit</i>(unit)</code>

continues on next page

Table 2.53 – continued from previous page

<code>set_spinbox_value(value, unit)</code>
<code>set_step_size(step)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>

continues on next page

Table 2.53 – continued from previous page

<code>tr(self, sourceText[, disambiguation, n])</code>
<code><i>unblock_combobox_signals()</i></code>
<code><i>unblock_spinbox_signals()</i></code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code><i>unlock</i>([notify_ao])</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>

continues on next page

Table 2.53 – continued from previous page

windowModality(self)	
windowOpacity(self)	
windowRole(self)	
windowState(self)	
windowTitle(self)	
windowType(self)	
x(self)	
y(self)	
Attributes	
DrawChildren	
DrawWindowBackground	
IgnoreMask	
PdmDepth	
PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	

continues on next page

Table 2.54 – continued from previous page

customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
<i>selected_unit</i>	
<hr/>	
staticMetaObject	
<hr/>	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

```

block_combobox_signals()
block_spinbox_signals()
connect_value_change(func)
disconnect_value_change()
eventFilter(self, a0: QObject, a1: QEvent) → bool
get_AO()
lock(notify_ao=True)
property selected_unit
set_AO(AO, notify_old_AO=True, notify_new_AO=True)
set_combobox_model(model)
set_limits(lower, upper)
set_num_decimals(decimals)
set_selected_unit(unit)
set_spinbox_value(value, unit)
set_step_size(step)
unblock_combobox_signals()
unblock_spinbox_signals()
unlock(notify_ao=True)

```

labscript_utils.qtwidgets.analogoutput.NoStealFocusDoubleSpinBox

```
class labscript_utils.qtwidgets.analogoutput.NoStealFocusDoubleSpinBox(*args, **kwargs)
```

Bases: PyQt5.QtWidgets.QDoubleSpinBox

A QDoubleSpinBox that doesn't steal focus as you scroll over it with a mouse wheel.

```
__init__(*args, **kwargs)
```

Methods

`__init__(*args, **kwargs)`

`acceptDrops(self)`

`accessibleDescription(self)`

`accessibleName(self)`

`actionEvent(self, a0)`

`actions(self)`

`activateWindow(self)`

`addAction(self, action)`

`addActions(self, actions)`

`adjustSize(self)`

`alignment(self)`

`autoFillBackground(self)`

`backgroundRole(self)`

`baseSize(self)`

`blockSignals(self, b)`

`buttonSymbols(self)`

`changeEvent(self, e)`

`childAt(-> QWidget)`

`childEvent(self, a0)`

`children(self)`

`childrenRect(self)`

`childrenRegion(self)`

`cleanText(self)`

`clear(self)`

continues on next page

Table 2.55 – continued from previous page

<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, e)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, e)</code>
<code>contextMenuPolicy(self)</code>
<code>correctionMode(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>decimals(self)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>

continues on next page

Table 2.55 – continued from previous page

<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, event)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>fixup(self, str)</code>
<code><i>focusInEvent</i>(self, e)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code><i>focusOutEvent</i>(self, e)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>

continues on next page

Table 2.55 – continued from previous page

fontInfo(self)
fontMetrics(self)
foregroundRole(self)
frameGeometry(self)
frameSize(self)
geometry(self)
getContentsMargins(self)
grab(self[, rectangle])
grabGesture(self, type[, flags])
grabKeyboard(self)
grabMouse()
grabShortcut(self, key[, context])
graphicsEffect(self)
graphicsProxyWidget(self)
hasAcceptableInput(self)
hasFocus(self)
hasFrame(self)
hasHeightForWidth(self)
hasMouseTracking(self)
hasTabletTracking(self)
height(self)
heightForWidth(self, a0)
heightMM(self)
hide(self)
hideEvent(self, e)

continues on next page

Table 2.55 – continued from previous page

<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>interpretText(self)</code>
<code>isAccelerated(self)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isGroupSeparatorShown(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isReadOnly(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>

continues on next page

Table 2.55 – continued from previous page

<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, e)</code>
<code>keyReleaseEvent(self, e)</code>
<code>keyboardGrabber()</code>
<code>keyboardTracking(self)</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>lineEdit(self)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>

continues on next page

Table 2.55 – continued from previous page

<code>mask(self)</code>
<code>maximum(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimum(self)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, e)</code>
<code>mousePressEvent(self, e)</code>
<code>mouseReleaseEvent(self, e)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>

continues on next page

Table 2.55 – continued from previous page

<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, e)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>prefix(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	

continues on next page

Table 2.55 – continued from previous page

<code>resize()</code>
<code>resizeEvent(self, e)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>selectAll(self)</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAccelerated(self, on)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAlignment(self, flag)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setButtonSymbols(self, bs)</code>
<code>setContentMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCorrectionMode(self, cm)</code>
<code>setCursor(self, a0)</code>
<code>setDecimals(self, prec)</code>
<code>setDisabled(self, a0)</code>

continues on next page

Table 2.55 – continued from previous page

<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setFrame(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setGroupSeparatorShown(self, shown)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setKeyboardTracking(self, kt)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLineEdit(self, e)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximum(self, max)</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>

continues on next page

Table 2.55 – continued from previous page

<code>setMinimum(self, min)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setPrefix(self, p)</code>
<code>setProperty(self, name, value)</code>
<code>setRange(self, min, max)</code>
<code>setReadOnly(self, r)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSingleStep(self, val)</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setSpecialValueText(self, s)</code>
<code>setStatusTip(self, a0)</code>
<code>setStepType(self, stepType)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setSuffix(self, s)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>

continues on next page

Table 2.55 – continued from previous page

<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setValue(self, val)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>setWrapping(self, w)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, e)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>

continues on next page

Table 2.55 – continued from previous page

<code>signalsBlocked(self)</code>
<code>singleStep(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>specialValueText(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>stepBy(self, steps)</code>
<code>stepDown(self)</code>
<code>stepEnabled(self)</code>
<code>stepType(self)</code>
<code>stepUp(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>suffix(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>text(self)</code>
<code>textFromValue(self, v)</code>
<code>thread(self)</code>
<code>timerEvent(self, e)</code>
<code>toolTip(self)</code>

continues on next page

Table 2.55 – continued from previous page

<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>validate(self, input, pos)</code>
<code>value(self)</code>
<code>valueFromText(self, text)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code><i>wheelEvent</i>(self, e)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>

continues on next page

Table 2.55 – continued from previous page

windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
wrapping(self)
x(self)
y(self)

Attributes

AdaptiveDecimalStepType
CorrectToNearestValue
CorrectToPreviousValue
DefaultStepType
DrawChildren
DrawWindowBackground
IgnoreMask
NoButtons
PdmDepth
PdmDevicePixelRatio
PdmDevicePixelRatioScaled
PdmDpiX
PdmDpiY

continues on next page

Table 2.56 – continued from previous page

PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
PlusMinus	
StepDownEnabled	
StepNone	
StepUpEnabled	
UpDownArrows	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
editingFinished	editingFinished(self) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
textChanged	str) [signal]
valueChanged	float) [signal] valueChanged(self, a0: str) [signal]
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

focusInEvent (*self*, *e*: *QFocusEvent*)

focusOutEvent (*self*, *e*: *QFocusEvent*)

wheelEvent (*self*, *e*: *QWheelEvent*)

labscript_utils.qtwidgets.ddsoutput

Classes

DDSOutput(hardware_name[, connection_name, ...])

labscript_utils.qtwidgets.ddsoutput.DDSOutput

```
class labscript_utils.qtwidgets.ddsoutput.DDSOutput(hardware_name, connection_name='-',
                                                    parent=None)
```

```
    Bases: PyQt5.QtWidgets.QWidget
```

```
    __init__(hardware_name, connection_name='-', parent=None)
```

Methods

```
__init__(hardware_name[, connection_name, ...])
```

```
acceptDrops(self)
```

```
accessibleDescription(self)
```

```
accessibleName(self)
```

```
actionEvent(self, a0)
```

```
actions(self)
```

```
activateWindow(self)
```

```
addAction(self, action)
```

```
addActions(self, actions)
```

```
adjustSize(self)
```

```
autoFillBackground(self)
```

```
backgroundRole(self)
```

```
baseSize(self)
```

```
blockSignals(self, b)
```

```
changeEvent(self, a0)
```

```
childAt(-> QWidget)
```

```
childEvent(self, a0)
```

```
children(self)
```

```
childrenRect(self)
```

```
childrenRegion(self)
```

continues on next page

Table 2.58 – continued from previous page

<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>

continues on next page

Table 2.58 – continued from previous page

<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>

continues on next page

Table 2.58 – continued from previous page

<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code><i>get_sub_widget</i>(subchnl)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code><i>hide_sub_widget</i>(subchnl)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>inputMethodEvent(self, a0)</code>

continues on next page

Table 2.58 – continued from previous page

<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>

continues on next page

Table 2.58 – continued from previous page

<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>

continues on next page

Table 2.58 – continued from previous page

<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>overrideWindowFlags(self, type)</code>
<code>overrideWindowState(self, state)</code>
<code>paintEngine(self)</code>
<code>paintEvent(self, a0)</code>
<code>paintingActive(self)</code>
<code>palette(self)</code>
<code>parent(self)</code>
<code>parentWidget(self)</code>
<code>physicalDpiX(self)</code>
<code>physicalDpiY(self)</code>
<code>pos(self)</code>
<code>previousInFocusChain(self)</code>

continues on next page

Table 2.58 – continued from previous page

property(self, name)	
pyqtConfigure(...)	Each keyword argument is either the name of a Qt property or a Qt signal.
raise_(self)	
receivers(self, signal)	
rect(self)	
releaseKeyboard(self)	
releaseMouse(self)	
releaseShortcut(self, id)	
removeAction(self, action)	
removeEventFilter(self, a0)	
render(, sourceRegion, flags, ...)	
repaint(-> None -> None)	
resize()	
resizeEvent(self, a0)	
restoreGeometry(self, geometry)	
saveGeometry(self)	
screen(self)	
scroll()	
sender(self)	
senderSignalIndex(self)	
setAcceptDrops(self, on)	
setAccessibleDescription(self, description)	
setAccessibleName(self, name)	
setAttribute(self, attribute[, on])	
setAutoFillBackground(self, enabled)	

continues on next page

Table 2.58 – continued from previous page

<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>

continues on next page

Table 2.58 – continued from previous page

<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>

continues on next page

Table 2.58 – continued from previous page

<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code><i>show_sub_widget</i>(subchnl)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>

continues on next page

Table 2.58 – continued from previous page

<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>

continues on next page

Table 2.58 – continued from previous page

windowFilePath(self)
windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

DrawChildren
DrawWindowBackground
IgnoreMask
PdmDepth
PdmDevicePixelRatio
PdmDevicePixelRatioScaled
PdmDpiX
PdmDpiY
PdmHeight
PdmHeightMM

continues on next page

Table 2.59 – continued from previous page

PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

```

get_sub_widget(subchnl)
hide_sub_widget(subchnl)
show_sub_widget(subchnl)

```

labscript_utils.qtwidgets.digitaloutput

Classes

```

DigitalOutput(*args, **kwargs)
InvertedDigitalOutput(*args, **kwargs)

```

labscript_utils.qtwidgets.digitaloutput.DigitalOutput

```

class labscript_utils.qtwidgets.digitaloutput.DigitalOutput(*args, **kwargs)
    Bases: PyQt5.QtWidgets.QPushButton
    __init__(*args, **kwargs)

```

Methods

`__init__(*args, **kwargs)`

`acceptDrops(self)`

`accessibleDescription(self)`

`accessibleName(self)`

`actionEvent(self, a0)`

`actions(self)`

`activateWindow(self)`

`addAction(self, action)`

`addActions(self, actions)`

`adjustSize(self)`

`animateClick(self[, msec])`

`autoDefault(self)`

`autoExclusive(self)`

`autoFillBackground(self)`

`autoRepeat(self)`

`autoRepeatDelay(self)`

`autoRepeatInterval(self)`

`backgroundRole(self)`

`baseSize(self)`

`blockSignals(self, b)`

`changeEvent(self, e)`

`checkStateSet(self)`

`childAt(-> QWidget)`

`childEvent(self, a0)`

continues on next page

Table 2.61 – continued from previous page

<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>click(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWin- dows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>

continues on next page

Table 2.61 – continued from previous page

<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, e)</code>
<code><i>eventFilter</i>(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>

continues on next page

Table 2.61 – continued from previous page

font(self)
fontInfo(self)
fontMetrics(self)
foregroundRole(self)
frameGeometry(self)
frameSize(self)
geometry(self)
getContentsMargins(self)
<i>get_D0()</i>
grab(self[, rectangle])
grabGesture(self, type[, flags])
grabKeyboard(self)
grabMouse()
grabShortcut(self, key[, context])
graphicsEffect(self)
graphicsProxyWidget(self)
group(self)
hasFocus(self)
hasHeightForWidth(self)
hasMouseTracking(self)
hasTabletTracking(self)
height(self)
heightForWidth(self, a0)
heightMM(self)
hide(self)

continues on next page

Table 2.61 – continued from previous page

<code>hideEvent(self, a0)</code>
<code>hitButton(self, pos)</code>
<code>icon(self)</code>
<code>iconSize(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isCheckable(self)</code>
<code>isChecked(self)</code>
<code>isDefault(self)</code>
<code>isDown(self)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFlat(self)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>

continues on next page

Table 2.61 – continued from previous page

<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, e)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code><i>lock</i>([notify_do])</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>

continues on next page

Table 2.61 – continued from previous page

<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>menu(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, e)</code>
<code>mousePressEvent(self, e)</code>
<code>mouseReleaseEvent(self, e)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>

continues on next page

Table 2.61 – continued from previous page

<code>nextCheckState(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	

continues on next page

Table 2.61 – continued from previous page

<code>removeEventFilter(self, a0)</code>
<code>render(, sourceRegion, flags, ...)</code>
<code>repaint(-> None -> None)</code>
<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoDefault(self, a0)</code>
<code>setAutoExclusive(self, a0)</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setAutoRepeat(self, a0)</code>
<code>setAutoRepeatDelay(self, a0)</code>
<code>setAutoRepeatInterval(self, a0)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setCheckable(self, a0)</code>
<code>setChecked(self, a0)</code>

continues on next page

Table 2.61 – continued from previous page

<code>setContentMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDefault(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDown(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFlat(self, a0)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setIcon(self, icon)</code>
<code>setIconSize(self, size)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>

continues on next page

Table 2.61 – continued from previous page

<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMenu(self, menu)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcut(self, key)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setText(self, text)</code>
<code>setToolTip(self, a0)</code>

continues on next page

Table 2.61 – continued from previous page

<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>set_DO(DO[, notify_old_DO, notify_new_DO])</code>
<code>sharedPainter(self)</code>
<code>shortcut(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMenu(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>

continues on next page

Table 2.61 – continued from previous page

<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>text(self)</code>
<code>thread(self)</code>
<code>timerEvent(self, e)</code>
<code>toggle(self)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code><i>unlock</i>([notify_do])</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>

continues on next page

Table 2.61 – continued from previous page

<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>
<code>windowModality(self)</code>
<code>windowOpacity(self)</code>
<code>windowRole(self)</code>
<code>windowState(self)</code>
<code>windowTitle(self)</code>
<code>windowType(self)</code>
<code>x(self)</code>
<code>y(self)</code>

property state

unlock(notify_do=True)

labscript_utils.qtwidgets.digitaloutput.InvertedDigitalOutput

class labscript_utils.qtwidgets.digitaloutput.InvertedDigitalOutput(*args, **kwargs)

Bases: *labscript_utils.qtwidgets.digitaloutput.DigitalOutput*

__init__(*args, **kwargs)

Methods

__init__(*args, **kwargs)

acceptDrops(self)

accessibleDescription(self)

accessibleName(self)

actionEvent(self, a0)

actions(self)

activateWindow(self)

addAction(self, action)

addActions(self, actions)

adjustSize(self)

animateClick(self[, msec])

autoDefault(self)

autoExclusive(self)

autoFillBackground(self)

autoRepeat(self)

autoRepeatDelay(self)

autoRepeatInterval(self)

backgroundRole(self)

baseSize(self)

continues on next page

Table 2.63 – continued from previous page

<code>blockSignals(self, b)</code>
<code>changeEvent(self, e)</code>
<code>checkStateSet(self)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>click(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>

continues on next page

Table 2.63 – continued from previous page

<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, e)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>

continues on next page

Table 2.63 – continued from previous page

<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>get_DO()</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>group(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>

continues on next page

Table 2.63 – continued from previous page

<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>hitButton(self, pos)</code>
<code>icon(self)</code>
<code>iconSize(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isCheckable(self)</code>
<code>isChecked(self)</code>
<code>isDefault(self)</code>
<code>isDown(self)</code>
<code>isEnabled(self)</code>

continues on next page

Table 2.63 – continued from previous page

<code>isEnabledTo(self, a0)</code>
<code>isFlat(self)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, e)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>lock([notify_do])</code>

continues on next page

Table 2.63 – continued from previous page

<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>menu(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, e)</code>
<code>mousePressEvent(self, e)</code>
<code>mouseReleaseEvent(self, e)</code>

continues on next page

Table 2.63 – continued from previous page

<code>move()</code>	
<code>moveEvent(self, a0)</code>	
<code>moveToThread(self, thread)</code>	
<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextCheckState(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	

continues on next page

Table 2.63 – continued from previous page

<code>rect(self)</code>
<code>releaseKeyboard(self)</code>
<code>releaseMouse(self)</code>
<code>releaseShortcut(self, id)</code>
<code>removeAction(self, action)</code>
<code>removeEventFilter(self, a0)</code>
<code>render(, sourceRegion, flags, ...)</code>
<code>repaint(-> None -> None)</code>
<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoDefault(self, a0)</code>
<code>setAutoExclusive(self, a0)</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setAutoRepeat(self, a0)</code>
<code>setAutoRepeatDelay(self, a0)</code>

continues on next page

Table 2.63 – continued from previous page

<code>setAutoRepeatInterval(self, a0)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setCheckable(self, a0)</code>
<code>setChecked(self, a0)</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDefault(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDown(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFlat(self, a0)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setIcon(self, icon)</code>

continues on next page

Table 2.63 – continued from previous page

<code>setIconSize(self, size)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMenu(self, menu)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcut(self, key)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>

continues on next page

Table 2.63 – continued from previous page

<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setText(self, text)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>set_DO(DO[, notify_old_DO, notify_new_DO])</code>
<code>sharedPainter(self)</code>
<code>shortcut(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>

continues on next page

Table 2.63 – continued from previous page

<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMenu(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>text(self)</code>
<code>thread(self)</code>
<code>timerEvent(self, e)</code>
<code>toggle(self)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>

continues on next page

Table 2.63 – continued from previous page

<code>ungrabGesture(self, type)</code>
<code>unlock([notify_do])</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>
<code>windowModality(self)</code>
<code>windowOpacity(self)</code>
<code>windowRole(self)</code>
<code>windowState(self)</code>

continues on next page

Table 2.63 – continued from previous page

windowTitle(self)	
windowType(self)	
x(self)	
y(self)	
Attributes	
DrawChildren	
DrawWindowBackground	
IgnoreMask	
PdmDepth	
PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
clicked	bool = False) [signal]
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
pressed	pressed(self) [signal]
released	released(self) [signal]
<i>state</i>	

continues on next page

Table 2.64 – continued from previous page

staticMetaObject	
toggled	bool) [signal]
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

property state

labscript_utils.qtwidgets.dragdroptab

Classes

<i>DragDropTabBar</i> (parent, group_id)	
<i>DragDropTabWidget</i> ([group_id, ...])	A tab widget that supports dragging and dropping of tabs between tab widgets that share a group_id.
<i>Tab</i> (widget, text, data, text_color, tooltip, ...)	
<i>TabAnimation</i> (parent)	
<i>debug</i> ()	

labscript_utils.qtwidgets.dragdroptab.DragDropTabBar

class labscript_utils.qtwidgets.dragdroptab.DragDropTabBar(*parent*, *group_id*)

Bases: labscript_utils.qtwidgets.dragdroptab._BaseDragDropTabBar

__init__(*parent*, *group_id*)

Methods

__init__ (parent, group_id)	
acceptDrops(self)	
accessibleDescription(self)	
accessibleName(self)	
accessibleTabName(self, index)	
actionEvent(self, a0)	
actions(self)	

continues on next page

Table 2.66 – continued from previous page

activateWindow(self)	
addAction(self, action)	
addActions(self, actions)	
addTab(-> int)	
add_dragged_tab(index, tab)	Insert the tab at the given index and set all of its configuration
adjustSize(self)	
autoFillBackground(self)	
autoHide(self)	
backgroundRole(self)	
baseSize(self)	
blockSignals(self, b)	
changeCurrentOnDrag(self)	
changeEvent(self, a0)	
childAt(-> QWidget)	
childEvent(self, a0)	
children(self)	
childrenRect(self)	
childrenRegion(self)	
clearFocus(self)	
clearMask(self)	
close(self)	
closeEvent(self, a0)	
colorCount(self)	
connectNotify(self, signal)	
contentsMargins(self)	

continues on next page

Table 2.66 – continued from previous page

<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>count(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>currentIndex(self)</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>documentMode(self)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>drawBase(self)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>

continues on next page

Table 2.66 – continued from previous page

<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>elideMode(self)</code>
<code>ensurePolished(self)</code>
<code><i>ensure_visible</i>(index[, prefer_left])</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>expanding(self)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>

continues on next page

Table 2.66 – continued from previous page

<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>iconSize(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option, tabIndex)</code>
<code>inputMethodEvent(self, a0)</code>

continues on next page

Table 2.66 – continued from previous page

<code>inputMethodHints(self)</code>	
<code>inputMethodQuery(self, a0)</code>	
<code>insertAction(self, before, action)</code>	
<code>insertActions(self, before, actions)</code>	
<code>insertTab(-> int)</code>	
<code><i>insertion_index_at</i>(pos)</code>	Compute at which index the tab with given upper left corner position in global coordinates should be inserted into the tabBar.
<code>installEventFilter(self, a0)</code>	
<code>isActiveWindow(self)</code>	
<code>isAncestorOf(self, child)</code>	
<code>isEnabled(self)</code>	
<code>isEnabledTo(self, a0)</code>	
<code>isFullScreen(self)</code>	
<code>isHidden(self)</code>	
<code>isLeftToRight(self)</code>	
<code>isMaximized(self)</code>	
<code>isMinimized(self)</code>	
<code>isModal(self)</code>	
<code>isMovable(self)</code>	
<code>isRightToLeft(self)</code>	
<code>isSignalConnected(self, signal)</code>	
<code>isTabEnabled(self, index)</code>	
<code>isTabVisible(self, index)</code>	
<code>isVisible(self)</code>	
<code>isVisibleTo(self, a0)</code>	
<code>isWidgetType(self)</code>	

continues on next page

Table 2.66 – continued from previous page

<code>isWindow(self)</code>	
<code>isWindowModified(self)</code>	
<code>isWindowType(self)</code>	
<code>is_dragged_tab(index)</code>	Return whether the tab at the given index is currently being dragged
<code>keyPressEvent(self, a0)</code>	
<code>keyReleaseEvent(self, a0)</code>	
<code>keyboardGrabber()</code>	
<code>killTimer(self, id)</code>	
<code>layout(self)</code>	
<code>layoutDirection(self)</code>	
<code>leaveEvent(self, a0)</code>	
<code>locale(self)</code>	
<code>logicalDpiX(self)</code>	
<code>logicalDpiY(self)</code>	
<code>lower(self)</code>	
<code>mapFrom(self, a0, a1)</code>	
<code>mapFromGlobal(self, a0)</code>	
<code>mapFromParent(self, a0)</code>	
<code>mapTo(self, a0, a1)</code>	
<code>mapToGlobal(self, a0)</code>	
<code>mapToParent(self, a0)</code>	
<code>mask(self)</code>	
<code>maximumHeight(self)</code>	
<code>maximumSize(self)</code>	
<code>maximumWidth(self)</code>	

continues on next page

Table 2.66 – continued from previous page

<code>metaObject(self)</code>	
<code>metric(self, a0)</code>	
<code>minimumHeight(self)</code>	
<code>minimumSize(self)</code>	
<code><i>minimumSizeHint</i>(self)</code>	
<code>minimumTabSizeHint(self, index)</code>	
<code>minimumWidth(self)</code>	
<code>mouseDoubleClickEvent(self, a0)</code>	
<code>mouseGrabber()</code>	
<code><i>mouseMoveEvent</i>(event)</code>	Update the parent of the tab to be the DragDropTabWidget under the mouse, if any, otherwise update it to the limbo object.
<code><i>mousePressEvent</i>(event)</code>	Take note of the tab that was clicked so it can be dragged on mouseMoveEvents
<code><i>mouseReleaseEvent</i>(event)</code>	Same as mouseMove event - update the DragDropTabWidget and position of the tab to the current mouse position.
<code>move()</code>	
<code>moveEvent(self, a0)</code>	
<code>moveTab(self, from_, to)</code>	
<code>moveToThread(self, thread)</code>	
<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code><i>on_scroll_button_clicked</i>(button)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	

continues on next page

Table 2.66 – continued from previous page

<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paint_tab(index, painter, option)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>removeTab(self, index)</code>	
<code>remove_dragged_tab(index)</code>	Remove the tab at the given index and return all its configuration
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	

continues on next page

Table 2.66 – continued from previous page

<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>selectionBehaviorOnRemove(self)</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAccessibleTabName(self, index, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setAutoHide(self, hide)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setChangeCurrentOnDrag(self, change)</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCurrentIndex(self, index)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDocumentMode(self, set)</code>

continues on next page

Table 2.66 – continued from previous page

<code>setDrawBase(self, drawTheBase)</code>
<code>setElideMode(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setExpanding(self, enabled)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setIconSize(self, size)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>

continues on next page

Table 2.66 – continued from previous page

<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setMovable(self, movable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setSelectionBehaviorOnRemove(self, behavior)</code>
<code>setShape(self, shape)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabButton(self, index, position, widget)</code>
<code>setTabData(self, index, data)</code>
<code>setTabEnabled(self, index, a1)</code>
<code>setTabIcon(self, index, icon)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabText(self, index, text)</code>
<code>setTabTextColor(self, index, color)</code>
<code>setTabToolTip(self, index, tip)</code>

continues on next page

Table 2.66 – continued from previous page

<code>setTabVisible(self, index, visible)</code>	
<code>setTabWhatsThis(self, index, text)</code>	
<code>setTabletTracking(self, enable)</code>	
<code>setTabsClosable(self, closable)</code>	
<code>setToolTip(self, a0)</code>	
<code>setToolTipDuration(self, msec)</code>	
<code>setUpdatesEnabled(self, enable)</code>	
<code><i>setUsesScrollButtons</i>(self, useButtons)</code>	
<code>setVisible(self, visible)</code>	
<code>setWhatsThis(self, a0)</code>	
<code>setWindowFilePath(self, filePath)</code>	
<code>setWindowFlag(self, a0[, on])</code>	
<code>setWindowFlags(self, type)</code>	
<code>setWindowIcon(self, icon)</code>	
<code>setWindowIconText(self, a0)</code>	
<code>setWindowModality(self, windowModality)</code>	
<code>setWindowModified(self, a0)</code>	
<code>setWindowOpacity(self, level)</code>	
<code>setWindowRole(self, a0)</code>	
<code>setWindowState(self, state)</code>	
<code>setWindowTitle(self, a0)</code>	
<code><i>set_tab_parent</i>(dest[, index, pos])</code>	Move the tab to the given parent DragDropTabBar if it's not already there.
<code>shape(self)</code>	
<code>sharedPainter(self)</code>	
<code>show(self)</code>	

continues on next page

Table 2.66 – continued from previous page

<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code><i>sizeHint</i>(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code><i>tabAt</i>(self, pos)</code>
<code>tabButton(self, index, position)</code>
<code>tabData(self, index)</code>
<code>tabIcon(self, index)</code>
<code><i>tabInserted</i>(self, index)</code>
<code><i>tabLayoutChange</i>(self)</code>
<code><i>tabRect</i>(self, index)</code>
<code><i>tabRemoved</i>(self, index)</code>
<code>tabSizeHint(self, index)</code>
<code>tabText(self, index)</code>

continues on next page

Table 2.66 – continued from previous page

<code>tabTextColor(self, index)</code>	
<code>tabToolTip(self, index)</code>	
<code>tabWhatsThis(self, index)</code>	
<code>tabletEvent(self, a0)</code>	
<code>tabsClosable(self)</code>	
<code>testAttribute(self, attribute)</code>	
<code>thread(self)</code>	
<code>timerEvent(self, event)</code>	
<code>toolTip(self)</code>	
<code>toolTipDuration(self)</code>	
<code>tr(self, sourceText[, disambiguation, n])</code>	
<code>underMouse(self)</code>	
<code>ungrabGesture(self, type)</code>	
<code>unsetCursor(self)</code>	
<code>unsetLayoutDirection(self)</code>	
<code>unsetLocale(self)</code>	
<code><i>update</i>(-> None -> None)</code>	
<code>updateGeometry(self)</code>	
<code>updateMicroFocus(self)</code>	
<code><i>update_dragged_tab_animation_pos</i>(pos)</code>	
<code><i>update_scroll_button_state</i>()</code>	
<code><i>update_tab_index</i>(index, pos)</code>	Check if the tab at the given index, being dragged by the mouse at the given position, needs to be moved.
<code>updatesEnabled(self)</code>	
<code>usesScrollButtons(self)</code>	
<code>visibleRegion(self)</code>	

continues on next page

Table 2.66 – continued from previous page

<code>whatsThis(self)</code>	
<code>wheelEvent(self, event)</code>	
<code>widgetAt(pos)</code>	If the given position is over a DragDropTabBar belonging to the current group, return the DragDropTabBar.
<code>width(self)</code>	
<code>widthMM(self)</code>	
<code>winId(self)</code>	
<code>window(self)</code>	
<code>windowFilePath(self)</code>	
<code>windowFlags(self)</code>	
<code>windowHandle(self)</code>	
<code>windowIcon(self)</code>	
<code>windowIconText(self)</code>	
<code>windowModality(self)</code>	
<code>windowOpacity(self)</code>	
<code>windowRole(self)</code>	
<code>windowState(self)</code>	
<code>windowTitle(self)</code>	
<code>windowType(self)</code>	
<code>x(self)</code>	
<code>y(self)</code>	

Attributes

DrawChildren

DrawWindowBackground

FLUSH_GAP

IgnoreMask

LeftSide

PdmDepth

PdmDevicePixelRatio

PdmDevicePixelRatioScaled

PdmDpiX

PdmDpiY

PdmHeight

PdmHeightMM

PdmNumColors

PdmPhysicalDpiX

PdmPhysicalDpiY

PdmWidth

PdmWidthMM

RightSide

RoundedEast

RoundedNorth

RoundedSouth

RoundedWest

SCROLL_BUTTON_GAP

SCROLL_BUTTON_WIDTH

continues on next page

Table 2.67 – continued from previous page

SelectLeftTab	
SelectPreviousTab	
SelectRightTab	
TriangularEast	
TriangularNorth	
TriangularSouth	
TriangularWest	
currentChanged	int) [signal]
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
<i>drag_in_progress</i>	
dragged_tab_grab_point	
<i>dragged_tab_index</i>	
<i>dragged_tab_parent</i>	
<i>limbo</i>	
objectNameChanged	str) [signal]
staticMetaObject	
tabBarClicked	int) [signal]
tabBarDoubleClicked	int) [signal]
tabCloseRequested	int) [signal]
tabMoved	int) [signal]
<i>tab_widgets</i>	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

FLUSH_GAP = 5

SCROLL_BUTTON_GAP = 2

SCROLL_BUTTON_WIDTH = 15

property drag_in_progress

property dragged_tab_index

property dragged_tab_parent

ensure_visible(*index*, *prefer_left=True*)

insertion_index_at(*pos*)
 Compute at which index the tab with given upper left corner position in global coordinates should be inserted into the tabBar.

is_dragged_tab(*index*)
 Return whether the tab at the given index is currently being dragged

limbo = None

minimumSizeHint(*self*) → QSize

mouseMoveEvent(*event*)
 Update the parent of the tab to be the DragDropTabWidget under the mouse, if any, otherwise update it to the limbo object. Update the position of the tab in the widget it's in.

mousePressEvent(*event*)
 Take note of the tab that was clicked so it can be dragged on mouseMoveEvents

mouseReleaseEvent(*event*)
 Same as mouseMove event - update the DragDropTabWidget and position of the tab to the current mouse position. Unless the mouse position is outside of any widgets at the time of mouse release, in which case move the tab to its last known parent and position.

on_scroll_button_clicked(*button*)

paintEvent(*self*, *a0*: QPaintEvent)

paint_tab(*index*, *painter*, *option*)

setUsesScrollButtons(*self*, *useButtons*: bool)

set_tab_parent(*dest*, *index*=None, *pos*=None)
 Move the tab to the given parent DragDropTabBar if it's not already there. *index*=None will determined the insertion index from the given mouse position.

sizeHint(*self*) → QSize

tabAt(*self*, *pos*: QPoint) → int

tabInserted(*self*, *index*: int)

tabLayoutChange(*self*)

tabRect(*self*, *index*: int) → QRect

tabRemoved(*self*, *index*: int)

tab_widgets = {}

update(*self*)

update(*self*, *a0*: QRect) → None

update(*self*, *a0*: QRegion) → None

update(*self*, *ax*: int, *ay*: int, *aw*: int, *ah*: int) → None

update_dragged_tab_animation_pos(*pos*)

update_scroll_button_state()

update_tab_index(*index*, *pos*)
 Check if the tab at the given index, being dragged by the mouse at the given position, needs to be moved. Move it and return the new index.

widgetAt (*pos*)

If the given position is over a DragDropTabBar belonging to the current group, return the DragDropTabBar. If it is over a TabWidget in the same group that has no tabs, or the dragged tab as its only tab, return its DragDropTabBar. Otherwise return the limbo object.

labscript_utils.qtwidgets.dragdroptab.DragDropTabWidget

class labscript_utils.qtwidgets.dragdroptab.DragDropTabWidget (*group_id=None, accept_drops_bar_only=False*)

Bases: PyQt5.QtWidgets.QTabWidget

A tab widget that supports dragging and dropping of tabs between tab widgets that share a *group_id*. a *group_id* of None indicates that tab dragging is disabled.

__init__ (*group_id=None, accept_drops_bar_only=False*)

Methods

__init__ ([*group_id, accept_drops_bar_only*])

acceptDrops (self)

accessibleDescription (self)

accessibleName (self)

actionEvent (self, a0)

actions (self)

activateWindow (self)

addAction (self, action)

addActions (self, actions)

addTab (-> int)

adjustSize (self)

autoFillBackground (self)

backgroundRole (self)

baseSize (self)

blockSignals (self, b)

changeEvent (self, a0)

continues on next page

Table 2.68 – continued from previous page

<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clear(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>cornerWidget(self[, corner])</code>
<code>count(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>currentIndex(self)</code>
<code>currentWidget(self)</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>

continues on next page

Table 2.68 – continued from previous page

<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>documentMode(self)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>elideMode(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>

continues on next page

Table 2.68 – continued from previous page

<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>

continues on next page

Table 2.68 – continued from previous page

<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, width)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>iconSize(self)</code>
<code>indexOf(self, widget)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>insertTab(-> int)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>

continues on next page

Table 2.68 – continued from previous page

<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isMovable(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isTabEnabled(self, index)</code>
<code>isTabVisible(self, index)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>

continues on next page

Table 2.68 – continued from previous page

<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>

continues on next page

Table 2.68 – continued from previous page

<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	

continues on next page

Table 2.68 – continued from previous page

<code>removeAction(self, action)</code>
<code>removeEventFilter(self, a0)</code>
<code>removeTab(self, index)</code>
<code>render(, sourceRegion, flags, ...)</code>
<code>repaint(-> None -> None)</code>
<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCornerWidget(self, widget[, corner])</code>
<code>setCurrentIndex(self, index)</code>
<code>setCurrentWidget(self, widget)</code>

continues on next page

Table 2.68 – continued from previous page

<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDocumentMode(self, set)</code>
<code><i>setElideMode</i>(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setIconSize(self, size)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>

continues on next page

Table 2.68 – continued from previous page

<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setMovable(self, movable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabBar(self, a0)</code>
<code>setTabBarAutoHide(self, enabled)</code>
<code>setTabEnabled(self, index, a1)</code>
<code>setTabIcon(self, index, icon)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabPosition(self, a0)</code>
<code>setTabShape(self, s)</code>
<code>setTabText(self, index, a1)</code>
<code>setTabToolTip(self, index, tip)</code>

continues on next page

Table 2.68 – continued from previous page

<code>setTabVisible(self, index, visible)</code>
<code>setTabWhatsThis(self, index, text)</code>
<code>setTabletTracking(self, enable)</code>
<code>setTabsClosable(self, closeable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code><i>setUsesScrollButtons</i>(self, useButtons)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>

continues on next page

Table 2.68 – continued from previous page

<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabBar(self)</code>
<code>tabBarAutoHide(self)</code>
<code>tabIcon(self, index)</code>
<code>tabInserted(self, index)</code>
<code>tabPosition(self)</code>
<code>tabRemoved(self, index)</code>
<code>tabShape(self)</code>
<code>tabText(self, index)</code>
<code>tabToolTip(self, index)</code>
<code>tabWhatsThis(self, index)</code>
<code>tabletEvent(self, a0)</code>
<code>tabsClosable(self)</code>

continues on next page

Table 2.68 – continued from previous page

<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>usesScrollButtons(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>widget(self, index)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>

continues on next page

Table 2.68 – continued from previous page

windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

DrawChildren
DrawWindowBackground
East
IgnoreMask
North
PdmDepth
PdmDevicePixelRatio
PdmDevicePixelRatioScaled
PdmDpiX
PdmDpiY
PdmHeight

continues on next page

Table 2.69 – continued from previous page

PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
Rounded	
South	
Triangular	
West	
currentChanged	int) [signal]
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
tabBarClicked	int) [signal]
tabBarDoubleClicked	int) [signal]
tabCloseRequested	int) [signal]
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

setElideMode(*self*, *a0*: *Qt.TextElideMode*)

setUsesScrollButtons(*self*, *useButtons*: *bool*)

labscript_utils.qtwidgets.dragdroptab.Tab

class labscript_utils.qtwidgets.dragdroptab.Tab(*widget*, *text*, *data*, *text_color*, *tooltip*, *whats_this*,
button_left, *button_right*, *icon*)

Bases: *tuple*

__init__()

Methods

<code>__init__()</code>	
<code>count(value, /)</code>	Return number of occurrences of value.
<code>index(value[, start, stop])</code>	Return first index of value.

Attributes

<code>button_left</code>	Alias for field number 6
<code>button_right</code>	Alias for field number 7
<code>data</code>	Alias for field number 2
<code>icon</code>	Alias for field number 8
<code>text</code>	Alias for field number 1
<code>text_color</code>	Alias for field number 3
<code>tooltip</code>	Alias for field number 4
<code>whats_this</code>	Alias for field number 5
<code>widget</code>	Alias for field number 0

property button_left
Alias for field number 6

property button_right
Alias for field number 7

property data
Alias for field number 2

property icon
Alias for field number 8

property text
Alias for field number 1

property text_color
Alias for field number 3

property tooltip
Alias for field number 4

property whats_this
Alias for field number 5

property widget
Alias for field number 0

labscript_utils.qtwidgets.dragdroptab.TabAnimation

class labscript_utils.qtwidgets.dragdroptab.TabAnimation(*parent*)

Bases: PyQt5.QtCore.QAbstractAnimation

`__init__(parent)`

Methods

`__init__(parent)`

`animate_limbo(limbo, index)`

If the floating tab in limbo is being sucked back into one of our tabs, then we can animate that by hiding the relevant tab rect off to the side somewhere whilst the floating tab swoops in.

`blockSignals(self, b)`

`childEvent(self, a0)`

`children(self)`

`connectNotify(self, signal)`

`currentLoop(self)`

`currentLoopTime(self)`

`currentTime(self)`

`customEvent(self, a0)`

`deleteLater(self)`

`direction(self)`

`disconnect(-> bool)`

`disconnectNotify(self, signal)`

`dumpObjectInfo(self)`

`dumpObjectTree(self)`

`duration(self)`

`dynamicPropertyNames(self)`

`ensure_running()`

`event(self, event)`

continues on next page

Table 2.72 – continued from previous page

<code>eventFilter(self, a0, a1)</code>	
<code>findChild(-> QObject)</code>	
<code>findChildren(...)</code>	
<code>group(self)</code>	
<code>inherits(self, classname)</code>	
<code>installEventFilter(self, a0)</code>	
<code>isSignalConnected(self, signal)</code>	
<code>isWidgetType(self)</code>	
<code>isWindowType(self)</code>	
<code>killTimer(self, id)</code>	
<code>loopCount(self)</code>	
<code>metaObject(self)</code>	
<code>moveToThread(self, thread)</code>	
<code>objectName(self)</code>	
<code><i>on_tab_moved</i>(source_index, dest_index)</code>	
<code>parent(self)</code>	
<code>pause(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>receivers(self, signal)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>resume(self)</code>	
<code>sender(self)</code>	
<code>senderSignalIndex(self)</code>	
<code>setCurrentTime(self, msecs)</code>	

continues on next page

Table 2.72 – continued from previous page

<code>setDirection(self, direction)</code>	
<code>setLoopCount(self, loopCount)</code>	
<code>setObjectName(self, name)</code>	
<code>setParent(self, a0)</code>	
<code>setPaused(self, a0)</code>	
<code>setProperty(self, name, value)</code>	
<code>signalsBlocked(self)</code>	
<code>start(self[, policy])</code>	
<code>startTimer(self, interval[, timerType])</code>	
<code>state(self)</code>	
<code>stop(self)</code>	
<code><i>tabInserted</i>(index)</code>	
<code><i>tabRemoved</i>(index)</code>	
<code><i>target</i>(i)</code>	Return the target position we are animating toward for a tab
<code>thread(self)</code>	
<code>timerEvent(self, a0)</code>	
<code>totalDuration(self)</code>	
<code>tr(self, sourceText[, disambiguation, n])</code>	
<code><i>updateCurrentTime</i>(self, currentTime)</code>	
<code>updateDirection(self, direction)</code>	
<code>updateState(self, newState, oldState)</code>	

Attributes

Backward	
DeleteWhenStopped	
Forward	
KeepWhenStopped	
Paused	
Running	
Stopped	
currentLoopChanged	int [signal]
destroyed	typing.Optional[QObject] = None [signal]
directionChanged	QAbstractAnimation.Direction [signal]
finished	finished(self) [signal]
objectNameChanged	str [signal]
stateChanged	QAbstractAnimation.State [signal]
staticMetaObject	
<i>tau</i>	

animate_limbo(*limbo*, *index*)

If the floating tab in limbo is being sucked back into one of our tabs, then we can animate that by hiding the relevant tab rect off to the side somewhere whilst the floating tab swoops in.

duration(*self*) → int

ensure_running()

on_tab_moved(*source_index*, *dest_index*)

tabInserted(*index*)

tabRemoved(*index*)

target(*i*)

Return the target position we are animating toward for a tab

tau = 60.0

updateCurrentTime(*self*, *currentTime*: int)

labscript_utils.qtwidgets.dragdroptab.debug

```
class labscript_utils.qtwidgets.dragdroptab.debug
```

```
    Bases: object
```

```
    __init__()
```

Methods

```
__init__()
```

```
trace(f)                                decorator to print function entries and exits
```

Attributes

```
DEBUG
```

```
depth
```

```
DEBUG = False
```

```
depth = 0
```

```
classmethod trace(f)
    decorator to print function entries and exits
```

labscript_utils.qtwidgets.elide_label**Functions**

```
elide_label(label, layout, elide_mode)    Take an existing label that is in a layout, and wrap it in
                                           our widgets that elide the text, and insert it back into the
                                           layout.
```

labscript_utils.qtwidgets.elide_label.elide_label

```
labscript_utils.qtwidgets.elide_label.elide_label(label, layout, elide_mode)
```

Take an existing label that is in a layout, and wrap it in our widgets that elide the text, and insert it back into the layout. This is a hack that allows us to elide a QLabel with a single line of (possibly rich) text, a task that seems pretty much impossible to do in any kosher way.

This function is for modifying an existing label already in a layout, but if you are programatically creating a label, then you can wrap it in `ElidedLabelContainer(label)` before inserting it into a layout or other container widget, which is more flexible than this function which only works if the label is in a `QBoxLayout`

Classes

<i>ElideScrollArea</i> (*args, **kwargs)	A ScrollArea for containing a label that we want to elide.
<i>ElidedLabelContainer</i> (label)	A QWidget to contain a QLabel with a single line of (possibly rich) text that we want to elide.

labscript_utils.qtwidgets.elide_label.ElideScrollArea

class labscript_utils.qtwidgets.elide_label.**ElideScrollArea**(*args, **kwargs)

Bases: PyQt5.QtWidgets.QScrollArea

A ScrollArea for containing a label that we want to elide. The elision is attained by just letting the text we don't want to see be scrolled off to the side with the scrollbars hidden.

__init__(*args, **kwargs)

Methods

__init__(*args, **kwargs)

acceptDrops(self)

accessibleDescription(self)

accessibleName(self)

actionEvent(self, a0)

actions(self)

activateWindow(self)

addAction(self, action)

addActions(self, actions)

addScrollBarWidget(self, widget, alignment)

adjustSize(self)

alignment(self)

autoFillBackground(self)

backgroundRole(self)

baseSize(self)

blockSignals(self, b)

continues on next page

Table 2.78 – continued from previous page

<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>cornerWidget(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWin- dows])</code>
<code>devType(self)</code>

continues on next page

Table 2.78 – continued from previous page

<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>drawFrame(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>ensureVisible(self, x, y[, xMargin, yMargin])</code>
<code>ensureWidgetVisible(self, childWidget[, ...])</code>
<code>enterEvent(self, a0)</code>
<code><i>event</i>(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>

continues on next page

Table 2.78 – continued from previous page

<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameRect(self)</code>
<code>frameShadow(self)</code>
<code>frameShape(self)</code>
<code>frameSize(self)</code>
<code>frameStyle(self)</code>
<code>frameWidth(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>

continues on next page

Table 2.78 – continued from previous page

<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>horizontalScrollBar(self)</code>
<code>horizontalScrollBarPolicy(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>

continues on next page

Table 2.78 – continued from previous page

<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>lineWidth(self)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>

continues on next page

Table 2.78 – continued from previous page

<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumViewportSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>midLineWidth(self)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code><i>minimumSizeHint</i>(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>

continues on next page

Table 2.78 – continued from previous page

<code>moveEvent(self, a0)</code>	
<code>moveToThread(self, thread)</code>	
<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	

continues on next page

Table 2.78 – continued from previous page

<code>releaseMouse(self)</code>
<code>releaseShortcut(self, id)</code>
<code>removeAction(self, action)</code>
<code>removeEventFilter(self, a0)</code>
<code>render(, sourceRegion, flags, ...)</code>
<code>repaint(-> None -> None)</code>
<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>scrollBarWidgets(self, alignment)</code>
<code>scrollContentsBy(self, dx, dy)</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAlignment(self, a0)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>

continues on next page

Table 2.78 – continued from previous page

<code>setContextMenuPolicy(self, policy)</code>
<code>setCornerWidget(self, widget)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code><i>setElideMode</i>(elideMode)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setFrameRect(self, a0)</code>
<code>setFrameShadow(self, a0)</code>
<code>setFrameShape(self, a0)</code>
<code>setFrameStyle(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setHorizontalScrollBar(self, scrollbar)</code>
<code>setHorizontalScrollBarPolicy(self, a0)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>

continues on next page

Table 2.78 – continued from previous page

<code>setLayoutDirection(self, direction)</code>
<code>setLineWidth(self, a0)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMidLineWidth(self, a0)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeAdjustPolicy(self, policy)</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>

continues on next page

Table 2.78 – continued from previous page

<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVerticalScrollBar(self, scrollbar)</code>
<code>setVerticalScrollBarPolicy(self, a0)</code>
<code>setViewport(self, widget)</code>
<code>setViewportMargins()</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code><i>setWidget</i>(self, w)</code>
<code>setWidgetResizable(self, resizable)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>setupViewport(self, viewport)</code>
<code>sharedPainter(self)</code>

continues on next page

Table 2.78 – continued from previous page

<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeAdjustPolicy(self)</code>
<code><i>sizeHint</i>(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>takeWidget(self)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>

continues on next page

Table 2.78 – continued from previous page

<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>verticalScrollBar(self)</code>
<code>verticalScrollBarPolicy(self)</code>
<code>viewport(self)</code>
<code>viewportEvent(self, a0)</code>
<code>viewportMargins(self)</code>
<code>viewportSizeHint(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>widget(self)</code>
<code>widgetResizable(self)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>

continues on next page

Table 2.78 – continued from previous page

windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

AdjustIgnored
AdjustToContents
AdjustToContentsOnFirstShow
Box
DrawChildren
DrawWindowBackground
HLine
IgnoreMask
NoFrame
Panel
PdmDepth

continues on next page

Table 2.79 – continued from previous page

PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
Plain	
Raised	
Shadow_Mask	
Shape_Mask	
StyledPanel	
Sunken	
VLine	
WinPanel	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

event(*self*, *a0*: *QEvent*) → bool

minimumSizeHint(*self*) → QSize

setElideMode(*elideMode*)

setWidget(*self*, *w*: *QWidget*)

sizeHint(*self*) → *QSize*

labscript_utils.qtwidgets.elide_label.ElidedLabelContainer

class `labscript_utils.qtwidgets.elide_label.ElidedLabelContainer`(*label*)

Bases: `PyQt5.QtWidgets.QWidget`

A `QWidget` to contain a `QLabel` with a single line of (possibly rich) text that we want to elide. The elision is obtained by putting the `QLabel` in a `QScrollArea` and having the `QScrollArea` only show the part of the text we want to see. An extra label with the elision indication “...” is also inserted next to the `QScrollArea`.

__init__(*label*)

Methods

`__init__`(*label*)

`acceptDrops`(*self*)

`accessibleDescription`(*self*)

`accessibleName`(*self*)

`actionEvent`(*self*, *a0*)

`actions`(*self*)

`activateWindow`(*self*)

`addAction`(*self*, *action*)

`addActions`(*self*, *actions*)

`adjustSize`(*self*)

`autoFillBackground`(*self*)

`backgroundRole`(*self*)

`baseSize`(*self*)

`blockSignals`(*self*, *b*)

`changeEvent`(*self*, *a0*)

`childAt`(-> *QWidget*)

`childEvent`(*self*, *a0*)

continues on next page

Table 2.80 – continued from previous page

<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWin- dows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>

continues on next page

Table 2.80 – continued from previous page

<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code><i>elideMode()</i></code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code><i>event</i>(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>

continues on next page

Table 2.80 – continued from previous page

<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>inherits(self, classname)</code>

continues on next page

Table 2.80 – continued from previous page

<code>initPainter(self, painter)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>

continues on next page

Table 2.80 – continued from previous page

<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>

continues on next page

Table 2.80 – continued from previous page

<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>overrideWindowFlags(self, type)</code>
<code>overrideWindowState(self, state)</code>
<code>paintEngine(self)</code>
<code>paintEvent(self, a0)</code>
<code>paintingActive(self)</code>
<code>palette(self)</code>
<code>parent(self)</code>
<code>parentWidget(self)</code>
<code>physicalDpiX(self)</code>
<code>physicalDpiY(self)</code>

continues on next page

Table 2.80 – continued from previous page

<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	
<code>resize()</code>	
<code>resizeEvent(self, a0)</code>	
<code>restoreGeometry(self, geometry)</code>	
<code>saveGeometry(self)</code>	
<code>screen(self)</code>	
<code>scroll()</code>	
<code>sender(self)</code>	
<code>senderSignalIndex(self)</code>	
<code>setAcceptDrops(self, on)</code>	
<code>setAccessibleDescription(self, description)</code>	
<code>setAccessibleName(self, name)</code>	

continues on next page

Table 2.80 – continued from previous page

<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code><i>setElideMode</i>(elideMode)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>

continues on next page

Table 2.80 – continued from previous page

<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>

continues on next page

Table 2.80 – continued from previous page

<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code><i>sizeHint</i>(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>

continues on next page

Table 2.80 – continued from previous page

<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code><i>update_elide_widget()</i></code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>

continues on next page

Table 2.80 – continued from previous page

widthMM(self)
winId(self)
window(self)
windowFilePath(self)
windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

DrawChildren
DrawWindowBackground
IgnoreMask
PdmDepth
PdmDevicePixelRatio
PdmDevicePixelRatioScaled
PdmDpiX

continues on next page

Table 2.81 – continued from previous page

PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

elideMode()

event(*self*, *a0*: *QEvent*) → bool

minimumSizeHint(*self*) → QSize

setElideMode(*elideMode*)

sizeHint(*self*) → QSize

update_elide_widget()

labscript_utils.qtwidgets.enumoutput

Classes

EnumOutput(hardware_name[, connection_name, ...])

labscript_utils.qtwidgets.enumoutput.EnumOutput

class labscript_utils.qtwidgets.enumoutput.**EnumOutput** (*hardware_name*, *connection_name*='-',
display_name=None,
horizontal_alignment=False, *parent*=None)

Bases: PyQt5.QtWidgets.QWidget

__init__ (*hardware_name*, *connection_name*='-', *display_name*=None, *horizontal_alignment*=False,
parent=None)

Methods

__init__(hardware_name[, connection_name, ...])

acceptDrops(self)

accessibleDescription(self)

accessibleName(self)

actionEvent(self, a0)

actions(self)

activateWindow(self)

addAction(self, action)

addActions(self, actions)

adjustSize(self)

autoFillBackground(self)

backgroundRole(self)

baseSize(self)

blockSignals(self, b)

block_combobox_signals()

changeEvent(self, a0)

childAt(-> QWidget)

childEvent(self, a0)

children(self)

continues on next page

Table 2.83 – continued from previous page

<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code><i>connect_value_change</i>(func)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>

continues on next page

Table 2.83 – continued from previous page

<code>disconnectNotify(self, signal)</code>
<code><i>disconnect_value_change()</i></code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code><i>eventFilter</i>(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>

continues on next page

Table 2.83 – continued from previous page

<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentMargins(self)</code>
<code>get_EO()</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>

continues on next page

Table 2.83 – continued from previous page

<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>

continues on next page

Table 2.83 – continued from previous page

<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code><i>lock</i>([notify_eo])</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>

continues on next page

Table 2.83 – continued from previous page

<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>overrideWindowFlags(self, type)</code>
<code>overrideWindowState(self, state)</code>
<code>paintEngine(self)</code>
<code>paintEvent(self, a0)</code>
<code>paintingActive(self)</code>
<code>palette(self)</code>
<code>parent(self)</code>
<code>parentWidget(self)</code>

continues on next page

Table 2.83 – continued from previous page

<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	
<code>resize()</code>	
<code>resizeEvent(self, a0)</code>	
<code>restoreGeometry(self, geometry)</code>	
<code>saveGeometry(self)</code>	
<code>screen(self)</code>	
<code>scroll()</code>	
<code>sender(self)</code>	
<code>senderSignalIndex(self)</code>	
<code>setAcceptDrops(self, on)</code>	

continues on next page

Table 2.83 – continued from previous page

<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>

continues on next page

Table 2.83 – continued from previous page

<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>

continues on next page

Table 2.83 – continued from previous page

<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>set_EO(EO[, notify_old_EO, notify_new_EO])</code>
<code>set_combobox_model(model)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>

continues on next page

Table 2.83 – continued from previous page

<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code><i>unblock_combobox_signals()</i></code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code><i>unlock</i>([notify_eo])</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>

continues on next page

Table 2.83 – continued from previous page

<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>
<code>windowModality(self)</code>
<code>windowOpacity(self)</code>
<code>windowRole(self)</code>
<code>windowState(self)</code>
<code>windowTitle(self)</code>
<code>windowType(self)</code>
<code>x(self)</code>
<code>y(self)</code>

Attributes

<code>DrawChildren</code>
<code>DrawWindowBackground</code>
<code>IgnoreMask</code>

continues on next page

Table 2.84 – continued from previous page

PdmDepth	
PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
<i>selected_index</i>	
<i>selected_option</i>	
staticMetaObject	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

```

block_combobox_signals()
connect_value_change(func)
disconnect_value_change()
eventFilter(self, a0: QObject, a1: QEvent) → bool
get_EO()
lock(notify_eo=True)
property selected_index
property selected_option
set_EO(EO, notify_old_EO=True, notify_new_EO=True)
set_combobox_model(model)

```

`unblock_combobox_signals()`

`unlock(notify_eo=True)`

labscript_utils.qtwidgets.fingertab

Classes

FingerTabBarWidget([parent])

FingerTabWidget(parent, *args)

A QTabWidget equivalent which uses our FingerTabBarWidget

labscript_utils.qtwidgets.fingertab.FingerTabBarWidget

class labscript_utils.qtwidgets.fingertab.FingerTabBarWidget(*parent=None, *args, **kwargs*)

Bases: `PyQt5.QtWidgets.QTabBar`

`__init__`(*parent=None, *args, **kwargs*)

Methods

`__init__`([parent])

`acceptDrops`(self)

`accessibleDescription`(self)

`accessibleName`(self)

`accessibleTabName`(self, index)

`actionEvent`(self, a0)

`actions`(self)

`activateWindow`(self)

`addAction`(self, action)

`addActions`(self, actions)

`addTab`(-> int)

`adjustSize`(self)

`autoFillBackground`(self)

continues on next page

Table 2.86 – continued from previous page

<code>autoHide(self)</code>
<code>backgroundRole(self)</code>
<code>baseSize(self)</code>
<code>blockSignals(self, b)</code>
<code>changeCurrentOnDrag(self)</code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>count(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>currentIndex(self)</code>

continues on next page

Table 2.86 – continued from previous page

cursor(self)
customEvent(self, a0)
deleteLater(self)
depth(self)
destroy(self[, destroyWindow, destroySubWindows])
devType(self)
devicePixelRatio(self)
devicePixelRatioF(self)
devicePixelRatioFScale()
disconnect(-> bool)
disconnectNotify(self, signal)
documentMode(self)
dragEnterEvent(self, a0)
dragLeaveEvent(self, a0)
dragMoveEvent(self, a0)
drawBase(self)
dropEvent(self, a0)
dumpObjectInfo(self)
dumpObjectTree(self)
dynamicPropertyNames(self)
effectiveWinId(self)
elideMode(self)
ensurePolished(self)
enterEvent(self, a0)
event(self, a0)

continues on next page

Table 2.86 – continued from previous page

<code>eventFilter(self, a0, a1)</code>
<code>expanding(self)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>

continues on next page

Table 2.86 – continued from previous page

<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>iconSize(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option, tabIndex)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>insertTab(-> int)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>

continues on next page

Table 2.86 – continued from previous page

<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isMovable(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isTabEnabled(self, index)</code>
<code>isTabVisible(self, index)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>

continues on next page

Table 2.86 – continued from previous page

<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumTabSizeHint(self, index)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>

continues on next page

Table 2.86 – continued from previous page

<code>mousePressEvent(self, a0)</code>	
<code>mouseReleaseEvent(self, a0)</code>	
<code>move()</code>	
<code>moveEvent(self, a0)</code>	
<code>moveTab(self, from_, to)</code>	
<code>moveToThread(self, thread)</code>	
<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code><i>paintEvent</i>(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.

continues on next page

Table 2.86 – continued from previous page

<code>raise_(self)</code>
<code>receivers(self, signal)</code>
<code>rect(self)</code>
<code>releaseKeyboard(self)</code>
<code>releaseMouse(self)</code>
<code>releaseShortcut(self, id)</code>
<code>removeAction(self, action)</code>
<code>removeEventFilter(self, a0)</code>
<code>removeTab(self, index)</code>
<code>render(, sourceRegion, flags, ...)</code>
<code>repaint(-> None -> None)</code>
<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>selectionBehaviorOnRemove(self)</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAccessibleTabName(self, index, name)</code>
<code>setAttribute(self, attribute[, on])</code>

continues on next page

Table 2.86 – continued from previous page

<code>setAutoFillBackground(self, enabled)</code>
<code>setAutoHide(self, hide)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setChangeCurrentOnDrag(self, change)</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCurrentIndex(self, index)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDocumentMode(self, set)</code>
<code>setDrawBase(self, drawTheBase)</code>
<code>setElideMode(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setExpanding(self, enabled)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>

continues on next page

Table 2.86 – continued from previous page

<code>setHidden(self, hidden)</code>
<code>setIconSize(self, size)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setMovable(self, movable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setSelectionBehaviorOnRemove(self, behavior)</code>
<code>setShape(self, shape)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>

continues on next page

Table 2.86 – continued from previous page

<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabButton(self, index, position, widget)</code>
<code>setTabData(self, index, data)</code>
<code>setTabEnabled(self, index, a1)</code>
<code>setTabIcon(self, index, icon)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabText(self, index, text)</code>
<code>setTabTextColor(self, index, color)</code>
<code>setTabToolTip(self, index, tip)</code>
<code>setTabVisible(self, index, visible)</code>
<code>setTabWhatsThis(self, index, text)</code>
<code>setTabletTracking(self, enable)</code>
<code>setTabsClosable(self, closable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setUsesScrollButtons(self, useButtons)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>

continues on next page

Table 2.86 – continued from previous page

<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>shape(self)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>

continues on next page

Table 2.86 – continued from previous page

<code>tabAt(self, pos)</code>
<code>tabButton(self, index, position)</code>
<code>tabData(self, index)</code>
<code>tabIcon(self, index)</code>
<code>tabInserted(self, index)</code>
<code>tabLayoutChange(self)</code>
<code>tabRect(self, index)</code>
<code>tabRemoved(self, index)</code>
<code><i>tabSizeHint</i>(self, index)</code>
<code>tabText(self, index)</code>
<code>tabTextColor(self, index)</code>
<code>tabToolTip(self, index)</code>
<code>tabWhatsThis(self, index)</code>
<code>tabletEvent(self, a0)</code>
<code>tabsClosable(self)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, event)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>

continues on next page

Table 2.86 – continued from previous page

<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>usesScrollButtons(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, event)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>
<code>windowModality(self)</code>
<code>windowOpacity(self)</code>
<code>windowRole(self)</code>
<code>windowState(self)</code>
<code>windowTitle(self)</code>
<code>windowType(self)</code>
<code>x(self)</code>

continues on next page

Table 2.86 – continued from previous page

y(self)	
Attributes	
DrawChildren	
DrawWindowBackground	
IgnoreMask	
LeftSide	
PdmDepth	
PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
RightSide	
RoundedEast	
RoundedNorth	
RoundedSouth	
RoundedWest	
SelectLeftTab	

continues on next page

Table 2.87 – continued from previous page

SelectPreviousTab	
SelectRightTab	
TriangularEast	
TriangularNorth	
TriangularSouth	
TriangularWest	
currentChanged	int) [signal]
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
tabBarClicked	int) [signal]
tabBarDoubleClicked	int) [signal]
tabCloseRequested	int) [signal]
tabMoved	int) [signal]
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

paintEvent(self, a0: *QPaintEvent*)

tabSizeHint(self, index: int) → QSize

labscript_utils.qtwidgets.fingertab.FingerTabWidget

class labscript_utils.qtwidgets.fingertab.FingerTabWidget(parent, *args)

Bases: PyQt5.QtWidgets.QTabWidget

A QTabWidget equivalent which uses our FingerTabBarWidget

__init__(parent, *args)

Methods

__init__(parent, *args)

acceptDrops(self)

accessibleDescription(self)

accessibleName(self)

continues on next page

Table 2.88 – continued from previous page

<code>actionEvent(self, a0)</code>
<code>actions(self)</code>
<code>activateWindow(self)</code>
<code>addAction(self, action)</code>
<code>addActions(self, actions)</code>
<code>addTab(-> int)</code>
<code>adjustSize(self)</code>
<code>autoFillBackground(self)</code>
<code>backgroundRole(self)</code>
<code>baseSize(self)</code>
<code>blockSignals(self, b)</code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clear(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>

continues on next page

Table 2.88 – continued from previous page

<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>cornerWidget(self[, corner])</code>
<code>count(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>currentIndex(self)</code>
<code>currentWidget(self)</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>documentMode(self)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>

continues on next page

Table 2.88 – continued from previous page

<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>elideMode(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>

continues on next page

Table 2.88 – continued from previous page

<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, width)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>iconSize(self)</code>
<code>indexOf(self, widget)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>

continues on next page

Table 2.88 – continued from previous page

<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>insertTab(-> int)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isMovable(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isTabEnabled(self, index)</code>
<code>isTabVisible(self, index)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>

continues on next page

Table 2.88 – continued from previous page

<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>

continues on next page

Table 2.88 – continued from previous page

<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>overrideWindowFlags(self, type)</code>
<code>overrideWindowState(self, state)</code>
<code>paintEngine(self)</code>
<code>paintEvent(self, a0)</code>
<code>paintingActive(self)</code>
<code>palette(self)</code>
<code>parent(self)</code>
<code>parentWidget(self)</code>

continues on next page

Table 2.88 – continued from previous page

<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>removeTab(self, index)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	
<code>resize()</code>	
<code>resizeEvent(self, a0)</code>	
<code>restoreGeometry(self, geometry)</code>	
<code>saveGeometry(self)</code>	
<code>screen(self)</code>	
<code>scroll()</code>	
<code>sender(self)</code>	
<code>senderSignalIndex(self)</code>	

continues on next page

Table 2.88 – continued from previous page

<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCornerWidget(self, widget[, corner])</code>
<code>setCurrentIndex(self, index)</code>
<code>setCurrentWidget(self, widget)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDocumentMode(self, set)</code>
<code>setElideMode(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>

continues on next page

Table 2.88 – continued from previous page

<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setIconSize(self, size)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setMovable(self, movable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>

continues on next page

Table 2.88 – continued from previous page

<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabBar(self, a0)</code>
<code>setTabBarAutoHide(self, enabled)</code>
<code>setTabEnabled(self, index, a1)</code>
<code>setTabIcon(self, index, icon)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabPosition(self, a0)</code>
<code>setTabShape(self, s)</code>
<code>setTabText(self, index, a1)</code>
<code>setTabToolTip(self, index, tip)</code>
<code>setTabVisible(self, index, visible)</code>
<code>setTabWhatsThis(self, index, text)</code>
<code>setTabletTracking(self, enable)</code>
<code>setTabsClosable(self, closeable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setUsesScrollButtons(self, useButtons)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>

continues on next page

Table 2.88 – continued from previous page

<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>

continues on next page

Table 2.88 – continued from previous page

<code>tabBar(self)</code>
<code>tabBarAutoHide(self)</code>
<code>tabIcon(self, index)</code>
<code>tabInserted(self, index)</code>
<code>tabPosition(self)</code>
<code>tabRemoved(self, index)</code>
<code>tabShape(self)</code>
<code>tabText(self, index)</code>
<code>tabToolTip(self, index)</code>
<code>tabWhatsThis(self, index)</code>
<code>tabletEvent(self, a0)</code>
<code>tabsClosable(self)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>

continues on next page

Table 2.88 – continued from previous page

<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>usesScrollButtons(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>widget(self, index)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>
<code>windowModality(self)</code>
<code>windowOpacity(self)</code>
<code>windowRole(self)</code>
<code>windowState(self)</code>
<code>windowTitle(self)</code>
<code>windowType(self)</code>
<code>x(self)</code>
<code>y(self)</code>

Attributes

DrawChildren	
DrawWindowBackground	
East	
IgnoreMask	
North	
PdmDepth	
PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
Rounded	
South	
Triangular	
West	
currentChanged	int) [signal]
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
tabBarClicked	int) [signal]

continues on next page

Table 2.89 – continued from previous page

tabBarDoubleClicked	int) [signal]
tabCloseRequested	int) [signal]
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

labscript_utils.qtwidgets.headerview_with_widgets

Classes

<code>HorizontalHeaderViewWithWidgets(model[, parent])</code>	A QHeaderView that supports inserting arbitrary widgets into sections.
---	--

labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderViewWithWidgets

class `labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderViewWithWidgets`(*model*, *parent=*`None`)

Bases: `PyQt5.QtWidgets.QHeaderView`

A QHeaderView that supports inserting arbitrary widgets into sections. Use `addWidget(logical_index, widget)` to set and `addWidget(logical_index, None)` to unset. Decorations, checkboxes or anything other than text in the headers containing widgets is unsupported, and may result in garbled output

`__init__(model, parent=``None``)`

Methods

<code>__init__(model[, parent])</code>
<code>acceptDrops(self)</code>
<code>accessibleDescription(self)</code>
<code>accessibleName(self)</code>
<code>actionEvent(self, a0)</code>
<code>actions(self)</code>
<code>activateWindow(self)</code>
<code>addAction(self, action)</code>
<code>addActions(self, actions)</code>
<code>addScrollBarWidget(self, widget, alignment)</code>

continues on next page

Table 2.91 – continued from previous page

<code>adjustSize(self)</code>
<code>alternatingRowColors(self)</code>
<code>autoFillBackground(self)</code>
<code>autoScrollMargin(self)</code>
<code>backgroundRole(self)</code>
<code>baseSize(self)</code>
<code>blockSignals(self, b)</code>
<code>cascadingSectionResizes(self)</code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>clearSelection(self)</code>
<code>close(self)</code>
<code>closeEditor(self, editor, hint)</code>
<code>closeEvent(self, a0)</code>
<code>closePersistentEditor(self, index)</code>
<code>colorCount(self)</code>
<code>commitData(self, editor)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>

continues on next page

Table 2.91 – continued from previous page

<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>cornerWidget(self)</code>
<code>count(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>currentChanged(self, current, old)</code>
<code>currentIndex(self)</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>dataChanged(self, topLeft, bottomRight[, roles])</code>
<code>defaultAlignment(self)</code>
<code>defaultDropAction(self)</code>
<code>defaultSectionSize(self)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWin- dows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>dirtyRegionOffset(self)</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>

continues on next page

Table 2.91 – continued from previous page

<code>do_update_widget_positions()</code>	
<code>dragDropMode(self)</code>	
<code>dragDropOverwriteMode(self)</code>	
<code>dragEnabled(self)</code>	
<code>dragEnterEvent(self, e)</code>	
<code>dragLeaveEvent(self, e)</code>	
<code>dragMoveEvent(self, e)</code>	
<code>drawFrame(self, a0)</code>	
<code>dropEvent(self, e)</code>	
<code>dropIndicatorPosition(self)</code>	
<code>dumpObjectInfo(self)</code>	
<code>dumpObjectTree(self)</code>	
<code>dynamicPropertyNames(self)</code>	
<code>edit()</code>	
<code>editTriggers(self)</code>	
<code>editorDestroyed(self, editor)</code>	
<code>effectiveWinId(self)</code>	
<code>ensurePolished(self)</code>	
<code>enterEvent(self, a0)</code>	
<code>event(self, e)</code>	
<code>eventFilter(target, event)</code>	Ensure we don't leave the cursor set as a resize handle when the mouse moves onto a child widget:
<code>executeDelayedItemsLayout(self)</code>	
<code>find(a0)</code>	
<code>findChild(-> QObject)</code>	
<code>findChildren(...)</code>	

continues on next page

Table 2.91 – continued from previous page

<code>focusInEvent(self, e)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, e)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameRect(self)</code>
<code>frameShadow(self)</code>
<code>frameShape(self)</code>
<code>frameSize(self)</code>
<code>frameStyle(self)</code>
<code>frameWidth(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>

continues on next page

Table 2.91 – continued from previous page

<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasAutoScroll(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>headerDataChanged(self, orientation, ...)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hiddenSectionCount(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code><i>hideSection</i>(self, alogicalIndex)</code>
<code>highlightSections(self)</code>
<code>horizontalOffset(self)</code>
<code>horizontalScrollBar(self)</code>
<code>horizontalScrollBarPolicy(self)</code>
<code>horizontalScrollMode(self)</code>
<code>horizontalScrollbarAction(self, action)</code>
<code>horizontalScrollbarValueChanged(self, value)</code>
<code>iconSize(self)</code>
<code>indexAt(self, p)</code>

continues on next page

Table 2.91 – continued from previous page

<code>indexWidget(self, index)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>initialize(self)</code>
<code>initializeSections()</code>
<code>inputMethodEvent(self, event)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, query)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFirstSectionMovable(self)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isIndexHidden(self, index)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isPersistentEditorOpen(self, index)</code>

continues on next page

Table 2.91 – continued from previous page

<code>isRightToLeft(self)</code>
<code>isSectionHidden(self, logicalIndex)</code>
<code>isSignalConnected(self, signal)</code>
<code>isSortIndicatorShown(self)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>itemDelegate(-> QAbstractItemDelegate)</code>
<code>itemDelegateForColumn(self, column)</code>
<code>itemDelegateForRow(self, row)</code>
<code>keyPressEvent(self, e)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>keyboardSearch(self, search)</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>length(self)</code>
<code>lineWidth(self)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>

continues on next page

Table 2.91 – continued from previous page

<code>logicalDpiY(self)</code>
<code>logicalIndex(self, visualIndex)</code>
<code>logicalIndexAt(-> int -> int)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSectionSize(self)</code>
<code>maximumSize(self)</code>
<code>maximumViewportSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>midLineWidth(self)</code>
<code>minimumHeight(self)</code>
<code>minimumSectionSize(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>model(self)</code>

continues on next page

Table 2.91 – continued from previous page

<code>mouseDoubleClickEvent(self, e)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, e)</code>
<code>mousePressEvent(self, e)</code>
<code>mouseReleaseEvent(self, e)</code>
<code>move()</code>
<code>moveCursor(self, a0, a1)</code>
<code>moveEvent(self, a0)</code>
<code>moveSection(self, from_, to)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>offset(self)</code>
<code><i>on_columnsInserted</i>(parent, logical_first, ...)</code>
<code><i>on_columnsRemoved</i>(parent, logical_first, ...)</code>
<code>openPersistentEditor(self, index)</code>
<code>orientation(self)</code>
<code>overrideWindowFlags(self, type)</code>
<code>overrideWindowState(self, state)</code>
<code>paintEngine(self)</code>
<code>paintEvent(self, e)</code>
<code>paintSection(self, painter, rect, logicalIndex)</code>

continues on next page

Table 2.91 – continued from previous page

paintingActive(self)	
palette(self)	
parent(self)	
parentWidget(self)	
physicalDpiX(self)	
physicalDpiY(self)	
pos(self)	
previousInFocusChain(self)	
property(self, name)	
pyqtConfigure(...)	Each keyword argument is either the name of a Qt property or a Qt signal.
raise_(self)	
receivers(self, signal)	
rect(self)	
releaseKeyboard(self)	
releaseMouse(self)	
releaseShortcut(self, id)	
removeAction(self, action)	
removeEventFilter(self, a0)	
render(, sourceRegion, flags, ...)	
repaint(-> None -> None)	
reset(self)	
resetDefaultSectionSize(self)	
resetHorizontalScrollMode(self)	
resetVerticalScrollMode(self)	
resize()	

continues on next page

Table 2.91 – continued from previous page

<code>resizeContentsPrecision(self)</code>
<code>resizeEvent(self, e)</code>
<code>resizeSection(self, logicalIndex, size)</code>
<code>resizeSections()</code>
<code>restoreGeometry(self, geometry)</code>
<code>restoreState(self, state)</code>
<code>rootIndex(self)</code>
<code>rowsAboutToBeRemoved(self, parent, start, end)</code>
<code>rowsInserted(self, parent, start, end)</code>
<code>saveGeometry(self)</code>
<code>saveState(self)</code>
<code>scheduleDelayedItemsLayout(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>scrollBarWidgets(self, alignment)</code>
<code>scrollContentsBy(self, dx, dy)</code>
<code>scrollDirtyRegion(self, dx, dy)</code>
<code>scrollTo(self, index, hint)</code>
<code>scrollToBottom(self)</code>
<code>scrollToTop(self)</code>
<code>sectionPosition(self, logicalIndex)</code>
<code>sectionResizeMode(self, logicalIndex)</code>
<code>sectionSize(self, logicalIndex)</code>
<code><i>sectionSizeFromContents</i>(self, logicalIndex)</code>
<code>sectionSizeHint(self, logicalIndex)</code>

continues on next page

Table 2.91 – continued from previous page

<code>sectionViewportPosition(self, logicalIndex)</code>
<code>sectionsAboutToBeRemoved(self, parent, ...)</code>
<code>sectionsClickable(self)</code>
<code>sectionsHidden(self)</code>
<code>sectionsInserted(self, parent, logicalFirst, ...)</code>
<code>sectionsMovable(self)</code>
<code>sectionsMoved(self)</code>
<code>selectAll(self)</code>
<code>selectedIndexes(self)</code>
<code>selectionBehavior(self)</code>
<code>selectionChanged(self, selected, deselected)</code>
<code>selectionCommand(self, index[, event])</code>
<code>selectionMode(self)</code>
<code>selectionModel(self)</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAlternatingRowColors(self, enable)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setAutoScroll(self, enable)</code>
<code>setAutoScrollMargin(self, margin)</code>
<code>setBackgroundRole(self, a0)</code>

continues on next page

Table 2.91 – continued from previous page

<code>setBaseSize()</code>
<code>setCascadingSectionResizes(self, enable)</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCornerWidget(self, widget)</code>
<code>setCurrentIndex(self, index)</code>
<code>setCursor(self, a0)</code>
<code>setDefaultAlignment(self, alignment)</code>
<code>setDefaultDropAction(self, dropAction)</code>
<code>setDefaultSectionSize(self, size)</code>
<code>setDirtyRegion(self, region)</code>
<code>setDisabled(self, a0)</code>
<code>setDragDropMode(self, behavior)</code>
<code>setDragDropOverwriteMode(self, overwrite)</code>
<code>setDragEnabled(self, enable)</code>
<code>setDropIndicatorShown(self, enable)</code>
<code>setEditTriggers(self, triggers)</code>
<code>setEnabled(self, a0)</code>
<code>setFirstSectionMovable(self, movable)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>

continues on next page

Table 2.91 – continued from previous page

<code>setFont(self, a0)</code>
<code>setForegroundColor(self, a0)</code>
<code>setFrameRect(self, a0)</code>
<code>setFrameShadow(self, a0)</code>
<code>setFrameShape(self, a0)</code>
<code>setFrameStyle(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setHighlightSections(self, highlight)</code>
<code>setHorizontalScrollBar(self, scrollbar)</code>
<code>setHorizontalScrollBarPolicy(self, a0)</code>
<code>setHorizontalScrollMode(self, mode)</code>
<code>setIconSize(self, size)</code>
<code>setIndexWidget(self, index, widget)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setItemDelegate(self, delegate)</code>
<code>setItemDelegateForColumn(self, column, dele- gate)</code>
<code>setItemDelegateForRow(self, row, delegate)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLineWidth(self, a0)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>

continues on next page

Table 2.91 – continued from previous page

<code>setMaximumSectionSize(self, size)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMidLineWidth(self, a0)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSectionSize(self, size)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setModel(self, model)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setOffset(self, offset)</code>
<code>setOffsetToLastSection(self)</code>
<code>setOffsetToSectionPosition(self, visualIndex)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setResizeContentsPrecision(self, precision)</code>
<code>setResizeMode(*args)</code>
<code>setRootIndex(self, index)</code>
<code><i>setSectionHidden</i>(self, logicalIndex, hide)</code>
<code>setSectionResizeMode()</code>
<code>setSectionsClickable(self, clickable)</code>
<code>setSectionsMovable(self, movable)</code>
<code>setSelection(self, rect, flags)</code>

continues on next page

Table 2.91 – continued from previous page

<code>setSelectionBehavior(self, behavior)</code>
<code>setSelectionMode(self, mode)</code>
<code>setSelectionModel(self, selectionModel)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeAdjustPolicy(self, policy)</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setSortIndicator(self, logicalIndex, order)</code>
<code>setSortIndicatorShown(self, show)</code>
<code>setState(self, state)</code>
<code>setStatusTip(self, a0)</code>
<code>setStretchLastSection(self, stretch)</code>
<code>setStyle(self, a0)</code>
<code><i>setStyleSheet</i>(self, styleSheet)</code>
<code>setTabKeyNavigation(self, enable)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setTextElideMode(self, mode)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVerticalScrollBar(self, scrollbar)</code>
<code>setVerticalScrollBarPolicy(self, a0)</code>
<code>setVerticalScrollMode(self, mode)</code>

continues on next page

Table 2.91 – continued from previous page

<code>setViewport(self, widget)</code>
<code>setViewportMargins()</code>
<code>setVisible(self, v)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWidget(logical_index[, widget])</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>setupViewport(self, viewport)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showDropIndicator(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>

continues on next page

Table 2.91 – continued from previous page

<code>showSection(self, logicalIndex)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeAdjustPolicy(self)</code>
<code>sizeHint(self)</code>
<code>sizeHintForColumn(self, column)</code>
<code>sizeHintForIndex(self, index)</code>
<code>sizeHintForRow(self, row)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>sortIndicatorOrder(self)</code>
<code>sortIndicatorSection(self)</code>
<code>stackUnder(self, a0)</code>
<code>startDrag(self, supportedActions)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>state(self)</code>
<code>statusTip(self)</code>
<code>stretchLastSection(self)</code>
<code>stretchSectionCount(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>swapSections(self, first, second)</code>
<code>tabKeyNavigation(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>

continues on next page

Table 2.91 – continued from previous page

<code>textElideMode(self)</code>
<code>thread(self)</code>
<code>timerEvent(self, e)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update()</code>
<code>updateEditorData(self)</code>
<code>updateEditorGeometries(self)</code>
<code>updateGeometries(self)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updateSection(self, logicalIndex)</code>
<code><i>update_indents()</i></code>
<code><i>update_widget_positions()</i></code>
<code>updatesEnabled(self)</code>
<code>verticalOffset(self)</code>
<code>verticalScrollBar(self)</code>
<code>verticalScrollBarPolicy(self)</code>
<code>verticalScrollMode(self)</code>

continues on next page

Table 2.91 – continued from previous page

verticalScrollbarAction(self, action)
verticalScrollbarValueChanged(self, value)
viewOptions(self)
viewport(self)
<i>viewportEvent</i> (self, e)
viewportMargins(self)
viewportSizeHint(self)
visibleRegion(self)
visualIndex(self, logicalIndex)
visualIndexAt(self, position)
visualRect(self, index)
visualRegionForSelection(self, selection)
whatsThis(self)
wheelEvent(self, a0)
width(self)
widthMM(self)
winId(self)
window(self)
windowFilePath(self)
windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)

continues on next page

Table 2.91 – continued from previous page

windowRole(self)	
windowState(self)	
windowTitle(self)	
windowType(self)	
x(self)	
y(self)	
<hr/>	
Attributes	
<hr/>	
AboveItem	
AdjustIgnored	
AdjustToContents	
AdjustToContentsOnFirstShow	
AllEditTriggers	
AnimatingState	
AnyKeyPressed	
BelowItem	
Box	
CollapsingState	
ContiguousSelection	
CurrentChanged	
Custom	
DoubleClicked	
DragDrop	
DragOnly	
DragSelectingState	

continues on next page

Table 2.92 – continued from previous page

DraggingState
DrawChildren
DrawWindowBackground
DropOnly
EditKeyPressed
EditingState
EnsureVisible
ExpandingState
ExtendedSelection
Fixed
HLine
IgnoreMask
Interactive
InternalMove
MoveDown
MoveEnd
MoveHome
MoveLeft
MoveNext
MovePageDown
MovePageUp
MovePrevious
MoveRight
MoveUp
MultiSelection

continues on next page

Table 2.92 – continued from previous page

NoDragDrop
NoEditTriggers
NoFrame
NoSelection
NoState
OnItem
OnViewport
Panel
PdmDepth
PdmDevicePixelRatio
PdmDevicePixelRatioScaled
PdmDpiX
PdmDpiY
PdmHeight
PdmHeightMM
PdmNumColors
PdmPhysicalDpiX
PdmPhysicalDpiY
PdmWidth
PdmWidthMM
Plain
PositionAtBottom
PositionAtCenter
PositionAtTop
Raised

continues on next page

Table 2.92 – continued from previous page

ResizeToContents	
ScrollPerItem	
ScrollPerPixel	
SelectColumns	
SelectItems	
SelectRows	
SelectedClicked	
Shadow_Mask	
Shape_Mask	
SingleSelection	
Stretch	
StyledPanel	
Sunken	
VLine	
WinPanel	
activated	QModelIndex) [signal]
clicked	QModelIndex) [signal]
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
doubleClicked	QModelIndex) [signal]
entered	QModelIndex) [signal]
geometriesChanged	geometriesChanged(self) [signal]
iconSizeChanged	QSize) [signal]
objectNameChanged	str) [signal]
pressed	QModelIndex) [signal]
sectionClicked	int) [signal]
sectionCountChanged	int) [signal]
sectionDoubleClicked	int) [signal]
sectionEntered	int) [signal]
sectionHandleDoubleClicked	int) [signal]
sectionMoved	int, newVisualIndex: int) [signal]
sectionPressed	int) [signal]
sectionResized	int, newSize: int) [signal]
sortIndicatorChanged	Qt.SortOrder) [signal]
staticMetaObject	

continues on next page

Table 2.92 – continued from previous page

<i>stylesheet</i>	
<i>thinspace</i>	
<code>viewportEntered</code>	<code>viewportEntered(self)</code> [signal]
<code>windowIconChanged</code>	<code>QIcon</code> [signal]
<code>windowIconTextChanged</code>	<code>str</code> [signal]
<code>windowTitleChanged</code>	<code>str</code> [signal]

`do_update_widget_positions()`

`eventFilter(target, event)`

Ensure we don't leave the cursor set as a resize handle when the mouse moves onto a child widget:

`hideSection(self, logicalIndex: int)`

`on_columnsInserted(parent, logical_first, logical_last)`

`on_columnsRemoved(parent, logical_first, logical_last)`

`sectionSizeFromContents(self, logicalIndex: int) → QSize`

`setSectionHidden(self, logicalIndex: int, hide: bool)`

`setStyleSheet(self, styleSheet: str)`

`setWidget(logical_index, widget=None)`

`showEvent(self, a0: QShowEvent)`

`showSection(self, logicalIndex: int)`

```
stylesheet = '\n QHeaderView::section {\n /* Will be set dynamically: */\n
padding-top: %dpx;\n padding-bottom: %dpx;\n /* Required, otherwise set to zero
upon setting any stylesheet at all: */\n padding-left: 4px;\n /* Required for some
reason, otherwise other settings ignored: */\n color: black;\n }\n\n /* Any other
style goes here: */\n %s\n '
```

`thinspace = '\u2009'`

`update_indents()`

`update_widget_positions()`

`viewportEvent(self, e: QEvent) → bool`

labscript_utils.qtwidgets.imageoutput

Classes

`BrowseButton(image_output, *args, **kwargs)`

`ImageOutput(name, width, height, *args, **kwargs)`

`ImageView(*args, **kwargs)`

labscript_utils.qtwidgets.imageoutput.BrowseButton

```
class labscript_utils.qtwidgets.imageoutput.BrowseButton(image_output, *args, **kwargs)
    Bases: PyQt5.QtWidgets.QPushButton
    __init__(image_output, *args, **kwargs)
```

Methods

```
__init__(image_output, *args, **kwargs)
```

```
acceptDrops(self)
```

```
accessibleDescription(self)
```

```
accessibleName(self)
```

```
actionEvent(self, a0)
```

```
actions(self)
```

```
activateWindow(self)
```

```
addAction(self, action)
```

```
addActions(self, actions)
```

```
adjustSize(self)
```

```
animateClick(self[, msec])
```

```
autoDefault(self)
```

```
autoExclusive(self)
```

```
autoFillBackground(self)
```

```
autoRepeat(self)
```

```
autoRepeatDelay(self)
```

```
autoRepeatInterval(self)
```

```
backgroundRole(self)
```

```
baseSize(self)
```

```
blockSignals(self, b)
```

```
browse()
```

continues on next page

Table 2.94 – continued from previous page

<code>changeEvent(self, e)</code>
<code>checkStateSet(self)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>click(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>

continues on next page

Table 2.94 – continued from previous page

<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, e)</code>
<code><i>eventFilter</i>(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>

continues on next page

Table 2.94 – continued from previous page

<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameSize(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>group(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>

continues on next page

Table 2.94 – continued from previous page

<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>hitButton(self, pos)</code>
<code>icon(self)</code>
<code>iconSize(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isCheckable(self)</code>
<code>isChecked(self)</code>
<code>isDefault(self)</code>
<code>isDown(self)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFlat(self)</code>

continues on next page

Table 2.94 – continued from previous page

<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, e)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>

continues on next page

Table 2.94 – continued from previous page

<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>menu(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, e)</code>
<code>mousePressEvent(self, e)</code>
<code>mouseReleaseEvent(self, e)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>

continues on next page

Table 2.94 – continued from previous page

<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextCheckState(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	

continues on next page

Table 2.94 – continued from previous page

<code>releaseShortcut(self, id)</code>
<code>removeAction(self, action)</code>
<code>removeEventFilter(self, a0)</code>
<code>render(, sourceRegion, flags, ...)</code>
<code>repaint(-> None -> None)</code>
<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoDefault(self, a0)</code>
<code>setAutoExclusive(self, a0)</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setAutoRepeat(self, a0)</code>
<code>setAutoRepeatDelay(self, a0)</code>
<code>setAutoRepeatInterval(self, a0)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>

continues on next page

Table 2.94 – continued from previous page

<code>setCheckable(self, a0)</code>
<code>setChecked(self, a0)</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDefault(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDown(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFlat(self, a0)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setIcon(self, icon)</code>
<code>setIconSize(self, size)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>

continues on next page

Table 2.94 – continued from previous page

<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMenu(self, menu)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcut(self, key)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>

continues on next page

Table 2.94 – continued from previous page

<code>setText(self, text)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>sharedPainter(self)</code>
<code>shortcut(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMenu(self)</code>
<code>showMinimized(self)</code>

continues on next page

Table 2.94 – continued from previous page

<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>text(self)</code>
<code>thread(self)</code>
<code>timerEvent(self, e)</code>
<code>toggle(self)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>

continues on next page

Table 2.94 – continued from previous page

update(-> None -> None)
updateGeometry(self)
updateMicroFocus(self)
updatesEnabled(self)
visibleRegion(self)
whatsThis(self)
wheelEvent(self, a0)
width(self)
widthMM(self)
winId(self)
window(self)
windowFilePath(self)
windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

DrawChildren	
DrawWindowBackground	
IgnoreMask	
PdmDepth	
PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
clicked	bool = False) [signal]
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
pressed	pressed(self) [signal]
released	released(self) [signal]
staticMetaObject	
toggled	bool) [signal]
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

browse()

eventFilter(self, a0: *QObject*, a1: *QEvent*) → bool

labscript_utils.qtwidgets.imageoutput.ImageOutput

class labscript_utils.qtwidgets.imageoutput.**ImageOutput**(*name, width, height, *args, **kwargs*)

Bases: `PyQt5.QtWidgets.QWidget`

`__init__`(*name, width, height, *args, **kwargs*)

Methods

`__init__`(*name, width, height, *args, **kwargs*)

`acceptDrops`(*self*)

`accessibleDescription`(*self*)

`accessibleName`(*self*)

`actionEvent`(*self, a0*)

`actions`(*self*)

`activateWindow`(*self*)

`addAction`(*self, action*)

`addActions`(*self, actions*)

`adjustSize`(*self*)

`autoFillBackground`(*self*)

`backgroundRole`(*self*)

`baseSize`(*self*)

`blockSignals`(*self, b*)

`changeEvent`(*self, a0*)

`childAt`(*-> QWidget*)

`childEvent`(*self, a0*)

`children`(*self*)

`childrenRect`(*self*)

`childrenRegion`(*self*)

`clearFocus`(*self*)

continues on next page

Table 2.96 – continued from previous page

<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>

continues on next page

Table 2.96 – continued from previous page

<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code><i>eventFilter</i>(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>

continues on next page

Table 2.96 – continued from previous page

frameSize(self)
geometry(self)
getContentsMargins(self)
<i>get_Image()</i>
grab(self[, rectangle])
grabGesture(self, type[, flags])
grabKeyboard(self)
grabMouse()
grabShortcut(self, key[, context])
graphicsEffect(self)
graphicsProxyWidget(self)
hasFocus(self)
hasHeightForWidth(self)
hasMouseTracking(self)
hasTabletTracking(self)
height(self)
heightForWidth(self, a0)
heightMM(self)
hide(self)
hideEvent(self, a0)
inherits(self, classname)
initPainter(self, painter)
inputMethodEvent(self, a0)
inputMethodHints(self)
inputMethodQuery(self, a0)

continues on next page

Table 2.96 – continued from previous page

<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>

continues on next page

Table 2.96 – continued from previous page

layout(self)
layoutDirection(self)
leaveEvent(self, a0)
locale(self)
<i>lock</i> ([notify_Image])
logicalDpiX(self)
logicalDpiY(self)
lower(self)
mapFrom(self, a0, a1)
mapFromGlobal(self, a0)
mapFromParent(self, a0)
mapTo(self, a0, a1)
mapToGlobal(self, a0)
mapToParent(self, a0)
mask(self)
maximumHeight(self)
maximumSize(self)
maximumWidth(self)
metaObject(self)
metric(self, a0)
minimumHeight(self)
minimumSize(self)
minimumSizeHint(self)
minimumWidth(self)
mouseDoubleClickEvent(self, a0)

continues on next page

Table 2.96 – continued from previous page

<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>overrideWindowFlags(self, type)</code>
<code>overrideWindowState(self, state)</code>
<code>paintEngine(self)</code>
<code>paintEvent(self, a0)</code>
<code>paintingActive(self)</code>
<code>palette(self)</code>
<code>parent(self)</code>
<code>parentWidget(self)</code>
<code>physicalDpiX(self)</code>
<code>physicalDpiY(self)</code>
<code>pos(self)</code>
<code>previousInFocusChain(self)</code>
<code>property(self, name)</code>

continues on next page

Table 2.96 – continued from previous page

<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	
<code>resize()</code>	
<code>resizeEvent(self, a0)</code>	
<code>restoreGeometry(self, geometry)</code>	
<code>saveGeometry(self)</code>	
<code>screen(self)</code>	
<code>scroll()</code>	
<code>sender(self)</code>	
<code>senderSignalIndex(self)</code>	
<code>setAcceptDrops(self, on)</code>	
<code>setAccessibleDescription(self, description)</code>	
<code>setAccessibleName(self, name)</code>	
<code>setAttribute(self, attribute[, on])</code>	
<code>setAutoFillBackground(self, enabled)</code>	
<code>setBackgroundRole(self, a0)</code>	

continues on next page

Table 2.96 – continued from previous page

<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>

continues on next page

Table 2.96 – continued from previous page

<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>

continues on next page

Table 2.96 – continued from previous page

<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>set_Image(Image[, notify_old_Image, ...])</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>

continues on next page

Table 2.96 – continued from previous page

<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code><i>unlock</i>([notify_Image])</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>

continues on next page

Table 2.96 – continued from previous page

windowFilePath(self)
windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

DrawChildren
DrawWindowBackground
IgnoreMask
PdmDepth
PdmDevicePixelRatio
PdmDevicePixelRatioScaled
PdmDpiX
PdmDpiY
PdmHeight
PdmHeightMM

continues on next page

Table 2.97 – continued from previous page

PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
<i>imageUpdated</i>	
objectNameChanged	str) [signal]
staticMetaObject	
<i>value</i>	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

eventFilter(*self*, *a0*: *QObject*, *a1*: *QEvent*) → bool

get_Image()

imageUpdated

lock(*notify_Image=True*)

set_Image(*Image*, *notify_old_Image=True*, *notify_new_Image=True*)

unlock(*notify_Image=True*)

property value

labscript_utils.qtwidgets.imageoutput.ImageView

class labscript_utils.qtwidgets.imageoutput.**ImageView**(*args, **kwargs)

Bases: PyQt5.QtWidgets.QGraphicsView

__init__(*args, **kwargs)

Methods

__init__(*args, **kwargs)

acceptDrops(self)

accessibleDescription(self)

continues on next page

Table 2.98 – continued from previous page

<code>accessibleName(self)</code>
<code>actionEvent(self, a0)</code>
<code>actions(self)</code>
<code>activateWindow(self)</code>
<code>addAction(self, action)</code>
<code>addActions(self, actions)</code>
<code>addScrollBarWidget(self, widget, alignment)</code>
<code>adjustSize(self)</code>
<code>alignment(self)</code>
<code>autoFillBackground(self)</code>
<code>backgroundBrush(self)</code>
<code>backgroundRole(self)</code>
<code>baseSize(self)</code>
<code>blockSignals(self, b)</code>
<code>cacheMode(self)</code>
<code>centerOn(-> None)</code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>

continues on next page

Table 2.98 – continued from previous page

<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, event)</code>
<code>contextMenuPolicy(self)</code>
<code>cornerWidget(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, event)</code>
<code>dragLeaveEvent(self, event)</code>
<code>dragMode(self)</code>
<code>dragMoveEvent(self, event)</code>

continues on next page

Table 2.98 – continued from previous page

<code>drawBackground(self, painter, rect)</code>
<code>drawForeground(self, painter, rect)</code>
<code>drawFrame(self, a0)</code>
<code>dropEvent(self, event)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>ensureVisible(-> None)</code>
<code>enterEvent(self, a0)</code>
<code>event(self, event)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>fitInView(-> None)</code>
<code>focusInEvent(self, event)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, event)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>

continues on next page

Table 2.98 – continued from previous page

font(self)
fontInfo(self)
fontMetrics(self)
foregroundBrush(self)
foregroundRole(self)
frameGeometry(self)
frameRect(self)
frameShadow(self)
frameShape(self)
frameSize(self)
frameStyle(self)
frameWidth(self)
geometry(self)
getContentsMargins(self)
grab(self[, rectangle])
grabGesture(self, type[, flags])
grabKeyboard(self)
grabMouse()
grabShortcut(self, key[, context])
graphicsEffect(self)
graphicsProxyWidget(self)
hasFocus(self)
hasHeightForWidth(self)
hasMouseTracking(self)
hasTabletTracking(self)

continues on next page

Table 2.98 – continued from previous page

<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>horizontalScrollBar(self)</code>
<code>horizontalScrollBarPolicy(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, event)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, query)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code>installEventFilter(self, a0)</code>
<code>invalidateScene(self[, rect, layers])</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isInteractive(self)</code>
<code>isLeftToRight(self)</code>

continues on next page

Table 2.98 – continued from previous page

<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isTransformed(self)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>itemAt(-> QGraphicsItem)</code>
<code>items(...)</code>
<code>keyPressEvent(self, event)</code>
<code>keyReleaseEvent(self, event)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>lineWidth(self)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>

continues on next page

Table 2.98 – continued from previous page

<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapFromScene(...)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mapToScene(...)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumViewportSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>midLineWidth(self)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, event)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, event)</code>
<code>mousePressEvent(self, event)</code>

continues on next page

Table 2.98 – continued from previous page

<code>mouseReleaseEvent(self, event)</code>	
<code>move()</code>	
<code>moveEvent(self, a0)</code>	
<code>moveToThread(self, thread)</code>	
<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>optimizationFlags(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code>paintEvent(self, event)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	

continues on next page

Table 2.98 – continued from previous page

<code>receivers(self, signal)</code>
<code>rect(self)</code>
<code>releaseKeyboard(self)</code>
<code>releaseMouse(self)</code>
<code>releaseShortcut(self, id)</code>
<code>removeAction(self, action)</code>
<code>removeEventFilter(self, a0)</code>
<code>render(self, painter[, target, source, mode])</code>
<code>renderHints(self)</code>
<code>repaint(-> None -> None)</code>
<code>resetCachedContent(self)</code>
<code>resetTransform(self)</code>
<code>resize()</code>
<code>resizeAnchor(self)</code>
<code>resizeEvent(self, event)</code>
<code>restoreGeometry(self, geometry)</code>
<code>rotate(self, angle)</code>
<code>rubberBandRect(self)</code>
<code>rubberBandSelectionMode(self)</code>
<code>saveGeometry(self)</code>
<code>scale(self, sx, sy)</code>
<code>scene(self)</code>
<code>sceneRect(self)</code>
<code>screen(self)</code>
<code>scroll()</code>

continues on next page

Table 2.98 – continued from previous page

<code>scrollBarWidgets(self, alignment)</code>
<code>scrollContentsBy(self, dx, dy)</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAlignment(self, alignment)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundBrush(self, brush)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setCacheMode(self, mode)</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCornerWidget(self, widget)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setDragMode(self, mode)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>

continues on next page

Table 2.98 – continued from previous page

<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundBrush(self, brush)</code>
<code>setForegroundRole(self, a0)</code>
<code>setFrameRect(self, a0)</code>
<code>setFrameShadow(self, a0)</code>
<code>setFrameShape(self, a0)</code>
<code>setFrameStyle(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setHorizontalScrollBar(self, scrollbar)</code>
<code>setHorizontalScrollBarPolicy(self, a0)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setInteractive(self, allowed)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLineWidth(self, a0)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMidLineWidth(self, a0)</code>

continues on next page

Table 2.98 – continued from previous page

<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setOptimizationFlag(self, flag[, enabled])</code>
<code>setOptimizationFlags(self, flags)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setRenderHint(self, hint[, on])</code>
<code>setRenderHints(self, hints)</code>
<code>setResizeAnchor(self, anchor)</code>
<code>setRubberBandSelectionMode(self, mode)</code>
<code>setScene(self, scene)</code>
<code>setSceneRect()</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeAdjustPolicy(self, policy)</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>

continues on next page

Table 2.98 – continued from previous page

<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setTransform(self, matrix[, combine])</code>
<code>setTransformationAnchor(self, anchor)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVerticalScrollBar(self, scrollbar)</code>
<code>setVerticalScrollBarPolicy(self, a0)</code>
<code>setViewport(self, widget)</code>
<code>setViewportMargins()</code>
<code>setViewportUpdateMode(self, mode)</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>setupViewport(self, widget)</code>

continues on next page

Table 2.98 – continued from previous page

<code>sharedPainter(self)</code>
<code>shear(self, sh, sv)</code>
<code>show(self)</code>
<code>showEvent(self, event)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeAdjustPolicy(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>

continues on next page

Table 2.98 – continued from previous page

<code>tr(self, sourceText[, disambiguation, n])</code>
<code>transform(self)</code>
<code>transformationAnchor(self)</code>
<code>translate(self, dx, dy)</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code>updateScene(self, rects)</code>
<code>updateSceneRect(self, rect)</code>
<code>updatesEnabled(self)</code>
<code>verticalScrollBar(self)</code>
<code>verticalScrollBarPolicy(self)</code>
<code>viewport(self)</code>
<code>viewportEvent(self, event)</code>
<code>viewportMargins(self)</code>
<code>viewportSizeHint(self)</code>
<code>viewportTransform(self)</code>
<code>viewportUpdateMode(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>

continues on next page

Table 2.98 – continued from previous page

wheelEvent(self, event)
width(self)
widthMM(self)
winId(self)
window(self)
windowFilePath(self)
windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

AdjustIgnored
AdjustToContents
AdjustToContentsOnFirstShow
AnchorUnderMouse
AnchorViewCenter

continues on next page

Table 2.99 – continued from previous page

BoundingRectViewportUpdate
Box
CacheBackground
CacheNone
DontAdjustForAntialiasing
DontClipPainter
DontSavePainterState
DrawChildren
DrawWindowBackground
FullViewportUpdate
HLine
IgnoreMask
MinimalViewportUpdate
NoAnchor
NoDrag
NoFrame
NoViewportUpdate
Panel
PdmDepth
PdmDevicePixelRatio
PdmDevicePixelRatioScaled
PdmDpiX
PdmDpiY
PdmHeight
PdmHeightMM

continues on next page

Table 2.99 – continued from previous page

PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
Plain	
Raised	
RubberBandDrag	
ScrollHandDrag	
Shadow_Mask	
Shape_Mask	
SmartViewportUpdate	
StyledPanel	
Sunken	
VLine	
WinPanel	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
rubberBandChanged	Union[QPointF, QPoint], toScenePoint: Union[QPointF, QPoint]) [signal]
staticMetaObject	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

contextMenuEvent (*self*, *event*: *QContextMenuEvent*)

labscript_utils.qtwidgets.outputbox**Classes**

<code>OutputBox(container[, scrollbar_lines])</code>	A subclass of <code>qtutils.outputbox.OutputBox</code> configured with security from <code>labconfig</code> .
--	---

labscript_utils.qtwidgets.outputbox.OutputBox

class `labscript_utils.qtwidgets.outputbox.OutputBox`(*container*, *scrollback_lines=1000*)

Bases: `qtutils.outputbox.OutputBox`

A subclass of `qtutils.outputbox.OutputBox` configured with security from `labconfig`.

__init__(*container*, *scrollback_lines=1000*)

Instantiate an `outputBox` and insert into `container` widget. Set the number of lines of scrollback to keep. Set a `zmq_context` for creating sockets, otherwise `zmq.Context.instance()` will be used. set `bind_address`, defaulting to the local interface.

Methods

<code>__init__(container[, scrollbar_lines])</code>	Instantiate an <code>outputBox</code> and insert into <code>container</code> widget.
<code>add_text()</code>	
<code>close()</code>	
<code>fileno()</code>	
<code>flush()</code>	
<code>isatty()</code>	
<code>mainloop(socket)</code>	
<code>new_socket()</code>	
<code>output(text[, red])</code>	Wrapper around <code>write()</code> with only option for normal text or bold red text, retained for backward compatibility but deprecated.
<code>print(*values, **kwargs)</code>	Print to the output box.
<code>shutdown()</code>	Stop the <code>mainloop</code> .
<code>write(text[, color, bold, italic, charformat])</code>	Write to the output box as if it were a file.

Attributes

LINE_MID

LINE_NEW

LINE_START

MAX_LINES_BATCH

supports_rich_write

labscript_utils.qtwidgets.toolpalette

Classes

ToolPalette(parent, name, *args, **kwargs)

ToolPaletteGroup(*args, **kwargs)

labscript_utils.qtwidgets.toolpalette.ToolPalette

class labscript_utils.qtwidgets.toolpalette.**ToolPalette**(parent, name, *args, **kwargs)

Bases: `PyQt5.QtWidgets.QScrollArea`

`__init__`(parent, name, *args, **kwargs)

Methods

`__init__`(parent, name, *args, **kwargs)

`acceptDrops`(self)

`accessibleDescription`(self)

`accessibleName`(self)

`actionEvent`(self, a0)

`actions`(self)

`activateWindow`(self)

`addAction`(self, action)

continues on next page

Table 2.104 – continued from previous page

<code>addAction(self, actions)</code>
<code>addScrollBarWidget(self, widget, alignment)</code>
<code>addWidget(widget[, force_relayout])</code>
<code>adjustSize(self)</code>
<code>alignment(self)</code>
<code>autoFillBackground(self)</code>
<code>backgroundRole(self)</code>
<code>baseSize(self)</code>
<code>blockSignals(self, b)</code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>
<code>contextMenuPolicy(self)</code>

continues on next page

Table 2.104 – continued from previous page

<code>cornerWidget(self)</code>
<code>create(self[, window, initializeWindow, ...])</code>
<code>createWindowContainer(window[, parent, flags])</code>
<code>cursor(self)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>depth(self)</code>
<code>destroy(self[, destroyWindow, destroySubWindows])</code>
<code>devType(self)</code>
<code>devicePixelRatio(self)</code>
<code>devicePixelRatioF(self)</code>
<code>devicePixelRatioFScale()</code>
<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dragEnterEvent(self, a0)</code>
<code>dragLeaveEvent(self, a0)</code>
<code>dragMoveEvent(self, a0)</code>
<code>drawFrame(self, a0)</code>
<code>dropEvent(self, a0)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>effectiveWinId(self)</code>
<code>ensurePolished(self)</code>
<code>ensureVisible(self, x, y[, xMargin, yMargin])</code>

continues on next page

Table 2.104 – continued from previous page

<code>ensureWidgetVisible(self, childWidget[, ...])</code>
<code>enterEvent(self, a0)</code>
<code>event(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameRect(self)</code>
<code>frameShadow(self)</code>
<code>frameShape(self)</code>
<code>frameSize(self)</code>
<code>frameStyle(self)</code>

continues on next page

Table 2.104 – continued from previous page

<code>frameWidth(self)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>horizontalScrollBar(self)</code>
<code>horizontalScrollBarPolicy(self)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>

continues on next page

Table 2.104 – continued from previous page

<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>
<code><i>insertWidget</i>(index, widget[, force_relayout])</code>
<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>

continues on next page

Table 2.104 – continued from previous page

<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>
<code>leaveEvent(self, a0)</code>
<code>lineWidth(self)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumViewportSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>midLineWidth(self)</code>

continues on next page

Table 2.104 – continued from previous page

<code>minimumHeight(self)</code>
<code><i>minimumSize</i>(self)</code>
<code><i>minimumSizeHint</i>(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>
<code>mouseMoveEvent(self, a0)</code>
<code>mousePressEvent(self, a0)</code>
<code>mouseReleaseEvent(self, a0)</code>
<code>move()</code>
<code>moveEvent(self, a0)</code>
<code>moveToThread(self, thread)</code>
<code>nativeEvent(self, eventType, message)</code>
<code>nativeParentWidget(self)</code>
<code>nextInFocusChain(self)</code>
<code>normalGeometry(self)</code>
<code>objectName(self)</code>
<code>overrideWindowFlags(self, type)</code>
<code>overrideWindowState(self, state)</code>
<code>paintEngine(self)</code>
<code>paintEvent(self, a0)</code>
<code>paintingActive(self)</code>
<code>palette(self)</code>
<code>parent(self)</code>
<code>parentWidget(self)</code>

continues on next page

Table 2.104 – continued from previous page

<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.
<code>raise_(self)</code>	
<code>receivers(self, signal)</code>	
<code>rect(self)</code>	
<code>releaseKeyboard(self)</code>	
<code>releaseMouse(self)</code>	
<code>releaseShortcut(self, id)</code>	
<code>removeAction(self, action)</code>	
<code>removeEventFilter(self, a0)</code>	
<code>render(, sourceRegion, flags, ...)</code>	
<code>repaint(-> None -> None)</code>	
<code>resize()</code>	
<code><i>resizeEvent</i>(self, a0)</code>	
<code>restoreGeometry(self, geometry)</code>	
<code>saveGeometry(self)</code>	
<code>screen(self)</code>	
<code>scroll()</code>	
<code>scrollBarWidgets(self, alignment)</code>	
<code>scrollContentsBy(self, dx, dy)</code>	
<code>sender(self)</code>	

continues on next page

Table 2.104 – continued from previous page

<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAlignment(self, a0)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>
<code>setContentsMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCornerWidget(self, widget)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setFrameRect(self, a0)</code>
<code>setFrameShadow(self, a0)</code>

continues on next page

Table 2.104 – continued from previous page

<code>setFrameShape(self, a0)</code>
<code>setFrameStyle(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setHorizontalScrollBar(self, scrollbar)</code>
<code>setHorizontalScrollBarPolicy(self, a0)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLineWidth(self, a0)</code>
<code>setLocale(self, locale)</code>
<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMidLineWidth(self, a0)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>

continues on next page

Table 2.104 – continued from previous page

<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeAdjustPolicy(self, policy)</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>
<code>setVerticalScrollBar(self, scrollbar)</code>
<code>setVerticalScrollBarPolicy(self, a0)</code>
<code>setViewport(self, widget)</code>
<code>setViewportMargins()</code>
<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWidget(self, w)</code>
<code>setWidgetResizable(self, resizable)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>

continues on next page

Table 2.104 – continued from previous page

<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>setupViewport(self, viewport)</code>
<code>sharedPainter(self)</code>
<code>show(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeAdjustPolicy(self)</code>
<code><i>sizeHint</i>(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>
<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>

continues on next page

Table 2.104 – continued from previous page

<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>takeWidget(self)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code><i>updateMinimumSize()</i></code>
<code>updatesEnabled(self)</code>
<code>verticalScrollBar(self)</code>
<code>verticalScrollBarPolicy(self)</code>
<code>viewport(self)</code>
<code>viewportEvent(self, a0)</code>
<code>viewportMargins(self)</code>
<code>viewportSizeHint(self)</code>

continues on next page

Table 2.104 – continued from previous page

<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>
<code>widget(self)</code>
<code>widgetResizable(self)</code>
<code>width(self)</code>
<code>widthMM(self)</code>
<code>winId(self)</code>
<code>window(self)</code>
<code>windowFilePath(self)</code>
<code>windowFlags(self)</code>
<code>windowHandle(self)</code>
<code>windowIcon(self)</code>
<code>windowIconText(self)</code>
<code>windowModality(self)</code>
<code>windowOpacity(self)</code>
<code>windowRole(self)</code>
<code>windowState(self)</code>
<code>windowTitle(self)</code>
<code>windowType(self)</code>
<code>x(self)</code>
<code>y(self)</code>

Attributes

AdjustIgnored

AdjustToContents

AdjustToContentsOnFirstShow

Box

DrawChildren

DrawWindowBackground

HLine

IgnoreMask

NoFrame

Panel

PdmDepth

PdmDevicePixelRatio

PdmDevicePixelRatioScaled

PdmDpiX

PdmDpiY

PdmHeight

PdmHeightMM

PdmNumColors

PdmPhysicalDpiX

PdmPhysicalDpiY

PdmWidth

PdmWidthMM

Plain

Raised

continues on next page

Table 2.105 – continued from previous page

Shadow_Mask	
Shape_Mask	
StyledPanel	
Sunken	
VLine	
WinPanel	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
objectNameChanged	str) [signal]
staticMetaObject	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

addWidget(*widget*, *force_relayout=True*)
event(*self*, *a0: QEvent*) → bool
insertWidget(*index*, *widget*, *force_relayout=True*)
minimumSize(*self*) → QSize
minimumSizeHint(*self*) → QSize
resizeEvent(*self*, *a0: QResizeEvent*)
sizeHint(*self*) → QSize
updateMinimumSize()

labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup

class labscript_utils.qtwidgets.toolpalette.**ToolPaletteGroup**(*args, **kwargs)
 Bases: PyQt5.QtWidgets.QVBoxLayout
__init__(*args, **kwargs)

Methods

__init__(*args, **kwargs)
activate(self)
addChildLayout(self, l)

continues on next page

Table 2.106 – continued from previous page

<code>addChildWidget(self, w)</code>
<code>addItem(self, a0)</code>
<code>addLayout(self, layout[, stretch])</code>
<code>addSpacerItem(self, spacerItem)</code>
<code>addSpacing(self, size)</code>
<code>addStretch(self[, stretch])</code>
<code>addStrut(self, a0)</code>
<code>addWidget(self, a0[, stretch, alignment])</code>
<code><i>add_to_linked_width_group</i>(width_group_name, name)</code>
<code>alignment(self)</code>
<code>alignmentRect(self, a0)</code>
<code><i>append_new_palette</i>(name, *args, **kwargs)</code>
<code>blockSignals(self, b)</code>
<code>childEvent(self, e)</code>
<code>children(self)</code>
<code>closestAcceptableSize(w, s)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>controlTypes(self)</code>
<code>count(self)</code>
<code><i>create_linked_width_group</i>(width_group_name, ...)</code>
<code>customEvent(self, a0)</code>
<code>deleteLater(self)</code>
<code>direction(self)</code>

continues on next page

Table 2.106 – continued from previous page

<code>disconnect(-> bool)</code>
<code>disconnectNotify(self, signal)</code>
<code>dumpObjectInfo(self)</code>
<code>dumpObjectTree(self)</code>
<code>dynamicPropertyNames(self)</code>
<code>event(self, a0)</code>
<code>eventFilter(self, a0, a1)</code>
<code>expandingDirections(self)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>geometry(self)</code>
<code>getContentsMargins(self)</code>
<code><i>get_index_from_name</i>(name)</code>
<code><i>get_name_from_index</i>(index)</code>
<code><i>get_palette</i>(name)</code>
<code><i>get_palette_by_index</i>(index)</code>
<code>hasHeightForWidth(self)</code>
<code><i>has_palette</i>(name)</code>
<code>heightForWidth(self, a0)</code>
<code><i>hide_palette</i>(name)</code>
<code><i>hide_palette_by_index</i>(index)</code>
<code>indexOf(-> int)</code>
<code>inherits(self, classname)</code>
<code>insertItem(self, index, a1)</code>
<code>insertLayout(self, index, layout[, stretch])</code>

continues on next page

Table 2.106 – continued from previous page

<code>insertSpacerItem(self, index, spacerItem)</code>
<code>insertSpacing(self, index, size)</code>
<code>insertStretch(self, index[, stretch])</code>
<code>insertWidget(self, index, widget[, stretch, ...])</code>
<code><i>insert_new_palette</i>(index, name, *args, **kwargs)</code>
<code>installEventFilter(self, a0)</code>
<code>invalidate(self)</code>
<code>isEmpty(self)</code>
<code>isEnabled(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isWidgetType(self)</code>
<code>isWindowType(self)</code>
<code>itemAt(self, a0)</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>maximumSize(self)</code>
<code>menuBar(self)</code>
<code>metaObject(self)</code>
<code>minimumHeightForWidth(self, a0)</code>
<code>minimumSize(self)</code>
<code>moveToThread(self, thread)</code>
<code>objectName(self)</code>
<code>parent(self)</code>
<code>parentWidget(self)</code>
<code>property(self, name)</code>

continues on next page

Table 2.106 – continued from previous page

pyqtConfigure(...)	Each keyword argument is either the name of a Qt property or a Qt signal.
receivers(self, signal)	
<i>remove</i> (name)	
removeEventFilter(self, a0)	
removeItem(self, a0)	
removeWidget(self, w)	
<i>remove_by_index</i> (index)	
<i>remove_from_linked_width_group</i> (...)	
<i>reorder_palette</i> (name, new_index)	
<i>reorder_palette_by_index</i> (old_index, new_index)	
replaceWidget(self, from_, to[, options])	
sender(self)	
senderSignalIndex(self)	
setAlignment(-> bool -> bool)	
setContentMargins()	
setDirection(self, a0)	
setEnabled(self, a0)	
setGeometry(self, a0)	
setMenuBar(self, w)	
setObjectName(self, name)	
setParent(self, a0)	
setProperty(self, name, value)	
setSizeConstraint(self, a0)	
setSpacing(self, spacing)	
setStretch(self, index, stretch)	

continues on next page

Table 2.106 – continued from previous page

<code>setStretchFactor(-> bool)</code>
<code><i>show_palette</i>(name)</code>
<code><i>show_palette_by_index</i>(index)</code>
<code>signalsBlocked(self)</code>
<code>sizeConstraint(self)</code>
<code>sizeHint(self)</code>
<code>spacerItem(self)</code>
<code>spacing(self)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>stretch(self, index)</code>
<code>takeAt(self, a0)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>totalHeightForWidth(self, w)</code>
<code>totalMaximumSize(self)</code>
<code>totalMinimumSize(self)</code>
<code>totalSizeHint(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>update(self)</code>
<code>widget(self)</code>
<code>widgetEvent(self, a0)</code>

Attributes

BottomToTop	
Down	
LeftToRight	
RightToLeft	
SetDefaultConstraint	
SetFixedSize	
SetMaximumSize	
SetMinAndMaxSize	
SetMinimumSize	
SetNoConstraint	
TopToBottom	
Up	
destroyed	typing.Optional[QObject] = None [signal]
objectNameChanged	str) [signal]
staticMetaObject	
<i>widths_linked</i>	

```

add_to_linked_width_group(width_group_name, name)
append_new_palette(name, *args, **kwargs)
create_linked_width_group(width_group_name, names)
get_index_from_name(name)
get_name_from_index(index)
get_palette(name)
get_palette_by_index(index)
has_palette(name)
hide_palette(name)
hide_palette_by_index(index)
insert_new_palette(index, name, *args, **kwargs)
remove(name)
remove_by_index(index)

```

```

remove_from_linked_width_group(width_group_name, name)
reorder_palette(name, new_index)
reorder_palette_by_index(old_index, new_index)
show_palette(name)
show_palette_by_index(index)
property widths_linked

```

2.1.16 labscript_utils.remote

Script to run a zprocess.remote server configured according to LabConfig. Run with:

```
python -m labscript_utils.remote [--daemon] [--no-tui]
```

If `--daemon` is specified, the server will be started in the background. If `--no-tui` is specified, the server will run with ordinary terminal output instead of with the interactive text-based user interface (TUI).

Functions

```
main()
```

labscript_utils.remote.main

```
labscript_utils.remote.main()
```

2.1.17 labscript_utils.settings

Classes

```
Settings([storage, file, parent, page_classes])
```

labscript_utils.settings.Settings

```

class labscript_utils.settings.Settings(storage='hdf5', file=None, parent=None, page_classes=[])
    Bases: object
    __init__(storage='hdf5', file=None, parent=None, page_classes=[])

```

Methods

`__init__`([storage, file, parent, page_classes])

`add_settings_interface`(setting_class)

`close`(*args, **kwargs)

`create_dialog`([goto_page])

`get_value`(settings_class, value_name)

`load`(name)

`on_cancel`(*args, **kwargs)

`on_save`(*args, **kwargs)

`register_callback`(callback)

`remove_callback`(callback)

add_settings_interface(*setting_class*)

close(**args, **kwargs*)

create_dialog(*goto_page=None*)

get_value(*settings_class, value_name*)

load(*name*)

on_cancel(**args, **kwargs*)

on_save(**args, **kwargs*)

register_callback(*callback*)

remove_callback(*callback*)

2.1.18 labscript_utils.setup_logging

Functions

`setup_logging`(program_name[, log_level, ...])

labscript_utils.setup_logging.setup_logging

labscript_utils.setup_logging.**setup_logging**(*program_name*, *log_level=10*, *terminal_level=20*,
maxBytes=52428800, *backupCount=1*)

Classes

LessThanFilter(*less_than*)

labscript_utils.setup_logging.LessThanFilter

class labscript_utils.setup_logging.**LessThanFilter**(*less_than*)

Bases: `logging.Filter`

__init__(*less_than*)

Initialize a filter.

Initialize with the name of the logger which, together with its children, will have its events allowed through the filter. If no name is specified, allow every event.

Methods

__init__(*less_than*)

Initialize a filter.

filter(*record*)

Determine if the specified record is to be logged.

filter(*record*)

Determine if the specified record is to be logged.

Is the specified record to be logged? Returns 0 for no, nonzero for yes. If deemed appropriate, the record may be modified in-place.

2.1.19 labscript_utils.shared_drive**Functions**

path_to_agnostic(*path*)

path_to_local(*path*)

labscript_utils.shared_drive.path_to_agnostic

labscript_utils.shared_drive.path_to_agnostic(*path*)

labscript_utils.shared_drive.path_to_local

labscript_utils.shared_drive.path_to_local(*path*)

2.1.20 labscript_utils.shot_utils

Functions

<code>get_shot_globals(filepath)</code>	Returns the evaluated globals for a shot, for use by labscript or lyse.
---	---

labscript_utils.shot_utils.get_shot_globals

labscript_utils.shot_utils.get_shot_globals(*filepath*)

Returns the evaluated globals for a shot, for use by labscript or lyse. Simple dictionary access as in `dict(h5py.File(filepath).attrs)` would be fine except we want to apply some hacks, so it's best to do that in one place.

2.1.21 labscript_utils.splash

Classes

Splash(*imagepath*)

labscript_utils.splash.Splash

class labscript_utils.splash.Splash(*imagepath*)

Bases: `PyQt5.QtWidgets.QFrame`

`__init__`(*imagepath*)

Methods

`__init__`(*imagepath*)

`acceptDrops`(*self*)

`accessibleDescription`(*self*)

`accessibleName`(*self*)

continues on next page

Table 2.117 – continued from previous page

<code>actionEvent(self, a0)</code>
<code>actions(self)</code>
<code>activateWindow(self)</code>
<code>addAction(self, action)</code>
<code>addActions(self, actions)</code>
<code>adjustSize(self)</code>
<code>autoFillBackground(self)</code>
<code>backgroundRole(self)</code>
<code>baseSize(self)</code>
<code>blockSignals(self, b)</code>
<code>changeEvent(self, a0)</code>
<code>childAt(-> QWidget)</code>
<code>childEvent(self, a0)</code>
<code>children(self)</code>
<code>childrenRect(self)</code>
<code>childrenRegion(self)</code>
<code>clearFocus(self)</code>
<code>clearMask(self)</code>
<code>close(self)</code>
<code>closeEvent(self, a0)</code>
<code>colorCount(self)</code>
<code>connectNotify(self, signal)</code>
<code>contentsMargins(self)</code>
<code>contentsRect(self)</code>
<code>contextMenuEvent(self, a0)</code>

continues on next page

Table 2.117 – continued from previous page

contextMenuPolicy(self)
create(self[, window, initializeWindow, ...])
createWindowContainer(window[, parent, flags])
cursor(self)
customEvent(self, a0)
deleteLater(self)
depth(self)
destroy(self[, destroyWindow, destroySubWindows])
devType(self)
devicePixelRatio(self)
devicePixelRatioF(self)
devicePixelRatioFScale()
disconnect(-> bool)
disconnectNotify(self, signal)
dragEnterEvent(self, a0)
dragLeaveEvent(self, a0)
dragMoveEvent(self, a0)
drawFrame(self, a0)
dropEvent(self, a0)
dumpObjectInfo(self)
dumpObjectTree(self)
dynamicPropertyNames(self)
effectiveWinId(self)
ensurePolished(self)
enterEvent(self, a0)

continues on next page

Table 2.117 – continued from previous page

<code>event(self, e)</code>
<code>eventFilter(self, a0, a1)</code>
<code>find(a0)</code>
<code>findChild(-> QObject)</code>
<code>findChildren(...)</code>
<code>focusInEvent(self, a0)</code>
<code>focusNextChild(self)</code>
<code>focusNextPrevChild(self, next)</code>
<code>focusOutEvent(self, a0)</code>
<code>focusPolicy(self)</code>
<code>focusPreviousChild(self)</code>
<code>focusProxy(self)</code>
<code>focusWidget(self)</code>
<code>font(self)</code>
<code>fontInfo(self)</code>
<code>fontMetrics(self)</code>
<code>foregroundRole(self)</code>
<code>frameGeometry(self)</code>
<code>frameRect(self)</code>
<code>frameShadow(self)</code>
<code>frameShape(self)</code>
<code>frameSize(self)</code>
<code>frameStyle(self)</code>
<code>frameWidth(self)</code>
<code>geometry(self)</code>

continues on next page

Table 2.117 – continued from previous page

<code>getContentsMargins(self)</code>
<code>grab(self[, rectangle])</code>
<code>grabGesture(self, type[, flags])</code>
<code>grabKeyboard(self)</code>
<code>grabMouse()</code>
<code>grabShortcut(self, key[, context])</code>
<code>graphicsEffect(self)</code>
<code>graphicsProxyWidget(self)</code>
<code>hasFocus(self)</code>
<code>hasHeightForWidth(self)</code>
<code>hasMouseTracking(self)</code>
<code>hasTabletTracking(self)</code>
<code>height(self)</code>
<code>heightForWidth(self, a0)</code>
<code>heightMM(self)</code>
<code>hide(self)</code>
<code>hideEvent(self, a0)</code>
<code>inherits(self, classname)</code>
<code>initPainter(self, painter)</code>
<code>initStyleOption(self, option)</code>
<code>inputMethodEvent(self, a0)</code>
<code>inputMethodHints(self)</code>
<code>inputMethodQuery(self, a0)</code>
<code>insertAction(self, before, action)</code>
<code>insertActions(self, before, actions)</code>

continues on next page

Table 2.117 – continued from previous page

<code>installEventFilter(self, a0)</code>
<code>isActiveWindow(self)</code>
<code>isAncestorOf(self, child)</code>
<code>isEnabled(self)</code>
<code>isEnabledTo(self, a0)</code>
<code>isFullScreen(self)</code>
<code>isHidden(self)</code>
<code>isLeftToRight(self)</code>
<code>isMaximized(self)</code>
<code>isMinimized(self)</code>
<code>isModal(self)</code>
<code>isRightToLeft(self)</code>
<code>isSignalConnected(self, signal)</code>
<code>isVisible(self)</code>
<code>isVisibleTo(self, a0)</code>
<code>isWidgetType(self)</code>
<code>isWindow(self)</code>
<code>isWindowModified(self)</code>
<code>isWindowType(self)</code>
<code>keyPressEvent(self, a0)</code>
<code>keyReleaseEvent(self, a0)</code>
<code>keyboardGrabber()</code>
<code>killTimer(self, id)</code>
<code>layout(self)</code>
<code>layoutDirection(self)</code>

continues on next page

Table 2.117 – continued from previous page

<code>leaveEvent(self, a0)</code>
<code>lineWidth(self)</code>
<code>locale(self)</code>
<code>logicalDpiX(self)</code>
<code>logicalDpiY(self)</code>
<code>lower(self)</code>
<code>mapFrom(self, a0, a1)</code>
<code>mapFromGlobal(self, a0)</code>
<code>mapFromParent(self, a0)</code>
<code>mapTo(self, a0, a1)</code>
<code>mapToGlobal(self, a0)</code>
<code>mapToParent(self, a0)</code>
<code>mask(self)</code>
<code>maximumHeight(self)</code>
<code>maximumSize(self)</code>
<code>maximumWidth(self)</code>
<code>metaObject(self)</code>
<code>metric(self, a0)</code>
<code>midLineWidth(self)</code>
<code>minimumHeight(self)</code>
<code>minimumSize(self)</code>
<code>minimumSizeHint(self)</code>
<code>minimumWidth(self)</code>
<code>mouseDoubleClickEvent(self, a0)</code>
<code>mouseGrabber()</code>

continues on next page

Table 2.117 – continued from previous page

<code>mouseMoveEvent(self, a0)</code>	
<code>mousePressEvent(self, a0)</code>	
<code>mouseReleaseEvent(self, a0)</code>	
<code>move()</code>	
<code>moveEvent(self, a0)</code>	
<code>moveToThread(self, thread)</code>	
<code>nativeEvent(self, eventType, message)</code>	
<code>nativeParentWidget(self)</code>	
<code>nextInFocusChain(self)</code>	
<code>normalGeometry(self)</code>	
<code>objectName(self)</code>	
<code>overrideWindowFlags(self, type)</code>	
<code>overrideWindowState(self, state)</code>	
<code>paintEngine(self)</code>	
<code><i>paintEvent</i>(self, a0)</code>	
<code>paintingActive(self)</code>	
<code>palette(self)</code>	
<code>parent(self)</code>	
<code>parentWidget(self)</code>	
<code>physicalDpiX(self)</code>	
<code>physicalDpiY(self)</code>	
<code>pos(self)</code>	
<code>previousInFocusChain(self)</code>	
<code>property(self, name)</code>	
<code>pyqtConfigure(...)</code>	Each keyword argument is either the name of a Qt property or a Qt signal.

continues on next page

Table 2.117 – continued from previous page

<code>raise_(self)</code>
<code>receivers(self, signal)</code>
<code>rect(self)</code>
<code>releaseKeyboard(self)</code>
<code>releaseMouse(self)</code>
<code>releaseShortcut(self, id)</code>
<code>removeAction(self, action)</code>
<code>removeEventFilter(self, a0)</code>
<code>render(, sourceRegion, flags, ...)</code>
<code>repaint(-> None -> None)</code>
<code>resize()</code>
<code>resizeEvent(self, a0)</code>
<code>restoreGeometry(self, geometry)</code>
<code>saveGeometry(self)</code>
<code>screen(self)</code>
<code>scroll()</code>
<code>sender(self)</code>
<code>senderSignalIndex(self)</code>
<code>setAcceptDrops(self, on)</code>
<code>setAccessibleDescription(self, description)</code>
<code>setAccessibleName(self, name)</code>
<code>setAttribute(self, attribute[, on])</code>
<code>setAutoFillBackground(self, enabled)</code>
<code>setBackgroundRole(self, a0)</code>
<code>setBaseSize()</code>

continues on next page

Table 2.117 – continued from previous page

<code>setContentMargins()</code>
<code>setContextMenuPolicy(self, policy)</code>
<code>setCursor(self, a0)</code>
<code>setDisabled(self, a0)</code>
<code>setEnabled(self, a0)</code>
<code>setFixedHeight(self, h)</code>
<code>setFixedSize()</code>
<code>setFixedWidth(self, w)</code>
<code>setFocus()</code>
<code>setFocusPolicy(self, policy)</code>
<code>setFocusProxy(self, a0)</code>
<code>setFont(self, a0)</code>
<code>setForegroundRole(self, a0)</code>
<code>setFrameRect(self, a0)</code>
<code>setFrameShadow(self, a0)</code>
<code>setFrameShape(self, a0)</code>
<code>setFrameStyle(self, a0)</code>
<code>setGeometry()</code>
<code>setGraphicsEffect(self, effect)</code>
<code>setHidden(self, hidden)</code>
<code>setInputMethodHints(self, hints)</code>
<code>setLayout(self, a0)</code>
<code>setLayoutDirection(self, direction)</code>
<code>setLineWidth(self, a0)</code>
<code>setLocale(self, locale)</code>

continues on next page

Table 2.117 – continued from previous page

<code>setMask()</code>
<code>setMaximumHeight(self, maxh)</code>
<code>setMaximumSize()</code>
<code>setMaximumWidth(self, maxw)</code>
<code>setMidLineWidth(self, a0)</code>
<code>setMinimumHeight(self, minh)</code>
<code>setMinimumSize()</code>
<code>setMinimumWidth(self, minw)</code>
<code>setMouseTracking(self, enable)</code>
<code>setObjectName(self, name)</code>
<code>setPalette(self, a0)</code>
<code>setParent()</code>
<code>setProperty(self, name, value)</code>
<code>setShortcutAutoRepeat(self, id[, enabled])</code>
<code>setShortcutEnabled(self, id[, enabled])</code>
<code>setSizeIncrement()</code>
<code>setSizePolicy()</code>
<code>setStatusTip(self, a0)</code>
<code>setStyle(self, a0)</code>
<code>setStyleSheet(self, styleSheet)</code>
<code>setTabOrder(a0, a1)</code>
<code>setTabletTracking(self, enable)</code>
<code>setToolTip(self, a0)</code>
<code>setToolTipDuration(self, msec)</code>
<code>setUpdatesEnabled(self, enable)</code>

continues on next page

Table 2.117 – continued from previous page

<code>setVisible(self, visible)</code>
<code>setWhatsThis(self, a0)</code>
<code>setWindowFilePath(self, filePath)</code>
<code>setWindowFlag(self, a0[, on])</code>
<code>setWindowFlags(self, type)</code>
<code>setWindowIcon(self, icon)</code>
<code>setWindowIconText(self, a0)</code>
<code>setWindowModality(self, windowModality)</code>
<code>setWindowModified(self, a0)</code>
<code>setWindowOpacity(self, level)</code>
<code>setWindowRole(self, a0)</code>
<code>setWindowState(self, state)</code>
<code>setWindowTitle(self, a0)</code>
<code>sharedPainter(self)</code>
<code><i>show</i>(self)</code>
<code>showEvent(self, a0)</code>
<code>showFullScreen(self)</code>
<code>showMaximized(self)</code>
<code>showMinimized(self)</code>
<code>showNormal(self)</code>
<code>signalsBlocked(self)</code>
<code>size(self)</code>
<code>sizeHint(self)</code>
<code>sizeIncrement(self)</code>
<code>sizePolicy(self)</code>

continues on next page

Table 2.117 – continued from previous page

<code>stackUnder(self, a0)</code>
<code>startTimer(self, interval[, timerType])</code>
<code>statusTip(self)</code>
<code>style(self)</code>
<code>styleSheet(self)</code>
<code>tabletEvent(self, a0)</code>
<code>testAttribute(self, attribute)</code>
<code>thread(self)</code>
<code>timerEvent(self, a0)</code>
<code>toolTip(self)</code>
<code>toolTipDuration(self)</code>
<code>tr(self, sourceText[, disambiguation, n])</code>
<code>underMouse(self)</code>
<code>ungrabGesture(self, type)</code>
<code>unsetCursor(self)</code>
<code>unsetLayoutDirection(self)</code>
<code>unsetLocale(self)</code>
<code>update(-> None -> None)</code>
<code>updateGeometry(self)</code>
<code>updateMicroFocus(self)</code>
<code><i>update_text</i>(text)</code>
<code>updatesEnabled(self)</code>
<code>visibleRegion(self)</code>
<code>whatsThis(self)</code>
<code>wheelEvent(self, a0)</code>

continues on next page

Table 2.117 – continued from previous page

width(self)
widthMM(self)
winId(self)
window(self)
windowFilePath(self)
windowFlags(self)
windowHandle(self)
windowIcon(self)
windowIconText(self)
windowModality(self)
windowOpacity(self)
windowRole(self)
windowState(self)
windowTitle(self)
windowType(self)
x(self)
y(self)

Attributes

<i>BG</i>
Box
DrawChildren
DrawWindowBackground
HLine
IgnoreMask

continues on next page

Table 2.118 – continued from previous page

NoFrame	
Panel	
PdmDepth	
PdmDevicePixelRatio	
PdmDevicePixelRatioScaled	
PdmDpiX	
PdmDpiY	
PdmHeight	
PdmHeightMM	
PdmNumColors	
PdmPhysicalDpiX	
PdmPhysicalDpiY	
PdmWidth	
PdmWidthMM	
Plain	
Raised	
Shadow_Mask	
Shape_Mask	
StyledPanel	
Sunken	
VLine	
WinPanel	
<i>alpha</i>	
customContextMenuRequested	QPoint) [signal]
destroyed	typing.Optional[QObject] = None) [signal]
<i>h</i>	

continues on next page

Table 2.118 – continued from previous page

<i>icon_frac</i>	
<i>imheight</i>	
<i>imwidth</i>	
objectNameChanged	str) [signal]
staticMetaObject	
<i>w</i>	
windowIconChanged	QIcon) [signal]
windowIconTextChanged	str) [signal]
windowTitleChanged	str) [signal]

```

BG = '#ffffff'
alpha = 0.875
h = 230
icon_frac = 0.65
imheight = 150
imwidth = 150
paintEvent(self, a0: QPaintEvent)
show(self)
update_text(text)
w = 250

```

2.1.22 labscript_utils.testing_utils

Classes

<i>Any</i> ([types])	A class whose instances equal any object of the given type or tuple of types.
<i>ThreadTestCase</i> (*args, **kwargs)	Test case that runs tests in a new thread whilst providing a mainloop that allows running scripts in the current thread.
<i>dotdict</i>	dot.notation access to dictionary attributes
<i>monkeypatch</i> (obj, name, mocked_attr)	Context manager to temporarily monkeypatch an object attribute with some mocked attribute

labscript_utils.testing_utils.Any

class labscript_utils.testing_utils.**Any**(types=<class 'object'>)

Bases: `object`

A class whose instances equal any object of the given type or tuple of types. For use with `mock.Mock.assert_called_with` when you don't care what some of the arguments are

`__init__`(types=<class 'object'>)

Methods

`__init__`([types])

labscript_utils.testing_utils.ThreadTestCase

class labscript_utils.testing_utils.**ThreadTestCase**(*args, **kwargs)

Bases: `unittest.case.TestCase`

Test case that runs tests in a new thread whilst providing a mainloop that allows running scripts in the current thread. Those scripts can then be tested from the testing thread.

`__init__`(*args, **kwargs)

Create an instance of the class that will use the named test method when executed. Raises a `ValueError` if the instance does not have a method with the specified name.

Methods

<code>__init__</code> (*args, **kwargs)	Create an instance of the class that will use the named test method when executed.
<code>addCleanup</code> (**kwargs)	Add a function, with arguments, to be called when the test is completed.
<code>addTypeEqualityFunc</code> (typeobj, function)	Add a type specific <code>assertEqual</code> style function to compare a type.
<code>assertAlmostEqual</code> (first, second[, places, ...])	Fail if the two objects are unequal as determined by their difference rounded to the given number of decimal places (default 7) and comparing to zero, or by comparing that the difference between the two objects is more than the given delta.
<code>assertAlmostEquals</code> (**kwargs)	
<code>assertCountEqual</code> (first, second[, msg])	An unordered sequence comparison asserting that the same elements, regardless of order.
<code>assertDictContainsSubset</code> (subset, dictionary)	Checks whether dictionary is a superset of subset.
<code>assertDictEqual</code> (d1, d2[, msg])	
<code>assertEqual</code> (first, second[, msg])	Fail if the two objects are unequal as determined by the <code>'=='</code> operator.
<code>assertEquals</code> (**kwargs)	

continues on next page

Table 2.121 – continued from previous page

<code>assertFalse(expr[, msg])</code>	Check that the expression is false.
<code>assertGreater(a, b[, msg])</code>	Just like <code>self.assertTrue(a > b)</code> , but with a nicer default message.
<code>assertGreaterEqual(a, b[, msg])</code>	Just like <code>self.assertTrue(a >= b)</code> , but with a nicer default message.
<code>assertIn(member, container[, msg])</code>	Just like <code>self.assertTrue(a in b)</code> , but with a nicer default message.
<code>assertIs(expr1, expr2[, msg])</code>	Just like <code>self.assertTrue(a is b)</code> , but with a nicer default message.
<code>assertIsInstance(obj, cls[, msg])</code>	Same as <code>self.assertTrue(isinstance(obj, cls))</code> , with a nicer default message.
<code>assertIsNone(obj[, msg])</code>	Same as <code>self.assertTrue(obj is None)</code> , with a nicer default message.
<code>assertIsNot(expr1, expr2[, msg])</code>	Just like <code>self.assertTrue(a is not b)</code> , but with a nicer default message.
<code>assertIsNotNone(obj[, msg])</code>	Included for symmetry with <code>assertIsNone</code> .
<code>assertLess(a, b[, msg])</code>	Just like <code>self.assertTrue(a < b)</code> , but with a nicer default message.
<code>assertLessEqual(a, b[, msg])</code>	Just like <code>self.assertTrue(a <= b)</code> , but with a nicer default message.
<code>assertListEqual(list1, list2[, msg])</code>	A list-specific equality assertion.
<code>assertLogs([logger, level])</code>	Fail unless a log message of level <i>level</i> or higher is emitted on <i>logger_name</i> or its children.
<code>assertMultiLineEqual(first, second[, msg])</code>	Assert that two multi-line strings are equal.
<code>assertNotAlmostEqual(first, second[, ...])</code>	Fail if the two objects are equal as determined by their difference rounded to the given number of decimal places (default 7) and comparing to zero, or by comparing that the difference between the two objects is less than the given delta.
<code>assertNotAlmostEquals(**kwargs)</code>	
<code>assertNotEqual(first, second[, msg])</code>	Fail if the two objects are equal as determined by the <code>'!='</code> operator.
<code>assertNotEquals(**kwargs)</code>	
<code>assertNotIn(member, container[, msg])</code>	Just like <code>self.assertTrue(a not in b)</code> , but with a nicer default message.
<code>assertNotIsInstance(obj, cls[, msg])</code>	Included for symmetry with <code>assertIsInstance</code> .
<code>assertNotRegex(text, unexpected_regex[, msg])</code>	Fail the test if the text matches the regular expression.
<code>assertNotRegexMatches(**kwargs)</code>	
<code>assertRaises(expected_exception, *args, **kwargs)</code>	Fail unless an exception of class <code>expected_exception</code> is raised by the callable when invoked with specified positional and keyword arguments.
<code>assertRaisesRegex(expected_exception, ...)</code>	Asserts that the message in a raised exception matches a regex.
<code>assertRaisesRegexp(**kwargs)</code>	
<code>assertRegex(text, expected_regex[, msg])</code>	Fail the test unless the text matches the regular expression.

continues on next page

Table 2.121 – continued from previous page

<code>assertRegexpMatches(**kwargs)</code>	
<code>assertSequenceEqual(seq1, seq2[, msg, seq_type])</code>	An equality assertion for ordered sequences (like lists and tuples).
<code>assertSetEqual(set1, set2[, msg])</code>	A set-specific equality assertion.
<code>assertTrue(expr[, msg])</code>	Check that the expression is true.
<code>assertTupleEqual(tuple1, tuple2[, msg])</code>	A tuple-specific equality assertion.
<code>assertWarns(expected_warning, *args, **kwargs)</code>	Fail unless a warning of class <code>warnClass</code> is triggered by the callable when invoked with specified positional and keyword arguments.
<code>assertWarnsRegex(expected_warning, ...)</code>	Asserts that the message in a triggered warning matches a regexp.
<code>assert_(**kwargs)</code>	
<code>countTestCases()</code>	
<code>debug()</code>	Run the test without collecting errors in a <code>TestResult</code>
<code>defaultTestResult()</code>	
<code>doCleanups()</code>	Execute all cleanup functions.
<code>fail([msg])</code>	Fail immediately, with the given message.
<code>failIf(**kwargs)</code>	
<code>failIfAlmostEqual(**kwargs)</code>	
<code>failIfEqual(**kwargs)</code>	
<code>failUnless(**kwargs)</code>	
<code>failUnlessAlmostEqual(**kwargs)</code>	
<code>failUnlessEqual(**kwargs)</code>	
<code>failUnlessRaises(**kwargs)</code>	
<code>id()</code>	
<code>quit_mainloop()</code>	
<code>run(*args, **kwargs)</code>	
<code>run_script_as_main(filepath)</code>	
<code>setUp()</code>	Hook method for setting up the test fixture before exercising it.
<code>setUpClass()</code>	Hook method for setting up class fixture before running tests in the class.
<code>shortDescription()</code>	Returns a one-line description of the test, or <code>None</code> if no description has been provided.
<code>skipTest(reason)</code>	Skip this test.

continues on next page

Table 2.121 – continued from previous page

<code>subTest([msg])</code>	Return a context manager that will return the enclosed block of code in a subtest identified by the optional message and keyword parameters.
<code>tearDown()</code>	Hook method for deconstructing the test fixture after testing it.
<code>tearDownClass()</code>	Hook method for deconstructing the class fixture after running all tests in the class.
<code>wait_for(condition_func[, timeout, ...])</code>	Busy wait for a condition to be true.

Attributes

`longMessage`

`maxDiff`

`quit_mainloop()`

`run(*args, **kwargs)`

`run_script_as_main(filepath)`

`static wait_for(condition_func, timeout=5, initial_poll_interval=0.005, max_poll_interval=0.5)`

Busy wait for a condition to be true. Uses exponential backoff so it's fast when things are fast and not a complete hog when they're not

labscript_utils.testing_utils.dotdict

class `labscript_utils.testing_utils.dotdict`

Bases: `dict`

`dot`.notation access to dictionary attributes

`__init__(*args, **kwargs)`

Methods

`__init__(*args, **kwargs)`

`clear()`

`copy()`

`fromkeys([value])`

Create a new dictionary with keys from iterable and values set to value.

`get(key[, default])`

Return the value for key if key is in the dictionary, else default.

`items()`

continues on next page

Table 2.123 – continued from previous page

<code>keys()</code>	
<code>pop(k[,d])</code>	If key is not found, d is returned if given, otherwise <code>KeyError</code> is raised
<code>popitem()</code>	2-tuple; but raise <code>KeyError</code> if D is empty.
<code>setdefault(key[, default])</code>	Insert key with a value of default if key is not in the dictionary.
<code>update([E,]**F)</code>	If E is present and has a <code>.keys()</code> method, then does: for k in E: <code>D[k] = E[k]</code> If E is present and lacks a <code>.keys()</code> method, then does: for k, v in E: <code>D[k] = v</code> In either case, this is followed by: for k in F: <code>D[k] = F[k]</code>
<code>values()</code>	

labscript_utils.testing_utils.monkeypatch

class `labscript_utils.testing_utils.monkeypatch(obj, name, mocked_attr)`

Bases: `object`

Context manager to temporarily monkeypatch an object attribute with some mocked attribute

`__init__(obj, name, mocked_attr)`

Methods

`__init__(obj, name, mocked_attr)`

2.1.23 labscript_utils.tracelog

Functions

<code>log([log_path, module_names, sub, all, mode])</code>	Trace and log Python execution.
--	---------------------------------

labscript_utils.tracelog.log

`labscript_utils.tracelog.log(log_path=None, module_names=(), sub=False, all=False, mode='w')`

Trace and log Python execution.

output includes the time, thread name, containing function name, line number and source line. Indentation before the thread name represents stack depth, indentation before source line is as in the source line itself.

`log_path`: the path of the desired output file to write to, or `None` for stdout (default=`None`) `module_names`: list of module names that tracing is desired for (default=`()`) `sub`: whether submodules of the above modules should be traced (default=`False`) `all`: whether all modules should be traced, in which case `module_names` is ignored (default=`False`) `mode`: mode to open the output file in, if `log_path` is not `None` (default=`'w'`)

2.1.24 labscript_utils.unitconversions

Sub-Modules

labscript_utils.unitconversions.

NovaTechDDS9m

labscript_utils.unitconversions.

UnitConversionBase

labscript_utils.unitconversions.aom

labscript_utils.unitconversions.detuning

labscript_utils.unitconversions.example

labscript_utils.unitconversions.

Generic frequency conversion

generic_frequency

labscript_utils.unitconversions.

linear_coil_driver

labscript_utils.unitconversions.

optotunelens

labscript_utils.unitconversions.

quad_driver

labscript_utils.unitconversions.

quad_monitor

labscript_utils.unitconversions.test

labscript_utils.unitconversions.NovaTechDDS9m

Classes

NovaTechDDS9mAmpConversion(...)

NovaTechDDS9mFreqConversion(...)

labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mAmpConversion

class labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mAmpConversion(*calibration_parameters=None*)

Bases: *labscript_utils.unitconversions.UnitConversionBase*.*UnitConversion*

__init__(*calibration_parameters=None*)

Methods

`__init__`([calibration_parameters])

`hardware_from_base`(arb)

`hardware_to_base`(hardware)

Attributes

`base_unit`

`unit_list`

`base_unit` = 'Arb'

`hardware_from_base`(arb)

`hardware_to_base`(hardware)

labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mFreqConversion

class `labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mFreqConversion`(*calibration_parameters=None*)
Bases: `labscript_utils.unitconversions.UnitConversionBase.UnitConversion`

`__init__`(*calibration_parameters=None*)

Methods

`MHz_from_base`(Hz)

`MHz_to_base`(MHz)

`__init__`([calibration_parameters])

Attributes

`base_unit`

`unit_list`

`MHz_from_base`(Hz)

`MHz_to_base`(MHz)

```
base_unit = 'Hz'
```

labscript_utils.unitconversions.UnitConversionBase

Functions

```
vectorise(method)
```

labscript_utils.unitconversions.UnitConversionBase.vectorise

```
labscript_utils.unitconversions.UnitConversionBase.vectorise(method)
```

Classes

```
UnitConversion(params)
```

labscript_utils.unitconversions.UnitConversionBase.UnitConversion

```
class labscript_utils.unitconversions.UnitConversionBase.UnitConversion(params)
```

```
    Bases: object
```

```
    __init__(params)
```

Methods

```
__init__(params)
```

Attributes

```
unit_list
```

```
unit_list = {'G': 1000000000.0, 'M': 1000000.0, 'T': 1000000000000.0, 'k': 1000.0,  
'm': 0.001, 'n': 1e-09, 'p': 1e-12, 'u': 1e-06}
```

labscript_utils.unitconversions.aom

Classes

<i>SineAom</i> ([calibration_parameters])	AOM calibration P(A) is very close to a sine for dipole trap AOM!
---	---

labscript_utils.unitconversions.aom.SineAom

class labscript_utils.unitconversions.aom.**SineAom**(*calibration_parameters=None*)
Bases: *labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mAmpConversion*
AOM calibration P(A) is very close to a sine for dipole trap AOM!
__init__(*calibration_parameters=None*)

Methods

Power_from_base(amp)

Power_to_base(power)

__init__([calibration_parameters])

fraction_from_base(amp)

fraction_to_base(fraction)

hardware_from_base(arb)

hardware_to_base(hardware)

Attributes

base_unit

unit_list

Power_from_base(amp)

Power_to_base(power)

base_unit = 'Arb'

fraction_from_base(amp)

fraction_to_base(fraction)

labscript_utils.unitconversions.detuning**Classes**

detuning([calibration_parameters])

labscript_utils.unitconversions.detuning.detuning

class labscript_utils.unitconversions.detuning.**detuning**(*calibration_parameters=None*)

Bases: *labscript_utils.unitconversions.UnitConversionBase.UnitConversion*

__init__(*calibration_parameters=None*)

Methods

MHz_from_base(*aom_frequency*)

MHz_to_base(*aom_frequency_MHz*)

__init__([*calibration_parameters*])

d_MHz_from_base(*aom_frequency*)

d_MHz_to_base(*detuning_MHz*)

linewidths_from_base(*aom_frequency*)

linewidths_to_base(*linewidths*)

Attributes

base_unit

derived_units

unit_list

MHz_from_base(*aom_frequency*)

MHz_to_base(*aom_frequency_MHz*)

base_unit = 'Hz'

d_MHz_from_base(*aom_frequency*)

d_MHz_to_base(*detuning_MHz*)

derived_units = ['MHz', 'd_MHz', 'linewidths']

`linewidths_from_base(aom_frequency)`

`linewidths_to_base(linewidths)`

labscript_utils.unitconversions.example

Classes

`example1([calibration_parameters])`

`example2([calibration_parameters])`

`example3([calibration_parameters])`

labscript_utils.unitconversions.example.example1

class `labscript_utils.unitconversions.example.example1(calibration_parameters=None)`
Bases: `labscript_utils.unitconversions.UnitConversionBase.UnitConversion`
`__init__(calibration_parameters=None)`

Methods

`A_from_base(volts)`

`A_to_base(amps)`

`Gauss_from_base(volts)`

`Gauss_to_base(gauss)`

`__init__([calibration_parameters])`

Attributes

`base_unit`

`unit_list`

A_from_base(*volts*)

A_to_base(*amps*)

Gauss_from_base(*volts*)

Gauss_to_base(*gauss*)

```
base_unit = 'V'
```

labscript_utils.unitconversions.example.example2

```
class labscript_utils.unitconversions.example.example2(calibration_parameters=None)
    Bases: labscript_utils.unitconversions.UnitConversionBase, UnitConversion
    __init__(calibration_parameters=None)
```

Methods

```
__init__([calibration_parameters])
```

```
detuned_MHz_from_base(mhz)
```

```
detuned_MHz_to_base(d_mhz)
```

Attributes

```
base_unit
```

```
unit_list
```

```
base_unit = 'MHz'
```

```
detuned_MHz_from_base(mhz)
```

```
detuned_MHz_to_base(d_mhz)
```

labscript_utils.unitconversions.example.example3

```
class labscript_utils.unitconversions.example.example3(calibration_parameters=None)
    Bases: labscript_utils.unitconversions.UnitConversionBase, UnitConversion
    __init__(calibration_parameters=None)
```

Methods

```
W_from_base(vpp)
```

```
W_to_base(watts)
```

```
__init__([calibration_parameters])
```

Attributes

base_unit

unit_list

W_from_base(*vpp*)

W_to_base(*watts*)

base_unit = 'Vpp'

labscript_utils.unitconversions.generic_frequency

Generic frequency conversion

Classes

<i>FreqConversion</i> ([<i>calibration_parameters</i>])	A Generic frequency conversion class that covers standard SI prefixes from a base of Hz.
---	--

labscript_utils.unitconversions.generic_frequency.FreqConversion

class `labscript_utils.unitconversions.generic_frequency.FreqConversion`(*calibration_parameters=None*)

Bases: `labscript_utils.unitconversions.UnitConversionBase`, `UnitConversion`

A Generic frequency conversion class that covers standard SI prefixes from a base of Hz.

__init__(*calibration_parameters=None*)

Methods

GHz_from_base(Hz)

GHz_to_base(GHz)

MHz_from_base(Hz)

MHz_to_base(MHz)

__init__([*calibration_parameters*])

kHz_from_base(Hz)

kHz_to_base(kHz)

Attributes

base_unit

unit_list

GHz_from_base(Hz)

GHz_to_base(GHz)

MHz_from_base(Hz)

MHz_to_base(MHz)

base_unit = 'Hz'

kHz_from_base(Hz)

kHz_to_base(kHz)

labscript_utils.unitconversions.linear_coil_driver**Classes**

BidirectionalCoilDriver([calibration_parameters])

UnidirectionalCoilDriver([...])

labscript_utils.unitconversions.linear_coil_driver.BidirectionalCoilDriver**class** labscript_utils.unitconversions.linear_coil_driver.**BidirectionalCoilDriver**(*calibration_parameters=None*)
Bases: *labscript_utils.unitconversions.UnitConversionBase.UnitConversion**__init__*(*calibration_parameters=None*)**Methods**

A_from_base(volts)

A_to_base(amps)

__init__([calibration_parameters])

Attributes

base_unit

derived_units

unit_list

A_from_base(volts)

A_to_base(amps)

base_unit = 'V'

derived_units = ['A']

labscript_utils.unitconversions.linear_coil_driver.UnidirectionalCoilDriver

class labscript_utils.unitconversions.linear_coil_driver.**UnidirectionalCoilDriver**(calibration_parameters=None)

Bases: *labscript_utils.unitconversions.linear_coil_driver.BidirectionalCoilDriver*

__init__(calibration_parameters=None)

Methods

A_from_base(volts)

A_to_base(amps)

__init__([calibration_parameters])

Attributes

base_unit

derived_units

unit_list

A_from_base(volts)

A_to_base(amps)

labscript_utils.unitconversions.optotunelens**Classes**

OptotuneLens([calibration_parameters])

labscript_utils.unitconversions.optotunelens.OptotuneLens

class labscript_utils.unitconversions.optotunelens.**OptotuneLens**(*calibration_parameters=None*)

Bases: *labscript_utils.unitconversions.UnitConversionBase.UnitConversion*

__init__(*calibration_parameters=None*)

Methods

I_from_base(volts)

I_to_base(current)

__init__([calibration_parameters])

distance_from_base(volts)

distance_to_base(percentage)

Attributes

base_unit

derived_units

unit_list

I_from_base(*volts*)

I_to_base(*current*)

base_unit = 'V'

derived_units = ['distance', 'I']

distance_from_base(*volts*)

distance_to_base(*percentage*)

labscript_utils.unitconversions.quad_driver

Classes

quad_driver([calibration_parameters])

labscript_utils.unitconversions.quad_driver.quad_driver

class labscript_utils.unitconversions.quad_driver.**quad_driver**(*calibration_parameters*={'A_min': - 0.09, 'A_offset': - 0.642724, 'A_per_V': 19.9757, 'Gcm_per_A': 1.88679})

Bases: *labscript_utils.unitconversions.UnitConversionBase.UnitConversion*

__init__(*calibration_parameters*={'A_min': - 0.09, 'A_offset': - 0.642724, 'A_per_V': 19.9757, 'Gcm_per_A': 1.88679})

Methods

A_from_base(volts)

A_to_base(arg)

Gcm_from_base(volts)

Gcm_to_base(gauss_per_cm)

__init__([calibration_parameters])

Attributes

base_unit

derived_units

unit_list

A_from_base(*volts*)

A_to_base(*arg*)

Gcm_from_base(*volts*)

Gcm_to_base(*gauss_per_cm*)

base_unit = 'V'

derived_units = ['A', 'Gcm']

labscript_utils.unitconversions.quad_monitor**Classes**

quad_monitor([calibration_parameters])

labscript_utils.unitconversions.quad_monitor.quad_monitor

class labscript_utils.unitconversions.quad_monitor.**quad_monitor**(*calibration_parameters*={'A_offset': -0.043200000000000016, 'A_per_V': 20.032, 'Gcm_per_A': 1.88679})

Bases: *labscript_utils.unitconversions.UnitConversionBase.UnitConversion*

__init__(*calibration_parameters*={'A_offset': -0.043200000000000016, 'A_per_V': 20.032, 'Gcm_per_A': 1.88679})

Methods

A_from_base(volts)

A_to_base(amps)

Gcm_from_base(volts)

Gcm_to_base(gauss_per_cm)

__init__([calibration_parameters])

Attributes

base_unit

derived_units

unit_list

A_from_base(*volts*)

A_to_base(*amps*)

Gcm_from_base(*volts*)

Gcm_to_base(*gauss_per_cm*)

base_unit = 'V'

derived_units = ['A', 'Gcm']

labscript_utils.unitconversions.test

Classes

test([calibration_parameters])

labscript_utils.unitconversions.test.test

class labscript_utils.unitconversions.test.test(*calibration_parameters=None*)
Bases: *labscript_utils.unitconversions.UnitConversionBase.UnitConversion*
__init__(*calibration_parameters=None*)

Methods

A_from_base(volts)

A_to_base(amps)

Gauss_from_base(volts)

Gauss_to_base(gauss)

__init__([calibration_parameters])

Attributes

base_unit

derived_units

unit_list

A_from_base(*volts*)

A_to_base(*amps*)

Gauss_from_base(*volts*)

Gauss_to_base(*gauss*)

base_unit = 'MHz'

derived_units = ['A', 'Gauss']

Functions

<code>get_unit_conversion_class(fullname)</code>	import and return the unit conversion class with the given name.
--	--

labscript_utils.unitconversions.get_unit_conversion_class

`labscript_utils.unitconversions.get_unit_conversion_class(fullname)`

import and return the unit conversion class with the given name. Ideally this is a fully qualified class name with an absolute import path, i.e. `path.to.some.module.ClassName`. But if it is just a single name, we fall back to looking through all classes defined in submodules. This allows backward compatibility with old shot files that do not have the full name saved.

2.1.25 labscript_utils.versions

Functions

<code>check_version(module_name, at_least, less_than)</code>	Checks if a module version is within specified bounds.
<code>get_import_path(import_name)</code>	Get which entry in <code>sys.path</code> a module would be imported from, without importing it.
<code>get_version(import_name[, project_name, ...])</code>	Try very hard to get the version of a package without importing it.

labscript_utils.versions.check_version

`labscript_utils.versions.check_version(module_name, at_least, less_than, version=None, project_name=None)`

Checks if a module version is within specified bounds.

Checks that the version of the given module is at least and less than the given version strings. This function uses `get_version()` to determine version numbers without importing modules. In order to do this, `project_name` must be provided if it differs from `module_name`. For example, `pyserial` is imported as `'serial'`, but the project name, as passed to a `'pip install'` command, is `'pyserial'`. Therefore to check the version of `pyserial`, pass in `module_name='serial'` and `project_name='pyserial'`. You can also pass in a version string yourself, in which case no inspection of packages will take place.

Parameters

- **module_name** (*str*) – The name of the module to check.
- **at_least** (*str*) – The minimum acceptable module version.
- **less_than** (*str*) – The minimum unacceptable module version. Usually this would be the next major version if the package follows `semver`.
- **version** (*str*, *optional*) – The current version of the installed package. Useful when the package version is stored in a non-standard location.
- **project_name** (*str*, *optional*) – The package name (e.g. the name used when pip installing the package). This must be specified if it does not match the module name.

Raises `VersionException` – if the module was not found or its version could not be determined.

labscript_utils.versions.get_import_path

labscript_utils.versions.get_import_path(*import_name*)

Get which entry in sys.path a module would be imported from, without importing it.

Parameters *import_name* (*str*) – The module name.

Raises

- **ModuleNotFoundError** – Raised if the module is not installed.
- **NotImplementedError** – Raised if the module is a “namespace package”. Support for namespace packages is not currently available.

Returns The path to the folder containing the module.

Return type *str*

labscript_utils.versions.get_version

labscript_utils.versions.get_version(*import_name*, *project_name=None*, *import_path=None*)

Try very hard to get the version of a package without importing it.

If *import_path* is not given, first find where it would be imported from, without importing it. Then look for metadata in the same import path with the given project name (note: this is not always the same as the import name, it is the name for example you would ask pip to install). If that is found, return the version info from it. Otherwise look for a `__version__.py` file in the package directory, or a `__version__ = <version>` literal defined in the package source (without executing it).

Parameters

- **import_name** (*str*) – The module name.
- **project_name** (*str*, *optional*) – The package name (e.g. the name used when pip installing the package). This must be specified if it does not match the module name.
- **import_path** (*str*, *optional*) – The path to the folder containing the installed package.

Raises **NotImplementedError** – Raised if the module name contains a period. Only top-level packages are supported at this time.

Returns The version literal of the package. If the package cannot be found, *NotFound* is returned. If the version cannot be obtained in the above way, or if the version was found but was *None*, *NoVersionInfo* is returned.

Classes

NoVersionInfo()

NotFound()

labscript_utils.versions.NoVersionInfo**class** labscript_utils.versions.NoVersionInfoBases: *object***__init__**()**Methods**

__init__()

labscript_utils.versions.NotFound**class** labscript_utils.versions.NotFoundBases: *object***__init__**()**Methods**

__init__()

Exceptions

BrokenInstall

VersionException

labscript_utils.versions.BrokenInstall**exception** labscript_utils.versions.BrokenInstall

labscript_utils.versions.VersionException

exception labscript_utils.versions.VersionException

2.1.26 labscript_utils.zlock

Script to run a zlock server configured according to LabConfig. Run with:

```
python -m labscript_utils.zlock [--daemon]
```

If `--daemon` is specified, the zlock server will be started in the background.

Functions

main()

labscript_utils.zlock.main

labscript_utils.zlock.**main()**

2.1.27 labscript_utils.zlog

Script to run a zlog server configured according to LabConfig. Run with:

```
python -m labscript_utils.zlog [--daemon]
```

If `--daemon` is specified, the zlog server will be started in the background.

Functions

main()

labscript_utils.zlog.main

labscript_utils.zlog.**main()**

Functions

<i>dedent</i> (s)	Remove leading spaces from the first line of a string, all common leading indentation (spaces only) from subsequent lines, strip trailing spaces from all lines and replace single newlines prior to lines with the common indentation with spaces.
<i>import_or_reload</i> (modulename)	Behaves like 'import modulename' would, excepts forces the imported script to be rerun

2.1.28 labscript_utils.dedent

labscript_utils.dedent(s)

Remove leading spaces from the first line of a string, all common leading indentation (spaces only) from subsequent lines, strip trailing spaces from all lines and replace single newlines prior to lines with the common indentation with spaces. Lines with additional indentation are kept verbatim. Good for unwrapping error messages etc that are in code as multiline triple-quoted strings.

2.1.29 labscript_utils.import_or_reload

labscript_utils.import_or_reload(modulename)

Behaves like 'import modulename' would, excepts forces the imported script to be rerun

2.2 labscript_profile

Sub-Modules

<i>labscript_profile.create</i>

2.2.1 labscript_profile.create

Functions

<i>create_profile</i> ()	
<i>make_labconfig_file</i> ()	
<i>make_shared_secret</i> (directory)	Create a new zprocess shared secret file in the given directory and return its filepath

labscript_profile.create.create_profile

labscript_profile.create.create_profile()

labscript_profile.create.make_labconfig_file

labscript_profile.create.make_labconfig_file()

labscript_profile.create.make_shared_secret

labscript_profile.create.make_shared_secret(*directory*)

Create a new zprocess shared secret file in the given directory and return its filepath

Functions

add_userlib_and_pythonlib()

Find the users's labconfig file, read the userlib and pythonlib keys, and add those directories to the Python search path.

default_labconfig_path()

hostname()

2.2.2 labscript_profile.add_userlib_and_pythonlib

labscript_profile.add_userlib_and_pythonlib()

Find the users's labconfig file, read the userlib and pythonlib keys, and add those directories to the Python search path. This function intentionally re-implements finding and reading the config file so as to not import labscript_utils, since we dont' want to import something like labscript_utils every time the interpreter starts up

2.2.3 labscript_profile.default_labconfig_path

labscript_profile.default_labconfig_path()

2.2.4 labscript_profile.hostname

labscript_profile.hostname()

LABSCRIPT SUITE COMPONENTS

The *labscript suite* is modular by design, and is comprised of:

Table 3.1: Python libraries

	labscript — Expressive composition of hardware-timed experiments
	labscript-devices — Plugin architecture for controlling experiment hardware
	labscript-utils — Shared modules used by the <i>labscript suite</i>

Table 3.2: Graphical applications

	runmanager — Graphical and remote interface to parameterized experiments
	blacs — Graphical interface to scientific instruments and experiment supervision
	lyse — Online analysis of live experiment data
	runviewer — Visualize hardware-timed experiment instructions

PYTHON MODULE INDEX

|

- labscript_profile, 388
- labscript_profile.create, 387
- labscript_utils, 386
- labscript_utils.camera_server, 6
- labscript_utils.connections, 8
- labscript_utils.device_registry, 11
- labscript_utils.dict_diff, 11
- labscript_utils.double_import_denier, 11
- labscript_utils.excepthook, 22
- labscript_utils.excepthook.tk_exception, 12
- labscript_utils.filewatcher, 23
- labscript_utils.h5_lock, 25
- labscript_utils.impprof, 28
- labscript_utils.labconfig, 28
- labscript_utils.ls_zprocess, 31
- labscript_utils.memprof, 37
- labscript_utils.modulewatcher, 38
- labscript_utils.properties, 39
- labscript_utils.qtwidgets, 345
- labscript_utils.qtwidgets.analoginput, 42
- labscript_utils.qtwidgets.analogoutput, 56
- labscript_utils.qtwidgets.ddsoutput, 86
- labscript_utils.qtwidgets.digitaloutput, 100
- labscript_utils.qtwidgets.dragdroptab, 131
- labscript_utils.qtwidgets.elide_label, 171
- labscript_utils.qtwidgets.enumoutput, 201
- labscript_utils.qtwidgets.fingertab, 216
- labscript_utils.qtwidgets.headerview_with_widgets, 248
- labscript_utils.qtwidgets.imageoutput, 273
- labscript_utils.qtwidgets.InputPlotWindow, 41
- labscript_utils.qtwidgets.outputbox, 321
- labscript_utils.qtwidgets.toolpalette, 322
- labscript_utils.remote, 345
- labscript_utils.settings, 345
- labscript_utils.setup_logging, 346
- labscript_utils.shared_drive, 347
- labscript_utils.shot_utils, 348
- labscript_utils.splash, 348
- labscript_utils.testing_utils, 363
- labscript_utils.tracelog, 368
- labscript_utils.unitconversions, 382
- labscript_utils.unitconversions.aom, 372
- labscript_utils.unitconversions.detuning, 373
- labscript_utils.unitconversions.example, 374
- labscript_utils.unitconversions.generic_frequency, 376
- labscript_utils.unitconversions.linear_coil_driver, 377
- labscript_utils.unitconversions.NovaTechDDS9m, 369
- labscript_utils.unitconversions.optotunelens, 379
- labscript_utils.unitconversions.quad_driver, 380
- labscript_utils.unitconversions.quad_monitor, 381
- labscript_utils.unitconversions.test, 382
- labscript_utils.unitconversions.UnitConversionBase, 371
- labscript_utils.versions, 383
- labscript_utils.zlock, 386
- labscript_utils.zlog, 386

INDEX

Symbols

- `__init__` () (*labscript_utils.camera_server.CameraServer* method), 6
- `__init__` () (*labscript_utils.camera_server.TubingenCameraServer* method), 7
- `__init__` () (*labscript_utils.connections.Connection* method), 9
- `__init__` () (*labscript_utils.connections.ConnectionTable* method), 10
- `__init__` () (*labscript_utils.double_import_denier.DoubleImportDenier* method), 12
- `__init__` () (*labscript_utils.excepthook.l* method), 23
- `__init__` () (*labscript_utils.excepthook.tk_exception.ErrorWindow* method), 12
- `__init__` () (*labscript_utils.filewatcher.FileWatcher* method), 23
- `__init__` () (*labscript_utils.h5_lock.File* method), 25
- `__init__` () (*labscript_utils.labconfig.EnvInterpolation* method), 29
- `__init__` () (*labscript_utils.labconfig.LabConfig* method), 30
- `__init__` () (*labscript_utils.ls_zprocess.Context* method), 34
- `__init__` () (*labscript_utils.ls_zprocess.ProcessTree* method), 35
- `__init__` () (*labscript_utils.ls_zprocess.ZMQClient* method), 36
- `__init__` () (*labscript_utils.ls_zprocess.ZMQServer* method), 37
- `__init__` () (*labscript_utils.memprof.MemoryProfiler* method), 38
- `__init__` () (*labscript_utils.modulewatcher.ModuleWatcher* method), 39
- `__init__` () (*labscript_utils.qtwidgets.InputPlotWindow.PlotWindow* method), 41
- `__init__` () (*labscript_utils.qtwidgets.analoginput.AnalogInput* method), 42
- `__init__` () (*labscript_utils.qtwidgets.analogoutput.AnalogOutput* method), 56
- `__init__` () (*labscript_utils.qtwidgets.analogoutput.NoSteerFocusDoubleSpinBox* method), 70
- `__init__` () (*labscript_utils.qtwidgets.ddsoutput.DDSOutput* method), 87
- `__init__` () (*labscript_utils.qtwidgets.digitaloutput.DigitalOutput* method), 100
- `__init__` () (*labscript_utils.qtwidgets.digitaloutput.InvertedDigitalOutput* method), 116
- `__init__` () (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar* method), 131
- `__init__` () (*labscript_utils.qtwidgets.dragdroptab.DragDropTabWidget* method), 150
- `__init__` () (*labscript_utils.qtwidgets.dragdroptab.Tab* method), 165
- `__init__` () (*labscript_utils.qtwidgets.dragdroptab.TabAnimation* method), 167
- `__init__` () (*labscript_utils.qtwidgets.dragdroptab.debug* method), 171
- `__init__` () (*labscript_utils.qtwidgets.elide_label.ElideScrollArea* method), 172
- `__init__` () (*labscript_utils.qtwidgets.elide_label.ElidedLabelContainer* method), 188
- `__init__` () (*labscript_utils.qtwidgets.enumoutput.EnumOutput* method), 202
- `__init__` () (*labscript_utils.qtwidgets.fingertab.FingerTabBarWidget* method), 216
- `__init__` () (*labscript_utils.qtwidgets.fingertab.FingerTabWidget* method), 232
- `__init__` () (*labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderView* method), 248
- `__init__` () (*labscript_utils.qtwidgets.imageoutput.BrowseButton* method), 274
- `__init__` () (*labscript_utils.qtwidgets.imageoutput.ImageOutput* method), 289
- `__init__` () (*labscript_utils.qtwidgets.imageoutput.ImageView* method), 302
- `__init__` () (*labscript_utils.qtwidgets.outputbox.OutputBox* method), 321
- `__init__` () (*labscript_utils.qtwidgets.toolpalette.ToolPalette* method), 322
- `__init__` () (*labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup* method), 338
- `__init__` () (*labscript_utils.settings.Settings* method), 345
- `__init__` () (*labscript_utils.setup_logging.LessThanFilter* method), 345

- method), 347
- __init__ () (labscript_utils.splash.Splash method), 348
- __init__ () (labscript_utils.testing_utils.Any method), 364
- __init__ () (labscript_utils.testing_utils.ThreadTestCase method), 364
- __init__ () (labscript_utils.testing_utils.dotdict method), 367
- __init__ () (labscript_utils.testing_utils.monkeypatch method), 368
- __init__ () (labscript_utils.unitconversions.NovaTechDDSUnitBase() method), 369
- __init__ () (labscript_utils.unitconversions.NovaTechDDSUnitBase() method), 370
- __init__ () (labscript_utils.unitconversions.UnitConversionBase() method), 371
- __init__ () (labscript_utils.unitconversions.aom.SineAom method), 372
- __init__ () (labscript_utils.unitconversions.detuning.detuning method), 373
- __init__ () (labscript_utils.unitconversions.example.example1 method), 374
- __init__ () (labscript_utils.unitconversions.example.example2 method), 375
- __init__ () (labscript_utils.unitconversions.example.example3 method), 375
- __init__ () (labscript_utils.unitconversions.generic_frequency_converter.GenericFrequencyConverter method), 376
- __init__ () (labscript_utils.unitconversions.linear_coil_driver.BidirectionalCoilDriver method), 377
- __init__ () (labscript_utils.unitconversions.linear_coil_driver.UnidirectionalCoilDriver method), 378
- __init__ () (labscript_utils.unitconversions.optotunelens.OptotuneLens method), 379
- __init__ () (labscript_utils.unitconversions.quad_driver.quad_driver method), 380
- __init__ () (labscript_utils.unitconversions.quad_monitor.quad_monitor method), 381
- __init__ () (labscript_utils.unitconversions.test.test method), 382
- __init__ () (labscript_utils.versions.NoVersionInfo method), 385
- __init__ () (labscript_utils.versions.NotFound method), 385
- A**
- A_from_base() (labscript_utils.unitconversions.example.example1 method), 374
- A_from_base() (labscript_utils.unitconversions.linear_coil_driver.BidirectionalCoilDriver method), 378
- A_from_base() (labscript_utils.unitconversions.linear_coil_driver.UnidirectionalCoilDriver method), 378
- A_from_base() (labscript_utils.unitconversions.quad_driver.quad_driver method), 380
- A_from_base() (labscript_utils.unitconversions.quad_monitor.quad_monitor method), 381
- A_from_base() (labscript_utils.unitconversions.test.test method), 382
- A_to_base() (labscript_utils.unitconversions.example.example1 method), 374
- A_to_base() (labscript_utils.unitconversions.linear_coil_driver.BidirectionalCoilDriver method), 378
- A_to_base() (labscript_utils.unitconversions.linear_coil_driver.UnidirectionalCoilDriver method), 378
- A_to_base() (labscript_utils.unitconversions.quad_driver.quad_driver method), 380
- A_to_base() (labscript_utils.unitconversions.quad_monitor.quad_monitor method), 381
- A_to_base() (labscript_utils.unitconversions.test.test method), 382
- abort() (labscript_utils.camera_server.CameraServer method), 7
- abort() (labscript_utils.camera_server.TubingenCameraServer method), 8
- add_file() (labscript_utils.filewatcher.FileWatcher method), 24
- add_files() (labscript_utils.filewatcher.FileWatcher method), 24
- add_folder() (labscript_utils.filewatcher.FileWatcher method), 24
- add_folders() (labscript_utils.filewatcher.FileWatcher method), 24
- add_backend() (labscript_utils.settings.Settings method), 346
- add_to_linked_group() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
- add_userlib_and_pythonlib() (in module labscript_profile), 388
- addWidget() (labscript_utils.qtwidgets.toolpalette.ToolPalette method), 338
- alpha (labscript_utils.splash.Splash attribute), 363
- AnalogInput (class in labscript_utils.qtwidgets.analoginput), 42
- AnalogOutput (class in labscript_utils.qtwidgets.analogoutput), 56
- animate_limbo() (labscript_utils.qtwidgets.dragdroptab.TabAnimation method), 170
- Any (class in labscript_utils.testing_utils), 364
- append_new_palette() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 246
- assert_superset() (labscript_utils.connection.ConnectionTable method), 10

B

base_unit (labscript_utils.unitconversions.aom.SineAom attribute), 372

base_unit (labscript_utils.unitconversions.detuning.detuning attribute), 373

base_unit (labscript_utils.unitconversions.example.example1 attribute), 374

base_unit (labscript_utils.unitconversions.example.example2 attribute), 375

base_unit (labscript_utils.unitconversions.example.example3 attribute), 376

base_unit (labscript_utils.unitconversions.generic_frequency.FreqConversion attribute), 377

base_unit (labscript_utils.unitconversions.linear_coil_driver.BidirectionalCoilDriver attribute), 378

base_unit (labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mAmpConversion attribute), 370

base_unit (labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mFreqConversion attribute), 370

base_unit (labscript_utils.unitconversions.optotunelens.OptotuneLens attribute), 379

base_unit (labscript_utils.unitconversions.quad_driver.quad_driver attribute), 380

base_unit (labscript_utils.unitconversions.quad_monitor.quad_monitor attribute), 381

base_unit (labscript_utils.unitconversions.test.test attribute), 382

before_get() (labscript_utils.labconfig.EnvInterpolation method), 29

BG (labscript_utils.splash.Splash attribute), 363

BidirectionalCoilDriver (class in labscript_utils.unitconversions.linear_coil_driver), 377

block_combobox_signals() (labscript_utils.qtwidgets.analogoutput.AnalogOutput method), 70

block_combobox_signals() (labscript_utils.qtwidgets.enumoutput.EnumOutput method), 215

block_spinbox_signals() (labscript_utils.qtwidgets.analogoutput.AnalogOutput method), 70

BrokenInstall, 385

browse() (labscript_utils.qtwidgets.imageoutput.BrowseButton method), 288

BrowseButton (class in labscript_utils.qtwidgets.imageoutput), 274

button_left (labscript_utils.qtwidgets.dragdroptab.Tab property), 166

button_right (labscript_utils.qtwidgets.dragdroptab.Tab property), 166

C

CameraServer (class in labscript_utils.camera_server),

6

check() (in module labscript_utils.memprof), 37

check() (labscript_utils.filewatcher.FileWatcher method), 24

check() (labscript_utils.memprof.MemoryProfiler method), 38

check() (labscript_utils.modulewatcher.ModuleWatcher method), 39

check_version() (in module labscript_utils.versions), 383

close() (labscript_utils.h5_lock.File method), 28

close() (labscript_utils.settings.Settings method), 346

compare_to() (labscript_utils.connections.Connection method), 9

compare_to() (labscript_utils.connections.ConnectionTable method), 10

connect_to_zlock_server() (in module labscript_utils.ls_zprocess), 33

connect_value_change() (labscript_utils.qtwidgets.analogoutput.AnalogOutput method), 70

connect_value_change() (labscript_utils.qtwidgets.enumoutput.EnumOutput method), 215

Connection (class in labscript_utils.connections), 9

ConnectionTable (class in labscript_utils.connections), 10

Context (class in labscript_utils.ls_zprocess), 34

contextMenuEvent() (labscript_utils.qtwidgets.imageoutput.ImageView method), 320

copy() (labscript_utils.excepthook.tk_exception.ErrorWindow method), 22

count_types() (labscript_utils.memprof.MemoryProfiler method), 38

create_dialog() (labscript_utils.settings.Settings method), 346

create_linked_width_group() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344

create_profile() (in module labscript_profile.create), 388

D

d_MHz_from_base() (labscript_utils.unitconversions.detuning.detuning method), 373

d_MHz_to_base() (labscript_utils.unitconversions.detuning.detuning method), 373

data (labscript_utils.qtwidgets.dragdroptab.Tab property), 166

DDSOutput (class in labscript_utils.qtwidgets.ddsoutput), 87

- debug (class in labscript_utils.qtwidgets.dragdroptab), 171
- DEBUG (labscript_utils.qtwidgets.dragdroptab.debug attribute), 171
- dedent() (in module labscript_utils), 387
- default_labconfig_path() (in module labscript_profile), 388
- depth (labscript_utils.qtwidgets.dragdroptab.debug attribute), 171
- derived_units (labscript_utils.unitconversions.detuning.detuning script_utils.qtwidgets.dragdroptab), 150
- derived_units (labscript_utils.unitconversions.linear_coil_driver.BidirectionalCoilDriver dragdroptab.DragDropTabBar attribute), 378
- derived_units (labscript_utils.unitconversions.optotunelens.OptotuneLens dragdroptab.DragDropTabBar attribute), 379
- derived_units (labscript_utils.unitconversions.quad_driver.QuadDriver dragdroptab.DragDropTabBar attribute), 380
- derived_units (labscript_utils.unitconversions.quad_monitor.QuadMonitor dragdroptab.DragDropTabBar attribute), 381
- derived_units (labscript_utils.unitconversions.test.test attribute), 382
- deserialise() (in module labscript_utils.properties), 40
- detuned_MHz_from_base() (labscript_utils.unitconversions.example.example2 method), 375
- detuned_MHz_to_base() (labscript_utils.unitconversions.example.example2 method), 375
- detuning (class in labscript_utils.unitconversions.detuning), 373
- dict_diff() (in module labscript_utils.dict_diff), 11
- diff() (labscript_utils.connections.Connection method), 9
- DigitalOutput (class in labscript_utils.qtwidgets.digitaloutput), 100
- disable() (in module labscript_utils.double_import_denier), 11
- disable() (in module labscript_utils.impprof), 28
- disconnect_value_change() (labscript_utils.qtwidgets.analogoutput.AnalogOutput method), 70
- disconnect_value_change() (labscript_utils.qtwidgets.enumoutput.EnumOutput method), 215
- distance_from_base() (labscript_utils.unitconversions.optotunelens.OptotuneLens method), 379
- distance_to_base() (labscript_utils.unitconversions.optotunelens.OptotuneLens method), 379
- do_update_widget_positions() (labscript_utils.qtwidgets.headerview_with_widgets.HeaderviewWithWidgets method), 273
- dotdict (class in labscript_utils.testing_utils), 367
- DoubleImportDenier (class in labscript_utils.double_import_denier), 12
- drag_in_progress (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar property), 148
- DragDropTabBar (class in labscript_utils.qtwidgets.dragdroptab), 131
- DragDropTabWidget (class in labscript_utils.qtwidgets.dragdroptab), 150
- dragged_tab_index (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar property), 148
- dragged_tab_parent (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar property), 148
- duration() (labscript_utils.qtwidgets.dragdroptab.TabAnimation method), 170
- ## E
- elide_label() (in module labscript_utils.qtwidgets.elide_label), 171
- ElidedLabelContainer (class in labscript_utils.qtwidgets.elide_label), 188
- elideMode() (labscript_utils.qtwidgets.elide_label.ElidedLabelContainer method), 201
- ElideScrollArea (class in labscript_utils.qtwidgets.elide_label), 172
- enable() (in module labscript_utils.double_import_denier), 11
- enable() (in module labscript_utils.impprof), 28
- ensure_connected_to_zlog() (in module labscript_utils.ls_zprocess), 33
- ensure_running() (labscript_utils.qtwidgets.dragdroptab.TabAnimation method), 170
- ensure_visible() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 148
- EnumOutput (class in labscript_utils.qtwidgets.enumoutput), 202
- EnvInterpolation (class in labscript_utils.labconfig), 29
- ErrorWindow (class in labscript_utils.excepthook.tk_exception), 12
- Event() (in module labscript_utils.ls_zprocess), 32
- event() (labscript_utils.qtwidgets.elide_label.ElidedLabelContainer method), 201
- event() (labscript_utils.qtwidgets.elide_label.ElideScrollArea method), 187
- event() (labscript_utils.qtwidgets.toolpalette.ToolPalette method), 338
- eventFilter() (labscript_utils.qtwidgets.analogoutput.AnalogOutput method), 70

eventFilter() (*labscript_utils.qtwidgets.digitaloutput.DigitalOutput* method), 115
 eventFilter() (*labscript_utils.qtwidgets.enumoutput.EnumOutput* method), 215
 eventFilter() (*labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderViewWithWidgets.example1* method), 273
 eventFilter() (*labscript_utils.qtwidgets.imageoutput.BroadcastButton* method), 288
 eventFilter() (*labscript_utils.qtwidgets.imageoutput.ImageOutput* method), 302
 example1 (class in *labscript_utils.unitconversions.example*), 374
 example2 (class in *labscript_utils.unitconversions.example*), 375
 example3 (class in *labscript_utils.unitconversions.example*), 375
F
 File (class in *labscript_utils.h5_lock*), 25
 FileWatcher (class in *labscript_utils.filewatcher*), 23
 filter() (*labscript_utils.setup_logging.LessThanFilter* method), 347
 find_by_name() (*labscript_utils.connections.Connection* method), 9
 find_by_name() (*labscript_utils.connections.ConnectionTable* method), 10
 find_child() (*labscript_utils.connections.Connection* method), 9
 find_child() (*labscript_utils.connections.ConnectionTable* method), 10
 find_spec() (*labscript_utils.double_import_denier.DoubleImportDenier* method), 12
 FingerTabBarWidget (class in *labscript_utils.qtwidgets.fingertab*), 216
 FingerTabWidget (class in *labscript_utils.qtwidgets.fingertab*), 232
 FLUSH_GAP (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar* attribute), 148
 focusInEvent() (*labscript_utils.qtwidgets.analogoutput.NoStealFocusDoubleSpinBox* method), 86
 focusOutEvent() (*labscript_utils.qtwidgets.analogoutput.NoStealFocusDoubleSpinBox* method), 86
 fraction_from_base() (*labscript_utils.unitconversions.aom.SineAom* method), 372
 fraction_to_base() (*labscript_utils.unitconversions.aom.SineAom* method), 372
 FreqConversion (class in *labscript_utils.unitconversions.generic_frequency*), 376
G
 Gauss_from_base() (*labscript_utils.unitconversions.test.test* method), 374
 Gauss_to_base() (*labscript_utils.unitconversions.example.example1* method), 374
 Gauss_to_base() (*labscript_utils.unitconversions.test.test* method), 382
 Gcm_from_base() (*labscript_utils.unitconversions.quad_driver.quad_driver* method), 380
 Gcm_from_base() (*labscript_utils.unitconversions.quad_monitor.quad_monitor* method), 381
 Gcm_to_base() (*labscript_utils.unitconversions.quad_driver.quad_driver* method), 380
 Gcm_to_base() (*labscript_utils.unitconversions.quad_monitor.quad_monitor* method), 381
 get() (in module *labscript_utils.properties*), 40
 get_AI() (*labscript_utils.qtwidgets.analoginput.AnalogInput* method), 55
 get_AO() (*labscript_utils.qtwidgets.analogoutput.AnalogOutput* method), 70
 get_attached_devices() (*labscript_utils.connections.ConnectionTable* method), 10
 get_attribute() (in module *labscript_utils.properties*), 40
 get_attributes() (in module *labscript_utils.properties*), 40
 get_clean_modified_info() (*labscript_utils.filewatcher.FileWatcher* method), 24
 get_config() (in module *labscript_utils.ls_zprocess*), 33
 get_D0() (*labscript_utils.qtwidgets.digitaloutput.DigitalOutput* method), 115
 get_D0() (*labscript_utils.qtwidgets.enumoutput.EnumOutput* method), 215
 get_Image() (*labscript_utils.qtwidgets.imageoutput.ImageOutput* method), 302
 get_import_path() (in module *labscript_utils.versions*), 384
 get_index_from_name() (*labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup* method), 344

get_modified_info() (labscript_utils.filewatcher.FileWatcher method), 24
 get_modified_times() (labscript_utils.filewatcher.FileWatcher method), 24
 get_name_from_index() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
 get_palette() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
 get_palette_by_index() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
 get_shot_globals() (in module labscript_utils.shot_utils), 348
 get_sub_widget() (labscript_utils.qtwidgets.ddsoutput.DDSOutput method), 100
 get_unit_conversion_class() (in module labscript_utils.unitconversions), 383
 get_value() (labscript_utils.settings.Settings method), 346
 get_version() (in module labscript_utils.versions), 384
 GHz_from_base() (labscript_utils.unitconversions.generic_frequency.FreqConversion method), 377
 GHz_to_base() (labscript_utils.unitconversions.generic_frequency.FreqConversion method), 377

H

h (labscript_utils.splash.Splash attribute), 363
 hack_locks_onto_h5py() (in module labscript_utils.h5_lock), 25
 Handler() (in module labscript_utils.ls_zprocess), 32
 handler() (labscript_utils.camera_server.CameraServer method), 7
 hardware_from_base() (labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9m.AmpCoef method), 370
 hardware_to_base() (labscript_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9m.AmpCoef method), 370
 has_palette() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
 hide_palette() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
 hide_palette_by_index() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
 hide_sub_widget() (labscript_utils.qtwidgets.ddsoutput.DDSOutput method), 100
 hideSection() (labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderViewWithWidgets method), 273
 HorizontalHeaderViewWithWidgets (class in labscript_utils.qtwidgets.headerview_with_widgets), 248
 hostname() (in module labscript_profile), 388
 I_from_base() (labscript_utils.unitconversions.optotunelens.OptotuneLens method), 379
 I_to_base() (labscript_utils.unitconversions.optotunelens.OptotuneLens method), 379
 icon (labscript_utils.qtwidgets.dragdroptab.Tab property), 166
 icon_frac (labscript_utils.splash.Splash attribute), 363
 ImageOutput (class in labscript_utils.qtwidgets.imageoutput), 289
 imageUpdated (labscript_utils.qtwidgets.imageoutput.ImageOutput attribute), 302
 ImageView (class in labscript_utils.qtwidgets.imageoutput), 302
 imheight (labscript_utils.splash.Splash attribute), 363
 import_or_reload() (in module labscript_utils), 387
 imwidth (labscript_utils.splash.Splash attribute), 363
 insert_new_palette() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
 insertNewIndexBar() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 148
 insertWidget() (labscript_utils.qtwidgets.toolpalette.ToolPalette method), 338
 install_thread_excepthook() (in module labscript_utils.excepthook), 22
 instance() (labscript_utils.ls_zprocess.Context class method), 35
 instance() (labscript_utils.ls_zprocess.ProcessTree class method), 36
 instance() (labscript_utils.ls_zprocess.ZMQClient class method), 36
 InverseDigitalOutput (class in labscript_utils.qtwidgets.digitaloutput), 116
 is_dagged_tab() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 149
 is_json() (in module labscript_utils.properties), 40

K

kHz_from_base() (labscript_utils.unitconversions.generic_frequency.FreqConversion method), 377
 kHz_to_base() (labscript_utils.unitconversions.generic_frequency.FreqConversion method), 377

L

l (class in *labscript_utils.excepthook*), 23
LabConfig (class in *labscript_utils.labconfig*), 30
LabConfig.NoOptionError, 31
LabConfig.NoSectionError, 31
labscript_profile
 module, 388
labscript_profile.create
 module, 387
labscript_utils
 module, 386
labscript_utils.camera_server
 module, 6
labscript_utils.connections
 module, 8
labscript_utils.device_registry
 module, 11
labscript_utils.dict_diff
 module, 11
labscript_utils.double_import_denier
 module, 11
labscript_utils.excepthook
 module, 22
labscript_utils.excepthook.tk_exception
 module, 12
labscript_utils.filewatcher
 module, 23
labscript_utils.h5_lock
 module, 25
labscript_utils.impprof
 module, 28
labscript_utils.labconfig
 module, 28
labscript_utils.ls_zprocess
 module, 31
labscript_utils.memprof
 module, 37
labscript_utils.modulewatcher
 module, 38
labscript_utils.properties
 module, 39
labscript_utils.qtwidgets
 module, 345
labscript_utils.qtwidgets.analoginput
 module, 42
labscript_utils.qtwidgets.analogoutput
 module, 56
labscript_utils.qtwidgets.ddsoutput
 module, 86
labscript_utils.qtwidgets.digitaloutput
 module, 100
labscript_utils.qtwidgets.dragdroptab
 module, 131
labscript_utils.qtwidgets.elide_label
 module, 171
labscript_utils.qtwidgets.enumoutput
 module, 201
labscript_utils.qtwidgets.fingertab
 module, 216
labscript_utils.qtwidgets.headerview_with_widgets
 module, 248
labscript_utils.qtwidgets.imageoutput
 module, 273
labscript_utils.qtwidgets.InputPlotWindow
 module, 41
labscript_utils.qtwidgets.outputbox
 module, 321
labscript_utils.qtwidgets.toolpalette
 module, 322
labscript_utils.remote
 module, 345
labscript_utils.settings
 module, 345
labscript_utils.setup_logging
 module, 346
labscript_utils.shared_drive
 module, 347
labscript_utils.shot_utils
 module, 348
labscript_utils.splash
 module, 348
labscript_utils.testing_utils
 module, 363
labscript_utils.tracelog
 module, 368
labscript_utils.unitconversions
 module, 382
labscript_utils.unitconversions.aom
 module, 372
labscript_utils.unitconversions.detuning
 module, 373
labscript_utils.unitconversions.example
 module, 374
labscript_utils.unitconversions.generic_frequency
 module, 376
labscript_utils.unitconversions.linear_coil_driver
 module, 377
labscript_utils.unitconversions.NovaTechDDS9m
 module, 369
labscript_utils.unitconversions.optotunelens
 module, 379
labscript_utils.unitconversions.quad_driver
 module, 380
labscript_utils.unitconversions.quad_monitor
 module, 381
labscript_utils.unitconversions.test
 module, 382
labscript_utils.unitconversions.UnitConversionBase

module, 371
 labscript_utils.versions
 module, 383
 labscript_utils.zlock
 module, 386
 labscript_utils.zlog
 module, 386
 LessThanFilter (class in lab-
 script_utils.setup_logging), 347
 limbo (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar
 attribute), 149
 linewidths_from_base() (lab-
 script_utils.unitconversions.detuning.detuning
 method), 373
 linewidths_to_base() (lab-
 script_utils.unitconversions.detuning.detuning
 method), 374
 load() (labscript_utils.settings.Settings method), 346
 load_appconfig() (in module lab-
 script_utils.labconfig), 29
 Lock() (in module labscript_utils.ls_zprocess), 32
 lock() (labscript_utils.qtwidgets.analogoutput.AnalogOutput
 method), 70
 lock() (labscript_utils.qtwidgets.digitaloutput.DigitalOutput
 method), 115
 lock() (labscript_utils.qtwidgets.enumoutput.EnumOutput
 method), 215
 lock() (labscript_utils.qtwidgets.imageoutput.ImageOutput
 method), 302
 log() (in module labscript_utils.tracelog), 368
 logger (labscript_utils.excepthook.l attribute), 23
 logwarning() (in module labscript_utils.excepthook),
 22
M
 main() (in module labscript_utils.remote), 345
 main() (in module labscript_utils.zlock), 386
 main() (in module labscript_utils.zlog), 386
 mainloop() (labscript_utils.filewatcher.FileWatcher
 method), 24
 mainloop() (labscript_utils.modulewatcher.ModuleWatcher
 method), 39
 make_labconfig_file() (in module lab-
 script_profile.create), 388
 make_shared_secret() (in module lab-
 script_profile.create), 388
 MemoryProfiler (class in labscript_utils.memprof), 38
 MHz_from_base() (lab-
 script_utils.unitconversions.detuning.detuning
 method), 373
 MHz_from_base() (lab-
 script_utils.unitconversions.generic_frequency.FreqConversion
 method), 377
 MHz_from_base() (lab-
 script_utils.unitconversions.NovaTechDDS9m.NovaTechDDS9mF
 method), 370
 MHz_to_base() (labscript_utils.unitconversions.detuning.detuning
 method), 373
 MHz_to_base() (labscript_utils.unitconversions.generic_frequency.FreqC
 method), 377
 MHz_to_base() (labscript_utils.unitconversions.NovaTechDDS9m.NovaTe
 method), 370
 minimumSize() (labscript_utils.qtwidgets.toolpalette.ToolPalette
 method), 338
 minimumSizeHint() (lab-
 script_utils.qtwidgets.dragdroptab.DragDropTabBar
 method), 149
 minimumSizeHint() (lab-
 script_utils.qtwidgets.elide_label.ElidedLabelContainer
 method), 201
 minimumSizeHint() (lab-
 script_utils.qtwidgets.elide_label.ElidedScrollArea
 method), 187
 minimumSizeHint() (lab-
 script_utils.qtwidgets.toolpalette.ToolPalette
 method), 338
 module
 labscript_profile, 388
 labscript_profile.create, 387
 labscript_utils, 386
 labscript_utils.camera_server, 6
 labscript_utils.connections, 8
 labscript_utils.device_registry, 11
 labscript_utils.dict_diff, 11
 labscript_utils.double_import_denier, 11
 labscript_utils.excepthook, 22
 labscript_utils.excepthook.tk_exception,
 12
 labscript_utils.filewatcher, 23
 labscript_utils.h5_lock, 25
 labscript_utils.impprof, 28
 labscript_utils.labconfig, 28
 labscript_utils.ls_zprocess, 31
 labscript_utils.memprof, 37
 labscript_utils.modulewatcher, 38
 labscript_utils.properties, 39
 labscript_utils.qtwidgets, 345
 labscript_utils.qtwidgets.analoginput, 42
 labscript_utils.qtwidgets.analogoutput,
 56
 labscript_utils.qtwidgets.ddsoutput, 86
 labscript_utils.qtwidgets.digitaloutput,
 100
 labscript_utils.qtwidgets.dragdroptab,
 171
 labscript_utils.qtwidgets.elide_label,
 171

- labscript_utils.qtwidgets.enumoutput, 201
 - labscript_utils.qtwidgets.fingertab, 216
 - labscript_utils.qtwidgets.headerview_with_widgets, 248
 - labscript_utils.qtwidgets.imageoutput, 273
 - labscript_utils.qtwidgets.InputPlotWindow, 41
 - labscript_utils.qtwidgets.outputbox, 321
 - labscript_utils.qtwidgets.toolpalette, 322
 - labscript_utils.remote, 345
 - labscript_utils.settings, 345
 - labscript_utils.setup_logging, 346
 - labscript_utils.shared_drive, 347
 - labscript_utils.shot_utils, 348
 - labscript_utils.splash, 348
 - labscript_utils.testing_utils, 363
 - labscript_utils.tracelog, 368
 - labscript_utils.unitconversions, 382
 - labscript_utils.unitconversions.aom, 372
 - labscript_utils.unitconversions.detuning, 373
 - labscript_utils.unitconversions.example, 374
 - labscript_utils.unitconversions.generic_frequency, 376
 - labscript_utils.unitconversions.linear_coil_driver, 377
 - labscript_utils.unitconversions.NovaTechDDS9m, 369
 - labscript_utils.unitconversions.optotunelens, 379
 - labscript_utils.unitconversions.quad_driver, 380
 - labscript_utils.unitconversions.quad_monitor, 381
 - labscript_utils.unitconversions.test, 382
 - labscript_utils.unitconversions.UnitConversionBase, 371
 - labscript_utils.versions, 383
 - labscript_utils.zlock, 386
 - labscript_utils.zlog, 386
 - ModuleWatcher (class in labscript_utils.modulewatcher), 39
 - monkeypatch (class in labscript_utils.testing_utils), 368
 - mouseMoveEvent() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 149
 - mousePressEvent() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 149
 - mouseReleaseEvent() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 149
 - NoStealFocusDoubleSpinBox (class in labscript_utils.qtwidgets.analogoutput), 70
 - NotFound (class in labscript_utils.versions), 385
 - NovaTechDDS9mAmpConversion (class in labscript_utils.unitconversions.NovaTechDDS9m), 369
 - NovaTechDDS9mFreqConversion (class in labscript_utils.unitconversions.NovaTechDDS9m), 370
 - NoVersionInfo (class in labscript_utils.versions), 385
- ## O
- ok_clicked() (labscript_utils.excepthook.tk_exception.ErrorWindow method), 22
 - on_cancel() (labscript_utils.settings.Settings method), 346
 - on_columnsInserted() (labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeader method), 273
 - on_columnsRemoved() (labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeader method), 273
 - on_save() (labscript_utils.settings.Settings method), 346
 - on_scroll_button_clicked() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 149
 - on_tab_moved() (labscript_utils.qtwidgets.dragdroptab.TabAnimation method), 170
 - open_plot_window() (labscript_utils.qtwidgets.analoginput.AnalogInput method), 55
 - OptotuneLens (class in labscript_utils.unitconversions.optotunelens), 379
 - OutputBox (class in labscript_utils.qtwidgets.outputbox), 321
- ## P
- paint_tab() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 149
 - paintEvent() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar method), 149
 - paintEvent() (labscript_utils.qtwidgets.fingertab.FingerTabBarWidget method), 232
 - paintEvent() (labscript_utils.splash.Splash method), 363
 - path_to_agnostic() (in module labscript_utils.shared_drive), 348

- path_to_local() (in module labscript_utils.shared_drive), 348
- PlotWindow (class in labscript_utils.qtwidgets.InputPlotWindow), 41
- Power_from_base() (labscript_utils.unitconversions.aom.SineAom method), 372
- Power_to_base() (labscript_utils.unitconversions.aom.SineAom method), 372
- print_details() (labscript_utils.connections.Connection method), 9
- print_details() (labscript_utils.connections.ConnectionTable method), 10
- ProcessTree (class in labscript_utils.ls_zprocess), 35
- properties (labscript_utils.connections.Connection property), 9
- ## Q
- quad_driver (class in labscript_utils.unitconversions.quad_driver), 380
- quad_monitor (class in labscript_utils.unitconversions.quad_monitor), 381
- quit_mainloop() (labscript_utils.testing_utils.ThreadTestCase method), 367
- ## R
- register_callback() (labscript_utils.settings.Settings method), 346
- RemoteProcessClient() (in module labscript_utils.ls_zprocess), 32
- remove() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
- remove_by_index() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
- remove_callback() (labscript_utils.settings.Settings method), 346
- remove_device() (labscript_utils.connections.ConnectionTable method), 10
- remove_from_linked_width_group() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 344
- reorder_palette() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 345
- reorder_palette_by_index() (labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup method), 345
- resizeEvent() (labscript_utils.qtwidgets.toolpalette.ToolPalette method), 338
- run() (labscript_utils.qtwidgets.InputPlotWindow.PlotWindow method), 42
- run() (labscript_utils.testing_utils.ThreadTestCase method), 367
- run_script_as_main() (labscript_utils.testing_utils.ThreadTestCase method), 367
- ## S
- save_appconfig() (in module labscript_utils.labconfig), 29
- SCROLL_BUTTON_GAP (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar attribute), 148
- SCROLL_BUTTON_WIDTH (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar attribute), 148
- setSizeFromContents() (labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderView method), 273
- selected_index (labscript_utils.qtwidgets.enumoutput.EnumOutput property), 215
- selected_option (labscript_utils.qtwidgets.enumoutput.EnumOutput property), 215
- selected_unit (labscript_utils.qtwidgets.analogoutput.AnalogOutput property), 70
- serialise() (in module labscript_utils.properties), 40
- set_AI() (labscript_utils.qtwidgets.analoginput.AnalogInput method), 55
- set_A0() (labscript_utils.qtwidgets.analogoutput.AnalogOutput method), 70
- set_attributes() (in module labscript_utils.properties), 40
- set_combobox_model() (labscript_utils.qtwidgets.analogoutput.AnalogOutput method), 70
- set_combobox_model() (labscript_utils.qtwidgets.enumoutput.EnumOutput method), 215
- set_device_properties() (in module labscript_utils.properties), 40
- set_D0() (labscript_utils.qtwidgets.digitaloutput.DigitalOutput method), 115
- set_E0() (labscript_utils.qtwidgets.enumoutput.EnumOutput method), 215
- set_Image() (labscript_utils.qtwidgets.imageoutput.ImageOutput method), 302

set_limits() (*labscript_utils.qtwidgets.analogoutput.AnalogOutputscript_utils.qtwidgets.toolpalette.ToolPaletteGroup*
method), 70 *method*), 345
 set_logger() (*in module labscript_utils.excepthook*), 22 show_palette_by_index() (*lab-*
script_utils.qtwidgets.toolpalette.ToolPaletteGroup
method), 345
 set_num_decimals() (*lab-* show_sub_widget() (*lab-*
script_utils.qtwidgets.analogoutput.AnalogOutput *method*), 70 *script_utils.qtwidgets.ddsoutput.DDSOutput*
method), 100
 set_selected_unit() (*lab-* showEvent() (*labscript_utils.qtwidgets.headerview_with_widgets.Horizontal*
method), 70 *method*), 273
 set_spinbox_value() (*lab-* showSection() (*labscript_utils.qtwidgets.headerview_with_widgets.Horizontal*
script_utils.qtwidgets.analogoutput.AnalogOutput *method*), 70 *method*), 273
method), 70 SineAom (*class in labscript_utils.unitconversions.aom*),
 set_step_size() (*lab-* 372
script_utils.qtwidgets.analogoutput.AnalogOutputsizeHint() (labscript_utils.qtwidgets.dragdroptab.DragDropTabBar
method), 70 *method*), 149
 set_tab_parent() (*lab-* sizeHint() (*labscript_utils.qtwidgets.elide_label.ElidedLabelContainer*
script_utils.qtwidgets.dragdroptab.DragDropTabBar *method*), 149 *method*), 201
method), 149 sizeHint() (*labscript_utils.qtwidgets.elide_label.ElideScrollArea*
 set_value() (*labscript_utils.qtwidgets.analoginput.AnalogInput* *method*), 188
method), 55 sizeHint() (*labscript_utils.qtwidgets.toolpalette.ToolPalette*
 setElideMode() (*lab-* *method*), 338
script_utils.qtwidgets.dragdroptab.DragDropTabWidget widget() (*labscript_utils.ls_zprocess.Context* *method*),
method), 165 *method*), 35
 setElideMode() (*lab-* sockopts (*labscript_utils.ls_zprocess.Context* *attribute*),
script_utils.qtwidgets.elide_label.ElidedLabelContainer *method*), 201 *method*), 35
method), 201 Splash (*class in labscript_utils.splash*), 348
 setElideMode() (*lab-* start() (*in module labscript_utils.memprof*), 38
script_utils.qtwidgets.elide_label.ElideScrollArea start() (*labscript_utils.memprof.MemoryProfiler*
method), 187 *method*), 38
method), 187
 setSectionHidden() (*lab-* state (*labscript_utils.qtwidgets.digitaloutput.DigitalOutput*
script_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderWidgetWithWidgets
method), 273 *method*), 348
method), 273 state (*labscript_utils.qtwidgets.digitaloutput.InvertedDigitalOutput*
 setStyleSheet() (*lab-* *property*), 131
script_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderWidgetWithWidgets *method*), 273
method), 273 stop() (*labscript_utils.ls_zprocess.Context* *attribute*), 24
 Settings (*class in labscript_utils.settings*), 345 stylesheet (*labscript_utils.qtwidgets.headerview_with_widgets.Horizontal*
 setTopLevelWindow() (*lab-* *attribute*), 273
script_utils.qtwidgets.InputPlotWindow.PlotWindow
method), 42 **T**
 setup_logging() (*in module lab-* Tab (*class in labscript_utils.qtwidgets.dragdroptab*), 165
script_utils.setup_logging), 347 *method*), 165
 setUsesScrollButtons() (*lab-* tab_widgets (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar*
script_utils.qtwidgets.dragdroptab.DragDropTabBar *attribute*), 149 *method*), 149
method), 149 TabAnimation (*class in lab-*
method), 149 *script_utils.qtwidgets.dragdroptab*), 167
 setUsesScrollButtons() (*lab-* tabAt() (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar*
script_utils.qtwidgets.dragdroptab.DragDropTabWidget *method*), 165 *method*), 149
method), 165 tabInserted() (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar*
 setWidget() (*labscript_utils.qtwidgets.elide_label.ElideScrollArea* *method*), 149
method), 187 *method*), 149
 setWidget() (*labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderWidgetWithWidgets*
method), 273 *method*), 273
 show() (*labscript_utils.splash.Splash* *method*), 363 tabInserted() (*labscript_utils.qtwidgets.dragdroptab.TabAnimation*
 show_palette() (*lab-* *method*), 149
method), 149 tabLayoutChange() (*lab-*
method), 149 *method*), 149
method), 149 *method*), 149

tabRect() (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar* method), 70
 method), 149 *UnidirectionalCoilDriver* (class in *lab-*
 tabRemoved() (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar* method), 149 *script_utils.unitconversions.linear_coil_driver*),
 378
 tabRemoved() (*labscript_utils.qtwidgets.dragdroptab.TabAnimation* method), 170 *unitconversion_params* (lab-
 script_utils.connections.Connection property),
 tabSizeHint() (*labscript_utils.qtwidgets.fingertab.FingerTabBarWidget* method), 232 *unit_list* (*labscript_utils.unitconversions.UnitConversionBase.UnitConversion*)
 target() (*labscript_utils.qtwidgets.dragdroptab.TabAnimation* attribute), 371 *UnitConversion* (class in *lab-*
 method), 170 *script_utils.unitconversions.UnitConversionBase*),
 tau (*labscript_utils.qtwidgets.dragdroptab.TabAnimation* attribute), 170 371
 test (class in *labscript_utils.unitconversions.test*), 382
 text (*labscript_utils.qtwidgets.dragdroptab.Tab* prop-
 erty), 166
 text_color (*labscript_utils.qtwidgets.dragdroptab.Tab*
 property), 166
 thinspace (*labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeadViewWithWidgets*
 attribute), 273
 ThreadTestCase (class in *labscript_utils.testing_utils*),
 364
 tkhandler() (in module *labscript_utils.excepthook*), 22
 ToolPalette (class in *lab-*
 script_utils.qtwidgets.toolpalette), 322
 ToolPaletteGroup (class in *lab-*
 script_utils.qtwidgets.toolpalette), 338
 tooltip (*labscript_utils.qtwidgets.dragdroptab.Tab*
 property), 166
 trace() (*labscript_utils.qtwidgets.dragdroptab.debug*
 class method), 171
 transition_to_buffered() (lab-
 script_utils.camera_server.CameraServer
 method), 7
 transition_to_buffered() (lab-
 script_utils.camera_server.TubingenCameraServer
 method), 8
 transition_to_static() (lab-
 script_utils.camera_server.CameraServer
 method), 7
 transition_to_static() (lab-
 script_utils.camera_server.TubingenCameraServer
 method), 8
 TubingenCameraServer (class in *lab-*
 script_utils.camera_server), 7

U

unblock_combobox_signals() (lab-
 script_utils.qtwidgets.analogoutput.AnalogOutput
 method), 70
 unblock_combobox_signals() (lab-
 script_utils.qtwidgets.enumoutput.EnumOutput
 method), 215
 unblock_spinbox_signals() (lab-
 script_utils.qtwidgets.analogoutput.AnalogOutput
 method), 70
 update_files() (lab-
 script_utils.filewatcher.FileWatcher method),
 25
 update_indents() (lab-
 script_utils.qtwidgets.headerview_with_widgets.HorizontalHeadViewWithWidgets
 method), 273
 update_plot() (*labscript_utils.qtwidgets.InputPlotWindow.PlotWindow*
 method), 42
 update_scroll_button_state() (lab-
 script_utils.qtwidgets.dragdroptab.DragDropTabBar
 method), 149
 update_tab_index() (lab-
 script_utils.qtwidgets.dragdroptab.DragDropTabBar
 method), 149
 update_text() (*labscript_utils.splash.Splash* method),
 363
 update_widget_positions() (lab-
 script_utils.qtwidgets.headerview_with_widgets.HorizontalHeadViewWithWidgets
 method), 273
 updateCurrentTime() (lab-
 script_utils.qtwidgets.dragdroptab.TabAnimation
 method), 170
 updateMinimumSize() (lab-
 script_utils.qtwidgets.toolpalette.ToolPalette

method), 338

V

value (*labscript_utils.qtwidgets.imageoutput.ImageOutput* property), 302

vectorise() (in module *labscript_utils.unitconversions.UnitConversionBase*), 371

VersionException, 386

viewportEvent() (*labscript_utils.qtwidgets.headerview_with_widgets.HorizontalHeaderViewWithWidgets* method), 273

W

w (*labscript_utils.splash.Splash* attribute), 363

W_from_base() (*labscript_utils.unitconversions.example.example3* method), 376

W_to_base() (*labscript_utils.unitconversions.example.example3* method), 376

wait_for() (*labscript_utils.testing_utils.ThreadTestCase* static method), 367

whats_this (*labscript_utils.qtwidgets.dragdroptab.Tab* property), 166

wheelEvent() (*labscript_utils.qtwidgets.analogoutput.NoStealFocusDoubleSpinBox* method), 86

widget (*labscript_utils.qtwidgets.dragdroptab.Tab* property), 166

widgetAt() (*labscript_utils.qtwidgets.dragdroptab.DragDropTabBar* method), 149

widths_linked (*labscript_utils.qtwidgets.toolpalette.ToolPaletteGroup* property), 345

write_to_file() (*labscript_utils.memprof.MemoryProfiler* method), 38

Z

zmq_get() (in module *labscript_utils.ls_zprocess*), 33

zmq_get_multipart() (in module *labscript_utils.ls_zprocess*), 33

zmq_get_raw() (in module *labscript_utils.ls_zprocess*), 33

zmq_get_string() (in module *labscript_utils.ls_zprocess*), 33

zmq_push() (in module *labscript_utils.ls_zprocess*), 33

zmq_push_multipart() (in module *labscript_utils.ls_zprocess*), 33

zmq_push_raw() (in module *labscript_utils.ls_zprocess*), 34

zmq_push_string() (in module *labscript_utils.ls_zprocess*), 34

ZMQClient (class in *labscript_utils.ls_zprocess*), 36

ZMQServer (class in *labscript_utils.ls_zprocess*), 37