electrotestgtk

0.99

Generated by Doxygen 1.8.11

Contents

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Data Structure Documentation

3.1 gui_comp Struct Reference

Struct with pointers to all gui and data parts.

```
#include <electrotestgtk.h>
```

Data Fields

- GtkWidget ∗ voltage_box
- GtkWidget ∗ coupling_box
- GtkWidget ∗ resistor_box
- GtkWidget ∗ calc_result_box
- float ∗ resistor_values

3.1.1 Detailed Description

Struct with pointers to all gui and data parts.

The struct contains pointers to the three GtkWidget gui parts and the float array that contains the current resistor values.

3.1.2 Field Documentation

3.1.2.1 GtkWidget∗ gui_comp::calc_result_box

3.1.2.2 GtkWidget∗ gui_comp::coupling_box

3.1.2.3 GtkWidget∗ gui_comp::resistor_box

3.1.2.4 float∗ gui_comp::resistor_values

3.1.2.5 GtkWidget∗ gui_comp::voltage_box

The documentation for this struct was generated from the following file:

- electrotestgtk.h

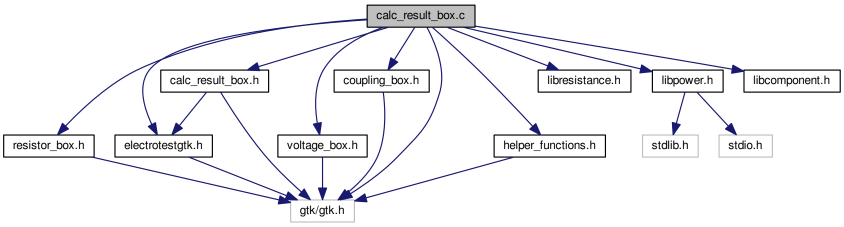
Chapter 4

File Documentation

4.1 calc_result_box.c File Reference

```
#include <gtk/gtk.h>
#include "electrotestgtk.h"
#include "helper_functions.h"
#include "resistor_box.h"
#include "voltage_box.h"
#include "coupling_box.h"
#include "calc_result_box.h"
#include "libresistance.h"
#include "libpower.h"
#include "libcomponent.h"
```

Include dependency graph for calc_result_box.c:



Functions

- void button_clicked (GtkWidget ∗button, struct gui_comp ∗gui)

    callback function for calculation button
- GtkWidget ∗ calc_result_box_new (struct gui_comp ∗gui)

    Constructor of the lower GUI part.

Variables

- GtkWidget ∗ result_power
- GtkWidget ∗ result_resistance

4.1.1 Function Documentation

4.1.1.1 void button_clicked (GtkWidget ∗ *button,* struct gui_comp ∗ *gui*)

callback function for calculation button

Function uses the electrotest libraries to calculate the replacement resistor and power. Currently, no additional error handling over the libraries is implemented. The function takes a struct with pointers to the relevant gui parts and the data array that contains the resistor values.

Parameters

button	The button widget to which the callback relates.
gui	A struct that contains pointers to all gui parts and to the array which contains the resistor values.

4.1.1.2 GtkWidget∗ calc_result_box_new (struct gui_comp ∗ *gui*)

Constructor of the lower GUI part.

This part of the GUI contains the calculation button and the result boxes.

Parameters

gui	A struct that contains pointers to all gui parts and to the array which contains the resistor values.

Returns

GtkWidget∗ Pointer to the constructed and wired-up GUI part

4.1.2 Variable Documentation

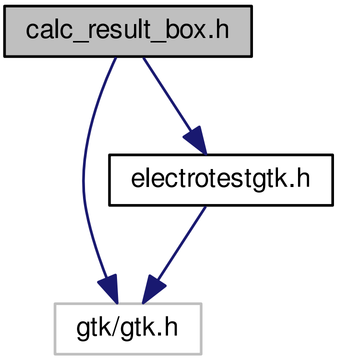
4.1.2.1 GtkWidget∗ result_power

4.1.2.2 GtkWidget∗ result_resistance

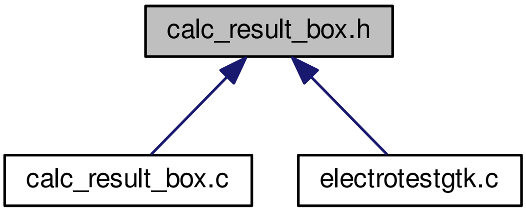
4.2 calc_result_box.h File Reference

```
#include <gtk/gtk.h>
#include "electrotestgtk.h"
```

Include dependency graph for calc_result_box.h:



This graph shows which files directly or indirectly include this file:



Functions

- GtkWidget ∗ calc_result_box_new (struct gui_comp ∗gui)

  Constructor of the lower GUI part.

4.2.1 Function Documentation

4.2.1.1 GtkWidget∗ calc_result_box_new (struct **gui_comp** ∗ *gui*)

Constructor of the lower GUI part.

This part of the GUI contains the calculation button and the result boxes.

Parameters

gui	A struct that contains pointers to all gui parts and to the array which contains the resistor values.

Returns

calc_result_box The constructed and wired GUI part

This part of the GUI contains the calculation button and the result boxes.

Parameters

gui	A struct that contains pointers to all gui parts and to the array which contains the resistor values.

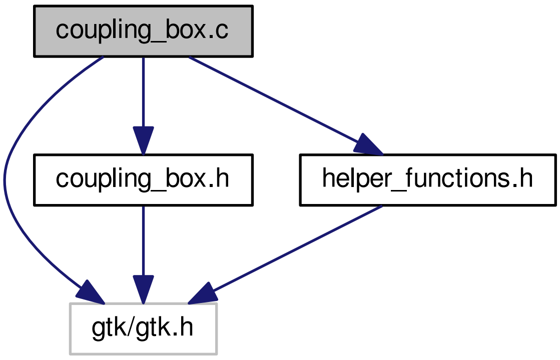
Returns

GtkWidget∗ Pointer to the constructed and wired-up GUI part

4.3 coupling_box.c File Reference

```
#include <gtk/gtk.h>
#include "coupling_box.h"
#include "helper_functions.h"
```

Include dependency graph for coupling_box.c:



Functions

- GtkWidget ∗ coupling_box_new (void)

  Constructor for the Resistor Coupling Gui part.
- int get_coupling (GtkWidget ∗coupling_box)

  Getter for mode of resistor coupling.

4.3.1 Function Documentation

4.3.1.1 GtkWidget∗ coupling_box_new (void)

Constructor for the Resistor Coupling Gui part.

The Resistor coupling GUI part allows to choose between serial/parallel coupling of the resistors.

Returns

GtkWidget∗ pointer to the new created Gui part.

4.3.1.2 int get_coupling (GtkWidget ∗ *coupling_box*)

Getter for mode of resistor coupling.

Function that queries the selected resitor coupling mode in the gui and returns 1 for serial and 0 for parallel.

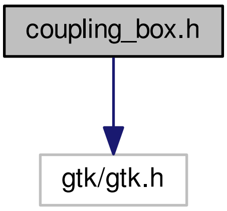
Returns

1 = serial, 0 = parallel

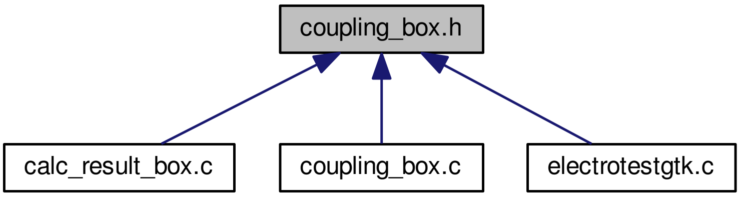
4.4 coupling_box.h File Reference

```
#include <gtk/gtk.h>
```

Include dependency graph for coupling_box.h:



This graph shows which files directly or indirectly include this file:



Functions

- GtkWidget ∗ coupling_box_new (void)

  Constructor for the Resistor Coupling Gui part.
- int get_coupling (GtkWidget ∗coupling_box)

  Getter for mode of resistor coupling.

4.4.1 Function Documentation

4.4.1.1 GtkWidget∗ coupling_box_new (void)

Constructor for the Resistor Coupling Gui part.

The Resistor coupling GUI part allows to choose between serial/parallel coupling of the resistors.

Returns

GtkWidget∗ pointer to the new created Gui part.

4.4.1.2 int get_coupling (GtkWidget ∗ *coupling_box*)

Getter for mode of resistor coupling.

Function that queries the selected resitor coupling mode in the gui and returns 1 for serial and 0 for parallel.

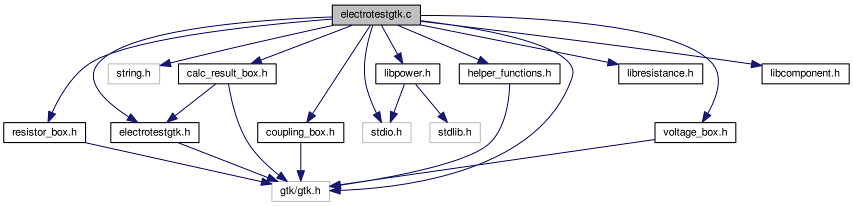
Returns

1 = serial, 0 = parallel

4.5 electrotestgtk.c File Reference

```
#include <gtk/gtk.h>
#include <stdio.h>
#include <string.h>
#include "electrotestgtk.h"
#include "voltage_box.h"
#include "resistor_box.h"
#include "coupling_box.h"
#include "calc_result_box.h"
#include "helper_functions.h"
#include "libresistance.h"
#include "libpower.h"
#include "libcomponent.h"
```

Include dependency graph for electrotestgtk.c:



Functions

- void closeApp (GtkWidget ∗window, gpointer data)

  closeApp callback function
- gint main (gint argc, gchar ∗argv[ ])

  Main of electrotestgtk Application.

4.5.1 Function Documentation

4.5.1.1 void closeApp (GtkWidget ∗ *window,* gpointer *data*)

closeApp callback function

Function to close main GTK application window.

Parameters

window,pointer	to the main window widget
data,pointer	for extra data, not used

4.5.1.2 gint main (gint *argc,* gchar ∗ *argv[ ]*)

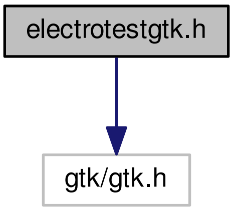
Main of electrotestgtk Application.

Application GTK+ gui frontend for electrotest libraries. The Main file constructs the GUI by calling the three respective constructor functions. Interfacing to the libraries happens mostly in the calc_results_box gui part. Currently the application does not implement advanced user input validation and relies on the functionality in the libraries. ∗

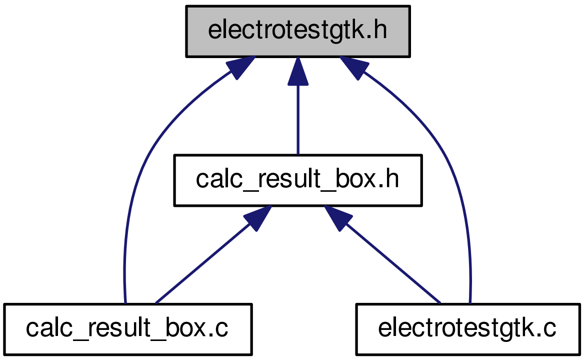
4.6 electrotestgtk.h File Reference

```
#include <gtk/gtk.h>
```

Include dependency graph for electrotestgtk.h:



This graph shows which files directly or indirectly include this file:



Data Structures

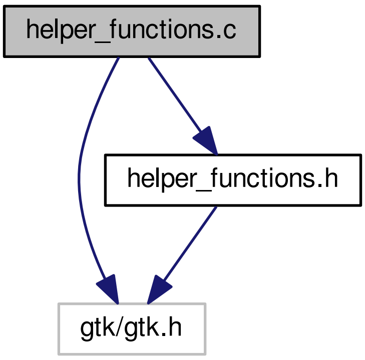
- struct gui_comp

  Struct with pointers to all gui and data parts.

4.7 helper_functions.c File Reference

```
#include <gtk/gtk.h>
#include "helper_functions.h"
```

Include dependency graph for helper_functions.c:



Functions

- GtkWidget ∗ add_widget_with_label_vbox (GtkContainer ∗box, gchar ∗caption, GtkWidget ∗widget)

  Function constructs a label above a provided widget.
- GtkWidget ∗ add_widget_with_label_hbox (GtkContainer ∗box, gchar ∗caption, GtkWidget ∗widget)

  Function constructs a label to the left of a provided widget.
- GtkWidget ∗ gtk_entry_with_name_new (gchar ∗name)

  Helper to create entry widget with a name.
- GtkWidget ∗ find_child (GtkWidget ∗parent, const gchar ∗name)

  Find recursively child widgets of a GTK container by name.

4.7.1 Function Documentation

4.7.1.1 GtkWidget∗ add_widget_with_label_hbox (GtkContainer ∗ *box,* gchar ∗ *caption,* GtkWidget ∗ *widget*)

Function constructs a label to the left of a provided widget.

The function creates a new hbox and a label. The provided widget is put in the hbox together with the new label which results in a label to the left of the widget. The hbox is then put into the provided container/box.

Parameters

box	Gtk Container into which the hbox with label and widget shall be added
caption	String used to construct the caption
widget	widget to which a label shall be constructed

Returns

GtkWidget∗ returns pointer to the given box/container. This is not always needed but sometimes of advantage when multiple widgets shall be constructed in a loop with a deeply nested constructor.

4.7.1.2 GtkWidget∗ add_widget_with_label_vbox (GtkContainer ∗ *box,* gchar ∗ *caption,* GtkWidget ∗ *widget*)

Function constructs a label above a provided widget.

The function creates a new vbox and a label. The provided widget is put in the vbox together with the new label which results in a label above the widget. The vbox is then put into the provided container/box.

Parameters

box	Gtk Container into which the vbox with label and widget shall be added
caption	String used to construct the caption
widget	widget to which a label shall be constructed

Returns

GtkWidget∗ returns pointer to the given box/container. This is not always needed but sometimes of advantage when multiple widgets shall be constructed in a loop with a deeply nested constructor.

4.7.1.3 GtkWidget∗ find_child (GtkWidget ∗ *parent,* const gchar ∗ *name*)

Find recursively child widgets of a GTK container by name.

Recursive search of child widgets in a GTK container structure. The function returns, when found a pointer to the respective widget, else NULL.

Parameters

parent	Widget to start the search at.
name	String of the name to search for

Returns

pointer to the found widget or NULL

4.7.1.4 GtkWidget∗ gtk_entry_with_name_new (gchar ∗ *name*)

Helper to create entry widget with a name.

Naming widgets is handy as they can be searched easily with find_child() also provided in this helper function collection.

Parameters

name	string on how to name the new entry widget

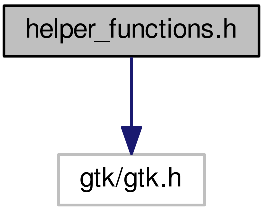
Returns

GtkWidget∗ pointer to the constructed and named gtk_entry widget

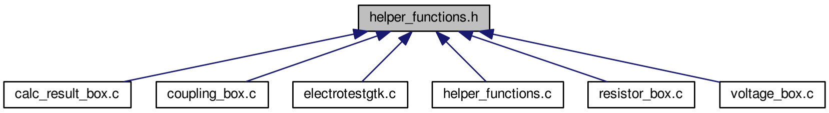
4.8 helper_functions.h File Reference

```
#include <gtk/gtk.h>
```

Include dependency graph for helper_functions.h:



This graph shows which files directly or indirectly include this file:



Functions

- GtkWidget ∗ add_widget_with_label_vbox (GtkContainer ∗box, gchar ∗caption, GtkWidget ∗widget)

  Function constructs a label above a provided widget.
- GtkWidget ∗ add_widget_with_label_hbox (GtkContainer ∗box, gchar ∗caption, GtkWidget ∗widget)

  Function constructs a label to the left of a provided widget.
- GtkWidget ∗ find_child (GtkWidget ∗parent, const gchar ∗name)

  Find recursively child widgets of a GTK container by name.
- GtkWidget ∗ gtk_entry_with_name_new (gchar ∗name)

  Helper to create entry widget with a name.

4.8.1 Function Documentation

4.8.1.1 GtkWidget∗ add_widget_with_label_hbox (GtkContainer ∗ *box,* gchar ∗ *caption,* GtkWidget ∗ *widget*)

Function constructs a label to the left of a provided widget.

The function creates a new hbox and a label. The provided widget is put in the hbox together with the new label which results in a label to the left of the widget. The hbox is then put into the provided container/box.

Parameters

box	Gtk Container into which the hbox with label and widget shall be added
caption	String used to construct the caption
widget	widget to which a label shall be constructed

Returns

GtkWidget∗ returns pointer to the given box/container. This is not always needed but sometimes of advantage when multiple widgets shall be constructed in a loop with a deeply nested constructor.

4.8.1.2 GtkWidget∗ add_widget_with_label_vbox (GtkContainer ∗ *box,* gchar ∗ *caption,* GtkWidget ∗ *widget*)

Function constructs a label above a provided widget.

The function creates a new vbox and a label. The provided widget is put in the vbox together with the new label which results in a label above the widget. The vbox is then put into the provided container/box.

Parameters

box	Gtk Container into which the vbox with label and widget shall be added
caption	String used to construct the caption
widget	widget to which a label shall be constructed

Returns

GtkWidget∗ returns pointer to the given box/container. This is not always needed but sometimes of advantage when multiple widgets shall be constructed in a loop with a deeply nested constructor.

4.8.1.3 GtkWidget∗ find_child (GtkWidget ∗ *parent,* const gchar ∗ *name*)

Find recursively child widgets of a GTK container by name.

Recursive search of child widgets in a GTK container structure. The function returns, when found a pointer to the respective widget, else NULL.

Parameters

parent	Widget to start the search at.
name	String of the name to search for

Returns

pointer to the found widget or NULL

4.8.1.4 GtkWidget∗ gtk_entry_with_name_new (gchar ∗ *name*)

Helper to create entry widget with a name.

Naming widgets is handy as they can be searched easily with find_child() also provided in this helper function collection.

Parameters

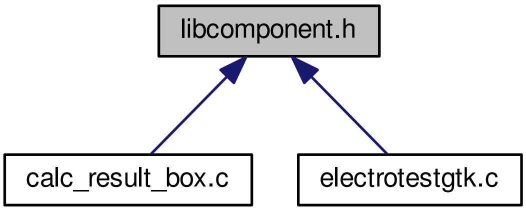
name	string on how to name the new entry widget

Returns

GtkWidget∗ pointer to the constructed and named gtk_entry widget

4.9 libcomponent.h File Reference

This graph shows which files directly or indirectly include this file:



Functions

- int findresistors (float orig_resistance, float ∗res_array)

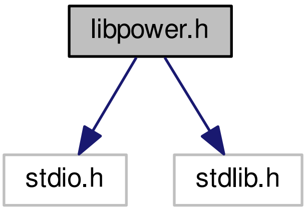
4.9.1 Function Documentation

4.9.1.1 int findresistors (float *orig_resistance,* float ∗ *res_array*)

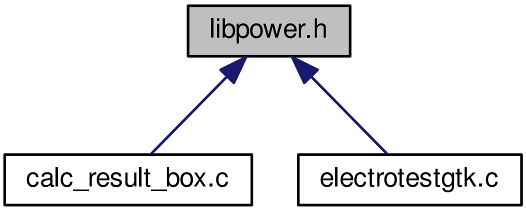
4.10 libpower.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
```

Include dependency graph for libpower.h:



This graph shows which files directly or indirectly include this file:



Functions

- float calc_power_r (float volt, float resistance)
- float calc_power_i (float volt, float current)

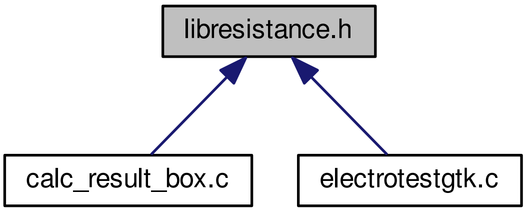
4.10.1 Function Documentation

4.10.1.1 float calc_power_i (float *volt,* float *current*)

4.10.1.2 float calc_power_r (float *volt,* float *resistance*)

4.11 libresistance.h File Reference

This graph shows which files directly or indirectly include this file:



Functions

- float calc_resistance (int count, char conn, float ∗array)

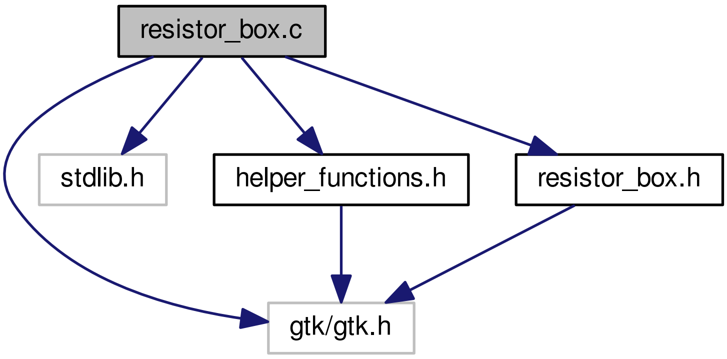
4.11.1 Function Documentation

4.11.1.1 float calc_resistance (int *count,* char *conn,* float ∗ *array*)

4.12 resistor_box.c File Reference

```
#include <gtk/gtk.h>
#include <stdlib.h>
#include "helper_functions.h"
#include "resistor_box.h"
```

Include dependency graph for resistor_box.c:



Functions

- GtkWidget ∗ resistor_box_new (void)

  Constructor for resistor_box GUI part.
- gfloat ∗ update_resistor_values (GtkWidget ∗resistor_box, gfloat ∗value_array)

  Updates the value_array with values from the GUI.

Variables

- char ∗const resistor_names [ ] = { "res1", "res2", "res3" }
- char ∗const resistor_labels [ ] = {"1:", "2:", "3:"}

4.12.1 Function Documentation

4.12.1.1 GtkWidget∗ resistor_box_new (void)

Constructor for resistor_box GUI part.

This GUI part contains three resistor input field.

Returns

GtkWidget Contains the constructed GUI part.

4.12.1.2 gfloat∗ update_resistor_values (GtkWidget ∗ *resistor_box*, gfloat ∗ *value_array*)

Updates the value_array with values from the GUI.

The Function assumes a gfloat array of size 3.

Parameters

resistor_box	The GUI part to access the user input. Float array of size 3 that contains the resistor values

4.12.2 Variable Documentation

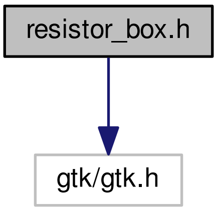
4.12.2.1 char∗ const resistor_labels[ ] = {"1:", "2:", "3:"}

4.12.2.2 char∗ const resistor_names[ ] = { "res1", "res2", "res3" }

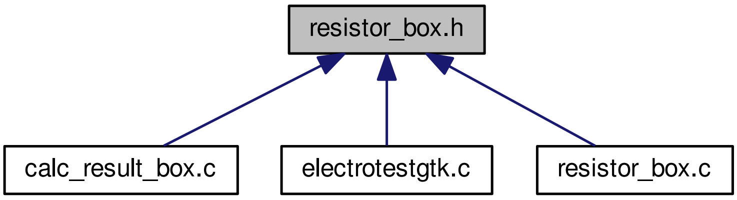
4.13 resistor_box.h File Reference

```
#include <gtk/gtk.h>
```

Include dependency graph for resistor_box.h:



This graph shows which files directly or indirectly include this file:



Functions

- GtkWidget ∗ resistor_box_new (void)

  Constructor for resistor_box GUI part.
- float ∗ update_resistor_values (GtkWidget ∗resistor_box, float ∗value_array)

  Updates the value_array with values from the GUI.

4.13.1 Function Documentation

4.13.1.1 GtkWidget∗ resistor_box_new (void)

Constructor for resistor_box GUI part.

This GUI part contains three resistor input field.

Returns

GtkWidget Contains the constructed GUI part.

4.13.1.2 float∗ update_resistor_values (GtkWidget ∗ *resistor_box,* float ∗ *value_array*)

Updates the value_array with values from the GUI.

The Function assumes a gfloat array of size 3.

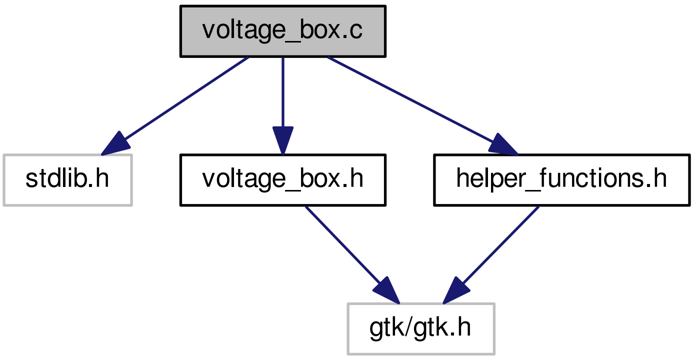
Parameters

resistor_box	The GUI part to access the user input. Float array of size 3 that contains the resistor values

4.14 voltage_box.c File Reference

```
#include <stdlib.h>
#include "voltage_box.h"
#include "helper_functions.h"
```

Include dependency graph for voltage_box.c:



Functions

- GtkWidget ∗ voltage_box_new (void)

    Constructor for voltage_box GUI element.
- float get_voltage (GtkWidget ∗voltage_box)

    Retrieve voltage value from GUI entry field.

4.14.1 Function Documentation

4.14.1.1 float get_voltage (GtkWidget ∗ *voltage_box*)

Retrieve voltage value from GUI entry field.

Retrieves float value of the voltage input box. The function needs a reference/pointer to the GUI element.

Parameters

voltage_box	GtkWidget GUI element voltage_box

Returns

float value of the current set voltage value in the GUI

4.14.1.2 GtkWidget∗ voltage_box_new (void)

Constructor for voltage_box GUI element.

Constructs the voltage_box GUI element which contains the entry field for setting the voltage.

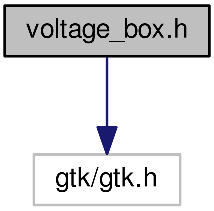
Returns

GtkWidget A voltage_box GUI Element

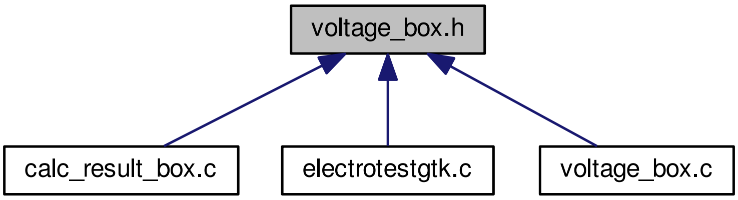
4.15 voltage_box.h File Reference

```
#include <gtk/gtk.h>
```

Include dependency graph for voltage_box.h:



This graph shows which files directly or indirectly include this file:



Functions

- GtkWidget ∗ voltage_box_new (void)

  Constructor for voltage_box GUI element.
- float get_voltage (GtkWidget ∗voltage_box)

  Retrieve voltage value from GUI entry field.

4.15.1 Function Documentation

4.15.1.1 float get_voltage (GtkWidget ∗ *voltage_box*)

Retrieve voltage value from GUI entry field.

Retrieves float value of the voltage input box. The function needs a reference/pointer to the GUI element.

Parameters

voltage_box	GtkWidget GUI element voltage_box

Returns

float value of the current set voltage value in the GUI

4.15.1.2 GtkWidget∗ voltage_box_new (void)

Constructor for voltage_box GUI element.

Constructs the voltage_box GUI element which contains the entry field for setting the voltage.

Returns

GtkWidget A voltage_box GUI Element