

gtkterm

Generated by Doxygen 1.9.4

1 GTKTerm: A GTK+ Serial Port Terminal	1
1.1 Usage	1
1.1.1 Keyboard Shortcuts	1
1.1.2 Command Line Options	1
1.1.3 Notes on RS485:	2
1.1.4 Scriptability with Signals	2
1.2 Installation	2
1.3 Uninstallation	3
1.4 License	3
2 Class Index	5
2.1 Class List	5
3 File Index	7
3.1 File List	7
4 Class Documentation	9
4.1 _GtkTerm Struct Reference	9
4.1.1 Detailed Description	10
4.1.2 Member Data Documentation	10
4.1.2.1 action_group	10
4.1.2.2 config	10
4.1.2.3 g_config_group	10
4.1.2.4 g_port_group	11
4.1.2.5 g_term_group	11
4.1.2.6 parent_instance	11
4.1.2.7 section	11
4.2 _GtkTermConfiguration Struct Reference	11
4.2.1 Member Data Documentation	12
4.2.1.1 parent_instance	12
4.3 _GtkTermConfigurationClass Struct Reference	12
4.3.1 Member Data Documentation	12
4.3.1.1 parent_class	12
4.4 _GtkTermSerialPort Struct Reference	13
4.4.1 Member Data Documentation	13
4.4.1.1 parent_instance	13
4.5 _GtkTermSerialPortClass Struct Reference	13
4.5.1 Member Data Documentation	14
4.5.1.1 parent_class	14
4.6 _GtkTermTerminal Struct Reference	14
4.6.1 Member Data Documentation	14
4.6.1.1 vte_object	14
4.7 _GtkTermTerminalClass Struct Reference	15

4.7.1 Member Data Documentation	15
4.7.1.1 <code>vte_class</code>	15
4.8 <code>_GtkTermWindow</code> Struct Reference	15
4.8.1 Detailed Description	17
4.8.2 Member Data Documentation	17
4.8.2.1 <code>action_group</code>	17
4.8.2.2 <code>fullscreen</code>	17
4.8.2.3 <code>height</code>	17
4.8.2.4 <code>infobar</code>	17
4.8.2.5 <code>maximized</code>	17
4.8.2.6 <code>menubutton</code>	18
4.8.2.7 <code>message</code>	18
4.8.2.8 <code>parent_instance</code>	18
4.8.2.9 <code>scrolled_window</code>	18
4.8.2.10 <code>search_bar</code>	18
4.8.2.11 <code>status_config</code>	18
4.8.2.12 <code>status_config_message</code>	19
4.8.2.13 <code>status_message</code>	19
4.8.2.14 <code>status_serial_signal</code>	19
4.8.2.15 <code>statusbox</code>	19
4.8.2.16 <code>terminal_window</code>	19
4.8.2.17 <code>toolmenu</code>	19
4.8.2.18 <code>width</code>	20
4.9 <code>GtkTermConfigurationPrivate</code> Struct Reference	20
4.9.1 Member Data Documentation	20
4.9.1.1 <code>config_file</code>	20
4.9.1.2 <code>key_file</code>	21
4.10 <code>GtkTermSerialPortPrivate</code> Struct Reference	21
4.10.1 Member Data Documentation	22
4.10.1.1 <code>port_conf</code>	22
4.10.1.2 <code>serial_port_fd</code>	22
4.10.1.3 <code>termios_save</code>	22
4.11 <code>GtkTermTerminalPrivate</code> Struct Reference	22
4.11.1 Member Data Documentation	23
4.11.1.1 <code>app</code>	23
4.11.1.2 <code>main_window</code>	23
4.11.1.3 <code>port_conf</code>	23
4.11.1.4 <code>section</code>	23
4.11.1.5 <code>serial_port</code>	24
4.11.1.6 <code>term_conf</code>	24
4.11.1.7 <code>view_mode</code>	24
4.12 <code>macro_t</code> Struct Reference	24

4.12.1 Detailed Description	25
4.12.2 Member Data Documentation	25
4.12.2.1 action	25
4.12.2.2 closure	25
4.12.2.3 shortcut	25
4.13 port_config_t Struct Reference	26
4.13.1 Member Data Documentation	26
4.13.1.1 baudrate	26
4.13.1.2 bits	27
4.13.1.3 disable_port_lock	27
4.13.1.4 flow_control	27
4.13.1.5 parity	27
4.13.1.6 port	27
4.13.1.7 rs485_rts_time_after_transmit	27
4.13.1.8 rs485_rts_time_before_transmit	27
4.13.1.9 stopbits	28
4.14 term_config_t Struct Reference	28
4.14.1 Member Data Documentation	29
4.14.1.1 background_color	29
4.14.1.2 block_cursor	29
4.14.1.3 char_queue	29
4.14.1.4 columns	29
4.14.1.5 crlfauto	29
4.14.1.6 delay	29
4.14.1.7 echo	29
4.14.1.8 font	30
4.14.1.9 foreground_color	30
4.14.1.10 rows	30
4.14.1.11 scrollbar	30
4.14.1.12 show_cursor	30
4.14.1.13 timestamp	30
4.14.1.14 visual_bell	30
5 File Documentation	31
5.1 README.md File Reference	31
5.2 README.source File Reference	31
5.3 buffer.c File Reference	31
5.3.1 Macro Definition Documentation	32
5.3.1.1 MAX_SECTION_LENGTH	32
5.3.1.2 TIMESTAMP_SIZE	32
5.3.2 Function Documentation	32
5.3.2.1 clear_buffer()	32

5.3.2.2 create_buffer()	33
5.3.2.3 delete_buffer()	33
5.3.2.4 insert_timestamp()	33
5.3.2.5 put_chars()	33
5.3.2.6 set_clear_func()	33
5.3.2.7 set_display_func()	33
5.3.2.8 unset_clear_func()	34
5.3.2.9 unset_display_func()	34
5.3.2.10 write_buffer()	34
5.3.2.11 write_buffer_with_func()	34
5.3.3 Variable Documentation	35
5.3.3.1 clear_func	35
5.3.3.2 overlapped	35
5.3.3.3 timestamp_on	35
5.3.3.4 virt_col_pos	35
5.3.3.5 write_func	35
5.4 buffer.h File Reference	36
5.4.1 Macro Definition Documentation	36
5.4.1.1 BUFFER_SIZE	36
5.4.2 Function Documentation	36
5.4.2.1 clear_buffer()	37
5.4.2.2 create_buffer()	37
5.4.2.3 delete_buffer()	37
5.4.2.4 put_chars()	37
5.4.2.5 set_clear_func()	37
5.4.2.6 set_display_func()	37
5.4.2.7 unset_clear_func()	38
5.4.2.8 unset_display_func()	38
5.4.2.9 write_buffer()	38
5.4.2.10 write_buffer_with_func()	38
5.5 buffer.h	39
5.6 cmdline.c File Reference	39
5.6.1 Function Documentation	40
5.6.1.1 gtkterm_add_cmdline_options()	40
5.7 cmdline.h File Reference	40
5.7.1 Function Documentation	40
5.7.1.1 gtkterm_add_cmdline_options()	40
5.7.2 Variable Documentation	41
5.7.2.1 g_term_group	41
5.8 cmdline.h	41
5.9 defaults.h File Reference	41
5.9.1 Macro Definition Documentation	42

5.9.1.1 BUFFER_LENGTH	42
5.9.1.2 DEFAULT_BAUDRATE	42
5.9.1.3 DEFAULT_BITS	42
5.9.1.4 DEFAULT_CHAR	43
5.9.1.5 DEFAULT_DELAY	43
5.9.1.6 DEFAULT_DELAY_RS485	43
5.9.1.7 DEFAULT_ECHO	43
5.9.1.8 DEFAULT_FLOW	43
5.9.1.9 DEFAULT_FONT	43
5.9.1.10 DEFAULT_PARITY	43
5.9.1.11 DEFAULT_PORT	44
5.9.1.12 DEFAULT_SCROLLBACK	44
5.9.1.13 DEFAULT_STOPBITS	44
5.9.1.14 DEFAULT_VISUAL_BELL	44
5.9.1.15 LINE_FEED	44
5.9.1.16 MAX_SECTION_LENGTH	44
5.9.1.17 POLL_DELAY	44
5.9.1.18 RECEIVE_BUFFER	45
5.9.1.19 TRANSMIT_BUFFER	45
5.10 defaults.h	45
5.11 files.c File Reference	45
5.11.1 Variable Documentation	46
5.11.1.1 default_filename	46
5.12 files.h File Reference	46
5.12.1 Function Documentation	46
5.12.1.1 add_input()	46
5.12.1.2 save_raw_file()	46
5.12.1.3 send_raw_file()	47
5.12.2 Variable Documentation	47
5.12.2.1 default_filename	47
5.12.2.2 waiting_for_char	47
5.13 files.h	47
5.14 gtkterm.c File Reference	48
5.14.1 Function Documentation	48
5.14.1.1 main()	48
5.14.1.2 set_window_title()	49
5.14.2 Variable Documentation	49
5.14.2.1 gtkterm_signals	49
5.15 gtkterm.h File Reference	49
5.15.1 Macro Definition Documentation	50
5.15.1.1 GTKTERM_TYPE_APP	50
5.15.1.2 GTKTERM_TYPE_GTKTERM_WINDOW	50

5.15.2 Typedef Documentation	50
5.15.2.1 GtkTerm	51
5.15.2.2 GtkTermWindow	51
5.15.3 Enumeration Type Documentation	51
5.15.3.1 anonymous enum	51
5.15.4 Variable Documentation	51
5.15.4.1 gtkterm_signals	51
5.16 gtkterm.h	51
5.17 interface.c File Reference	52
5.17.1 Function Documentation	53
5.17.1.1 show_message()	53
5.17.2 Variable Documentation	53
5.17.2.1 config	53
5.17.2.2 timestamp_on	54
5.17.2.3 virt_col_pos	54
5.18 interface.h File Reference	54
5.18.1 Macro Definition Documentation	54
5.18.1.1 ASCII_VIEW	55
5.18.1.2 HEXADECIMAL_VIEW	55
5.18.1.3 MSG_ERR	55
5.18.1.4 MSG_WRN	55
5.18.2 Function Documentation	55
5.18.2.1 show_message()	55
5.18.3 Variable Documentation	55
5.18.3.1 display	56
5.18.3.2 Text	56
5.19 interface.h	56
5.20 macros.c File Reference	56
5.20.1 Enumeration Type Documentation	57
5.20.1.1 anonymous enum	57
5.20.2 Function Documentation	57
5.20.2.1 convert_macros_to_string()	57
5.20.2.2 convert_string_to_macros()	58
5.20.2.3 get_shortcuts()	58
5.20.2.4 macro_count()	58
5.20.2.5 remove_shortcuts()	59
5.20.3 Variable Documentation	59
5.20.3.1 macros	59
5.20.3.2 nr_of_macros	59
5.21 macros.h File Reference	59
5.21.1 Function Documentation	60
5.21.1.1 add_shortcuts()	60

5.21.1.2 convert_macros_to_string()	60
5.21.1.3 convert_string_to_macros()	61
5.21.1.4 get_shortcuts()	61
5.21.1.5 macro_count()	61
5.21.1.6 remove_shortcuts()	61
5.21.2 Variable Documentation	62
5.21.2.1 macros	62
5.22 macros.h	62
5.23 resource_file.c File Reference	62
5.23.1 Macro Definition Documentation	63
5.23.1.1 BUFFER_LENGTH	64
5.23.1.2 CONFIGURATION_FILENAME	64
5.23.2 Function Documentation	64
5.23.2.1 check_keyfile()	64
5.23.2.2 gtkterm_configuration_default_configuration()	64
5.23.2.3 gtkterm_configuration_validate()	65
5.23.2.4 on_set_config_options()	65
5.23.3 Variable Documentation	66
5.23.3.1 GtkTermConfigurationItems	66
5.24 resource_file.h File Reference	67
5.24.1 Macro Definition Documentation	68
5.24.1.1 CONF_ITEM_LENGTH	68
5.24.1.2 DEFAULT_SECTION	68
5.24.1.3 GTKTERM_TYPE_CONFIGURATION	68
5.24.2 Typedef Documentation	68
5.24.2.1 GtkTermConfiguration	68
5.24.3 Enumeration Type Documentation	68
5.24.3.1 anonymous enum	68
5.24.4 Function Documentation	69
5.24.4.1 gtkterm_configuration_new()	69
5.24.4.2 on_set_config_options()	70
5.24.5 Variable Documentation	70
5.24.5.1 GtkTermConfigurationItems	70
5.25 resource_file.h	71
5.26 serial.c File Reference	71
5.26.1 Enumeration Type Documentation	72
5.26.1.1 anonymous enum	72
5.26.2 Function Documentation	73
5.26.2.1 gtkterm_serial_port_get_string()	73
5.26.2.2 gtkterm_serial_port_new()	73
5.26.2.3 gtkterm_serial_port_status()	73
5.27 serial.h File Reference	73

5.27.1 Macro Definition Documentation	74
5.27.1.1 GTKTERM_TYPE_SERIAL_PORT	74
5.27.2 Typedef Documentation	74
5.27.2.1 GtkTermSerialPort	74
5.27.3 Function Documentation	74
5.27.3.1 gtkterm_serial_port_get_string()	74
5.27.3.2 gtkterm_serial_port_new()	75
5.27.3.3 gtkterm_serial_port_status()	75
5.28 serial.h	75
5.29 terminal.c File Reference	76
5.29.1 Enumeration Type Documentation	76
5.29.1.1 anonymous enum	76
5.29.2 Function Documentation	77
5.29.2.1 gtkterm_terminal_new()	77
5.30 terminal.h File Reference	77
5.30.1 Macro Definition Documentation	78
5.30.1.1 GTKTERM_TYPE_TERMINAL	78
5.30.2 Function Documentation	78
5.30.2.1 gtkterm_terminal_new()	78
5.31 terminal.h	79
Index	81

Chapter 1

GTKTerm: A GTK+ Serial Port Terminal

GTKTerm is a simple, graphical serial port terminal emulator for Linux and possibly other POSIX-compliant operating systems. It can be used to communicate with all kinds of devices with a serial interface, such as embedded computers, microcontrollers, modems, GPS receivers, CNC machines and more.

1.1 Usage

1.1.1 Keyboard Shortcuts

As GTKTerm is often used like a terminal emulator, the shortcut keys are assigned to `<ctrl><shift>`, rather than just `<ctrl>`. This allows the user to send keystrokes of the form `<ctrl>X` and not have GTKTerm intercept them.

Key Combination	Effect
<code><ctrl><shift>L</code>	Clear screen
<code><ctrl><shift>R</code>	Send file
<code><ctrl><shift>Q</code>	Quit
<code><ctrl><shift>S</code>	Configure port
<code><ctrl><shift>V</code>	Paste
<code><ctrl><shift>C</code>	Copy
<code><ctrl><shift>F</code>	Find
<code><ctrl><shift>K</code>	Clear Scrollback
<code><ctrl><shift>A</code>	Select All
<code><ctrl><shift>B</code>	Send Break
<code><ctrl>B</code>	Send break
F5	Open Port
F6	Close Port
F7	Toggle DTR
F8	Toggle RTS

1.1.2 Command Line Options

See `man gtkterm` or `gtkterm --help` for more information on available command line interface options.

1.1.3 Notes on RS485:

The RS485 flow control is a software user-space emulation and therefore may not work for all configurations (won't respond quickly enough). If this is the case for your setup, you will need to either use a dedicated RS232 to RS485 converter, or look for a kernel level driver. This is an inherent limitation to user space programs.

1.1.4 Scriptability with Signals

Some microcontrollers and other embedded devices are flashed using the same serial interface that is also used for outputting debug information. To facilitate rapid development on these platforms, GTKTerm supports the following UNIX signals:

Signal	Action	Usage Example
SIGUSR1	Open Port	<code>killall -USR1 gtkterm</code>
SIGUSR2	Close Port	<code>killall -USR2 gtkterm</code>

You may find it useful to send these signals in your own firmware flashing scripts.

1.2 Installation

GTKTerm has a few dependencies-

- Gtk+4.0 (version 4.6 or higher)
- vte-gtk4 (version 0.68 or higher)
- intltool (version 0.40.0 or higher)
- libgudev (version 229 or higher)

Once these dependencies are installed, most people should simply run:

```
meson build
ninja -C build
```

To install GTKTerm system-wide, run:

```
ninja -C build install
gtk-update-icon-cache
```

If you wish to install GTKTerm someplace other than the default directory, e.g. in `/usr`, use:

```
meson build -Dprefix=/usr
```

Then build and install as usual.

1.3 Uninstallation

To uninstall GTKTerm, run:

```
ninja -C build uninstall
```

If you already deleted the `build` directory, just compile and install GTKTerm again as explained in the previous section with the same target location prefix (`-Dprefix`) and perform the uninstall step afterwards.

1.4 License

Original Code by: Julien Schmitt

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

_GtkTerm	
The main GtkTerm application class	9
_GtkTermConfiguration	11
_GtkTermConfigurationClass	12
_GtkTermSerialPort	13
_GtkTermSerialPortClass	13
_GtkTermTerminal	14
_GtkTermTerminalClass	15
_GtkTermWindow	
The main GtkTermWindow class	15
GtkTermConfigurationPrivate	20
GtkTermSerialPortPrivate	21
GtkTermTerminalPrivate	22
macro_t	
TODO: Migrate to GObject	24
port_config_t	26
term_config_t	28

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

README.source	31
buffer.c	31
buffer.h	36
cmdline.c	39
cmdline.h	40
defaults.h	41
files.c	45
files.h	46
gtkterm.c	48
gtkterm.h	49
interface.c	52
interface.h	54
macros.c	56
macros.h	59
resource_file.c	62
resource_file.h	67
serial.c	71
serial.h	73
terminal.c	76
terminal.h	77

Chapter 4

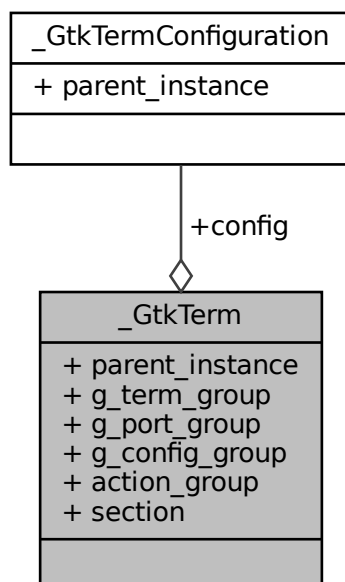
Class Documentation

4.1 `_GtkTerm` Struct Reference

The main `GtkTerm` application class.

```
#include <gtkterm.h>
```

Collaboration diagram for `_GtkTerm`:



Public Attributes

- GtkApplication [parent_instance](#)
- GOptionGroup * [g_term_group](#)
- GOptionGroup * [g_port_group](#)
- GOptionGroup * [g_config_group](#)
- GActionGroup * [action_group](#)
- [GtkTermConfiguration](#) * [config](#)
App action group.
- char * [section](#)
The Key file with the configurations.

4.1.1 Detailed Description

The main GtkTerm application class.

All application specific variables are defined here.

4.1.2 Member Data Documentation

4.1.2.1 `action_group`

```
GActionGroup* _GtkTerm::action_group
```

4.1.2.2 `config`

```
GtkTermConfiguration* _GtkTerm::config
```

App action group.

4.1.2.3 `g_config_group`

```
GOptionGroup* _GtkTerm::g_config_group
```

Referenced by [gtkterm_add_cmdline_options\(\)](#).

4.1.2.4 g_port_group

GOptionGroup* _GtkTerm::g_port_group

Referenced by [gtkterm_add_cmdline_options\(\)](#).

4.1.2.5 g_term_group

GOptionGroup* _GtkTerm::g_term_group

Referenced by [gtkterm_add_cmdline_options\(\)](#).

4.1.2.6 parent_instance

GtkApplication _GtkTerm::parent_instance

4.1.2.7 section

char* _GtkTerm::section

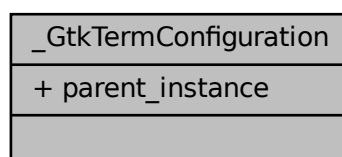
The Key file with the configurations.

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

- [gtkterm.h](#)

4.2 _GtkTermConfiguration Struct Reference

Collaboration diagram for _GtkTermConfiguration:



Public Attributes

- GObject [parent_instance](#)

4.2.1 Member Data Documentation

4.2.1.1 parent_instance

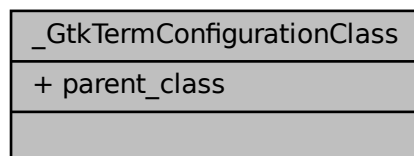
GObject `_GtkTermConfiguration::parent_instance`

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

- [resource_file.c](#)

4.3 _GtkTermConfigurationClass Struct Reference

Collaboration diagram for `_GtkTermConfigurationClass`:



Public Attributes

- GObjectClass [parent_class](#)

4.3.1 Member Data Documentation

4.3.1.1 parent_class

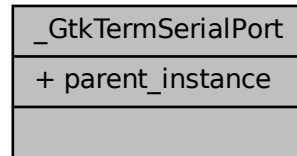
GObjectClass `_GtkTermConfigurationClass::parent_class`

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

- [resource_file.c](#)

4.4 _GtkTermSerialPort Struct Reference

Collaboration diagram for _GtkTermSerialPort:



Public Attributes

- GObject [parent_instance](#)

4.4.1 Member Data Documentation

4.4.1.1 parent_instance

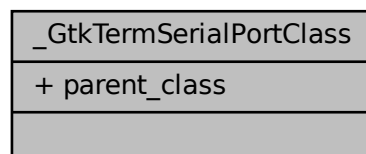
`GObject _GtkTermSerialPort::parent_instance`

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

- [serial.c](#)

4.5 _GtkTermSerialPortClass Struct Reference

Collaboration diagram for _GtkTermSerialPortClass:



Public Attributes

- GObjectClass [parent_class](#)

4.5.1 Member Data Documentation

4.5.1.1 parent_class

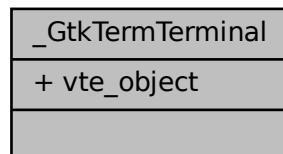
GObjectClass _GtkTermSerialPortClass::parent_class

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

- [serial.c](#)

4.6 _GtkTermTerminal Struct Reference

Collaboration diagram for _GtkTermTerminal:



Public Attributes

- VteTerminal [vte_object](#)

4.6.1 Member Data Documentation

4.6.1.1 vte_object

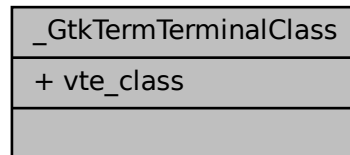
VteTerminal _GtkTermTerminal::vte_object

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

- [terminal.c](#)

4.7 _GtkTermTerminalClass Struct Reference

Collaboration diagram for _GtkTermTerminalClass:



Public Attributes

- VteTerminalClass [vte_class](#)

4.7.1 Member Data Documentation

4.7.1.1 vte_class

```
VteTerminalClass _GtkTermTerminalClass::vte_class
```

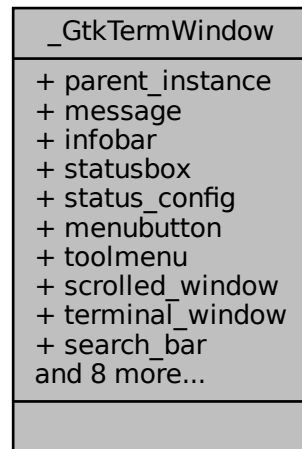
The documentation for this struct was generated from the following file:

- [terminal.c](#)

4.8 _GtkTermWindow Struct Reference

The main GtkTermWindow class.

Collaboration diagram for `_GtkTermWindow`:



Public Attributes

- `GtkApplicationWindow` [parent_instance](#)
- `GtkWidget` * [message](#)
- `GtkWidget` * [infobar](#)
Message for the infobar.
- `GtkBox` * [statusbox](#)
Infobar.
- `GtkBox` * [status_config](#)
Box for statusbar messages.
- `GtkWidget` * [menubutton](#)
Displays the actual used configuration.
- `GMenuModel` * [toolmenu](#)
Toolbar.
- `GtkScrolledWindow` * [scrolled_window](#)
Menu.
- `GtkTermTerminal` * [terminal_window](#)
Make the terminal window scrolled.
- `GtkWidget` * [search_bar](#)
The terminal window.
- `GActionGroup` * [action_group](#)
Searchbar.
- `GtkWidget` * [status_config_message](#) [3]
Window action group.
- `GtkWidget` * [status_serial_signal](#) [6]
- `GtkWidget` * [status_message](#)
- `int` [width](#)
- `int` [height](#)
- `bool` [maximized](#)
- `bool` [fullscreen](#)

4.8.1 Detailed Description

The main GtkTermWindow class.

MainWindow specific variables here.

4.8.2 Member Data Documentation

4.8.2.1 action_group

```
GActionGroup* _GtkTermWindow::action_group
```

Searchbar.

4.8.2.2 fullscreen

```
bool _GtkTermWindow::fullscreen
```

4.8.2.3 height

```
int _GtkTermWindow::height
```

4.8.2.4 infobar

```
GtkWidget* _GtkTermWindow::infobar
```

Message for the infobar.

4.8.2.5 maximized

```
bool _GtkTermWindow::maximized
```

4.8.2.6 menubutton

`GtkWidget* _GtkTermWindow::menubutton`

Displays the actual used configuration.

4.8.2.7 message

`GtkWidget* _GtkTermWindow::message`

4.8.2.8 parent_instance

`GtkApplicationWindow _GtkTermWindow::parent_instance`

4.8.2.9 scrolled_window

`GtkScrolledWindow* _GtkTermWindow::scrolled_window`

Menu.

4.8.2.10 search_bar

`GtkWidget* _GtkTermWindow::search_bar`

The terminal window.

4.8.2.11 status_config

`GtkBox* _GtkTermWindow::status_config`

Box for statusbar messages.

4.8.2.12 status_config_message

```
GtkWidget* _GtkTermWindow::status_config_message[3]
```

Window action group.

4.8.2.13 status_message

```
GtkWidget* _GtkTermWindow::status_message
```

4.8.2.14 status_serial_signal

```
GtkWidget* _GtkTermWindow::status_serial_signal[6]
```

4.8.2.15 statusbox

```
GtkBox* _GtkTermWindow::statusbox
```

Infobar.

4.8.2.16 terminal_window

```
GtkTermTerminal* _GtkTermWindow::terminal_window
```

Make the terminal window scrolled.

4.8.2.17 toolmenu

```
GMenuModel* _GtkTermWindow::toolmenu
```

Toolbar.

4.8.2.18 width

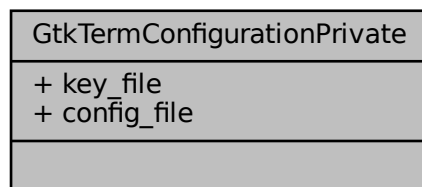
```
int _GtkTermWindow::width
```

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

- [gtkterm.c](#)

4.9 GtkTermConfigurationPrivate Struct Reference

Collaboration diagram for GtkTermConfigurationPrivate:



Public Attributes

- `GKeyFile *` [key_file](#)
- `GFile *` [config_file](#)
The memory loaded keyfile.

4.9.1 Member Data Documentation

4.9.1.1 config_file

```
GFile* GtkTermConfigurationPrivate::config_file
```

The memory loaded keyfile.

4.9.1.2 key_file

GKeyFile* GtkTermConfigurationPrivate::key_file

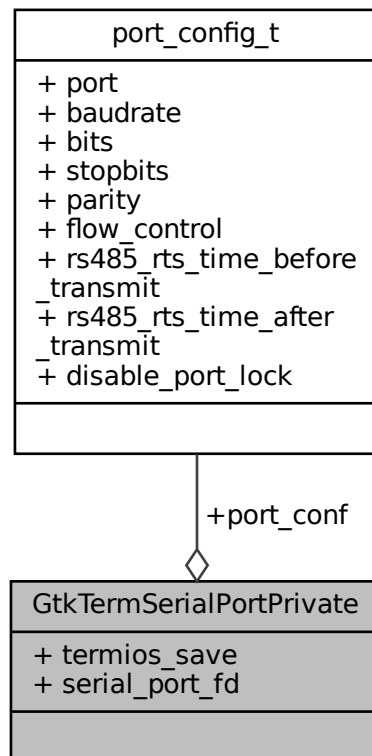
Referenced by [check_keyfile\(\)](#), [gtkterm_configuration_default_configuration\(\)](#), [gtkterm_configuration_validate\(\)](#), and [on_set_config_options\(\)](#).

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

- [resource_file.c](#)

4.10 GtkTermSerialPortPrivate Struct Reference

Collaboration diagram for GtkTermSerialPortPrivate:



Public Attributes

- [port_config_t](#) * [port_conf](#)
- struct termios [termios_save](#)
- int [serial_port_fd](#)

4.10.1 Member Data Documentation

4.10.1.1 port_conf

`port_config_t*` GtkTermSerialPortPrivate::port_conf

Referenced by [gtkterm_serial_port_get_string\(\)](#).

4.10.1.2 serial_port_fd

`int` GtkTermSerialPortPrivate::serial_port_fd

Referenced by [gtkterm_serial_port_get_string\(\)](#), and [gtkterm_serial_port_status\(\)](#).

4.10.1.3 termios_save

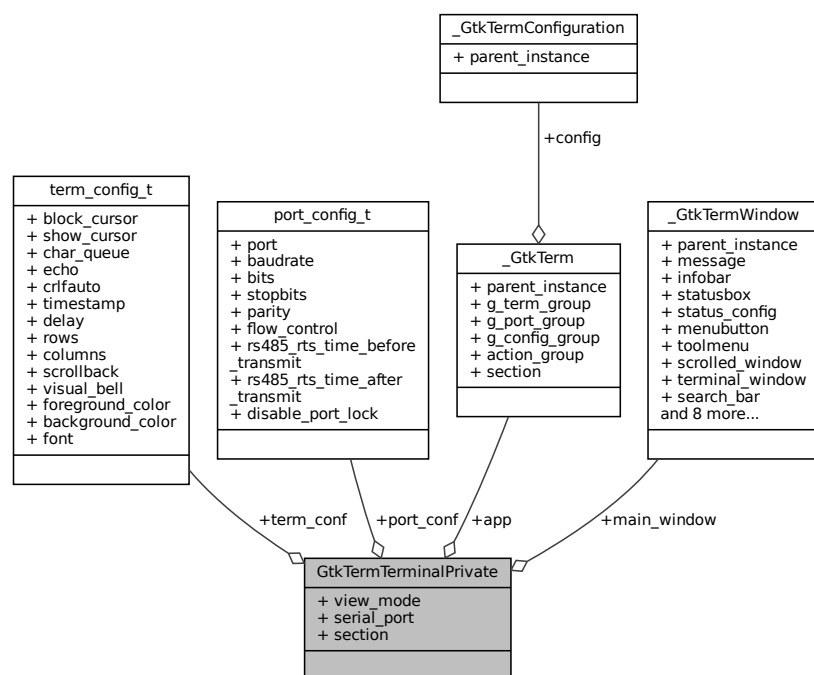
`struct termios` GtkTermSerialPortPrivate::termios_save

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

- [serial.c](#)

4.11 GtkTermTerminalPrivate Struct Reference

Collaboration diagram for GtkTermTerminalPrivate:



Public Attributes

- `uint8_t view_mode`
- `GtkTermSerialPort * serial_port`
ASCII or HEX view mode.
- `term_config_t * term_conf`
- `port_config_t * port_conf`
The configuration loaded from the keyfile.
- `char * section`
Port configuration used in this terminal.
- `GtkTerm * app`
Section used in this terminal for configuration from config file.
- `GtkTermWindow * main_window`
Pointer to the app for getting [section] and keyfile.

4.11.1 Member Data Documentation

4.11.1.1 app

`GtkTerm*` `GtkTermTerminalPrivate::app`

Section used in this terminal for configuration from config file.

4.11.1.2 main_window

`GtkTermWindow*` `GtkTermTerminalPrivate::main_window`

Pointer to the app for getting [section] and keyfile.

4.11.1.3 port_conf

`port_config_t*` `GtkTermTerminalPrivate::port_conf`

The configuration loaded from the keyfile.

4.11.1.4 section

`char*` `GtkTermTerminalPrivate::section`

Port configuration used in this terminal.

4.11.1.5 serial_port

`GtkTermSerialPort*` `GtkTermTerminalPrivate::serial_port`

ASCII or HEX view mode.

4.11.1.6 term_conf

`term_config_t*` `GtkTermTerminalPrivate::term_conf`

4.11.1.7 view_mode

`uint8_t` `GtkTermTerminalPrivate::view_mode`

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

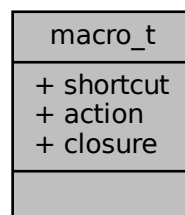
- [terminal.c](#)

4.12 macro_t Struct Reference

TODO: Migrate to GObject.

```
#include <macros.h>
```

Collaboration diagram for `macro_t`:



Public Attributes

- `char *` [shortcut](#)
- `char *` [action](#)
Shortcut of the macro.
- `GClosure *` [closure](#)
Command to perform.

4.12.1 Detailed Description

TODO: Migrate to GObject.

Define macro structure type

4.12.2 Member Data Documentation

4.12.2.1 action

```
char* macro_t::action
```

Shortcut of the macro.

Referenced by [convert_macros_to_string\(\)](#), and [convert_string_to_macros\(\)](#).

4.12.2.2 closure

```
GClosure* macro_t::closure
```

Command to perform.

4.12.2.3 shortcut

```
char* macro_t::shortcut
```

Referenced by [convert_macros_to_string\(\)](#), and [convert_string_to_macros\(\)](#).

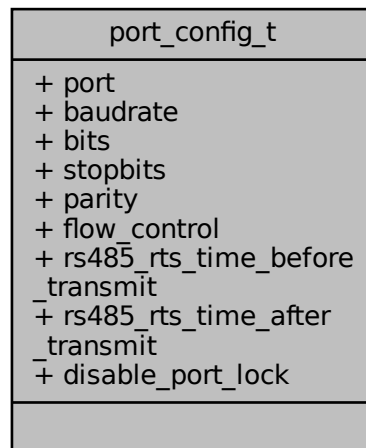
The documentation for this struct was generated from the following file:

- [macros.h](#)

4.13 port_config_t Struct Reference

```
#include <serial.h>
```

Collaboration diagram for port_config_t:



Public Attributes

- `char *` [port](#)
- `long int` [baudrate](#)
- `int` [bits](#)
- `int` [stopbits](#)
- `int` [parity](#)
- `int` [flow_control](#)
- `int` [rs485_rts_time_before_transmit](#)
- `int` [rs485_rts_time_after_transmit](#)
- `bool` [disable_port_lock](#)

4.13.1 Member Data Documentation

4.13.1.1 baudrate

```
long int port_config_t::baudrate
```

Referenced by [gtkterm_serial_port_get_string\(\)](#).

4.13.1.2 bits

```
int port_config_t::bits
```

Referenced by [gtkterm_serial_port_get_string\(\)](#).

4.13.1.3 disable_port_lock

```
bool port_config_t::disable_port_lock
```

4.13.1.4 flow_control

```
int port_config_t::flow_control
```

4.13.1.5 parity

```
int port_config_t::parity
```

Referenced by [gtkterm_serial_port_get_string\(\)](#).

4.13.1.6 port

```
char* port_config_t::port
```

Referenced by [gtkterm_serial_port_get_string\(\)](#).

4.13.1.7 rs485_rts_time_after_transmit

```
int port_config_t::rs485_rts_time_after_transmit
```

4.13.1.8 rs485_rts_time_before_transmit

```
int port_config_t::rs485_rts_time_before_transmit
```

4.13.1.9 stopbits

```
int port_config_t::stopbits
```

Referenced by [gtkterm_serial_port_get_string\(\)](#).

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

- [serial.h](#)

4.14 term_config_t Struct Reference

```
#include <terminal.h>
```

Collaboration diagram for term_config_t:



Public Attributes

- bool [block_cursor](#)
- bool [show_cursor](#)
- char [char_queue](#)
- bool [echo](#)
- bool [crlfauto](#)
- bool [timestamp](#)
- int [delay](#)
- int [rows](#)
- int [columns](#)
- int [scrollback](#)
- bool [visual_bell](#)
- GdkRGBA [foreground_color](#)
- GdkRGBA [background_color](#)
- PangoFontDescription * [font](#)

4.14.1 Member Data Documentation

4.14.1.1 background_color

GdkRGBA term_config_t::background_color

4.14.1.2 block_cursor

bool term_config_t::block_cursor

4.14.1.3 char_queue

char term_config_t::char_queue

4.14.1.4 columns

int term_config_t::columns

4.14.1.5 crlfauto

bool term_config_t::crlfauto

4.14.1.6 delay

int term_config_t::delay

4.14.1.7 echo

bool term_config_t::echo

4.14.1.8 font

PangoFontDescription* term_config_t::font

4.14.1.9 foreground_color

GdkRGBA term_config_t::foreground_color

4.14.1.10 rows

int term_config_t::rows

4.14.1.11 scrollbar

int term_config_t::scrollback

4.14.1.12 show_cursor

bool term_config_t::show_cursor

4.14.1.13 timestamp

bool term_config_t::timestamp

4.14.1.14 visual_bell

bool term_config_t::visual_bell

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

- [terminal.h](#)

Chapter 5

File Documentation

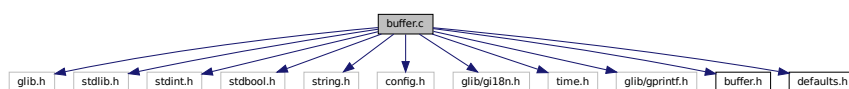
5.1 README.md File Reference

5.2 README.source File Reference

5.3 buffer.c File Reference

```
#include <glib.h>
#include <stdlib.h>
#include <stdint.h>
#include <stdbool.h>
#include <string.h>
#include <config.h>
#include <glib/gi18n.h>
#include <time.h>
#include <glib/gprintf.h>
#include "buffer.h"
#include "defaults.h"
```

Include dependency graph for buffer.c:



Macros

- #define MAX_SECTION_LENGTH 32
- #define TIMESTAMP_SIZE 50

Functions

- void [create_buffer](#) (void)
- void [delete_buffer](#) (void)
- unsigned int [insert_timestamp](#) (char *buffer)
- void [put_chars](#) (const char *chars, unsigned int size, bool crlf_auto)
- void [write_buffer](#) (void)
- void [write_buffer_with_func](#) (void(*func)(const char *, unsigned int))
- void [clear_buffer](#) (void)
- void [set_clear_func](#) (void(*func)(void))
- void [unset_clear_func](#) (void(*func)(void))
- void [set_display_func](#) (void(*func)(const char *, unsigned int))
- void [unset_display_func](#) (void(*func)(const char *, unsigned int))

Variables

- bool [timestamp_on](#)
- char [overlapped](#)
- unsigned int [virt_col_pos](#)
- void(* [write_func](#))(const char *, unsigned int) = NULL
- void(* [clear_func](#))(void) = NULL

5.3.1 Macro Definition Documentation

5.3.1.1 MAX_SECTION_LENGTH

```
#define MAX_SECTION_LENGTH 32
```

5.3.1.2 TIMESTAMP_SIZE

```
#define TIMESTAMP_SIZE 50
```

5.3.2 Function Documentation

5.3.2.1 clear_buffer()

```
void clear_buffer (  
    void )
```

References [clear_func](#).

5.3.2.2 create_buffer()

```
void create_buffer (  
    void )
```

5.3.2.3 delete_buffer()

```
void delete_buffer (  
    void )
```

5.3.2.4 insert_timestamp()

```
unsigned int insert_timestamp (  
    char * buffer )
```

5.3.2.5 put_chars()

```
void put_chars (  
    const char * chars,  
    unsigned int size,  
    bool crlf_auto )
```

References [RECEIVE_BUFFER](#), [timestamp_on](#), and [TIMESTAMP_SIZE](#).

5.3.2.6 set_clear_func()

```
void set_clear_func (  
    void(*) (void) func )
```

References [clear_func](#).

5.3.2.7 set_display_func()

```
void set_display_func (  
    void(*) (const char *, unsigned int) func )
```

References [write_func](#).

5.3.2.8 unset_clear_func()

```
void unset_clear_func (
    void(*) (void) func )
```

References [clear_func](#).

5.3.2.9 unset_display_func()

```
void unset_display_func (
    void(*) (const char *, unsigned int) func )
```

References [write_func](#).

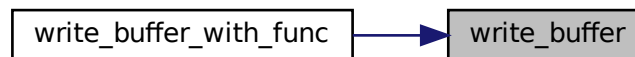
5.3.2.10 write_buffer()

```
void write_buffer (
    void )
```

References [overlapped](#), and [write_func](#).

Referenced by [write_buffer_with_func\(\)](#).

Here is the caller graph for this function:

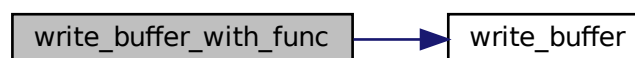


5.3.2.11 write_buffer_with_func()

```
void write_buffer_with_func (
    void(*) (const char *, unsigned int) func )
```

References [write_buffer\(\)](#), and [write_func](#).

Here is the call graph for this function:



5.3.3 Variable Documentation

5.3.3.1 clear_func

```
void(* clear_func) (void) (  
    void ) = NULL
```

Referenced by [clear_buffer\(\)](#), [set_clear_func\(\)](#), and [unset_clear_func\(\)](#).

5.3.3.2 overlapped

```
char overlapped
```

Referenced by [write_buffer\(\)](#).

5.3.3.3 timestamp_on

```
bool timestamp_on [extern]
```

Referenced by [put_chars\(\)](#).

5.3.3.4 virt_col_pos

```
unsigned int virt_col_pos [extern]
```

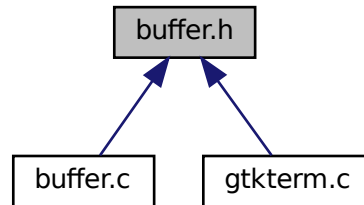
5.3.3.5 write_func

```
void(* write_func) (const char *, unsigned int) (  
    const char * ,  
    unsigned int ) = NULL
```

Referenced by [set_display_func\(\)](#), [unset_display_func\(\)](#), [write_buffer\(\)](#), and [write_buffer_with_func\(\)](#).

5.4 buffer.h File Reference

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



Macros

- `#define BUFFER_SIZE (128 * 1024)`

Functions

- void `create_buffer` (void)
- void `delete_buffer` (void)
- void `put_chars` (const char *, unsigned int, bool)
- void `clear_buffer` (void)
- void `write_buffer` (void)
- void `set_display_func` (void(*func)(const char *, uint32_t))
- void `unset_display_func` (void(*func)(const char *, uint32_t))
- void `set_clear_func` (void(*func)(void))
- void `unset_clear_func` (void(*func)(void))
- void `write_buffer_with_func` (void(*func)(const char *, uint32_t))

5.4.1 Macro Definition Documentation

5.4.1.1 BUFFER_SIZE

```
#define BUFFER_SIZE (128 * 1024)
```

5.4.2 Function Documentation

5.4.2.1 clear_buffer()

```
void clear_buffer (
    void )
```

References [clear_func](#).

5.4.2.2 create_buffer()

```
void create_buffer (
    void )
```

5.4.2.3 delete_buffer()

```
void delete_buffer (
    void )
```

5.4.2.4 put_chars()

```
void put_chars (
    const char * chars,
    unsigned int size,
    bool crlf_auto )
```

References [RECEIVE_BUFFER](#), [timestamp_on](#), and [TIMESTAMP_SIZE](#).

5.4.2.5 set_clear_func()

```
void set_clear_func (
    void(*) (void) func )
```

References [clear_func](#).

5.4.2.6 set_display_func()

```
void set_display_func (
    void(*) (const char *, uint32_t) func )
```

5.4.2.7 unset_clear_func()

```
void unset_clear_func (
    void(*) (void) func )
```

References [clear_func](#).

5.4.2.8 unset_display_func()

```
void unset_display_func (
    void(*) (const char *, uint32_t) func )
```

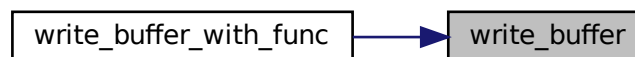
5.4.2.9 write_buffer()

```
void write_buffer (
    void )
```

References [overlapped](#), and [write_func](#).

Referenced by [write_buffer_with_func\(\)](#).

Here is the caller graph for this function:



5.4.2.10 write_buffer_with_func()

```
void write_buffer_with_func (
    void(*) (const char *, uint32_t) func )
```


5.5 buffer.h

[Go to the documentation of this file.](#)

```

1  /*****
2  /* buffer.h
3  /* -----
4  /*      GTKTerm Software
5  /*      (c) Julien Schmitt
6  /*
7  /* -----
8  /*
9  /* Purpose
10 /*      Management of a local buffer of data received
11 /*      - Header file -
12 /*
13 /*      ChangeLog
14 /*      - 0.99.7 : removed auto crlf stuff - (use macros instead)
15 /*      - 0.98.4 : file creation by Julien
16 /*
17 /*****/
18
19 #ifndef BUFFER_H_
20 #define BUFFER_H_
21
22 #define BUFFER_SIZE (128 * 1024)
23
24 void create_buffer(void);
25 void delete_buffer(void);
26 void put_chars(const char *, unsigned int, bool);
27 void clear_buffer(void);
28 void write_buffer(void);
29 void set_display_func(void (*func)(const char *, uint32_t));
30 void unset_display_func(void (*func)(const char *, uint32_t));
31 void set_clear_func(void (*func)(void));
32 void unset_clear_func(void (*func)(void));
33 void write_buffer_with_func(void (*func)(const char *, uint32_t));
34
35 #endif

```

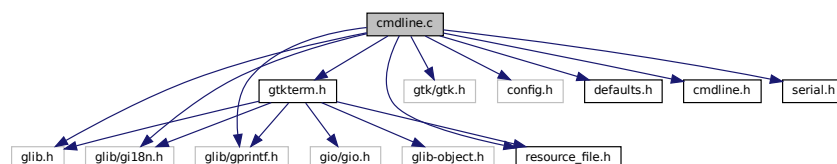
5.6 cmdline.c File Reference

```

#include <glib.h>
#include <glib/gi18n.h>
#include <gtk/gtk.h>
#include <glib/gprintf.h>
#include <config.h>
#include "defaults.h"
#include "gtkterm.h"
#include "resource_file.h"
#include "cmdline.h"
#include "serial.h"

```

Include dependency graph for cmdline.c:



Functions

- void [gtkterm_add_cmdline_options](#) (GtkTerm *app)

5.6.1 Function Documentation

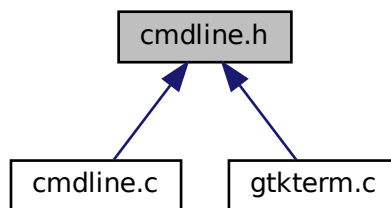
5.6.1.1 gtkterm_add_cmdline_options()

```
void gtkterm_add_cmdline_options (
    GtkTerm * app )
```

References [BUFFER_LENGTH](#), [_GtkTerm::g_config_group](#), [_GtkTerm::g_port_group](#), and [_GtkTerm::g_term_group](#).

5.7 cmdline.h File Reference

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



Functions

- void [gtkterm_add_cmdline_options](#) ([GtkTerm](#) *app)

Variables

- [GOptionGroup](#) * [g_term_group](#)

5.7.1 Function Documentation

5.7.1.1 gtkterm_add_cmdline_options()

```
void gtkterm_add_cmdline_options (
    GtkTerm * app )
```

References [BUFFER_LENGTH](#), [_GtkTerm::g_config_group](#), [_GtkTerm::g_port_group](#), and [_GtkTerm::g_term_group](#).

5.7.2 Variable Documentation

5.7.2.1 g_term_group

```
GOptionGroup* g_term_group [extern]
```

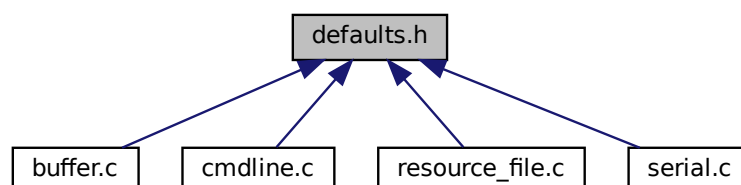
5.8 cmdline.h

[Go to the documentation of this file.](#)

```
1 /*****
2  /* cmdline.h
3  /* -----
4  /*      GTKTerm Software
5  /*      (c) Julien Schmitt
6  /*
7  /* -----
8  /*
9  /* Purpose
10 /*      Reads the command line
11 /*      - Header file -
12 /*
13 /*      ChangeLog
14 /*      - 2.0 : migrated to GTK4
15 /*      - 0.98 : file creation by Julien
16 /*
17 /*****/extern GOptionGroup
18 *g_term_group;
19 #ifndef CMDLINE_H
20 #define CMDLINE_H
21
22 void gtkterm_add_cmdline_options (GtkTerm *app);
23
24 #endif // CMDLINE_H
```

5.9 defaults.h File Reference

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



Macros

- #define `DEFAULT_FONT` "Monospace 12"
Default for VTE-terminal.
- #define `DEFAULT_SCROLLBACK` 10000
- #define `DEFAULT_DELAY` 0
- #define `DEFAULT_CHAR` -1
- #define `DEFAULT_DELAY_RS485` 30
- #define `DEFAULT_ECHO` "false"
- #define `DEFAULT_VISUAL_BELL` "false"
- #define `DEFAULT_PORT` "/dev/ttyS0"
Default for serial ports.
- #define `DEFAULT_BAUDRATE` 115200
- #define `DEFAULT_PARITY` "none"
- #define `DEFAULT_BITS` 8
- #define `DEFAULT_STOPBITS` 1
- #define `DEFAULT_FLOW` "none"
- #define `RECEIVE_BUFFER` 8192
- #define `TRANSMIT_BUFFER` 4096
- #define `LINE_FEED` 0x0A
- #define `POLL_DELAY` 100
- #define `BUFFER_LENGTH` 256
Generic defaults.
- #define `MAX_SECTION_LENGTH` 32

5.9.1 Macro Definition Documentation

5.9.1.1 BUFFER_LENGTH

```
#define BUFFER_LENGTH 256
```

Generic defaults.

5.9.1.2 DEFAULT_BAUDRATE

```
#define DEFAULT_BAUDRATE 115200
```

5.9.1.3 DEFAULT_BITS

```
#define DEFAULT_BITS 8
```

5.9.1.4 DEFAULT_CHAR

```
#define DEFAULT_CHAR -1
```

5.9.1.5 DEFAULT_DELAY

```
#define DEFAULT_DELAY 0
```

5.9.1.6 DEFAULT_DELAY_RS485

```
#define DEFAULT_DELAY_RS485 30
```

5.9.1.7 DEFAULT_ECHO

```
#define DEFAULT_ECHO "false"
```

5.9.1.8 DEFAULT_FLOW

```
#define DEFAULT_FLOW "none"
```

5.9.1.9 DEFAULT_FONT

```
#define DEFAULT_FONT "Monospace 12"
```

Default for VTE-terminal.

5.9.1.10 DEFAULT_PARITY

```
#define DEFAULT_PARITY "none"
```

5.9.1.11 DEFAULT_PORT

```
#define DEFAULT_PORT "/dev/ttyS0"
```

Default for serial ports.

5.9.1.12 DEFAULT_SCROLLBACK

```
#define DEFAULT_SCROLLBACK 10000
```

5.9.1.13 DEFAULT_STOPBITS

```
#define DEFAULT_STOPBITS 1
```

5.9.1.14 DEFAULT_VISUAL_BELL

```
#define DEFAULT_VISUAL_BELL "false"
```

5.9.1.15 LINE_FEED

```
#define LINE_FEED 0x0A
```

5.9.1.16 MAX_SECTION_LENGTH

```
#define MAX_SECTION_LENGTH 32
```

5.9.1.17 POLL_DELAY

```
#define POLL_DELAY 100
```

5.9.1.18 RECEIVE_BUFFER

```
#define RECEIVE_BUFFER 8192
```

5.9.1.19 TRANSMIT_BUFFER

```
#define TRANSMIT_BUFFER 4096
```

5.10 defaults.h

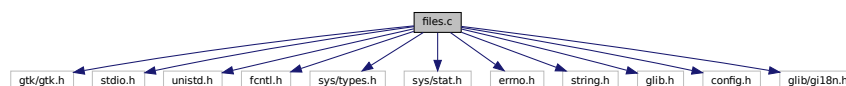
[Go to the documentation of this file.](#)

```
1 //! Default for VTE-terminal
2 #define DEFAULT_FONT           "Monospace 12"
3 #define DEFAULT_SCROLLBACK     10000
4 #define DEFAULT_DELAY          0
5 #define DEFAULT_CHAR           -1
6 #define DEFAULT_DELAY_RS485    30
7 #define DEFAULT_ECHO           "false"
8 #define DEFAULT_VISUAL_BELL    "false"
9
10 //! Default for serial ports
11 #define DEFAULT_PORT           "/dev/ttyS0"
12 #define DEFAULT_BAUDRATE       115200
13 #define DEFAULT_PARITY         "none"
14 #define DEFAULT_BITS           8
15 #define DEFAULT_STOPBITS       1
16 #define DEFAULT_FLOW           "none"
17
18 #define RECEIVE_BUFFER          8192
19 #define TRANSMIT_BUFFER         4096
20 #define LINE_FEED              0x0A
21 #define POLL_DELAY             100           //! in ms (for control signals)
22
23 //! Generic defaults
24 #define BUFFER_LENGTH           256
25 #define MAX_SECTION_LENGTH     32
```

5.11 files.c File Reference

```
#include <gtk/gtk.h>
#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <errno.h>
#include <string.h>
#include <glib.h>
#include <config.h>
#include <glib/glib.h>
```

Include dependency graph for files.c:



Variables

- char * [default_filename](#) = NULL

5.11.1 Variable Documentation

5.11.1.1 [default_filename](#)

```
char* default_filename = NULL
```

5.12 files.h File Reference

Functions

- void [send_raw_file](#) (GAction *action, gpointer data)
- void [save_raw_file](#) (GAction *action, gpointer data)
- void [add_input](#) (void)

Variables

- gboolean [waiting_for_char](#)
- char * [default_filename](#)

5.12.1 Function Documentation

5.12.1.1 [add_input\(\)](#)

```
void add_input (  
    void )
```

5.12.1.2 [save_raw_file\(\)](#)

```
void save_raw_file (  
    GAction * action,  
    gpointer data )
```


5.12.1.3 send_raw_file()

```
void send_raw_file (
    GAction * action,
    gpointer data )
```

5.12.2 Variable Documentation

5.12.2.1 default_filename

```
char* default_filename [extern]
```

5.12.2.2 waiting_for_char

```
gboolean waiting_for_char [extern]
```

5.13 files.h

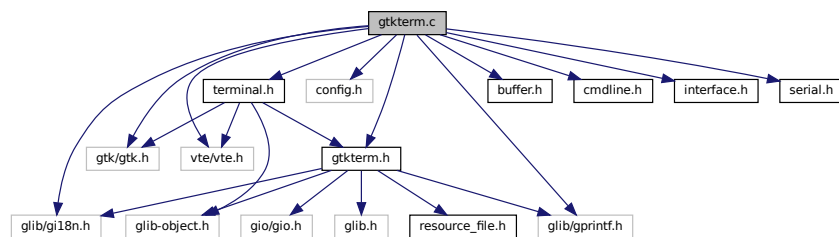
[Go to the documentation of this file.](#)

```
1 /*****
2  /* files.h
3  /* -----
4  /*      GTKTerm Software
5  /*      (c) Julien Schmitt
6  /*
7  /* -----
8  /*
9  /*      Purpose
10 /*      Raw / text file transfer management
11 /*      - Header file -
12 /*
13 /*****/
14
15 #ifndef FILES_H_
16 #define FILES_H_
17
18 void send_raw_file (GAction *action, gpointer data);
19 void save_raw_file (GAction *action, gpointer data);
20 void add_input(void);
21
22 extern gboolean waiting_for_char;
23 extern char *default_filename;
24
25 #endif
```

5.14 gtkterm.c File Reference

```
#include <gtk/gtk.h>
#include <vte/vte.h>
#include <glib/glib.h>
#include <glib/gprintf.h>
#include "config.h"
#include "gtkterm.h"
#include "terminal.h"
#include "buffer.h"
#include "cmdline.h"
#include "interface.h"
#include "serial.h"
```

Include dependency graph for gtkterm.c:



Classes

- struct [_GtkTermWindow](#)
The main GtkTermWindow class.

Functions

- void [set_window_title](#) ([GtkTermWindow](#) *, gpointer)
- int [main](#) (int argc, char *argv[])

Variables

- unsigned int [gtkterm_signals](#) [[LAST_GTKTERM_SIGNAL](#)]

5.14.1 Function Documentation

5.14.1.1 main()

```
int main (
    int argc,
    char * argv[] )
```

References [GTKTERM_TYPE_APP](#).

5.14.1.2 set_window_title()

```
void set_window_title (
    GtkTermWindow * window,
    gpointer serial_config_string )
```

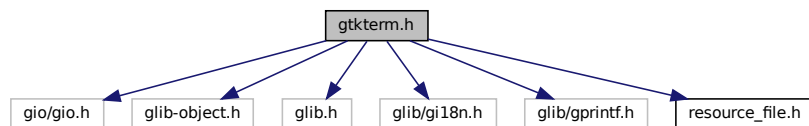
5.14.2 Variable Documentation

5.14.2.1 gtkterm_signals

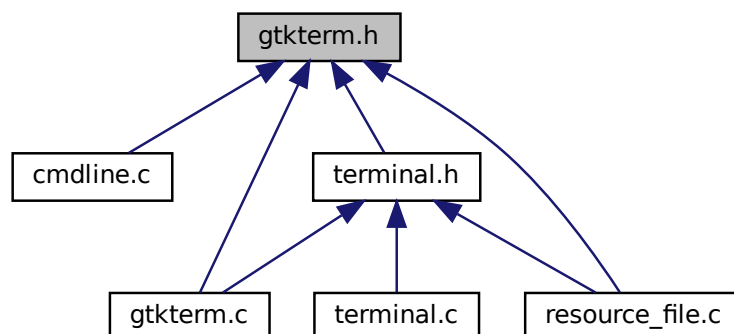
```
unsigned int gtkterm_signals[LAST\_GTKTERM\_SIGNAL]
```

5.15 gtkterm.h File Reference

```
#include <gio/gio.h>
#include <glib-object.h>
#include <glib.h>
#include <glib/gi18n.h>
#include <glib/gprintf.h>
#include "resource_file.h"
Include dependency graph for gtkterm.h:
```



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



Classes

- struct [_GtkTerm](#)

The main GtkTerm application class.

Macros

- #define [GTKTERM_TYPE_APP](#) [gtkterm_get_type\(\)](#)
- #define [GTKTERM_TYPE_GTKTERM_WINDOW](#) [gtkterm_window_get_type\(\)](#)

Typedefs

- typedef struct [_GtkTerm](#) [GtkTerm](#)
- typedef struct [_GtkTermWindow](#) [GtkTermWindow](#)

Enumerations

- enum {
 [SIGNAL_GTKTERM_LOAD_CONFIG](#) , [SIGNAL_GTKTERM_SAVE_CONFIG](#) , [SIGNAL_GTKTERM_REMOVE_SECTION](#)
 , [SIGNAL_GTKTERM_PRINT_SECTION](#) ,
 [SIGNAL_GTKTERM_CONFIG_TERMINAL](#) , [SIGNAL_GTKTERM_CONFIG_SERIAL](#) , [SIGNAL_GTKTERM_TERMINAL_CHA](#)
 , [LAST_GTKTERM_SIGNAL](#) }

Variables

- unsigned int [gtkterm_signals](#) []

5.15.1 Macro Definition Documentation

5.15.1.1 GTKTERM_TYPE_APP

```
#define GTKTERM_TYPE_APP gtkterm_get_type()
```

5.15.1.2 GTKTERM_TYPE_GTKTERM_WINDOW

```
#define GTKTERM_TYPE_GTKTERM_WINDOW gtkterm_window_get_type()
```

5.15.2 Typedef Documentation

5.15.2.1 GtkTerm

```
typedef struct _GtkTerm GtkTerm
```

5.15.2.2 GtkTermWindow

```
typedef struct _GtkTermWindow GtkTermWindow
```

5.15.3 Enumeration Type Documentation

5.15.3.1 anonymous enum

anonymous enum

Enumerator

SIGNAL_GTKTERM_LOAD_CONFIG	
SIGNAL_GTKTERM_SAVE_CONFIG	
SIGNAL_GTKTERM_REMOVE_SECTION	
SIGNAL_GTKTERM_PRINT_SECTION	
SIGNAL_GTKTERM_CONFIG_TERMINAL	
SIGNAL_GTKTERM_CONFIG_SERIAL	
SIGNAL_GTKTERM_TERMINAL_CHANGED	
LAST_GTKTERM_SIGNAL	

5.15.4 Variable Documentation

5.15.4.1 gtkterm_signals

```
unsigned int gtkterm_signals[] [extern]
```

5.16 gtkterm.h

[Go to the documentation of this file.](#)

```
1
2 #ifndef GTKTERM_H
3 #define GTKTERM_H
4
```

```

5 #include <gio/gio.h>
6 #include <glib-object.h>
7 #include <glib.h>
8 #include <glib/gi18n.h>
9 #include <glib/gprintf.h>
10
11 #include "resource_file.h"
12
13 enum {
14     SIGNAL_GTKTERM_LOAD_CONFIG,
15     SIGNAL_GTKTERM_SAVE_CONFIG,
16     SIGNAL_GTKTERM_REMOVE_SECTION,
17     SIGNAL_GTKTERM_PRINT_SECTION,
18     SIGNAL_GTKTERM_CONFIG_TERMINAL,
19     SIGNAL_GTKTERM_CONFIG_SERIAL,
20     SIGNAL_GTKTERM_TERMINAL_CHANGED,
21     LAST_GTKTERM_SIGNAL
22 };
23
24 extern unsigned int gtkterm_signals[];
25
26 G_BEGIN_DECLS
27
28 /*! @brief The main GtkTerm application class.
29  *! All application specific variables are defined here.
30  */
31 struct _GtkTerm {
32     GtkApplication parent_instance;
33
34     GOptionGroup *g_term_group;
35     GOptionGroup *g_port_group;
36     GOptionGroup *g_config_group;
37
38     GActionGroup *action_group;          /*! App action group
39
40     GtkTermConfiguration *config;        /*! The Key file with the configurations
41     char *section;                       /*! The section provided from the cli.
42     /*! Terminals have their own section pointer
43 };
44
45 #define GTKTERM_TYPE_APP gtkterm_get_type()
46 typedef struct _GtkTerm GtkTerm;
47 G_DECLARE_FINAL_TYPE (GtkTerm, gtkterm, GTKTERM, APP, GtkApplication)
48
49 #define GTKTERM_TYPE_GTKTERM_WINDOW gtkterm_window_get_type()
50 typedef struct _GtkTermWindow GtkTermWindow;
51 G_DECLARE_FINAL_TYPE (GtkTermWindow, gtkterm_window, GTKTERM, WINDOW, GtkApplicationWindow)
52
53 G_END_DECLS
54
55 #endif // GTKTERM_H

```

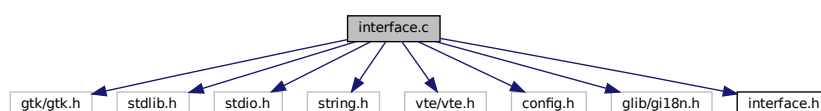
5.17 interface.c File Reference

```

#include <gtk/gtk.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <vte/vte.h>
#include <config.h>
#include <glib/gi18n.h>
#include "interface.h"

```

Include dependency graph for interface.c:



Functions

- void [show_message](#) (char *message, int type_msg)

Variables

- bool [timestamp_on](#) = 0
- struct configuration_port [config](#)
- int [virt_col_pos](#) = 0

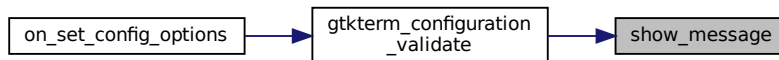
5.17.1 Function Documentation

5.17.1.1 [show_message\(\)](#)

```
void show_message (  
    char * message,  
    int type_msg )
```

Referenced by [gtkterm_configuration_validate\(\)](#).

Here is the caller graph for this function:



5.17.2 Variable Documentation

5.17.2.1 [config](#)

```
struct configuration_port config [extern]
```

Referenced by [on_set_config_options\(\)](#).

5.17.2.2 timestamp_on

```
bool timestamp_on = 0
```

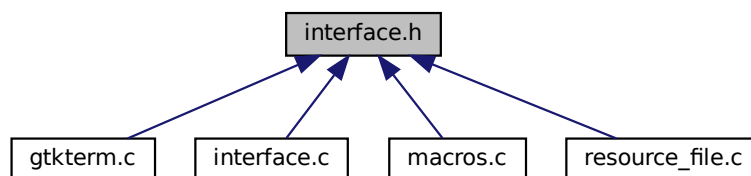
Referenced by [put_chars\(\)](#).

5.17.2.3 virt_col_pos

```
int virt_col_pos = 0
```

5.18 interface.h File Reference

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



Macros

- `#define` [MSG_WRN](#) 0
- `#define` [MSG_ERR](#) 1
- `#define` [ASCII_VIEW](#) 0
- `#define` [HEXADECIMAL_VIEW](#) 1

Functions

- void [show_message](#) (char *, int)

Variables

- GtkWidget * [Text](#)
- GtkWidget * [display](#)

5.18.1 Macro Definition Documentation

5.18.1.1 ASCII_VIEW

```
#define ASCII_VIEW 0
```

5.18.1.2 HEXADECIMAL_VIEW

```
#define HEXADECIMAL_VIEW 1
```

5.18.1.3 MSG_ERR

```
#define MSG_ERR 1
```

5.18.1.4 MSG_WRN

```
#define MSG_WRN 0
```

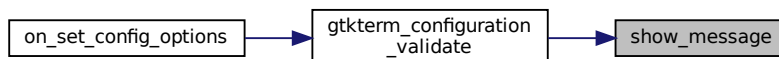
5.18.2 Function Documentation

5.18.2.1 show_message()

```
void show_message (  
    char * message,  
    int type_msg )
```

Referenced by [gtkterm_configuration_validate\(\)](#).

Here is the caller graph for this function:



5.18.3 Variable Documentation

5.18.3.1 display

```
GtkWidget* display [extern]
```

5.18.3.2 Text

```
GtkWidget* Text [extern]
```

5.19 interface.h

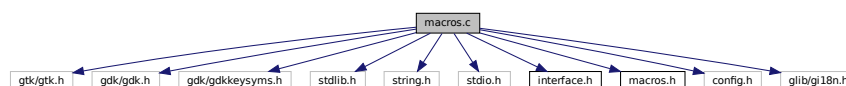
[Go to the documentation of this file.](#)

```
1 /*****
2  */ interface.h
3  */ -----
4  */          GTKTerm Software
5  */          (c) Julien Schmitt
6  */
7  */ -----
8  */
9  */ Purpose
10 */ Functions for the management of the GUI for the main window
11 */ - Header file -
12 */
13 /*****/
14
15 #ifndef WIDGETS_H_
16 #define WIDGETS_H_
17
18 #define MSG_WRN      0
19 #define MSG_ERR      1
20
21 #define ASCII_VIEW    0
22 #define HEXADECIMAL_VIEW 1
23
24 extern GtkWidget *Text;
25 extern GtkWidget *display;          // Serial terminal (vte)
26
27 void show_message(char *, int);
28
29 #endif
```

5.20 macros.c File Reference

```
#include <gtk/gtk.h>
#include <gdk/gdk.h>
#include <gdk/gdkkeysyms.h>
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
#include "interface.h"
#include "macros.h"
#include <config.h>
#include <glib/glib.h>
```

Include dependency graph for macros.c:



Enumerations

- enum { [COLUMN_SHORTCUT](#) , [COLUMN_ACTION](#) , [NUM_COLUMNS](#) }
- TODO: Migrate to GObject.*

Functions

- int [macro_count](#) ()
- void [convert_string_to_macros](#) (char **string_list, int size)
Convert the array of strings to macros.
- int [convert_macros_to_string](#) (char **string_list)
Convert the in memory macros to an array of strings for storage in file.
- [macro_t](#) * [get_shortcuts](#) (int *size)
- void [remove_shortcuts](#) (void)

Variables

- [macro_t](#) * [macros](#) = NULL
- int [nr_of_macros](#) = 0

5.20.1 Enumeration Type Documentation

5.20.1.1 anonymous enum

anonymous enum

TODO: Migrate to GObject.

Enumerator

COLUMN_SHORTCUT	
COLUMN_ACTION	
NUM_COLUMNS	

5.20.2 Function Documentation

5.20.2.1 [convert_macros_to_string\(\)](#)

```
int convert_macros_to_string (
    char ** string_list )
```

Convert the in memory macros to an array of strings for storage in file.

Must be NULL terminated

Number of strings is 2x the macros (shortcut and action)

References [macro_t::action](#), [macros](#), [nr_of_macros](#), and [macro_t::shortcut](#).

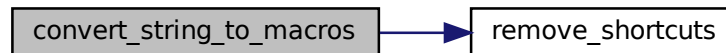
5.20.2.2 convert_string_to_macros()

```
void convert_string_to_macros (
    char ** string_list,
    int size )
```

Convert the array of strings to macros.

References [macro_t::action](#), [macros](#), [nr_of_macros](#), [remove_shortcuts\(\)](#), and [macro_t::shortcut](#).

Here is the call graph for this function:



5.20.2.3 get_shortcuts()

```
macro_t * get_shortcuts (
    int * size )
```

References [macros](#).

5.20.2.4 macro_count()

```
int macro_count ( )
```

References [nr_of_macros](#).

5.20.2.5 remove_shortcuts()

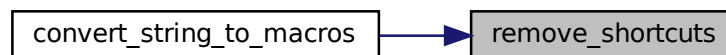
```
void remove_shortcuts (  
    void )
```

Clean up all macros

References [macros](#).

Referenced by [convert_string_to_macros\(\)](#).

Here is the caller graph for this function:



5.20.3 Variable Documentation

5.20.3.1 macros

```
macro_t* macros = NULL
```

Referenced by [convert_macros_to_string\(\)](#), [convert_string_to_macros\(\)](#), [get_shortcuts\(\)](#), and [remove_shortcuts\(\)](#).

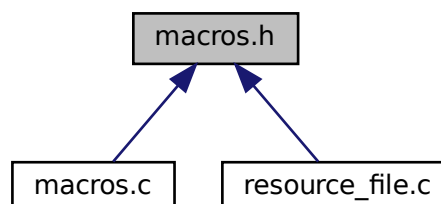
5.20.3.2 nr_of_macros

```
int nr_of_macros = 0
```

Referenced by [convert_macros_to_string\(\)](#), [convert_string_to_macros\(\)](#), and [macro_count\(\)](#).

5.21 macros.h File Reference

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



Classes

- struct [macro_t](#)
TODO: Migrate to GObject.

Functions

- void [remove_shortcuts](#) (void)
- void [add_shortcuts](#) (void)
Remove shortcuts from accel_group and free memory.
- [macro_t](#) * [get_shortcuts](#) (gint *)
- void [convert_string_to_macros](#) (char **, int)
Convert the array of strings to macros.
- int [convert_macros_to_string](#) (char **)
Convert the in memory macros to an array of strings for storage in file.
- int [macro_count](#) ()

Variables

- [macro_t](#) * [macros](#)

5.21.1 Function Documentation

5.21.1.1 [add_shortcuts\(\)](#)

```
void add_shortcuts (
    void )
```

Remove shortcuts from accel_group and free memory.

5.21.1.2 [convert_macros_to_string\(\)](#)

```
int convert_macros_to_string (
    char ** string_list )
```

Convert the in memory macros to an array of strings for storage in file.

Must be NULL terminated

Number of strings is 2x the macros (shortcut and action)

References [macro_t::action](#), [macros](#), [nr_of_macros](#), and [macro_t::shortcut](#).

5.21.1.3 convert_string_to_macros()

```
void convert_string_to_macros (
    char ** string_list,
    int size )
```

Convert the array of strings to macros.

References [macro_t::action](#), [macros](#), [nr_of_macros](#), [remove_shortcuts\(\)](#), and [macro_t::shortcut](#).

Here is the call graph for this function:



5.21.1.4 get_shortcuts()

```
macro_t * get_shortcuts (
    gint * )
```

5.21.1.5 macro_count()

```
int macro_count ( )
```

References [nr_of_macros](#).

5.21.1.6 remove_shortcuts()

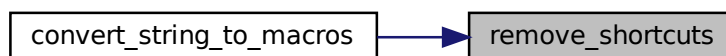
```
void remove_shortcuts (
    void )
```

Clean up all macros

References [macros](#).

Referenced by [convert_string_to_macros\(\)](#).

Here is the caller graph for this function:



5.21.2 Variable Documentation

5.21.2.1 macros

`macro_t* macros` [extern]

Referenced by [convert_macros_to_string\(\)](#), [convert_string_to_macros\(\)](#), [get_shortcuts\(\)](#), and [remove_shortcuts\(\)](#).

5.22 macros.h

[Go to the documentation of this file.](#)

```

1  /*****
2  * macros.h
3  * -----
4  *          GTKTerm Software
5  *          (c) Julien Schmitt
6  *
7  * -----
8  *
9  * \brief Purpose
10 *      Functions for the management of the macros
11 *      - Header file -
12 *
13 *****/
14
15 #ifndef MACROS_H_
16 #define MACROS_H_
17
18 //!  TODO: Migrate to GObject
19
20 //!  Define macro structure type
21 typedef struct
22 {
23     char *shortcut;    //!< Shortcut of the macro
24     char *action;      //!< Command to perform
25     GClosure *closure; //!<
26 }
27 macro_t;
28
29 //void config_macros(GtkAction *action, gpointer data);
30 void remove_shortcuts(void);           //!< Remove shortcuts from accel_group and free memory
31 void add_shortcuts(void);              //!<
32 macro_t *get_shortcuts(gint *);       //!<
33
34 void convert_string_to_macros (char **, int);
35 int  convert_macros_to_string (char **);
36
37 int macro_count ();
38
39 extern macro_t *macros;
40
41 #endif

```

5.23 resource_file.c File Reference

```

#include <stdio.h>
#include <stdbool.h>
#include <sys/stat.h>
#include <glib.h>
#include <glib/glib.h>
#include <glib/gprintf.h>
#include <glib-object.h>

```

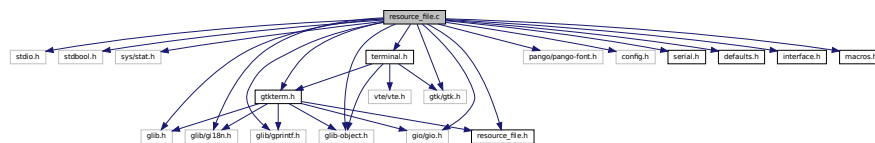


```

#include <gtk/gtk.h>
#include <gio/gio.h>
#include <pango/pango-font.h>
#include <config.h>
#include "gtkterm.h"
#include "serial.h"
#include "terminal.h"
#include "defaults.h"
#include "resource_file.h"
#include "interface.h"
#include "macros.h"

```

Include dependency graph for resource_file.c:



Classes

- struct [GtkTermConfigurationPrivate](#)
- struct [_GtkTermConfiguration](#)
- struct [_GtkTermConfigurationClass](#)

Macros

- #define [CONFIGURATION_FILENAME](#) "gtktermrc"
Default configuration filename.
- #define [BUFFER_LENGTH](#) 256

Functions

- void [gtkterm_configuration_default_configuration](#) ([GtkTermConfigurationPrivate](#) *priv, char *section)
Create a new <default> configuration.
- void [gtkterm_configuration_validate](#) ([GtkTermConfigurationPrivate](#) *priv, char *section)
validate the configuration, given by the section
- int [check_keyfile](#) ([GtkTermConfiguration](#) *self, char *section)
- bool [on_set_config_options](#) (const char *name, const char *value, gpointer data, GError **error)
Set the config option in the keyfile.

Variables

- const char [GtkTermConfigurationItems](#) [][[CONF_ITEM_LENGTH](#)]
Used configuration options to hold consistency between load/save functions.

5.23.1 Macro Definition Documentation

5.23.1.1 BUFFER_LENGTH

```
#define BUFFER_LENGTH 256
```

5.23.1.2 CONFIGURATION_FILENAME

```
#define CONFIGURATION_FILENAME ".gtktermrc"
```

Default configuration filename.

5.23.2 Function Documentation

5.23.2.1 check_keyfile()

```
int check_keyfile (
    GtkTermConfiguration * self,
    char * section )
```

Load keyfile if it is not loaded yet

Check if the <section> exists in the key file.

References [GtkTermConfigurationPrivate::key_file](#).

5.23.2.2 gtkterm_configuration_default_configuration()

```
void gtkterm_configuration_default_configuration (
    GtkTermConfigurationPrivate * priv,
    char * section )
```

Create a new <default> configuration.

References [CONF_ITEM_SERIAL_BAUDRATE](#), [CONF_ITEM_SERIAL_BITS](#), [CONF_ITEM_SERIAL_DISABLE_PORT_LOCK](#), [CONF_ITEM_SERIAL_FLOW_CONTROL](#), [CONF_ITEM_SERIAL_PARITY](#), [CONF_ITEM_SERIAL_PORT](#), [CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX](#), [CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX](#), [CONF_ITEM_SERIAL_STOPBITS](#), [CONF_ITEM_TERM_BACKGROUND_ALPHA](#), [CONF_ITEM_TERM_BACKGROUND_BLUE](#), [CONF_ITEM_TERM_BACKGROUND_GREEN](#), [CONF_ITEM_TERM_BACKGROUND_RED](#), [CONF_ITEM_TERM_BLOCK_CURSOR](#), [CONF_ITEM_TERM_COLS](#), [CONF_ITEM_TERM_CRLF_AUTO](#), [CONF_ITEM_TERM_ECHO](#), [CONF_ITEM_TERM_FONT](#), [CONF_ITEM_TERM_FOREGROUND_ALPHA](#), [CONF_ITEM_TERM_FOREGROUND_BLUE](#), [CONF_ITEM_TERM_FOREGROUND_GREEN](#), [CONF_ITEM_TERM_FOREGROUND_RED](#), [CONF_ITEM_TERM_MACROS](#), [CONF_ITEM_TERM_ROWS](#), [CONF_ITEM_TERM_SCROLLBACK](#), [CONF_ITEM_TERM_SHOW_CURSOR](#), [CONF_ITEM_TERM_VISUAL_BELL](#), [CONF_ITEM_TERM_WAIT_CHAR](#), [CONF_ITEM_TERM_WAIT_DELAY](#), [DEFAULT_BAUDRATE](#), [DEFAULT_BITS](#), [DEFAULT_CHAR](#), [DEFAULT_DELAY](#), [DEFAULT_DELAY_RS485](#), [DEFAULT_ECHO](#), [DEFAULT_FLOW](#), [DEFAULT_FONT](#), [DEFAULT_PARITY](#), [DEFAULT_PORT](#), [DEFAULT_SCROLLBACK](#), [DEFAULT_STOPBITS](#), [DEFAULT_VISUAL_BELL](#), [GtkTermConfigurationItems](#), and [GtkTermConfigurationPrivate::key_file](#).

5.23.2.3 gtkterm_configuration_validate()

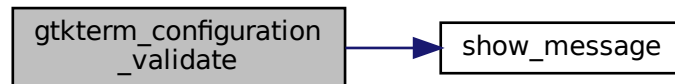
```
void gtkterm_configuration_validate (
    GtkTermConfigurationPrivate * priv,
    char * section )
```

validate the configuration, given by the section

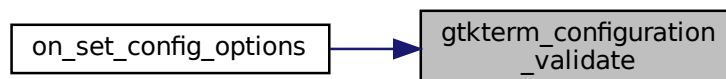
References [CONF_ITEM_SERIAL_BAUDRATE](#), [CONF_ITEM_SERIAL_BITS](#), [CONF_ITEM_SERIAL_STOPBITS](#), [CONF_ITEM_TERM_FONT](#), [CONF_ITEM_TERM_WAIT_DELAY](#), [DEFAULT_BITS](#), [DEFAULT_DELAY](#), [DEFAULT_FONT](#), [DEFAULT_STOPBITS](#), [GtkTermConfigurationItems](#), [GtkTermConfigurationPrivate::key_file](#), [MSG_ERR](#), and [show_message\(\)](#).

Referenced by [on_set_config_options\(\)](#).

Here is the call graph for this function:



Here is the caller graph for this function:



5.23.2.4 on_set_config_options()

```
bool on_set_config_options (
    const char * name,
    const char * value,
    gpointer data,
    GError ** error )
```

Set the config option in the keyfile.

Options are not saved. Point to the third charater ('-' in front of the cli option)

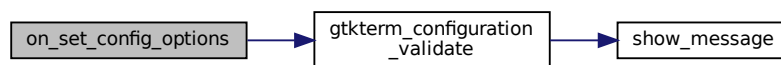
Search index for the option we want to set

Check for max path length. Exit if it is to long. Note: Serial port is also a path to a device.

We should not get here.

References [CONF_ITEM_LAST](#), [CONF_ITEM_SERIAL_BAUDRATE](#), [CONF_ITEM_SERIAL_BITS](#), [CONF_ITEM_SERIAL_DISABLE](#), [CONF_ITEM_SERIAL_FLOW_CONTROL](#), [CONF_ITEM_SERIAL_PARITY](#), [CONF_ITEM_SERIAL_PORT](#), [CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX](#), [CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX](#), [CONF_ITEM_SERIAL_STOPBITS](#), [CONF_ITEM_TERM_ECHO](#), [CONF_ITEM_TERM_RAW_FILENAME](#), [CONF_ITEM_TERM_WAIT_CHAR](#), [CONF_ITEM_TERM_WAIT_DELAY](#), [config](#), [gtkterm_configuration_validate\(\)](#), [GtkTermConfigurationItems](#), and [GtkTermConfigurationPrivate::key_file](#).

Here is the call graph for this function:



5.23.3 Variable Documentation

5.23.3.1 GtkTermConfigurationItems

```
const char GtkTermConfigurationItems[ ][CONF_ITEM_LENGTH]
```

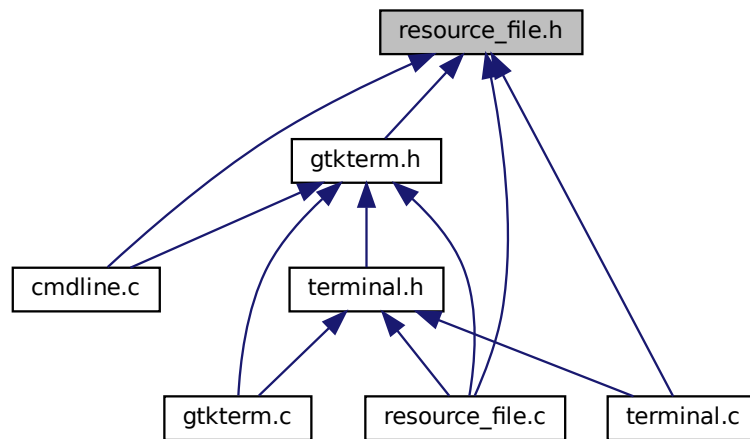
Used configuration options to hold consistency between load/save functions.

Configuration item names.

Referenced by [gtkterm_configuration_default_configuration\(\)](#), [gtkterm_configuration_validate\(\)](#), and [on_set_config_options\(\)](#).

5.24 resource_file.h File Reference

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



Macros

- `#define CONF_ITEM_LENGTH 32`
- `#define DEFAULT_SECTION "default"`
- `#define GTKTERM_TYPE_CONFIGURATION gtkterm_configuration_get_type ()`

Typedefs

- `typedef struct _GtkTermConfiguration GtkTermConfiguration`

Enumerations

- `enum {`
`CONF_ITEM_SERIAL_PORT , CONF_ITEM_SERIAL_BAUDRATE , CONF_ITEM_SERIAL_BITS ,`
`CONF_ITEM_SERIAL_STOPBITS ,`
`CONF_ITEM_SERIAL_PARITY , CONF_ITEM_SERIAL_FLOW_CONTROL , CONF_ITEM_TERM_WAIT_DELAY`
`, CONF_ITEM_TERM_WAIT_CHAR ,`
`CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX , CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX`
`, CONF_ITEM_TERM_MACROS , CONF_ITEM_TERM_RAW_FILENAME ,`
`CONF_ITEM_TERM_ECHO , CONF_ITEM_TERM_CRLF_AUTO , CONF_ITEM_SERIAL_DISABLE_PORT_LOCK`
`, CONF_ITEM_TERM_FONT ,`
`CONF_ITEM_TERM_TIMESTAMP , CONF_ITEM_TERM_BLOCK_CURSOR , CONF_ITEM_TERM_SHOW_CURSOR`
`, CONF_ITEM_TERM_ROWS ,`
`CONF_ITEM_TERM_COLS , CONF_ITEM_TERM_SCROLLBACK , CONF_ITEM_TERM_VISUAL_BELL ,`
`CONF_ITEM_TERM_FOREGROUND_RED ,`
`CONF_ITEM_TERM_FOREGROUND_GREEN , CONF_ITEM_TERM_FOREGROUND_BLUE , CONF_ITEM_TERM_FOREG`
`, CONF_ITEM_TERM_BACKGROUND_RED ,`
`CONF_ITEM_TERM_BACKGROUND_GREEN , CONF_ITEM_TERM_BACKGROUND_BLUE , CONF_ITEM_TERM_BACKG`
`, CONF_ITEM_LAST }`

Define all configuration items which are used in the resource file.

Functions

- [GtkTermConfiguration](#) * [gtkterm_configuration_new](#) (void)
- bool [on_set_config_options](#) (const char *, const char *, gpointer, GError **)
Set the config option in the keyfile.

Variables

- const char [GtkTermConfigurationItems](#) [[[CONF_ITEM_LENGTH](#)]
Configuration item names.

5.24.1 Macro Definition Documentation

5.24.1.1 CONF_ITEM_LENGTH

```
#define CONF_ITEM_LENGTH 32
```

5.24.1.2 DEFAULT_SECTION

```
#define DEFAULT_SECTION "default"
```

5.24.1.3 GTKTERM_TYPE_CONFIGURATION

```
#define GTKTERM_TYPE_CONFIGURATION gtkterm_configuration_get_type ()
```

5.24.2 Typedef Documentation

5.24.2.1 GtkTermConfiguration

```
typedef struct _GtkTermConfiguration GtkTermConfiguration
```

5.24.3 Enumeration Type Documentation

5.24.3.1 anonymous enum

```
anonymous enum
```

Define all configuration items which are used in the resource file.

it is an index to ConfigurationItem.

Enumerator

CONF_ITEM_SERIAL_PORT	
CONF_ITEM_SERIAL_BAUDRATE	
CONF_ITEM_SERIAL_BITS	
CONF_ITEM_SERIAL_STOPBITS	
CONF_ITEM_SERIAL_PARITY	
CONF_ITEM_SERIAL_FLOW_CONTROL	
CONF_ITEM_TERM_WAIT_DELAY	
CONF_ITEM_TERM_WAIT_CHAR	
CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX	
CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX	
CONF_ITEM_TERM_MACROS	
CONF_ITEM_TERM_RAW_FILENAME	
CONF_ITEM_TERM_ECHO	
CONF_ITEM_TERM_CRLF_AUTO	
CONF_ITEM_SERIAL_DISABLE_PORT_LOCK	
CONF_ITEM_TERM_FONT	
CONF_ITEM_TERM_TIMESTAMP	
CONF_ITEM_TERM_BLOCK_CURSOR	
CONF_ITEM_TERM_SHOW_CURSOR	
CONF_ITEM_TERM_ROWS	
CONF_ITEM_TERM_COLS	
CONF_ITEM_TERM_SCROLLBACK	
CONF_ITEM_TERM_VISUAL_BELL	
CONF_ITEM_TERM_FOREGROUND_RED	
CONF_ITEM_TERM_FOREGROUND_GREEN	
CONF_ITEM_TERM_FOREGROUND_BLUE	
CONF_ITEM_TERM_FOREGROUND_ALPHA	
CONF_ITEM_TERM_BACKGROUND_RED	
CONF_ITEM_TERM_BACKGROUND_GREEN	
CONF_ITEM_TERM_BACKGROUND_BLUE	
CONF_ITEM_TERM_BACKGROUND_ALPHA	
CONF_ITEM_LAST	Checking as last item in the list.

5.24.4 Function Documentation

5.24.4.1 gtkterm_configuration_new()

```

GtkTermConfiguration * gtkterm_configuration_new (
    void )

```

5.24.4.2 on_set_config_options()

```
bool on_set_config_options (
    const char * name,
    const char * value,
    gpointer data,
    GError ** error )
```

Set the config option in the keyfile.

Options are not saved. Point to the third character ('-' in front of the cli option)

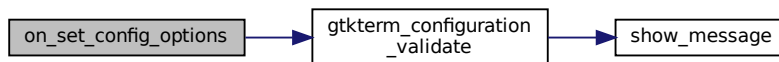
Search index for the option we want to set

Check for max path length. Exit if it is to long. Note: Serial port is also a path to a device.

We should not get here.

References [CONF_ITEM_LAST](#), [CONF_ITEM_SERIAL_BAUDRATE](#), [CONF_ITEM_SERIAL_BITS](#), [CONF_ITEM_SERIAL_DISABLE](#), [CONF_ITEM_SERIAL_FLOW_CONTROL](#), [CONF_ITEM_SERIAL_PARITY](#), [CONF_ITEM_SERIAL_PORT](#), [CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX](#), [CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX](#), [CONF_ITEM_SERIAL_STOPBITS](#), [CONF_ITEM_TERM_ECHO](#), [CONF_ITEM_TERM_RAW_FILENAME](#), [CONF_ITEM_TERM_WAIT_CHAR](#), [CONF_ITEM_TERM_WAIT_DELAY](#), [config](#), [gtkterm_configuration_validate\(\)](#), [GtkTermConfigurationItems](#), and [GtkTermConfigurationPrivate::key_file](#).

Here is the call graph for this function:



5.24.5 Variable Documentation

5.24.5.1 GtkTermConfigurationItems

```
const char GtkTermConfigurationItems[][CONF_ITEM_LENGTH] [extern]
```

Configuration item names.

Configuration item names.

Referenced by [gtkterm_configuration_default_configuration\(\)](#), [gtkterm_configuration_validate\(\)](#), and [on_set_config_options\(\)](#).

5.25 resource_file.h

[Go to the documentation of this file.](#)

```

1  /*****
2  * resource_file.h
3  * -----
4  *          GTKTerm Software
5  *          (c) Julien Schmitt
6  *
7  * -----
8  *
9  * \brief Purpose
10 *      Load and save configuration file
11 *      - Header file -
12 *
13 *****/
14
15 #ifndef RESOURCE_FILE_H_
16 #define RESOURCE_FILE_H_
17
18 #define CONF_ITEM_LENGTH      32
19 #define DEFAULT_SECTION      "default"      ///! Default section if not specified
20
21 ///! Define all configuration items which are used
22 ///! in the resource file. it is an index to ConfigurationItem.
23 enum {
24     CONF_ITEM_SERIAL_PORT,
25     CONF_ITEM_SERIAL_BAUDRATE,
26     CONF_ITEM_SERIAL_BITS,
27     CONF_ITEM_SERIAL_STOPBITS,
28     CONF_ITEM_SERIAL_PARITY,
29     CONF_ITEM_SERIAL_FLOW_CONTROL,
30     CONF_ITEM_TERM_WAIT_DELAY,
31     CONF_ITEM_TERM_WAIT_CHAR,
32     CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX,
33     CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX,
34     CONF_ITEM_TERM_MACROS,
35     CONF_ITEM_TERM_RAW_FILENAME,
36     CONF_ITEM_TERM_ECHO,
37     CONF_ITEM_TERM_CRLF_AUTO,
38     CONF_ITEM_SERIAL_DISABLE_PORT_LOCK,
39     CONF_ITEM_TERM_FONT,
40     CONF_ITEM_TERM_TIMESTAMP,
41     CONF_ITEM_TERM_BLOCK_CURSOR,
42     CONF_ITEM_TERM_SHOW_CURSOR,
43     CONF_ITEM_TERM_ROWS,
44     CONF_ITEM_TERM_COLS,
45     CONF_ITEM_TERM_SCROLLBACK,
46     CONF_ITEM_TERM_VISUAL_BELL,
47     CONF_ITEM_TERM_FOREGROUND_RED,
48     CONF_ITEM_TERM_FOREGROUND_GREEN,
49     CONF_ITEM_TERM_FOREGROUND_BLUE,
50     CONF_ITEM_TERM_FOREGROUND_ALPHA,
51     CONF_ITEM_TERM_BACKGROUND_RED,
52     CONF_ITEM_TERM_BACKGROUND_GREEN,
53     CONF_ITEM_TERM_BACKGROUND_BLUE,
54     CONF_ITEM_TERM_BACKGROUND_ALPHA,
55     CONF_ITEM_LAST          ///! Checking as last item in the list.
56 };
57
58 ///! Configuration item names.
59 extern const char GtkTermConfigurationItems[][CONF_ITEM_LENGTH];
60
61 G_BEGIN_DECLS
62
63 #define GTKTERM_TYPE_CONFIGURATION gtkterm_configuration_get_type ()
64 G_DECLARE_FINAL_TYPE (GtkTermConfiguration, gtkterm_configuration, GTKTERM, CONFIGURATION, GObject)
65 typedef struct _GtkTermConfiguration GtkTermConfiguration;
66
67 GtkTermConfiguration *gtkterm_configuration_new (void);
68
69 bool on_set_config_options (const char *, const char *, gpointer, GError **);
70
71 G_END_DECLS
72
73 #endif

```

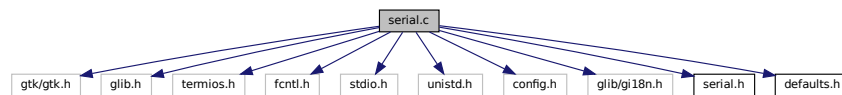
5.26 serial.c File Reference

```

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

```

```
#include <termios.h>
#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include <config.h>
#include <glib/gi18n.h>
#include "serial.h"
#include "defaults.h"
Include dependency graph for serial.c:
```



Classes

- struct [GtkTermSerialPortPrivate](#)
- struct [_GtkTermSerialPort](#)
- struct [_GtkTermSerialPortClass](#)

Enumerations

- enum { [PROP_0](#) , [PROP_PORT_CONFIG](#) , [N_PROPS](#) }

Functions

- [GtkTermSerialPort *](#) [gtkterm_serial_port_new](#) ([port_config_t *](#)port_conf)
- char * [gtkterm_serial_port_get_string](#) ([GtkTermSerialPort *](#)self)
- int [gtkterm_serial_port_status](#) ([GtkTermSerialPort *](#)self)

5.26.1 Enumeration Type Documentation

5.26.1.1 anonymous enum

anonymous enum

Enumerator

PROP_0	
PROP_PORT_CONFIG	
N_PROPS	

5.26.2 Function Documentation

5.26.2.1 gtkterm_serial_port_get_string()

```
char * gtkterm_serial_port_get_string (
    GtkTermSerialPort * self )
```

References [port_config_t::baudrate](#), [port_config_t::bits](#), [port_config_t::parity](#), [port_config_t::port](#), [GtkTermSerialPortPrivate::port_conf](#), [GtkTermSerialPortPrivate::serial_port_fd](#), and [port_config_t::stopbits](#).

5.26.2.2 gtkterm_serial_port_new()

```
GtkTermSerialPort * gtkterm_serial_port_new (
    port_config_t * port_conf )
```

References [GTKTERM_TYPE_SERIAL_PORT](#).

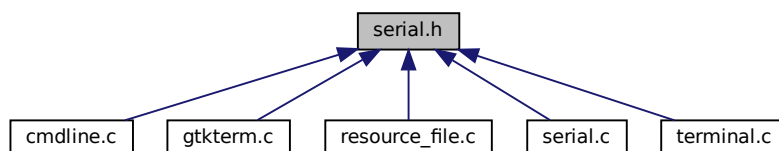
5.26.2.3 gtkterm_serial_port_status()

```
int gtkterm_serial_port_status (
    GtkTermSerialPort * self )
```

References [GtkTermSerialPortPrivate::serial_port_fd](#).

5.27 serial.h File Reference

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



Classes

- struct [port_config_t](#)

Macros

- `#define GTKTERM_TYPE_SERIAL_PORT gtkterm_serial_port_get_type ()`

Typedefs

- `typedef typedefG_BEGIN_DECLS struct _GtkTermSerialPort GtkTermSerialPort`

Functions

- `GtkTermSerialPort * gtkterm_serial_port_new (port_config_t *)`
- `G_END_DECLS char * gtkterm_serial_port_get_string (GtkTermSerialPort *)`
- `int gtkterm_serial_port_status (GtkTermSerialPort *)`

5.27.1 Macro Definition Documentation

5.27.1.1 GTKTERM_TYPE_SERIAL_PORT

```
#define GTKTERM_TYPE_SERIAL_PORT gtkterm_serial_port_get_type ()
```

5.27.2 Typedef Documentation

5.27.2.1 GtkTermSerialPort

```
typedef typedefG_BEGIN_DECLS struct _GtkTermSerialPort GtkTermSerialPort
```

5.27.3 Function Documentation

5.27.3.1 gtkterm_serial_port_get_string()

```
G_END_DECLS char * gtkterm_serial_port_get_string (  
    GtkTermSerialPort * self )
```

References [port_config_t::baudrate](#), [port_config_t::bits](#), [port_config_t::parity](#), [port_config_t::port](#), [GtkTermSerialPortPrivate::port_conf](#), [GtkTermSerialPortPrivate::serial_port_fd](#), and [port_config_t::stopbits](#).

5.27.3.2 gtkterm_serial_port_new()

```
GtkTermSerialPort * gtkterm_serial_port_new (
    port_config_t * port_conf )
```

References [GTKTERM_TYPE_SERIAL_PORT](#).

5.27.3.3 gtkterm_serial_port_status()

```
int gtkterm_serial_port_status (
    GtkTermSerialPort * self )
```

References [GtkTermSerialPortPrivate::serial_port_fd](#).

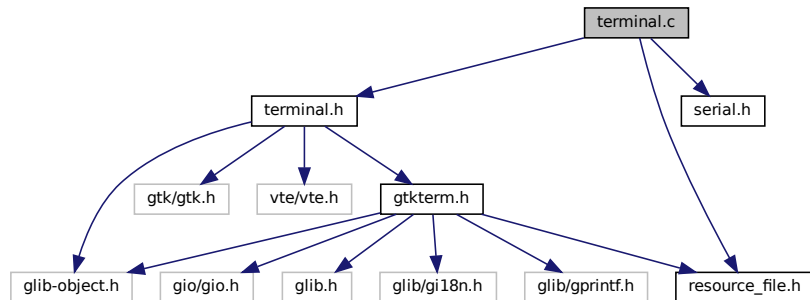
5.28 serial.h

[Go to the documentation of this file.](#)

```
1  /*****
2  /* serial.h
3  /* -----
4  /*      GTKTerm Software
5  /*      (c) Julien Schmitt
6  /*
7  /* -----
8  /*
9  /* Purpose
10 /*      Serial port access functions
11 /*      - Header file -
12 /*
13 /* *****/
14
15 #ifndef SERIAL_H_
16 #define SERIAL_H_
17
18 typedef struct
19 {
20     char *port;
21     long int baudrate;           // 300 - 600 - 1200 - ... - 2000000
22     int bits;                   // 5 - 6 - 7 - 8
23     int stopbits;               // 1 - 2
24     int parity;                 // 0 : None, 1 : Odd, 2 : Even
25     int flow_control;           // 0 : None, 1 : Xon/Xoff, 2 : RTS/CTS, 3 : RS485halfduplex
26     int rs485_rts_time_before_transmit;
27     int rs485_rts_time_after_transmit;
28     bool disable_port_lock;
29 } port_config_t;
30
31 #define G_BEGIN_DECLS
32
33 typedef struct _GtkTermSerialPort GtkTermSerialPort;
34
35 #define GTKTERM_TYPE_SERIAL_PORT gtkterm_serial_port_get_type ()
36 #define G_DECLARE_FINAL_TYPE (GtkTermSerialPort, gtkterm_serial_port, GTKTERM, SERIAL_PORT, GObject)
37
38 GtkTermSerialPort *gtkterm_serial_port_new (port_config_t *);
39
40 #define G_END_DECLS
41
42 char* gtkterm_serial_port_get_string (GtkTermSerialPort *);
43 int gtkterm_serial_port_status (GtkTermSerialPort *);
44
45 #endif
```

5.29 terminal.c File Reference

```
#include "terminal.h"
#include "serial.h"
#include "resource_file.h"
Include dependency graph for terminal.c:
```



Classes

- struct [GtkTermTerminalPrivate](#)
- struct [_GtkTermTerminal](#)
- struct [_GtkTermTerminalClass](#)

Enumerations

- enum {
[PROP_0](#) , [PROP_SECTION](#) , [PROP_GTKTERM_APP](#) , [PROP_MAIN_WINDOW](#) ,
[N_PROPS](#) }

Functions

- [GtkTermTerminal](#) * [gtkterm_terminal_new](#) (char *section, [GtkTerm](#) *gtkterm_app, [GtkTermWindow](#) *main↔_window)

5.29.1 Enumeration Type Documentation

5.29.1.1 anonymous enum

anonymous enum

Enumerator

PROP_0	
PROP_SECTION	
PROP_GTKTERM_APP	
PROP_MAIN_WINDOW	
N_PROPS	

5.29.2 Function Documentation

5.29.2.1 gtkterm_terminal_new()

```

GtkTermTerminal * gtkterm_terminal_new (
    char * section,
    GtkTerm * gtkterm_app,
    GtkTermWindow * main_window )

```

References [GTKTERM_TYPE_TERMINAL](#).

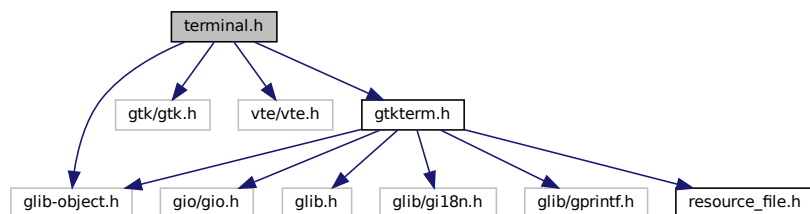
5.30 terminal.h File Reference

```

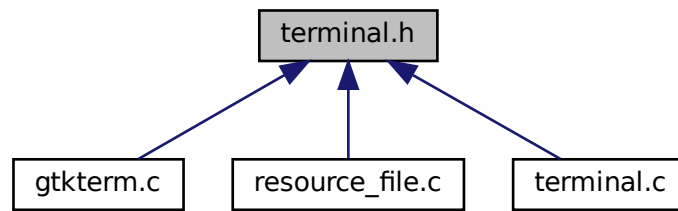
#include <glib-object.h>
#include <gtk/gtk.h>
#include <vte/vte.h>
#include "gtkterm.h"

```

Include dependency graph for terminal.h:



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



Classes

- struct [term_config_t](#)

Macros

- #define [GTKTERM_TYPE_TERMINAL](#) [gtkterm_terminal_get_type\(\)](#)

Functions

- GtkTermTerminal * [gtkterm_terminal_new](#) (char *, [GtkTerm](#) *, [GtkTermWindow](#) *)

5.30.1 Macro Definition Documentation

5.30.1.1 GTKTERM_TYPE_TERMINAL

```
#define GTKTERM_TYPE_TERMINAL gtkterm_terminal_get_type()
```

5.30.2 Function Documentation

5.30.2.1 gtkterm_terminal_new()

```
GtkTermTerminal * gtkterm_terminal_new (  
    char * section,  
    GtkTerm * gtkterm_app,  
    GtkTermWindow * main_window )
```

References [GTKTERM_TYPE_TERMINAL](#).

5.31 terminal.h

[Go to the documentation of this file.](#)

```

1  /*****
2  /* terminal.h
3  /* -----
4  /*          GTKTerm Software
5  /*          (c) Julien Schmitt
6  /*
7  /* -----
8  /*
9  /* Purpose
10 /*      Handles all VTE in/output to/from serial port
11 /*      - Header file -
12 /*
13 /*****/
14 #ifndef TERMINAL_H
15 #define TERMINAL_H
16
17 #include <glib-object.h>
18 #include <gtk/gtk.h>
19 #include <vte/vte.h>
20
21 #include "gtkterm.h"
22
23 typedef struct
24 {
25     bool block_cursor;
26     bool show_cursor;
27     char char_queue;           // character in queue
28     bool echo;                 // echo local
29     bool crlfauto;             // line feed auto
30     bool timestamp;
31     int delay;                 // end of char delay: in ms
32     int rows;
33     int columns;
34     int scrollbar;
35     bool visual_bell;
36     GdkRGBA foreground_color;
37     GdkRGBA background_color;
38     PangoFontDescription *font;
39 }
40 } term_config_t;
41
42 G_BEGIN_DECLS
43
44 #define GTKTERM_TYPE_TERMINAL gtkterm_terminal_get_type()
45 G_DECLARE_FINAL_TYPE (GtkTermTerminal, gtkterm_terminal, GTKTERM, TERMINAL, VteTerminal)
46
47 GtkTermTerminal *gtkterm_terminal_new (char *, GtkTerm *, GtkTermWindow *);
48
49 G_END_DECLS
50
51 #endif // TERMINAL_H

```


Index

- [_GtkTerm](#), [9](#)
 - [action_group](#), [10](#)
 - [config](#), [10](#)
 - [g_config_group](#), [10](#)
 - [g_port_group](#), [10](#)
 - [g_term_group](#), [11](#)
 - [parent_instance](#), [11](#)
 - [section](#), [11](#)
- [_GtkTermConfiguration](#), [11](#)
 - [parent_instance](#), [12](#)
- [_GtkTermConfigurationClass](#), [12](#)
 - [parent_class](#), [12](#)
- [_GtkTermSerialPort](#), [13](#)
 - [parent_instance](#), [13](#)
- [_GtkTermSerialPortClass](#), [13](#)
 - [parent_class](#), [14](#)
- [_GtkTermTerminal](#), [14](#)
 - [vte_object](#), [14](#)
- [_GtkTermTerminalClass](#), [15](#)
 - [vte_class](#), [15](#)
- [_GtkTermWindow](#), [15](#)
 - [action_group](#), [17](#)
 - [fullscreen](#), [17](#)
 - [height](#), [17](#)
 - [infobar](#), [17](#)
 - [maximized](#), [17](#)
 - [menubutton](#), [17](#)
 - [message](#), [18](#)
 - [parent_instance](#), [18](#)
 - [scrolled_window](#), [18](#)
 - [search_bar](#), [18](#)
 - [status_config](#), [18](#)
 - [status_config_message](#), [18](#)
 - [status_message](#), [19](#)
 - [status_serial_signal](#), [19](#)
 - [statusbox](#), [19](#)
 - [terminal_window](#), [19](#)
 - [toolmenu](#), [19](#)
 - [width](#), [19](#)
- [action](#)
 - [macro_t](#), [25](#)
- [action_group](#)
 - [_GtkTerm](#), [10](#)
 - [_GtkTermWindow](#), [17](#)
- [add_input](#)
 - [files.h](#), [46](#)
- [add_shortcuts](#)
 - [macros.h](#), [60](#)
- [app](#)
 - [GtkTermTerminalPrivate](#), [23](#)
- [ASCII_VIEW](#)
 - [interface.h](#), [54](#)
- [background_color](#)
 - [term_config_t](#), [29](#)
- [baudrate](#)
 - [port_config_t](#), [26](#)
- [bits](#)
 - [port_config_t](#), [26](#)
- [block_cursor](#)
 - [term_config_t](#), [29](#)
- [buffer.c](#), [31](#)
 - [clear_buffer](#), [32](#)
 - [clear_func](#), [35](#)
 - [create_buffer](#), [32](#)
 - [delete_buffer](#), [33](#)
 - [insert_timestamp](#), [33](#)
 - [MAX_SECTION_LENGTH](#), [32](#)
 - [overlapped](#), [35](#)
 - [put_chars](#), [33](#)
 - [set_clear_func](#), [33](#)
 - [set_display_func](#), [33](#)
 - [timestamp_on](#), [35](#)
 - [TIMESTAMP_SIZE](#), [32](#)
 - [unset_clear_func](#), [33](#)
 - [unset_display_func](#), [34](#)
 - [virt_col_pos](#), [35](#)
 - [write_buffer](#), [34](#)
 - [write_buffer_with_func](#), [34](#)
 - [write_func](#), [35](#)
- [buffer.h](#), [36](#), [39](#)
 - [BUFFER_SIZE](#), [36](#)
 - [clear_buffer](#), [36](#)
 - [create_buffer](#), [37](#)
 - [delete_buffer](#), [37](#)
 - [put_chars](#), [37](#)
 - [set_clear_func](#), [37](#)
 - [set_display_func](#), [37](#)
 - [unset_clear_func](#), [37](#)
 - [unset_display_func](#), [38](#)
 - [write_buffer](#), [38](#)
 - [write_buffer_with_func](#), [38](#)
- [BUFFER_LENGTH](#)
 - [defaults.h](#), [42](#)
 - [resource_file.c](#), [63](#)
- [BUFFER_SIZE](#)
 - [buffer.h](#), [36](#)
- [char_queue](#)

- term_config_t, 29
- check_keyfile
 - resource_file.c, 64
- clear_buffer
 - buffer.c, 32
 - buffer.h, 36
- clear_func
 - buffer.c, 35
- closure
 - macro_t, 25
- cmdline.c, 39
 - gtkterm_add_cmdline_options, 40
- cmdline.h, 40, 41
 - g_term_group, 41
 - gtkterm_add_cmdline_options, 40
- COLUMN_ACTION
 - macros.c, 57
- COLUMN_SHORTCUT
 - macros.c, 57
- columns
 - term_config_t, 29
- CONF_ITEM_LAST
 - resource_file.h, 69
- CONF_ITEM_LENGTH
 - resource_file.h, 68
- CONF_ITEM_SERIAL_BAUDRATE
 - resource_file.h, 69
- CONF_ITEM_SERIAL_BITS
 - resource_file.h, 69
- CONF_ITEM_SERIAL_DISABLE_PORT_LOCK
 - resource_file.h, 69
- CONF_ITEM_SERIAL_FLOW_CONTROL
 - resource_file.h, 69
- CONF_ITEM_SERIAL_PARITY
 - resource_file.h, 69
- CONF_ITEM_SERIAL_PORT
 - resource_file.h, 69
- CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX
 - resource_file.h, 69
- CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX
 - resource_file.h, 69
- CONF_ITEM_SERIAL_STOPBITS
 - resource_file.h, 69
- CONF_ITEM_TERM_BACKGROUND_ALPHA
 - resource_file.h, 69
- CONF_ITEM_TERM_BACKGROUND_BLUE
 - resource_file.h, 69
- CONF_ITEM_TERM_BACKGROUND_GREEN
 - resource_file.h, 69
- CONF_ITEM_TERM_BACKGROUND_RED
 - resource_file.h, 69
- CONF_ITEM_TERM_BLOCK_CURSOR
 - resource_file.h, 69
- CONF_ITEM_TERM_COLS
 - resource_file.h, 69
- CONF_ITEM_TERM_CRLF_AUTO
 - resource_file.h, 69
- CONF_ITEM_TERM_ECHO
 - resource_file.h, 69
- CONF_ITEM_TERM_FONT
 - resource_file.h, 69
- CONF_ITEM_TERM_FOREGROUND_ALPHA
 - resource_file.h, 69
- CONF_ITEM_TERM_FOREGROUND_BLUE
 - resource_file.h, 69
- CONF_ITEM_TERM_FOREGROUND_GREEN
 - resource_file.h, 69
- CONF_ITEM_TERM_FOREGROUND_RED
 - resource_file.h, 69
- CONF_ITEM_TERM_MACROS
 - resource_file.h, 69
- CONF_ITEM_TERM_RAW_FILENAME
 - resource_file.h, 69
- CONF_ITEM_TERM_ROWS
 - resource_file.h, 69
- CONF_ITEM_TERM_SCROLLBACK
 - resource_file.h, 69
- CONF_ITEM_TERM_SHOW_CURSOR
 - resource_file.h, 69
- CONF_ITEM_TERM_TIMESTAMP
 - resource_file.h, 69
- CONF_ITEM_TERM_VISUAL_BELL
 - resource_file.h, 69
- CONF_ITEM_TERM_WAIT_CHAR
 - resource_file.h, 69
- CONF_ITEM_TERM_WAIT_DELAY
 - resource_file.h, 69
- config
 - _GtkTerm, 10
 - interface.c, 53
- config_file
 - GtkTermConfigurationPrivate, 20
- CONFIGURATION_FILENAME
 - resource_file.c, 64
- convert_macros_to_string
 - macros.c, 57
 - macros.h, 60
- convert_string_to_macros
 - macros.c, 58
 - macros.h, 60
- create_buffer
 - buffer.c, 32
 - buffer.h, 37
- crlfauto
 - term_config_t, 29
- DEFAULT_BAUDRATE
 - defaults.h, 42
- DEFAULT_BITS
 - defaults.h, 42
- DEFAULT_CHAR
 - defaults.h, 42
- DEFAULT_DELAY
 - defaults.h, 43
- DEFAULT_DELAY_RS485
 - defaults.h, 43
- DEFAULT_ECHO

- defaults.h, 43
- default_filename
 - files.c, 46
 - files.h, 47
- DEFAULT_FLOW
 - defaults.h, 43
- DEFAULT_FONT
 - defaults.h, 43
- DEFAULT_PARITY
 - defaults.h, 43
- DEFAULT_PORT
 - defaults.h, 43
- DEFAULT_SCROLLBACK
 - defaults.h, 44
- DEFAULT_SECTION
 - resource_file.h, 68
- DEFAULT_STOPBITS
 - defaults.h, 44
- DEFAULT_VISUAL_BELL
 - defaults.h, 44
- defaults.h, 41, 45
 - BUFFER_LENGTH, 42
 - DEFAULT_BAUDRATE, 42
 - DEFAULT_BITS, 42
 - DEFAULT_CHAR, 42
 - DEFAULT_DELAY, 43
 - DEFAULT_DELAY_RS485, 43
 - DEFAULT_ECHO, 43
 - DEFAULT_FLOW, 43
 - DEFAULT_FONT, 43
 - DEFAULT_PARITY, 43
 - DEFAULT_PORT, 43
 - DEFAULT_SCROLLBACK, 44
 - DEFAULT_STOPBITS, 44
 - DEFAULT_VISUAL_BELL, 44
 - LINE_FEED, 44
 - MAX_SECTION_LENGTH, 44
 - POLL_DELAY, 44
 - RECEIVE_BUFFER, 44
 - TRANSMIT_BUFFER, 45
- delay
 - term_config_t, 29
- delete_buffer
 - buffer.c, 33
 - buffer.h, 37
- disable_port_lock
 - port_config_t, 27
- display
 - interface.h, 55
- echo
 - term_config_t, 29
- files.c, 45
 - default_filename, 46
- files.h, 46, 47
 - add_input, 46
 - default_filename, 47
 - save_raw_file, 46
 - send_raw_file, 46
 - waiting_for_char, 47
- flow_control
 - port_config_t, 27
- font
 - term_config_t, 29
- foreground_color
 - term_config_t, 30
- fullscreen
 - _GtkTermWindow, 17
- g_config_group
 - _GtkTerm, 10
- g_port_group
 - _GtkTerm, 10
- g_term_group
 - _GtkTerm, 11
 - cmdline.h, 41
- get_shortcuts
 - macros.c, 58
 - macros.h, 61
- GtkTerm
 - gtkterm.h, 50
- gtkterm.c, 48
 - gtkterm_signals, 49
 - main, 48
 - set_window_title, 48
- gtkterm.h, 49, 51
 - GtkTerm, 50
 - gtkterm_signals, 51
 - GTKTERM_TYPE_APP, 50
 - GTKTERM_TYPE_GTKTERM_WINDOW, 50
 - GtkTermWindow, 51
 - LAST_GTKTERM_SIGNAL, 51
 - SIGNAL_GTKTERM_CONFIG_SERIAL, 51
 - SIGNAL_GTKTERM_CONFIG_TERMINAL, 51
 - SIGNAL_GTKTERM_LOAD_CONFIG, 51
 - SIGNAL_GTKTERM_PRINT_SECTION, 51
 - SIGNAL_GTKTERM_REMOVE_SECTION, 51
 - SIGNAL_GTKTERM_SAVE_CONFIG, 51
 - SIGNAL_GTKTERM_TERMINAL_CHANGED, 51
- gtkterm_add_cmdline_options
 - cmdline.c, 40
 - cmdline.h, 40
- gtkterm_configuration_default_configuration
 - resource_file.c, 64
- gtkterm_configuration_new
 - resource_file.h, 69
- gtkterm_configuration_validate
 - resource_file.c, 64
- gtkterm_serial_port_get_string
 - serial.c, 73
 - serial.h, 74
- gtkterm_serial_port_new
 - serial.c, 73
 - serial.h, 74
- gtkterm_serial_port_status
 - serial.c, 73
 - serial.h, 75

- gtkterm_signals
 - gtkterm.c, [49](#)
 - gtkterm.h, [51](#)
- gtkterm_terminal_new
 - terminal.c, [77](#)
 - terminal.h, [78](#)
- GTKTERM_TYPE_APP
 - gtkterm.h, [50](#)
- GTKTERM_TYPE_CONFIGURATION
 - resource_file.h, [68](#)
- GTKTERM_TYPE_GTKTERM_WINDOW
 - gtkterm.h, [50](#)
- GTKTERM_TYPE_SERIAL_PORT
 - serial.h, [74](#)
- GTKTERM_TYPE_TERMINAL
 - terminal.h, [78](#)
- GtkTermConfiguration
 - resource_file.h, [68](#)
- GtkTermConfigurationItems
 - resource_file.c, [66](#)
 - resource_file.h, [70](#)
- GtkTermConfigurationPrivate, [20](#)
 - config_file, [20](#)
 - key_file, [20](#)
- GtkTermSerialPort
 - serial.h, [74](#)
- GtkTermSerialPortPrivate, [21](#)
 - port_conf, [22](#)
 - serial_port_fd, [22](#)
 - termios_save, [22](#)
- GtkTermTerminalPrivate, [22](#)
 - app, [23](#)
 - main_window, [23](#)
 - port_conf, [23](#)
 - section, [23](#)
 - serial_port, [23](#)
 - term_conf, [24](#)
 - view_mode, [24](#)
- GtkTermWindow
 - gtkterm.h, [51](#)
- height
 - _GtkTermWindow, [17](#)
- HEXADECIMAL_VIEW
 - interface.h, [55](#)
- infobar
 - _GtkTermWindow, [17](#)
- insert_timestamp
 - buffer.c, [33](#)
- interface.c, [52](#)
 - config, [53](#)
 - show_message, [53](#)
 - timestamp_on, [53](#)
 - virt_col_pos, [54](#)
- interface.h, [54](#), [56](#)
 - ASCII_VIEW, [54](#)
 - display, [55](#)
 - HEXADECIMAL_VIEW, [55](#)
 - MSG_ERR, [55](#)
 - MSG_WRN, [55](#)
 - show_message, [55](#)
 - Text, [56](#)
- key_file
 - GtkTermConfigurationPrivate, [20](#)
- LAST_GTKTERM_SIGNAL
 - gtkterm.h, [51](#)
- LINE_FEED
 - defaults.h, [44](#)
- macro_count
 - macros.c, [58](#)
 - macros.h, [61](#)
- macro_t, [24](#)
 - action, [25](#)
 - closure, [25](#)
 - shortcut, [25](#)
- macros
 - macros.c, [59](#)
 - macros.h, [62](#)
- macros.c, [56](#)
 - COLUMN_ACTION, [57](#)
 - COLUMN_SHORTCUT, [57](#)
 - convert_macros_to_string, [57](#)
 - convert_string_to_macros, [58](#)
 - get_shortcuts, [58](#)
 - macro_count, [58](#)
 - macros, [59](#)
 - nr_of_macros, [59](#)
 - NUM_COLUMNS, [57](#)
 - remove_shortcuts, [58](#)
- macros.h, [59](#), [62](#)
 - add_shortcuts, [60](#)
 - convert_macros_to_string, [60](#)
 - convert_string_to_macros, [60](#)
 - get_shortcuts, [61](#)
 - macro_count, [61](#)
 - macros, [62](#)
 - remove_shortcuts, [61](#)
- main
 - gtkterm.c, [48](#)
- main_window
 - GtkTermTerminalPrivate, [23](#)
- MAX_SECTION_LENGTH
 - buffer.c, [32](#)
 - defaults.h, [44](#)
- maximized
 - _GtkTermWindow, [17](#)
- menubutton
 - _GtkTermWindow, [17](#)
- message
 - _GtkTermWindow, [18](#)
- MSG_ERR
 - interface.h, [55](#)
- MSG_WRN
 - interface.h, [55](#)

- N_PROPS
 - serial.c, [72](#)
 - terminal.c, [77](#)
- nr_of_macros
 - macros.c, [59](#)
- NUM_COLUMNS
 - macros.c, [57](#)
- on_set_config_options
 - resource_file.c, [65](#)
 - resource_file.h, [69](#)
- overlapped
 - buffer.c, [35](#)
- parent_class
 - _GtkTermConfigurationClass, [12](#)
 - _GtkTermSerialPortClass, [14](#)
- parent_instance
 - _GtkTerm, [11](#)
 - _GtkTermConfiguration, [12](#)
 - _GtkTermSerialPort, [13](#)
 - _GtkTermWindow, [18](#)
- parity
 - port_config_t, [27](#)
- POLL_DELAY
 - defaults.h, [44](#)
- port
 - port_config_t, [27](#)
- port_conf
 - GtkTermSerialPortPrivate, [22](#)
 - GtkTermTerminalPrivate, [23](#)
- port_config_t, [26](#)
 - baudrate, [26](#)
 - bits, [26](#)
 - disable_port_lock, [27](#)
 - flow_control, [27](#)
 - parity, [27](#)
 - port, [27](#)
 - rs485_rts_time_after_transmit, [27](#)
 - rs485_rts_time_before_transmit, [27](#)
 - stopbits, [27](#)
- PROP_0
 - serial.c, [72](#)
 - terminal.c, [77](#)
- PROP_GTKTERM_APP
 - terminal.c, [77](#)
- PROP_MAIN_WINDOW
 - terminal.c, [77](#)
- PROP_PORT_CONFIG
 - serial.c, [72](#)
- PROP_SECTION
 - terminal.c, [77](#)
- put_chars
 - buffer.c, [33](#)
 - buffer.h, [37](#)
- README.md, [31](#)
- README.source, [31](#)
- RECEIVE_BUFFER
 - defaults.h, [44](#)
- remove_shortcuts
 - macros.c, [58](#)
 - macros.h, [61](#)
- resource_file.c, [62](#)
 - BUFFER_LENGTH, [63](#)
 - check_keyfile, [64](#)
 - CONFIGURATION_FILENAME, [64](#)
 - gtkterm_configuration_default_configuration, [64](#)
 - gtkterm_configuration_validate, [64](#)
 - GtkTermConfigurationItems, [66](#)
 - on_set_config_options, [65](#)
- resource_file.h, [67](#), [71](#)
 - CONF_ITEM_LAST, [69](#)
 - CONF_ITEM_LENGTH, [68](#)
 - CONF_ITEM_SERIAL_BAUDRATE, [69](#)
 - CONF_ITEM_SERIAL_BITS, [69](#)
 - CONF_ITEM_SERIAL_DISABLE_PORT_LOCK, [69](#)
 - CONF_ITEM_SERIAL_FLOW_CONTROL, [69](#)
 - CONF_ITEM_SERIAL_PARITY, [69](#)
 - CONF_ITEM_SERIAL_PORT, [69](#)
 - CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX, [69](#)
 - CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX, [69](#)
 - CONF_ITEM_SERIAL_STOPBITS, [69](#)
 - CONF_ITEM_TERM_BACKGROUND_ALPHA, [69](#)
 - CONF_ITEM_TERM_BACKGROUND_BLUE, [69](#)
 - CONF_ITEM_TERM_BACKGROUND_GREEN, [69](#)
 - CONF_ITEM_TERM_BACKGROUND_RED, [69](#)
 - CONF_ITEM_TERM_BLOCK_CURSOR, [69](#)
 - CONF_ITEM_TERM_COLS, [69](#)
 - CONF_ITEM_TERM_CRLF_AUTO, [69](#)
 - CONF_ITEM_TERM_ECHO, [69](#)
 - CONF_ITEM_TERM_FONT, [69](#)
 - CONF_ITEM_TERM_FOREGROUND_ALPHA, [69](#)
 - CONF_ITEM_TERM_FOREGROUND_BLUE, [69](#)
 - CONF_ITEM_TERM_FOREGROUND_GREEN, [69](#)
 - CONF_ITEM_TERM_FOREGROUND_RED, [69](#)
 - CONF_ITEM_TERM_MACROS, [69](#)
 - CONF_ITEM_TERM_RAW_FILENAME, [69](#)
 - CONF_ITEM_TERM_ROWS, [69](#)
 - CONF_ITEM_TERM_SCROLLBACK, [69](#)
 - CONF_ITEM_TERM_SHOW_CURSOR, [69](#)
 - CONF_ITEM_TERM_TIMESTAMP, [69](#)
 - CONF_ITEM_TERM_VISUAL_BELL, [69](#)
 - CONF_ITEM_TERM_WAIT_CHAR, [69](#)
 - CONF_ITEM_TERM_WAIT_DELAY, [69](#)
 - DEFAULT_SECTION, [68](#)
 - gtkterm_configuration_new, [69](#)
 - GTKTERM_TYPE_CONFIGURATION, [68](#)
 - GtkTermConfiguration, [68](#)
 - GtkTermConfigurationItems, [70](#)
 - on_set_config_options, [69](#)
- rows

- term_config_t, 30
- rs485_rts_time_after_transmit
 - port_config_t, 27
- rs485_rts_time_before_transmit
 - port_config_t, 27
- save_raw_file
 - files.h, 46
- scrollback
 - term_config_t, 30
- scrolled_window
 - _GtkTermWindow, 18
- search_bar
 - _GtkTermWindow, 18
- section
 - _GtkTerm, 11
 - GtkTermTerminalPrivate, 23
- send_raw_file
 - files.h, 46
- serial.c, 71
 - gtkterm_serial_port_get_string, 73
 - gtkterm_serial_port_new, 73
 - gtkterm_serial_port_status, 73
 - N_PROPS, 72
 - PROP_0, 72
 - PROP_PORT_CONFIG, 72
- serial.h, 73, 75
 - gtkterm_serial_port_get_string, 74
 - gtkterm_serial_port_new, 74
 - gtkterm_serial_port_status, 75
 - GTKTERM_TYPE_SERIAL_PORT, 74
 - GtkTermSerialPort, 74
- serial_port
 - GtkTermTerminalPrivate, 23
- serial_port_fd
 - GtkTermSerialPortPrivate, 22
- set_clear_func
 - buffer.c, 33
 - buffer.h, 37
- set_display_func
 - buffer.c, 33
 - buffer.h, 37
- set_window_title
 - gtkterm.c, 48
- shortcut
 - macro_t, 25
- show_cursor
 - term_config_t, 30
- show_message
 - interface.c, 53
 - interface.h, 55
- SIGNAL_GTKTERM_CONFIG_SERIAL
 - gtkterm.h, 51
- SIGNAL_GTKTERM_CONFIG_TERMINAL
 - gtkterm.h, 51
- SIGNAL_GTKTERM_LOAD_CONFIG
 - gtkterm.h, 51
- SIGNAL_GTKTERM_PRINT_SECTION
 - gtkterm.h, 51
- SIGNAL_GTKTERM_REMOVE_SECTION
 - gtkterm.h, 51
- SIGNAL_GTKTERM_SAVE_CONFIG
 - gtkterm.h, 51
- SIGNAL_GTKTERM_TERMINAL_CHANGED
 - gtkterm.h, 51
- status_config
 - _GtkTermWindow, 18
- status_config_message
 - _GtkTermWindow, 18
- status_message
 - _GtkTermWindow, 19
- status_serial_signal
 - _GtkTermWindow, 19
- statusbox
 - _GtkTermWindow, 19
- stopbits
 - port_config_t, 27
- term_conf
 - GtkTermTerminalPrivate, 24
- term_config_t, 28
 - background_color, 29
 - block_cursor, 29
 - char_queue, 29
 - columns, 29
 - crlfauto, 29
 - delay, 29
 - echo, 29
 - font, 29
 - foreground_color, 30
 - rows, 30
 - scrollback, 30
 - show_cursor, 30
 - timestamp, 30
 - visual_bell, 30
- terminal.c, 76
 - gtkterm_terminal_new, 77
 - N_PROPS, 77
 - PROP_0, 77
 - PROP_GTKTERM_APP, 77
 - PROP_MAIN_WINDOW, 77
 - PROP_SECTION, 77
- terminal.h, 77, 79
 - gtkterm_terminal_new, 78
 - GTKTERM_TYPE_TERMINAL, 78
- terminal_window
 - _GtkTermWindow, 19
- termios_save
 - GtkTermSerialPortPrivate, 22
- Text
 - interface.h, 56
- timestamp
 - term_config_t, 30
- timestamp_on
 - buffer.c, 35
 - interface.c, 53
- TIMESTAMP_SIZE
 - buffer.c, 32

- toolmenu
 - [_GtkTermWindow](#), 19
- TRANSMIT_BUFFER
 - [defaults.h](#), 45
- unset_clear_func
 - [buffer.c](#), 33
 - [buffer.h](#), 37
- unset_display_func
 - [buffer.c](#), 34
 - [buffer.h](#), 38
- view_mode
 - [GtkTermTerminalPrivate](#), 24
- virt_col_pos
 - [buffer.c](#), 35
 - [interface.c](#), 54
- visual_bell
 - [term_config_t](#), 30
- vte_class
 - [_GtkTermTerminalClass](#), 15
- vte_object
 - [_GtkTermTerminal](#), 14
- waiting_for_char
 - [files.h](#), 47
- width
 - [_GtkTermWindow](#), 19
- write_buffer
 - [buffer.c](#), 34
 - [buffer.h](#), 38
- write_buffer_with_func
 - [buffer.c](#), 34
 - [buffer.h](#), 38
- write_func
 - [buffer.c](#), 35