

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 config	10
4.1.2.2 g_config_group	10
4.1.2.3 g_port_group	10
4.1.2.4 g_term_group	10
4.1.2.5 initial_section	10
4.1.2.6 parent_instance	11
4.2 _GtkTermConfiguration Struct Reference	11
4.2.1 Member Data Documentation	11
4.2.1.1 parent_instance	11
4.3 _GtkTermConfigurationClass Struct Reference	12
4.3.1 Member Data Documentation	12
4.3.1.1 parent_class	12
4.4 _GtkTermTerminal Struct Reference	12
4.4.1 Member Data Documentation	14
4.4.1.1 active_section	14
4.4.1.2 filename	14
4.4.1.3 port_conf	14
4.4.1.4 term_background	14
4.4.1.5 term_conf	14
4.4.1.6 term_foreground	15
4.4.1.7 view_mode	15
4.4.1.8 vte_object	15
4.5 _GtkTermTerminalClass Struct Reference	15

4.5.1 Member Data Documentation	15
4.5.1.1 <code>vte_class</code>	16
4.6 <code>_GtkTermWindow</code> Struct Reference	16
4.6.1 Detailed Description	17
4.6.2 Member Data Documentation	17
4.6.2.1 <code>fullscreen</code>	17
4.6.2.2 <code>height</code>	17
4.6.2.3 <code>infobar</code>	17
4.6.2.4 <code>maximized</code>	17
4.6.2.5 <code>menubutton</code>	18
4.6.2.6 <code>message</code>	18
4.6.2.7 <code>parent_instance</code>	18
4.6.2.8 <code>scrolled_window</code>	18
4.6.2.9 <code>status</code>	18
4.6.2.10 <code>terminal_window</code>	18
4.6.2.11 <code>toolmenu</code>	19
4.6.2.12 <code>width</code>	19
4.7 <code>display_config_t</code> Struct Reference	19
4.7.1 Member Data Documentation	20
4.7.1.1 <code>active_section</code>	20
4.7.1.2 <code>background_color</code>	20
4.7.1.3 <code>block_cursor</code>	20
4.7.1.4 <code>char_queue</code>	20
4.7.1.5 <code>columns</code>	21
4.7.1.6 <code>crlfauto</code>	21
4.7.1.7 <code>default_filename</code>	21
4.7.1.8 <code>delay</code>	21
4.7.1.9 <code>echo</code>	21
4.7.1.10 <code>font</code>	21
4.7.1.11 <code>foreground_color</code>	21
4.7.1.12 <code>rows</code>	21
4.7.1.13 <code>scrollback</code>	22
4.7.1.14 <code>show_cursor</code>	22
4.7.1.15 <code>timestamp</code>	22
4.7.1.16 <code>visual_bell</code>	22
4.8 <code>GtkTermConfigurationPrivate</code> Struct Reference	22
4.8.1 Member Data Documentation	23
4.8.1.1 <code>config_file</code>	23
4.8.1.2 <code>key_file</code>	23
4.9 <code>macro_t</code> Struct Reference	23
4.9.1 Detailed Description	24
4.9.2 Member Data Documentation	24

4.9.2.1 action	24
4.9.2.2 closure	24
4.9.2.3 shortcut	24
4.10 port_config_t Struct Reference	25
4.10.1 Member Data Documentation	25
4.10.1.1 baudrate	25
4.10.1.2 bits	26
4.10.1.3 char_queue	26
4.10.1.4 disable_port_lock	26
4.10.1.5 flow_control	26
4.10.1.6 parity	26
4.10.1.7 port	26
4.10.1.8 rs485_rts_time_after_transmit	26
4.10.1.9 rs485_rts_time_before_transmit	26
4.10.1.10 stopbits	26
5 File Documentation	27
5.1 README.md File Reference	27
5.2 README.source File Reference	27
5.3 buffer.c File Reference	27
5.3.1 Macro Definition Documentation	28
5.3.1.1 TIMESTAMP_SIZE	28
5.3.2 Function Documentation	28
5.3.2.1 clear_buffer()	28
5.3.2.2 create_buffer()	28
5.3.2.3 delete_buffer()	29
5.3.2.4 insert_timestamp()	29
5.3.2.5 put_chars()	29
5.3.2.6 set_clear_func()	29
5.3.2.7 set_display_func()	29
5.3.2.8 unset_clear_func()	29
5.3.2.9 unset_display_func()	30
5.3.2.10 write_buffer()	30
5.3.2.11 write_buffer_with_func()	30
5.3.3 Variable Documentation	31
5.3.3.1 clear_func	31
5.3.3.2 overlapped	31
5.3.3.3 timestamp_on	31
5.3.3.4 virt_col_pos	31
5.3.3.5 write_func	31
5.4 buffer.h File Reference	32
5.4.1 Macro Definition Documentation	32

5.4.1.1 BUFFER_SIZE	32
5.4.2 Function Documentation	32
5.4.2.1 clear_buffer()	33
5.4.2.2 create_buffer()	33
5.4.2.3 delete_buffer()	33
5.4.2.4 put_chars()	33
5.4.2.5 set_clear_func()	33
5.4.2.6 set_display_func()	33
5.4.2.7 unset_clear_func()	34
5.4.2.8 unset_display_func()	34
5.4.2.9 write_buffer()	34
5.4.2.10 write_buffer_with_func()	34
5.5 buffer.h	35
5.6 cmdline.c File Reference	35
5.6.1 Macro Definition Documentation	36
5.6.1.1 BUFFER_LENGTH	36
5.6.1.2 MAX_SECTION_LENGTH	36
5.6.2 Function Documentation	36
5.6.2.1 gtkterm_add_cmdline_options()	36
5.7 cmdline.h File Reference	36
5.7.1 Function Documentation	37
5.7.1.1 gtkterm_add_cmdline_options()	37
5.7.2 Variable Documentation	37
5.7.2.1 g_term_group	37
5.8 cmdline.h	37
5.9 files.c File Reference	38
5.9.1 Variable Documentation	38
5.9.1.1 default_filename	38
5.10 files.h File Reference	38
5.10.1 Function Documentation	39
5.10.1.1 add_input()	39
5.10.1.2 save_raw_file()	39
5.10.1.3 send_raw_file()	39
5.10.2 Variable Documentation	39
5.10.2.1 default_filename	39
5.10.2.2 waiting_for_char	39
5.11 files.h	40
5.12 gtkterm.c File Reference	40
5.12.1 Macro Definition Documentation	41
5.12.1.1 GTKTERM_TYPE	41
5.12.1.2 GTKTERM_WINDOW_TYPE	41
5.12.1.3 NR_OF_SIGNALS	41

5.12.2 Function Documentation	41
5.12.2.1 main()	41
5.12.2.2 set_window_title()	42
5.12.3 Variable Documentation	42
5.12.3.1 gtkterm_signals	42
5.13 gtkterm.h File Reference	42
5.13.1 Typedef Documentation	43
5.13.1.1 GtkTerm	43
5.13.1.2 GtkTermWindow	44
5.13.2 Enumeration Type Documentation	44
5.13.2.1 anonymous enum	44
5.13.3 Variable Documentation	44
5.13.3.1 gtkterm_signals	44
5.14 gtkterm.h	44
5.15 interface.c File Reference	45
5.15.1 Function Documentation	46
5.15.1.1 show_message()	46
5.15.2 Variable Documentation	46
5.15.2.1 config	46
5.15.2.2 timestamp_on	46
5.15.2.3 virt_col_pos	47
5.16 interface.h File Reference	47
5.16.1 Macro Definition Documentation	47
5.16.1.1 ASCII_VIEW	47
5.16.1.2 HEXADECIMAL_VIEW	48
5.16.1.3 MSG_ERR	48
5.16.1.4 MSG_WRN	48
5.16.2 Function Documentation	48
5.16.2.1 show_message()	48
5.16.3 Variable Documentation	48
5.16.3.1 display	48
5.16.3.2 Text	49
5.17 interface.h	49
5.18 macros.c File Reference	49
5.18.1 Enumeration Type Documentation	50
5.18.1.1 anonymous enum	50
5.18.2 Function Documentation	50
5.18.2.1 convert_macros_to_string()	50
5.18.2.2 convert_string_to_macros()	51
5.18.2.3 get_shortcuts()	51
5.18.2.4 macro_count()	51
5.18.2.5 remove_shortcuts()	51

5.18.3 Variable Documentation	52
5.18.3.1 macros	52
5.18.3.2 nr_of_macros	52
5.19 macros.h File Reference	52
5.19.1 Function Documentation	53
5.19.1.1 add_shortcuts()	53
5.19.1.2 convert_macros_to_string()	53
5.19.1.3 convert_string_to_macros()	53
5.19.1.4 get_shortcuts()	54
5.19.1.5 macro_count()	54
5.19.1.6 remove_shortcuts()	54
5.19.2 Variable Documentation	54
5.19.2.1 macros	54
5.20 macros.h	55
5.21 resource_file.c File Reference	55
5.21.1 Macro Definition Documentation	56
5.21.1.1 BUFFER_LENGTH	56
5.21.1.2 CONFIGURATION_FILENAME	57
5.21.2 Function Documentation	57
5.21.2.1 gtkterm_configuration_default_configuration()	57
5.21.2.2 gtkterm_configuration_validate()	57
5.21.2.3 on_set_config_options()	58
5.21.3 Variable Documentation	59
5.21.3.1 GtkTermConfigurationItems	59
5.22 resource_file.h File Reference	59
5.22.1 Macro Definition Documentation	60
5.22.1.1 CONF_ITEM_LENGTH	60
5.22.1.2 DEFAULT_SECTION	61
5.22.1.3 GTKTERM_TYPE_CONFIGURATION	61
5.22.2 Typedef Documentation	61
5.22.2.1 GtkTermConfiguration	61
5.22.3 Enumeration Type Documentation	61
5.22.3.1 anonymous enum	61
5.22.4 Function Documentation	62
5.22.4.1 gtkterm_configuration_new()	62
5.22.4.2 on_set_config_options()	62
5.22.5 Variable Documentation	63
5.22.5.1 GtkTermConfigurationItems	63
5.23 resource_file.h	63
5.24 serial.c File Reference	64
5.24.1 Function Documentation	65
5.24.1.1 get_port_string()	65

5.24.2 Variable Documentation	65
5.24.2.1 port_conf	65
5.24.2.2 serial_port_fd	66
5.24.2.3 termios_save	66
5.25 serial.h File Reference	66
5.25.1 Macro Definition Documentation	67
5.25.1.1 DEFAULT_BAUDRATE	67
5.25.1.2 DEFAULT_BITS	67
5.25.1.3 DEFAULT_FLOW	67
5.25.1.4 DEFAULT_PARITY	67
5.25.1.5 DEFAULT_PORT	67
5.25.1.6 DEFAULT_STOPBITS	68
5.25.1.7 LINE_FEED	68
5.25.1.8 POLL_DELAY	68
5.25.1.9 RECEIVE_BUFFER	68
5.25.1.10 TRANSMIT_BUFFER	68
5.25.2 Function Documentation	68
5.25.2.1 get_port_string()	68
5.25.3 Variable Documentation	69
5.25.3.1 port_conf	69
5.25.3.2 serial_port_fd	69
5.26 serial.h	69
5.27 term_config.c File Reference	70
5.27.1 Variable Documentation	70
5.27.1.1 term_conf	70
5.28 term_config.h File Reference	70
5.28.1 Macro Definition Documentation	71
5.28.1.1 DEFAULT_CHAR	71
5.28.1.2 DEFAULT_DELAY	71
5.28.1.3 DEFAULT_DELAY_RS485	71
5.28.1.4 DEFAULT_ECHO	71
5.28.1.5 DEFAULT_FONT	72
5.28.1.6 DEFAULT_SCROLLBACK	72
5.28.1.7 DEFAULT_VISUAL_BELL	72
5.28.2 Variable Documentation	72
5.28.2.1 term_conf	72
5.29 term_config.h	72
5.30 terminal.c File Reference	73
5.31 terminal.h File Reference	73
5.31.1 Macro Definition Documentation	74
5.31.1.1 GTKTERM_TERMINAL_TYPE	74
5.32 terminal.h	75

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
_GtkTermTerminal	12
_GtkTermTerminalClass	15
_GtkTermWindow	
The main GtkTermWindow class	16
display_config_t	19
GtkTermConfigurationPrivate	22
macro_t	
TODO: Migrate to GObject	23
port_config_t	25

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

README.source	27
buffer.c	27
buffer.h	32
cmdline.c	35
cmdline.h	36
files.c	38
files.h	38
gtkterm.c	40
gtkterm.h	42
interface.c	45
interface.h	47
macros.c	49
macros.h	52
resource_file.c	55
resource_file.h	59
serial.c	64
serial.h	66
term_config.c	70
term_config.h	70
terminal.c	73
terminal.h	73

Chapter 4

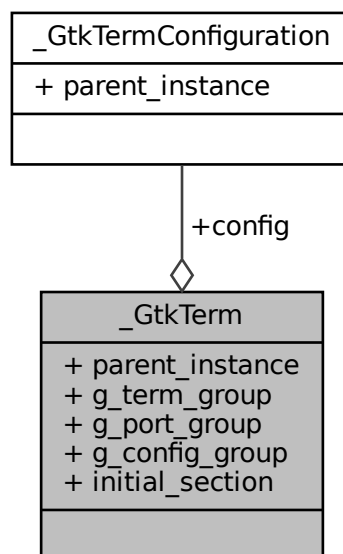
Class Documentation

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](#)
- [GtkTermConfiguration *](#) `config`
- `char *` [initial_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 config

`GtkTermConfiguration* _GtkTerm::config`

4.1.2.2 g_config_group

`GOptionGroup* _GtkTerm::g_config_group`

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

4.1.2.3 g_port_group

`GOptionGroup* _GtkTerm::g_port_group`

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

4.1.2.4 g_term_group

`GOptionGroup* _GtkTerm::g_term_group`

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

4.1.2.5 initial_section

`char* _GtkTerm::initial_section`

The Key file with the configurations.

4.1.2.6 parent_instance

GtkApplication _GtkTerm::parent_instance

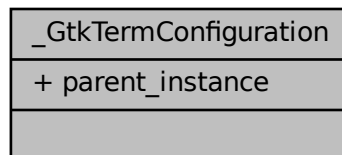
The documentation for this struct was generated from the following file:

- [gtkterm.h](#)

4.2 _GtkTermConfiguration Struct Reference

```
#include <resource_file.h>
```

Collaboration diagram for _GtkTermConfiguration:



Public Attributes

- GObject [parent_instance](#)

4.2.1 Member Data Documentation

4.2.1.1 parent_instance

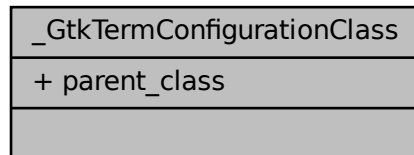
GObject _GtkTermConfiguration::parent_instance

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

- [resource_file.h](#)

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
```

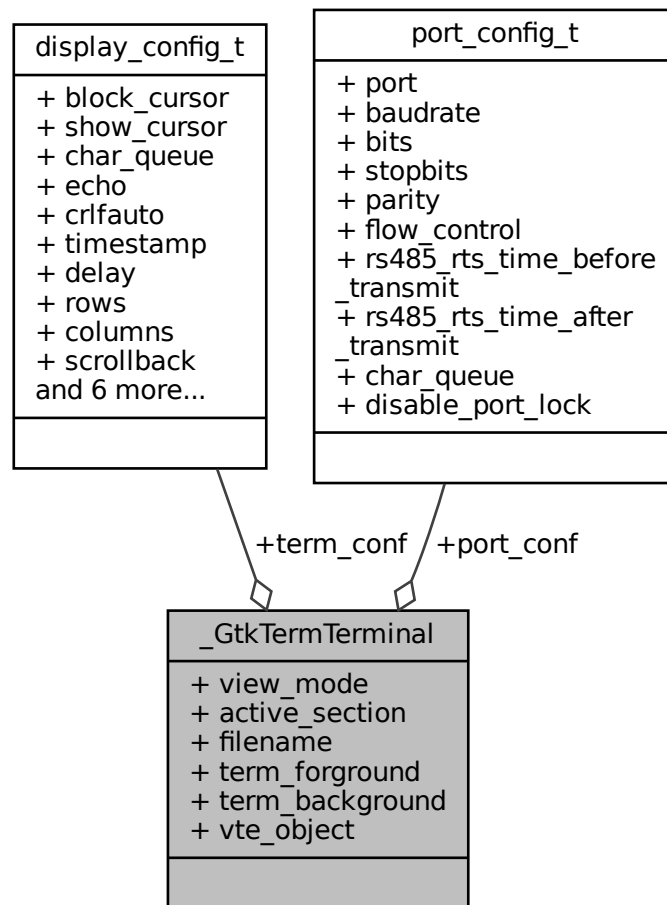
The documentation for this struct was generated from the following file:

- [resource_file.c](#)

4.4 `_GtkTermTerminal` Struct Reference

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

Collaboration diagram for _GtkTermTerminal:



Public Attributes

- `uint8_t view_mode`
- `display_config_t * term_conf`
ASCII or HEX view mode.
- `port_config_t * port_conf`
The configuration loaded from the keyfile.
- `char * active_section`
Port configuration used in this terminal.
- `char * filename`
Active section in this window from config file.
- `GdkRGBA * term_foreground`
File to send.
- `GdkRGBA * term_background`
Foreground (text) color of this terminal.
- `VteTerminal vte_object`
Background color.

4.4.1 Member Data Documentation

4.4.1.1 active_section

`char* _GtkTermTerminal::active_section`

Port configuration used in this terminal.

4.4.1.2 filename

`char* _GtkTermTerminal::filename`

Active section in this window from config file.

4.4.1.3 port_conf

`port_config_t* _GtkTermTerminal::port_conf`

The configuration loaded from the keyfile.

4.4.1.4 term_background

`GdkRGBA* _GtkTermTerminal::term_background`

Foreground (text) color of this terminal.

4.4.1.5 term_conf

`display_config_t* _GtkTermTerminal::term_conf`

ASCII or HEX view mode.

4.4.1.6 term_forground

```
GdkRGBA* _GtkTermTerminal::term_forground
```

File to send.

4.4.1.7 view_mode

```
uint8_t _GtkTermTerminal::view_mode
```

4.4.1.8 vte_object

```
VteTerminal _GtkTermTerminal::vte_object
```

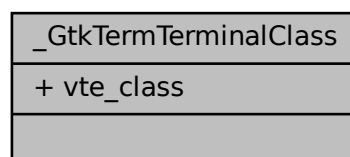
Background color.

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

- [terminal.h](#)

4.5 _GtkTermTerminalClass Struct Reference

Collaboration diagram for _GtkTermTerminalClass:



Public Attributes

- VteTerminalClass [vte_class](#)

4.5.1 Member Data Documentation

4.5.1.1 `vte_class`

`VteTerminalClass _GtkTermTerminalClass::vte_class`

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

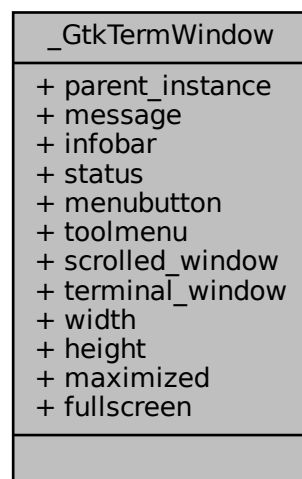
- [terminal.c](#)

4.6 `_GtkTermWindow` Struct Reference

The main `GtkTermWindow` class.

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

Collaboration diagram for `_GtkTermWindow`:



Public Attributes

- `GtkApplicationWindow` [parent_instance](#)
- `GtkWidget *` [message](#)
- `GtkWidget *` [infobar](#)
Message for the infobar.
- `GtkWidget *` [status](#)
Infobar.
- `GtkWidget *` [menubutton](#)
Statusbar.
- `GMenuModel *` [toolmenu](#)
Toolbar.

- GtkScrolledWindow * [scrolled_window](#)
Menu.
- GtkTermTerminal * [terminal_window](#)
Make the terminal window scrolled.
- int [width](#)
The terminal window.
- int [height](#)
- bool [maximized](#)
- bool [fullscreen](#)

4.6.1 Detailed Description

The main GtkTermWindow class.

MainWindow specific variables here.

4.6.2 Member Data Documentation

4.6.2.1 fullscreen

```
bool _GtkTermWindow::fullscreen
```

4.6.2.2 height

```
int _GtkTermWindow::height
```

4.6.2.3 infobar

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

Message for the infobar.

4.6.2.4 maximized

```
bool _GtkTermWindow::maximized
```

4.6.2.5 menubutton

`GtkWidget* _GtkTermWindow::menubutton`

Statusbar.

4.6.2.6 message

`GtkWidget* _GtkTermWindow::message`

4.6.2.7 parent_instance

`GtkApplicationWindow _GtkTermWindow::parent_instance`

4.6.2.8 scrolled_window

`GtkScrolledWindow* _GtkTermWindow::scrolled_window`

Menu.

4.6.2.9 status

`GtkWidget* _GtkTermWindow::status`

Infobar.

4.6.2.10 terminal_window

`GtkTermTerminal* _GtkTermWindow::terminal_window`

Make the terminal window scrolled.

4.6.2.11 toolmenu

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

Toolbar.

4.6.2.12 width

```
int _GtkTermWindow::width
```

The terminal window.

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

- [gtkterm.h](#)

4.7 display_config_t Struct Reference

```
#include <term_config.h>
```

Collaboration diagram for display_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](#)
- char * [active_section](#)
- char * [default_filename](#)

4.7.1 Member Data Documentation

4.7.1.1 active_section

`char* display_config_t::active_section`

4.7.1.2 background_color

`GdkRGBA display_config_t::background_color`

4.7.1.3 block_cursor

`bool display_config_t::block_cursor`

4.7.1.4 char_queue

`char display_config_t::char_queue`

4.7.1.5 columns

```
int display_config_t::columns
```

4.7.1.6 crlfauto

```
bool display_config_t::crlfauto
```

4.7.1.7 default_filename

```
char* display_config_t::default_filename
```

4.7.1.8 delay

```
int display_config_t::delay
```

4.7.1.9 echo

```
bool display_config_t::echo
```

4.7.1.10 font

```
PangoFontDescription* display_config_t::font
```

4.7.1.11 foreground_color

```
GdkRGBA display_config_t::foreground_color
```

4.7.1.12 rows

```
int display_config_t::rows
```

4.7.1.13 scrollbar

```
int display_config_t::scrollback
```

4.7.1.14 show_cursor

```
bool display_config_t::show_cursor
```

4.7.1.15 timestamp

```
bool display_config_t::timestamp
```

4.7.1.16 visual_bell

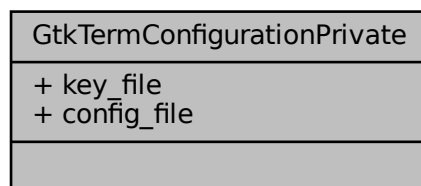
```
bool display_config_t::visual_bell
```

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

- [term_config.h](#)

4.8 GtkTermConfigurationPrivate Struct Reference

Collaboration diagram for GtkTermConfigurationPrivate:



Public Attributes

- GKeyFile * [key_file](#)
- GFile * [config_file](#)

The memory loaded keyfile.

4.8.1 Member Data Documentation

4.8.1.1 config_file

GFile* GtkTermConfigurationPrivate::config_file

The memory loaded keyfile.

4.8.1.2 key_file

GKeyFile* GtkTermConfigurationPrivate::key_file

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

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

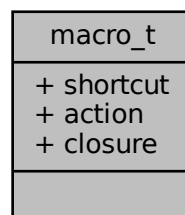
- [resource_file.c](#)

4.9 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.

4.9.1 Detailed Description

TODO: Migrate to GObject.

Define macro structure type

4.9.2 Member Data Documentation

4.9.2.1 action

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

Shortcut of the macro.

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

4.9.2.2 closure

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

Command to perform.

4.9.2.3 shortcut

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

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

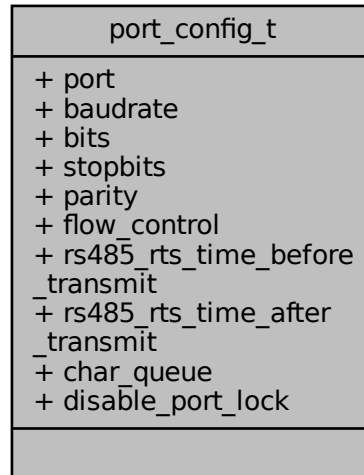
The documentation for this struct was generated from the following file:

- [macros.h](#)

4.10 port_config_t Struct Reference

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

Collaboration diagram for port_config_t:



Public Attributes

- char [port](#) [256]
- long int [baudrate](#)
- int [bits](#)
- int [stopbits](#)
- int [parity](#)
- int [flow_control](#)
- int [rs485_rts_time_before_transmit](#)
- int [rs485_rts_time_after_transmit](#)
- char [char_queue](#)
- bool [disable_port_lock](#)

4.10.1 Member Data Documentation

4.10.1.1 baudrate

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

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

4.10.1.2 bits

```
int port_config_t::bits
```

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

4.10.1.3 char_queue

```
char port_config_t::char_queue
```

4.10.1.4 disable_port_lock

```
bool port_config_t::disable_port_lock
```

4.10.1.5 flow_control

```
int port_config_t::flow_control
```

4.10.1.6 parity

```
int port_config_t::parity
```

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

4.10.1.7 port

```
char port_config_t::port[256]
```

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

4.10.1.8 rs485_rts_time_after_transmit

```
int port_config_t::rs485_rts_time_after_transmit
```

4.10.1.9 rs485_rts_time_before_transmit

```
int port_config_t::rs485_rts_time_before_transmit
```

4.10.1.10 stopbits

```
int port_config_t::stopbits
```

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

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

- [serial.h](#)

Chapter 5

File Documentation

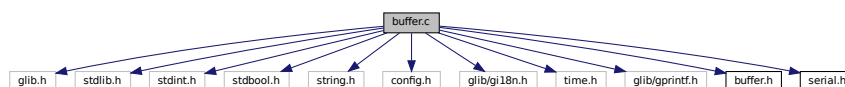
5.1 README.md File Reference

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 "serial.h"
```

Include dependency graph for buffer.c:



Macros

- `#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 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](#).

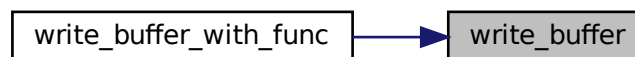
5.3.2.10 write_buffer()

```
void write_buffer (
    void )
```

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

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

Here is the caller graph for this function:

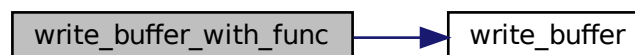


5.3.2.11 write_buffer_with_func()

```
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]
```

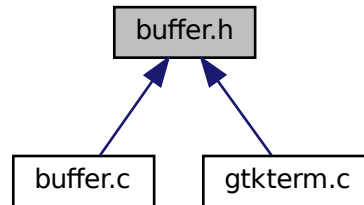
5.3.3.5 write_func

```
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 )
```

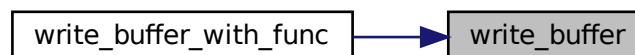
5.4.2.9 write_buffer()

```
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

```

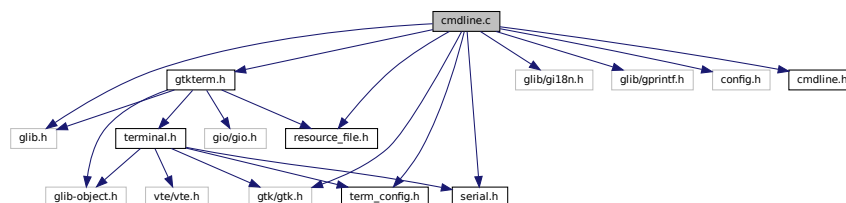
5.6 cmdline.c File Reference

```

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

```

Include dependency graph for cmdline.c:



Macros

- `#define BUFFER_LENGTH 256`
- `#define MAX_SECTION_LENGTH 32`

Functions

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

5.6.1 Macro Definition Documentation

5.6.1.1 BUFFER_LENGTH

```
#define BUFFER_LENGTH 256
```

5.6.1.2 MAX_SECTION_LENGTH

```
#define MAX_SECTION_LENGTH 32
```

5.6.2 Function Documentation

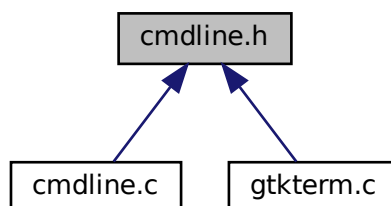
5.6.2.1 gtkterm_add_cmdline_options()

```
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

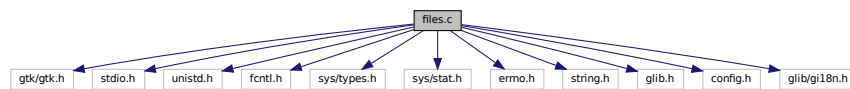
[Go to the documentation of this file.](#)

```
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
    *g_term_group;
18
19 #ifndef CMDLINE_H
20 #define CMDLINE_H
21
22 void gtkterm\_add\_cmdline\_options (GtkTerm *app);
23
24 #endif // CMDLINE_H
```

5.9 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/gi18n.h>
```

Include dependency graph for files.c:



Variables

- char * [default_filename](#) = NULL

5.9.1 Variable Documentation

5.9.1.1 default_filename

```
char* default_filename = NULL
```

5.10 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](#)

5.10.1 Function Documentation

5.10.1.1 add_input()

```
void add_input (
    void )
```

5.10.1.2 save_raw_file()

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

5.10.1.3 send_raw_file()

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

5.10.2 Variable Documentation

5.10.2.1 default_filename

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

5.10.2.2 waiting_for_char

```
gboolean waiting_for_char [extern]
```

5.11 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

```

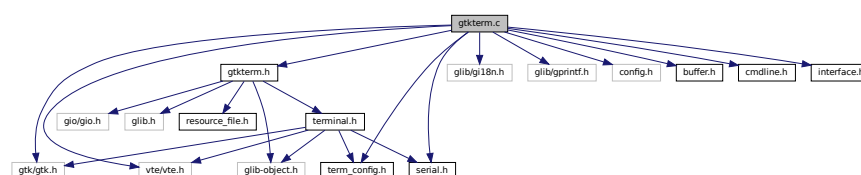
5.12 gtkterm.c File Reference

```

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

```

Include dependency graph for gtkterm.c:



Macros

- #define `NR_OF_SIGNALS` 4
- #define `GTKTERM_TYPE` `gtkterm_get_type()`
- #define `GTKTERM_WINDOW_TYPE` `gtkterm_window_get_type()`

Functions

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

Variables

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

5.12.1 Macro Definition Documentation

5.12.1.1 GTKTERM_TYPE

```
#define GTKTERM_TYPE gtkterm_get_type ()
```

5.12.1.2 GTKTERM_WINDOW_TYPE

```
#define GTKTERM_WINDOW_TYPE gtkterm_window_get_type ()
```

5.12.1.3 NR_OF_SIGNALS

```
#define NR_OF_SIGNALS 4
```

5.12.2 Function Documentation

5.12.2.1 main()

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

References [GTKTERM_TYPE](#).

5.12.2.2 set_window_title()

```
void set_window_title (
    GtkTermWindow * window )
```

References [get_port_string\(\)](#).

Here is the call graph for this function:



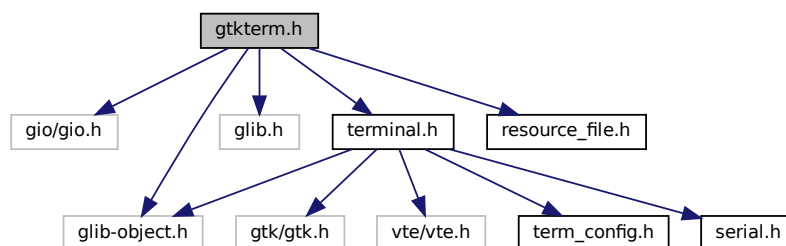
5.12.3 Variable Documentation

5.12.3.1 gtkterm_signals

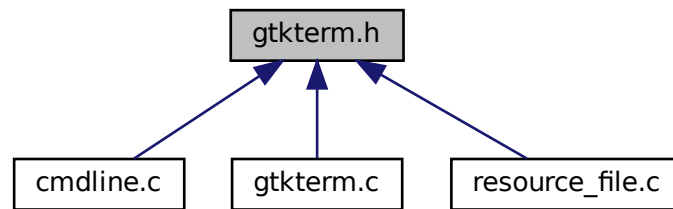
```
unsigned int gtkterm_signals[NR_OF_SIGNALS]
```

5.13 gtkterm.h File Reference

```
#include <gio/gio.h>
#include <glib-object.h>
#include <glib.h>
#include "terminal.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.
- struct [_GtkTermWindow](#)
The main GtkTermWindow class.

Typedefs

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

Enumerations

- enum { [SIGNAL_LOAD_CONFIG](#) , [SIGNAL_SAVE_CONFIG](#) , [SIGNAL_REMOVE_SECTION](#) , [SIGNAL_PRINT_SECTION](#) }

Variables

- unsigned int [gtkterm_signals](#) []

5.13.1 Typedef Documentation

5.13.1.1 GtkTerm

```
typedef struct \_GtkTerm GtkTerm
```

5.13.1.2 GtkTermWindow

```
typedef struct _GtkTermWindow GtkTermWindow
```

5.13.2 Enumeration Type Documentation

5.13.2.1 anonymous enum

anonymous enum

Enumerator

SIGNAL_LOAD_CONFIG	
SIGNAL_SAVE_CONFIG	
SIGNAL_REMOVE_SECTION	
SIGNAL_PRINT_SECTION	

5.13.3 Variable Documentation

5.13.3.1 gtkterm_signals

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

5.14 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
9 #include "terminal.h"
10 #include "resource_file.h"
11
12 enum {
13     SIGNAL_LOAD_CONFIG,
14     SIGNAL_SAVE_CONFIG,
15     SIGNAL_REMOVE_SECTION,
16     SIGNAL_PRINT_SECTION
17 };
18
19 extern unsigned int gtkterm_signals[];
20
21 G_BEGIN_DECLS
22
23 /*! @brief The main GtkTerm application class.
```

```

24 //! All application specific variables are defined here.
25 struct _GtkTerm {
26
27     GtkApplication parent_instance;
28
29     GOptionGroup *g_term_group;
30     GOptionGroup *g_port_group;
31     GOptionGroup *g_config_group;
32
33     GtkTermConfiguration *config; //! The Key file with the configurations
34     char *initial_section; //! The initial section provided from the cli.
35     //! Terminals have their own section pointer
36
37 };
38
39 typedef struct _GtkTerm GtkTerm;
40 G_DECLARE_FINAL_TYPE (GtkTerm, gtkterm, GTKTERM, APP, GtkApplication)
41
42 //! @brief The main GtkTermWindow class.
43 //! MainWindow specific variables here.
44 struct _GtkTermWindow {
45     GtkApplicationWindow parent_instance;
46
47     GtkWidget *message; //! Message for the infobar
48     GtkWidget *infobar; //! Infobar
49     GtkWidget *status; //! Statusbar
50     GtkWidget *menubutton; //! Toolbar
51     GMenuModel *toolmenu; //! Menu
52     GtkScrolledWindow *scrolled_window; //! Make the terminal window scrolled
53     GtkTerminal *terminal_window; //! The terminal window
54
55     int width;
56     int height;
57     bool maximized;
58     bool fullscreen;
59
60 };
61
62 typedef struct _GtkTermWindow GtkTermWindow;
63 G_DECLARE_FINAL_TYPE (GtkTermWindow, gtkterm_window, GTKTERM, WINDOW, GtkApplicationWindow)
64
65 G_END_DECLS
66
67 #endif // GTKTERM_H

```

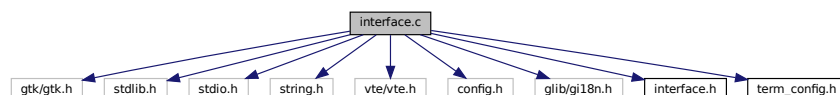
5.15 interface.c File Reference

```

#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 "term_config.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

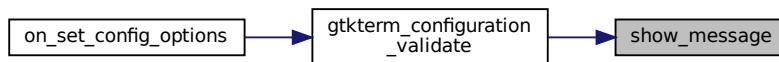
5.15.1 Function Documentation

5.15.1.1 `show_message()`

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

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

Here is the caller graph for this function:



5.15.2 Variable Documentation

5.15.2.1 `config`

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

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

5.15.2.2 `timestamp_on`

```
bool timestamp_on = 0
```

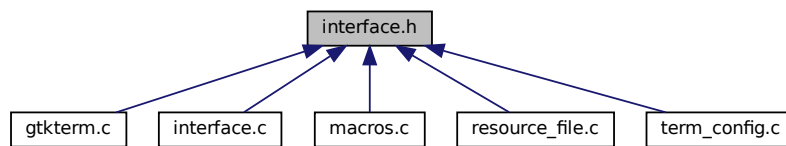
Referenced by [put_chars\(\)](#).

5.15.2.3 virt_col_pos

```
int virt_col_pos = 0
```

5.16 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](#)

5.16.1 Macro Definition Documentation

5.16.1.1 ASCII_VIEW

```
#define ASCII_VIEW 0
```

5.16.1.2 HEXADECIMAL_VIEW

```
#define HEXADECIMAL_VIEW 1
```

5.16.1.3 MSG_ERR

```
#define MSG_ERR 1
```

5.16.1.4 MSG_WRN

```
#define MSG_WRN 0
```

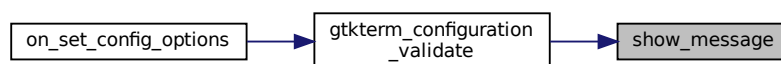
5.16.2 Function Documentation

5.16.2.1 show_message()

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

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

Here is the caller graph for this function:



5.16.3 Variable Documentation

5.16.3.1 display

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

5.16.3.2 Text

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

5.17 interface.h

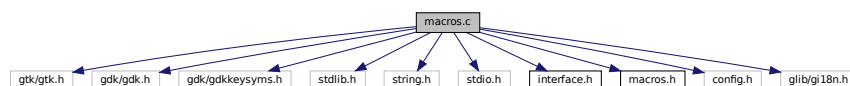
[Go to the documentation of this file.](#)

```
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
```

5.18 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/gi18n.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

5.18.1 Enumeration Type Documentation

5.18.1.1 anonymous enum

anonymous enum

TODO: Migrate to GObject.

Enumerator

COLUMN_SHORTCUT	
COLUMN_ACTION	
NUM_COLUMNS	

5.18.2 Function Documentation

5.18.2.1 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](#).

5.18.2.2 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:



5.18.2.3 get_shortcuts()

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

References [macros](#).

5.18.2.4 macro_count()

```
int macro_count ( )
```

References [nr_of_macros](#).

5.18.2.5 remove_shortcuts()

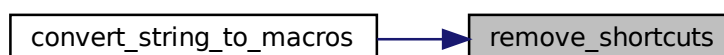
```
void remove_shortcuts (
    void )
```

Clean up all macros

References [macros](#).

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

Here is the caller graph for this function:



5.18.3 Variable Documentation

5.18.3.1 macros

```
macro_t* macros = NULL
```

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

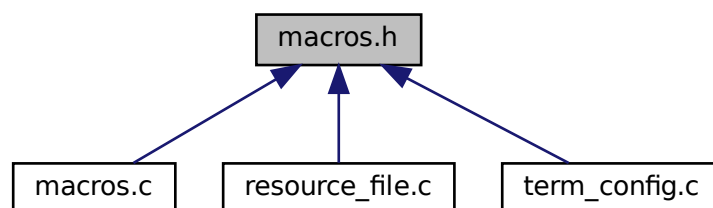
5.18.3.2 nr_of_macros

```
int nr_of_macros = 0
```

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

5.19 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](#)

5.19.1 Function Documentation

5.19.1.1 `add_shortcuts()`

```
void add_shortcuts (
    void )
```

Remove shortcuts from `accel_group` and free memory.

5.19.1.2 `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](#).

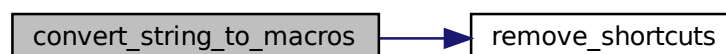
5.19.1.3 `convert_string_to_macros()`

```
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:



5.19.1.4 `get_shortcuts()`

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

5.19.1.5 `macro_count()`

```
int macro_count ( )
```

References [nr_of_macros](#).

5.19.1.6 `remove_shortcuts()`

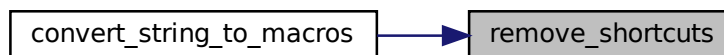
```
void remove_shortcuts (
    void )
```

Clean up all macros

References [macros](#).

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

Here is the caller graph for this function:



5.19.2 Variable Documentation

5.19.2.1 `macros`

```
macro_t* macros [extern]
```

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

5.20 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

```

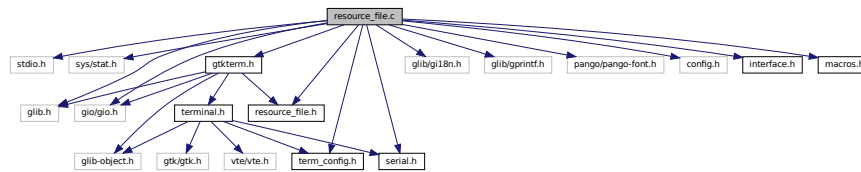
5.21 resource_file.c File Reference

```

#include <stdio.h>
#include <sys/stat.h>
#include <glib.h>
#include <glib/glib.h>
#include <glib/gprintf.h>
#include <gio/gio.h>
#include <pango/pango-font.h>
#include <config.h>
#include "gtkterm.h"
#include "serial.h"
#include "term_config.h"
#include "resource_file.h"
#include "interface.h"
#include "macros.h"

```

Include dependency graph for resource_file.c:



Classes

- struct [GtkTermConfigurationPrivate](#)
- 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
- 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.

5.21.1 Macro Definition Documentation

5.21.1.1 BUFFER_LENGTH

```
#define BUFFER_LENGTH 256
```

5.21.1.2 CONFIGURATION_FILENAME

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

Default configuration filename.

5.21.2 Function Documentation

5.21.2.1 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](#).

5.21.2.2 gtkterm_configuration_validate()

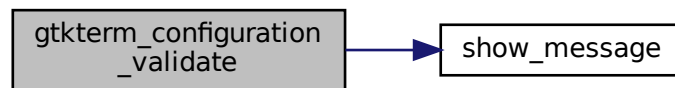
```
void gtkterm_configuration_validate (
    GtkTermConfigurationPrivate * priv,
    char * section )
```

validate the configuration, given by the section

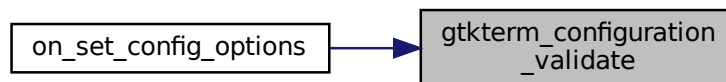
References [CONF_ITEM_SERIAL_BAUDRATE](#), [CONF_ITEM_SERIAL_BITS](#), [CONF_ITEM_SERIAL_STOPBITS](#), [CONF_ITEM_TERM_FONT](#), [CONF_ITEM_TERM_WAIT_DELAY](#), [DEFAULT_BITS](#), [DEFAULT_DELAY](#), [DEFAULT_FONT](#), [DEFAULT_STOPBITS](#), [GtkTermConfigurationItems](#), [GtkTermConfigurationPrivate::key_file](#),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



5.21.2.3 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)

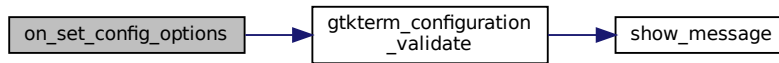
Search index for the option we want to set

Check for max path length. Exit if it is too 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:



5.21.3 Variable Documentation

5.21.3.1 GtkTermConfigurationItems

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

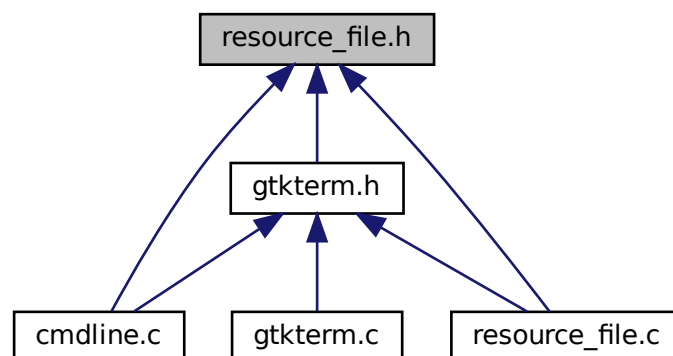
Used configuration options to hold consistency between load/save functions.

Configuration item names.

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

5.22 resource_file.h File Reference

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



Classes

- [struct _GtkTermConfiguration](#)

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.

5.22.1 Macro Definition Documentation

5.22.1.1 CONF_ITEM_LENGTH

```
#define CONF_ITEM_LENGTH 32
```

5.22.1.2 DEFAULT_SECTION

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

5.22.1.3 GTKTERM_TYPE_CONFIGURATION

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

5.22.2 Typedef Documentation

5.22.2.1 GtkTermConfiguration

```
typedef struct _GtkTermConfiguration GtkTermConfiguration
```

5.22.3 Enumeration Type Documentation

5.22.3.1 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	

Enumerator

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.

5.22.4 Function Documentation

5.22.4.1 gtkterm_configuration_new()

```
GtkTermConfiguration * gtkterm_configuration_new (
    void )
```

5.22.4.2 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 charater ('-' in front of the cli option)

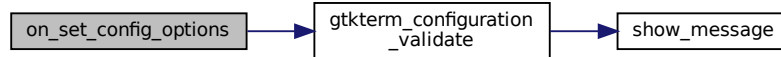
Search index for the option we want to set

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:



5.22.5 Variable Documentation

5.22.5.1 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\(\)](#).

5.23 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 //!

```

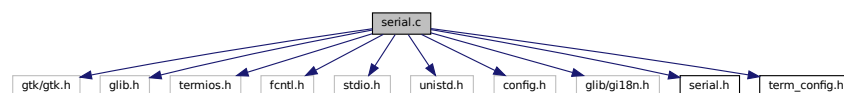
5.24 serial.c File Reference

```

#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 "term_config.h"

```

Include dependency graph for serial.c:



Functions

- char * [get_port_string](#) (void)

Variables

- [port_config_t](#) [port_conf](#)
- struct termios [termios_save](#)
- int [serial_port_fd](#) = -1

5.24.1 Function Documentation

5.24.1.1 [get_port_string\(\)](#)

```
char * get_port_string (  
    void )
```

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

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

Here is the caller graph for this function:



5.24.2 Variable Documentation

5.24.2.1 [port_conf](#)

```
port\_config\_t port\_conf
```

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

5.24.2.2 serial_port_fd

```
int serial_port_fd = -1
```

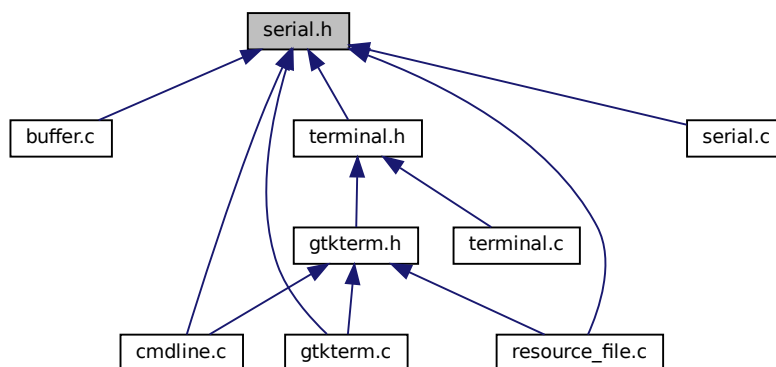
Referenced by [get_port_string\(\)](#).

5.24.2.3 termios_save

```
struct termios termios_save
```

5.25 serial.h File Reference

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



Classes

- struct [port_config_t](#)

Macros

- #define [DEFAULT_PORT](#) `"/dev/ttyS0"`
- #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 /* in ms (for control signals) */`

Functions

- char * [get_port_string](#) (void)

Variables

- int [serial_port_fd](#)
- [port_config_t](#) [port_conf](#)

5.25.1 Macro Definition Documentation

5.25.1.1 DEFAULT_BAUDRATE

```
#define DEFAULT_BAUDRATE 115200
```

5.25.1.2 DEFAULT_BITS

```
#define DEFAULT_BITS 8
```

5.25.1.3 DEFAULT_FLOW

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

5.25.1.4 DEFAULT_PARITY

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

5.25.1.5 DEFAULT_PORT

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

5.25.1.6 DEFAULT_STOPBITS

```
#define DEFAULT_STOPBITS 1
```

5.25.1.7 LINE_FEED

```
#define LINE_FEED 0x0A
```

5.25.1.8 POLL_DELAY

```
#define POLL_DELAY 100 /* in ms (for control signals) */
```

5.25.1.9 RECEIVE_BUFFER

```
#define RECEIVE_BUFFER 8192
```

5.25.1.10 TRANSMIT_BUFFER

```
#define TRANSMIT_BUFFER 4096
```

5.25.2 Function Documentation

5.25.2.1 get_port_string()

```
char * get_port_string (  
    void )
```

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

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

Here is the caller graph for this function:



5.25.3 Variable Documentation

5.25.3.1 port_conf

`port_config_t` `port_conf` [extern]

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

5.25.3.2 serial_port_fd

`int` `serial_port_fd` [extern]

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

5.26 serial.h

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

```

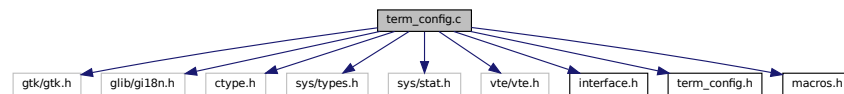
1  /*****
2  /* serial.h
3  /* -----
4  /*      GTKTerm Software
5  /*      (c) Julien Schmitt
6  /*
7  /* -----
8  /*
9  /*      Purpose
10 /*      Serial port access functions
11 /*      - Header file -
12 /*
13 /*****/
14
15 #ifndef SERIAL_H_
16 #define SERIAL_H_
17
18 #define DEFAULT_PORT      "/dev/ttyS0"
19 #define DEFAULT_BAUDRATE  115200
20 #define DEFAULT_PARITY    "none"
21 #define DEFAULT_BITS      8
22 #define DEFAULT_STOPBITS  1
23 #define DEFAULT_FLOW      "none"
24
25 typedef struct
26 {
27     char port[256];
28     long int baudrate;           // 300 - 600 - 1200 - ... - 2000000
29     int bits;                   // 5 - 6 - 7 - 8
30     int stopbits;               // 1 - 2
31     int parity;                 // 0 : None, 1 : Odd, 2 : Even
32     int flow_control;           // 0 : None, 1 : Xon/Xoff, 2 : RTS/CTS, 3 : RS485halfduplex
33     int rs485_rts_time_before_transmit;
34     int rs485_rts_time_after_transmit;
35     char char_queue;            // character in queue
36     bool disable_port_lock;
37 } port_config_t;
38
39 extern int serial_port_fd;
40 extern port_config_t port_conf;
41
42 char* get_port_string (void);
43
44 #define RECEIVE_BUFFER 8192
45 #define TRANSMIT_BUFFER 4096
46 #define LINE_FEED 0x0A
47 #define POLL_DELAY 100          /* in ms (for control signals) */
48
49 #endif

```

5.27 term_config.c File Reference

```
#include <gtk/gtk.h>
#include <glib/gi18n.h>
#include <ctype.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <vte/vte.h>
#include "interface.h"
#include "term_config.h"
#include "macros.h"
```

Include dependency graph for term_config.c:



Variables

- [display_config_t term_conf](#)

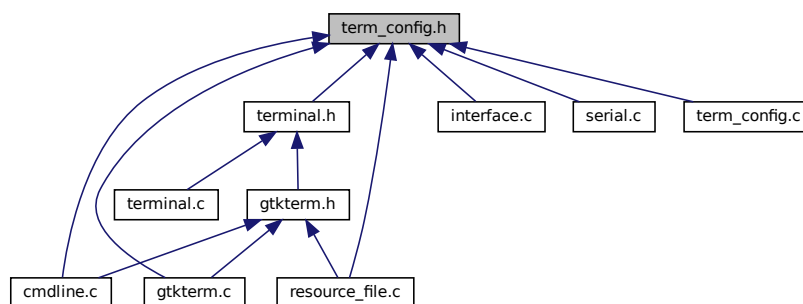
5.27.1 Variable Documentation

5.27.1.1 term_conf

```
display_config_t term_conf
```

5.28 term_config.h File Reference

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



Classes

- struct [display_config_t](#)

Macros

- #define [DEFAULT_FONT](#) "Monospace 12"
- #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"

Variables

- [display_config_t](#) term_conf

5.28.1 Macro Definition Documentation

5.28.1.1 DEFAULT_CHAR

```
#define DEFAULT_CHAR -1
```

5.28.1.2 DEFAULT_DELAY

```
#define DEFAULT_DELAY 0
```

5.28.1.3 DEFAULT_DELAY_RS485

```
#define DEFAULT_DELAY_RS485 30
```

5.28.1.4 DEFAULT_ECHO

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

5.28.1.5 DEFAULT_FONT

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

5.28.1.6 DEFAULT_SCROLLBACK

```
#define DEFAULT_SCROLLBACK 10000
```

5.28.1.7 DEFAULT_VISUAL_BELL

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

5.28.2 Variable Documentation

5.28.2.1 term_conf

```
display_config_t term_conf [extern]
```

5.29 term_config.h

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

```
1  /*****
2  /* term_config.h
3  /* -----
4  /*          GTKTerm Software
5  /*          (c) Julien Schmitt
6  /*
7  /* -----
8  /*
9  /* Purpose
10 /* Configuration of the serial port
11 /* - Header file -
12 /*
13 /*****/
14
15 #ifndef TERM_CONFIG_H_
16 #define TERM_CONFIG_H_
17
18 #define DEFAULT_FONT "Monospace 12"
19 #define DEFAULT_SCROLLBACK 10000
20
21 #define DEFAULT_DELAY 0
22 #define DEFAULT_CHAR -1
23 #define DEFAULT_DELAY_RS485 30
24 #define DEFAULT_ECHO "false"
25 #define DEFAULT_VISUAL_BELL "false"
26
27 typedef struct
28 {
29     bool block_cursor;
30     bool show_cursor;
31     char char_queue; // character in queue
32     bool echo; // echo local
```

```

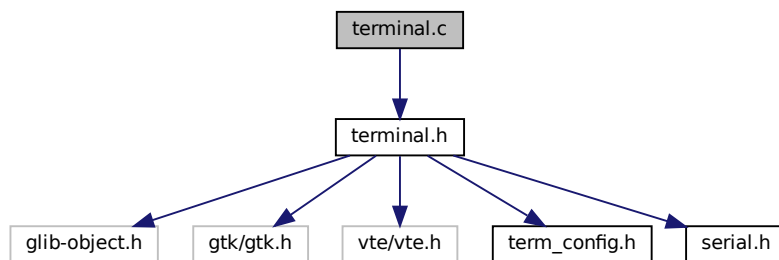
33     bool crlfauto;           // line feed auto
34     bool timestamp;
35     int delay;               // end of char delay: in ms
36     int rows;
37     int columns;
38     int scrollback;
39     bool visual_bell;
40     GdkRGBA foreground_color;
41     GdkRGBA background_color;
42     PangoFontDescription *font;
43     char *active_section;
44
45     char *default_filename;
46
47 } display_config_t;
48
49 // configuration for the terminal window
50 extern display_config_t term_conf;
51
52 #endif

```

5.30 terminal.c File Reference

```
#include "terminal.h"
```

Include dependency graph for terminal.c:



Classes

- [struct _GtkTermTerminalClass](#)

5.31 terminal.h File Reference

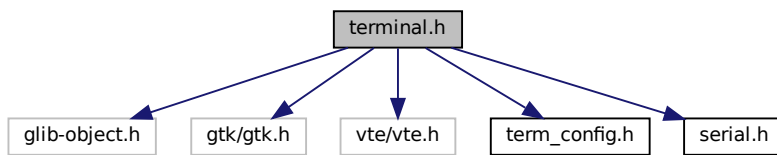
```

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

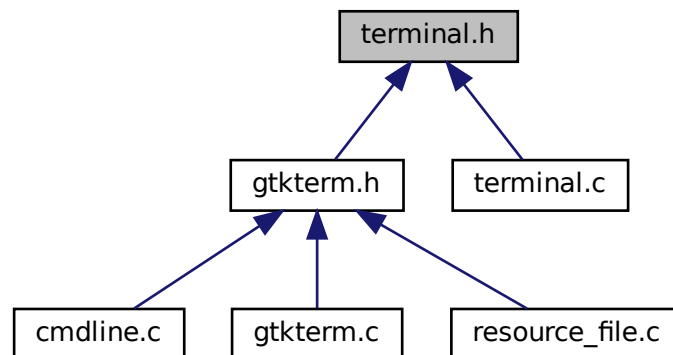
```

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

Include dependency graph for terminal.h:



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



Classes

- [struct _GtkTermTerminal](#)

Macros

- [#define GTKTERM_TERMINAL_TYPE](#) `gtkterm_terminal_get_type()`

5.31.1 Macro Definition Documentation

5.31.1.1 GTKTERM_TERMINAL_TYPE

```
#define GTKTERM_TERMINAL_TYPE gtkterm_terminal_get_type()
```

5.32 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 <term_config.h>
22 #include <serial.h>
23
24 G_BEGIN_DECLS
25
26 struct _GtkTermTerminal {
27     uint8_t view_mode;          /*!! ASCII or HEX view mode
28     //      GtkTermBuffer *term_buffer;
29     display_config_t *term_conf; /*!! The configuration loaded from the keyfile
30     port_config_t *port_conf;    /*!! Port configuration used in this terminal
31     char *active_section;        /*!! Active section in this window from config file
32
33     char *filename;             /*!! File to send
34
35     GdkRGBA *term_foreground;   /*!! Foreground (text) color of this terminal
36     GdkRGBA *term_background;   /*!! Background color
37
38     VteTerminal vte_object;     /*!! The actual terminal
39 };
40
41
42 #define GTKTERM_TERMINAL_TYPE gtkterm_terminal_get_type()
43 G_DECLARE_FINAL_TYPE (GtkTermTerminal, gtkterm_terminal, GTKTERM, TERMINAL, VteTerminal)
44
45 G_END_DECLS
46
47 #endif // TERMINAL_H

```


Index

- [_GtkTerm](#), [9](#)
 - [config](#), [10](#)
 - [g_config_group](#), [10](#)
 - [g_port_group](#), [10](#)
 - [g_term_group](#), [10](#)
 - [initial_section](#), [10](#)
 - [parent_instance](#), [10](#)
- [_GtkTermConfiguration](#), [11](#)
 - [parent_instance](#), [11](#)
- [_GtkTermConfigurationClass](#), [12](#)
 - [parent_class](#), [12](#)
- [_GtkTermTerminal](#), [12](#)
 - [active_section](#), [14](#)
 - [filename](#), [14](#)
 - [port_conf](#), [14](#)
 - [term_background](#), [14](#)
 - [term_conf](#), [14](#)
 - [term_foreground](#), [14](#)
 - [view_mode](#), [15](#)
 - [vte_object](#), [15](#)
- [_GtkTermTerminalClass](#), [15](#)
 - [vte_class](#), [15](#)
- [_GtkTermWindow](#), [16](#)
 - [fullscreen](#), [17](#)
 - [height](#), [17](#)
 - [infobar](#), [17](#)
 - [maximized](#), [17](#)
 - [menubutton](#), [17](#)
 - [message](#), [18](#)
 - [parent_instance](#), [18](#)
 - [scrolled_window](#), [18](#)
 - [status](#), [18](#)
 - [terminal_window](#), [18](#)
 - [toolmenu](#), [18](#)
 - [width](#), [19](#)
- [action](#)
 - [macro_t](#), [24](#)
- [active_section](#)
 - [_GtkTermTerminal](#), [14](#)
 - [display_config_t](#), [20](#)
- [add_input](#)
 - [files.h](#), [39](#)
- [add_shortcuts](#)
 - [macros.h](#), [53](#)
- [ASCII_VIEW](#)
 - [interface.h](#), [47](#)
- [background_color](#)
 - [display_config_t](#), [20](#)
- [baudrate](#)
 - [port_config_t](#), [25](#)
- [bits](#)
 - [port_config_t](#), [25](#)
- [block_cursor](#)
 - [display_config_t](#), [20](#)
- [buffer.c](#), [27](#)
 - [clear_buffer](#), [28](#)
 - [clear_func](#), [31](#)
 - [create_buffer](#), [28](#)
 - [delete_buffer](#), [28](#)
 - [insert_timestamp](#), [29](#)
 - [overlapped](#), [31](#)
 - [put_chars](#), [29](#)
 - [set_clear_func](#), [29](#)
 - [set_display_func](#), [29](#)
 - [timestamp_on](#), [31](#)
 - [TIMESTAMP_SIZE](#), [28](#)
 - [unset_clear_func](#), [29](#)
 - [unset_display_func](#), [29](#)
 - [virt_col_pos](#), [31](#)
 - [write_buffer](#), [30](#)
 - [write_buffer_with_func](#), [30](#)
 - [write_func](#), [31](#)
- [buffer.h](#), [32](#), [35](#)
 - [BUFFER_SIZE](#), [32](#)
 - [clear_buffer](#), [32](#)
 - [create_buffer](#), [33](#)
 - [delete_buffer](#), [33](#)
 - [put_chars](#), [33](#)
 - [set_clear_func](#), [33](#)
 - [set_display_func](#), [33](#)
 - [unset_clear_func](#), [33](#)
 - [unset_display_func](#), [34](#)
 - [write_buffer](#), [34](#)
 - [write_buffer_with_func](#), [34](#)
- [BUFFER_LENGTH](#)
 - [cmdline.c](#), [36](#)
 - [resource_file.c](#), [56](#)
- [BUFFER_SIZE](#)
 - [buffer.h](#), [32](#)
- [char_queue](#)
 - [display_config_t](#), [20](#)
 - [port_config_t](#), [26](#)
- [clear_buffer](#)
 - [buffer.c](#), [28](#)
 - [buffer.h](#), [32](#)
- [clear_func](#)
 - [buffer.c](#), [31](#)

- closure
 - macro_t, [24](#)
- cmdline.c, [35](#)
 - BUFFER_LENGTH, [36](#)
 - gtkterm_add_cmdline_options, [36](#)
 - MAX_SECTION_LENGTH, [36](#)
- cmdline.h, [36](#), [37](#)
 - g_term_group, [37](#)
 - gtkterm_add_cmdline_options, [37](#)
- COLUMN_ACTION
 - macros.c, [50](#)
- COLUMN_SHORTCUT
 - macros.c, [50](#)
- columns
 - display_config_t, [20](#)
- CONF_ITEM_LAST
 - resource_file.h, [62](#)
- CONF_ITEM_LENGTH
 - resource_file.h, [60](#)
- CONF_ITEM_SERIAL_BAUDRATE
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_BITS
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_DISABLE_PORT_LOCK
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_FLOW_CONTROL
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_PARITY
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_PORT
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX
 - resource_file.h, [61](#)
- CONF_ITEM_SERIAL_STOPBITS
 - resource_file.h, [61](#)
- CONF_ITEM_TERM_BACKGROUND_ALPHA
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_BACKGROUND_BLUE
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_BACKGROUND_GREEN
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_BACKGROUND_RED
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_BLOCK_CURSOR
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_COLS
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_CRLF_AUTO
 - resource_file.h, [61](#)
- CONF_ITEM_TERM_ECHO
 - resource_file.h, [61](#)
- CONF_ITEM_TERM_FONT
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_FOREGROUND_ALPHA
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_FOREGROUND_BLUE
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_FOREGROUND_GREEN
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_FOREGROUND_RED
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_MACROS
 - resource_file.h, [61](#)
- CONF_ITEM_TERM_RAW_FILENAME
 - resource_file.h, [61](#)
- CONF_ITEM_TERM_ROWS
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_SCROLLBACK
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_SHOW_CURSOR
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_TIMESTAMP
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_VISUAL_BELL
 - resource_file.h, [62](#)
- CONF_ITEM_TERM_WAIT_CHAR
 - resource_file.h, [61](#)
- CONF_ITEM_TERM_WAIT_DELAY
 - resource_file.h, [61](#)
- config
 - _GtkTerm, [10](#)
 - interface.c, [46](#)
- config_file
 - GtkTermConfigurationPrivate, [23](#)
- CONFIGURATION_FILENAME
 - resource_file.c, [56](#)
- convert_macros_to_string
 - macros.c, [50](#)
 - macros.h, [53](#)
- convert_string_to_macros
 - macros.c, [50](#)
 - macros.h, [53](#)
- create_buffer
 - buffer.c, [28](#)
 - buffer.h, [33](#)
- crlfauto
 - display_config_t, [21](#)
- DEFAULT_BAUDRATE
 - serial.h, [67](#)
- DEFAULT_BITS
 - serial.h, [67](#)
- DEFAULT_CHAR
 - term_config.h, [71](#)
- DEFAULT_DELAY
 - term_config.h, [71](#)
- DEFAULT_DELAY_RS485
 - term_config.h, [71](#)
- DEFAULT_ECHO
 - term_config.h, [71](#)
- default_filename
 - display_config_t, [21](#)
 - files.c, [38](#)
 - files.h, [39](#)
- DEFAULT_FLOW

- serial.h, 67
- DEFAULT_FONT
 - term_config.h, 71
- DEFAULT_PARITY
 - serial.h, 67
- DEFAULT_PORT
 - serial.h, 67
- DEFAULT_SCROLLBACK
 - term_config.h, 72
- DEFAULT_SECTION
 - resource_file.h, 60
- DEFAULT_STOPBITS
 - serial.h, 67
- DEFAULT_VISUAL_BELL
 - term_config.h, 72
- delay
 - display_config_t, 21
- delete_buffer
 - buffer.c, 28
 - buffer.h, 33
- disable_port_lock
 - port_config_t, 26
- display
 - interface.h, 48
- display_config_t, 19
 - active_section, 20
 - background_color, 20
 - block_cursor, 20
 - char_queue, 20
 - columns, 20
 - crlfauto, 21
 - default_filename, 21
 - delay, 21
 - echo, 21
 - font, 21
 - foreground_color, 21
 - rows, 21
 - scrollback, 21
 - show_cursor, 22
 - timestamp, 22
 - visual_bell, 22
- echo
 - display_config_t, 21
- filename
 - _GtkTermTerminal, 14
- files.c, 38
 - default_filename, 38
- files.h, 38, 40
 - add_input, 39
 - default_filename, 39
 - save_raw_file, 39
 - send_raw_file, 39
 - waiting_for_char, 39
- flow_control
 - port_config_t, 26
- font
 - display_config_t, 21
- foreground_color
 - display_config_t, 21
- fullscreen
 - _GtkTermWindow, 17
- g_config_group
 - _GtkTerm, 10
- g_port_group
 - _GtkTerm, 10
- g_term_group
 - _GtkTerm, 10
 - cmdline.h, 37
- get_port_string
 - serial.c, 65
 - serial.h, 68
- get_shortcuts
 - macros.c, 51
 - macros.h, 53
- GtkTerm
 - gtkterm.h, 43
- gtkterm.c, 40
 - gtkterm_signals, 42
 - GTKTERM_TYPE, 41
 - GTKTERM_WINDOW_TYPE, 41
 - main, 41
 - NR_OF_SIGNALS, 41
 - set_window_title, 41
- gtkterm.h, 42, 44
 - GtkTerm, 43
 - gtkterm_signals, 44
 - GtkTermWindow, 43
 - SIGNAL_LOAD_CONFIG, 44
 - SIGNAL_PRINT_SECTION, 44
 - SIGNAL_REMOVE_SECTION, 44
 - SIGNAL_SAVE_CONFIG, 44
- gtkterm_add_cmdline_options
 - cmdline.c, 36
 - cmdline.h, 37
- gtkterm_configuration_default_configuration
 - resource_file.c, 57
- gtkterm_configuration_new
 - resource_file.h, 62
- gtkterm_configuration_validate
 - resource_file.c, 57
- gtkterm_signals
 - gtkterm.c, 42
 - gtkterm.h, 44
- GTKTERM_TERMINAL_TYPE
 - terminal.h, 74
- GTKTERM_TYPE
 - gtkterm.c, 41
- GTKTERM_TYPE_CONFIGURATION
 - resource_file.h, 61
- GTKTERM_WINDOW_TYPE
 - gtkterm.c, 41
- GtkTermConfiguration
 - resource_file.h, 61
- GtkTermConfigurationItems
 - resource_file.c, 59

- resource_file.h, 63
- GtkTermConfigurationPrivate, 22
 - config_file, 23
 - key_file, 23
- GtkTermWindow
 - gtkterm.h, 43
- height
 - _GtkTermWindow, 17
- HEXADECIMAL_VIEW
 - interface.h, 47
- infobar
 - _GtkTermWindow, 17
- initial_section
 - _GtkTerm, 10
- insert_timestamp
 - buffer.c, 29
- interface.c, 45
 - config, 46
 - show_message, 46
 - timestamp_on, 46
 - virt_col_pos, 46
- interface.h, 47, 49
 - ASCII_VIEW, 47
 - display, 48
 - HEXADECIMAL_VIEW, 47
 - MSG_ERR, 48
 - MSG_WRN, 48
 - show_message, 48
 - Text, 48
- key_file
 - GtkTermConfigurationPrivate, 23
- LINE_FEED
 - serial.h, 68
- macro_count
 - macros.c, 51
 - macros.h, 54
- macro_t, 23
 - action, 24
 - closure, 24
 - shortcut, 24
- macros
 - macros.c, 52
 - macros.h, 54
- macros.c, 49
 - COLUMN_ACTION, 50
 - COLUMN_SHORTCUT, 50
 - convert_macros_to_string, 50
 - convert_string_to_macros, 50
 - get_shortcuts, 51
 - macro_count, 51
 - macros, 52
 - nr_of_macros, 52
 - NUM_COLUMNS, 50
 - remove_shortcuts, 51
- macros.h, 52, 55
 - add_shortcuts, 53
 - convert_macros_to_string, 53
 - convert_string_to_macros, 53
 - get_shortcuts, 53
 - macro_count, 54
 - macros, 54
 - remove_shortcuts, 54
- main
 - gtkterm.c, 41
- MAX_SECTION_LENGTH
 - cmdline.c, 36
- maximized
 - _GtkTermWindow, 17
- menubutton
 - _GtkTermWindow, 17
- message
 - _GtkTermWindow, 18
- MSG_ERR
 - interface.h, 48
- MSG_WRN
 - interface.h, 48
- nr_of_macros
 - macros.c, 52
- NR_OF_SIGNALS
 - gtkterm.c, 41
- NUM_COLUMNS
 - macros.c, 50
- on_set_config_options
 - resource_file.c, 58
 - resource_file.h, 62
- overlapped
 - buffer.c, 31
- parent_class
 - _GtkTermConfigurationClass, 12
- parent_instance
 - _GtkTerm, 10
 - _GtkTermConfiguration, 11
 - _GtkTermWindow, 18
- parity
 - port_config_t, 26
- POLL_DELAY
 - serial.h, 68
- port
 - port_config_t, 26
- port_conf
 - _GtkTermTerminal, 14
 - serial.c, 65
 - serial.h, 69
- port_config_t, 25
 - baudrate, 25
 - bits, 25
 - char_queue, 26
 - disable_port_lock, 26
 - flow_control, 26
 - parity, 26

- port, 26
- rs485_rts_time_after_transmit, 26
- rs485_rts_time_before_transmit, 26
- stopbits, 26
- put_chars
 - buffer.c, 29
 - buffer.h, 33
- README.md, 27
- README.source, 27
- RECEIVE_BUFFER
 - serial.h, 68
- remove_shortcuts
 - macros.c, 51
 - macros.h, 54
- resource_file.c, 55
 - BUFFER_LENGTH, 56
 - CONFIGURATION_FILENAME, 56
 - gtkterm_configuration_default_configuration, 57
 - gtkterm_configuration_validate, 57
 - GtkTermConfigurationItems, 59
 - on_set_config_options, 58
- resource_file.h, 59, 63
 - CONF_ITEM_LAST, 62
 - CONF_ITEM_LENGTH, 60
 - CONF_ITEM_SERIAL_BAUDRATE, 61
 - CONF_ITEM_SERIAL_BITS, 61
 - CONF_ITEM_SERIAL_DISABLE_PORT_LOCK, 61
 - CONF_ITEM_SERIAL_FLOW_CONTROL, 61
 - CONF_ITEM_SERIAL_PARITY, 61
 - CONF_ITEM_SERIAL_PORT, 61
 - CONF_ITEM_SERIAL_RS485_RTS_TIME_AFTER_TX, 61
 - CONF_ITEM_SERIAL_RS485_RTS_TIME_BEFORE_TX, 61
 - CONF_ITEM_SERIAL_STOPBITS, 61
 - CONF_ITEM_TERM_BACKGROUND_ALPHA, 62
 - CONF_ITEM_TERM_BACKGROUND_BLUE, 62
 - CONF_ITEM_TERM_BACKGROUND_GREEN, 62
 - CONF_ITEM_TERM_BACKGROUND_RED, 62
 - CONF_ITEM_TERM_BLOCK_CURSOR, 62
 - CONF_ITEM_TERM_COLS, 62
 - CONF_ITEM_TERM_CRLF_AUTO, 61
 - CONF_ITEM_TERM_ECHO, 61
 - CONF_ITEM_TERM_FONT, 62
 - CONF_ITEM_TERM_FOREGROUND_ALPHA, 62
 - CONF_ITEM_TERM_FOREGROUND_BLUE, 62
 - CONF_ITEM_TERM_FOREGROUND_GREEN, 62
 - CONF_ITEM_TERM_FOREGROUND_RED, 62
 - CONF_ITEM_TERM_MACROS, 61
 - CONF_ITEM_TERM_RAW_FILENAME, 61
 - CONF_ITEM_TERM_ROWS, 62
 - CONF_ITEM_TERM_SCROLLBACK, 62
 - CONF_ITEM_TERM_SHOW_CURSOR, 62
 - CONF_ITEM_TERM_TIMESTAMP, 62
 - CONF_ITEM_TERM_VISUAL_BELL, 62
 - CONF_ITEM_TERM_WAIT_CHAR, 61
 - CONF_ITEM_TERM_WAIT_DELAY, 61
 - DEFAULT_SECTION, 60
 - gtkterm_configuration_new, 62
 - GTKTERM_TYPE_CONFIGURATION, 61
 - GtkTermConfiguration, 61
 - GtkTermConfigurationItems, 63
 - on_set_config_options, 62
- rows
 - display_config_t, 21
- rs485_rts_time_after_transmit
 - port_config_t, 26
- rs485_rts_time_before_transmit
 - port_config_t, 26
- save_raw_file
 - files.h, 39
- scrollback
 - display_config_t, 21
- scrolled_window
 - _GtkTermWindow, 18
- send_raw_file
 - files.h, 39
- serial.c, 64
 - get_port_string, 65
 - port_conf, 65
 - serial_port_fd, 65
 - termios_save, 66
- serial.h, 66, 69
 - DEFAULT_BAUDRATE, 67
 - DEFAULT_BITS, 67
 - DEFAULT_FLOW, 67
 - DEFAULT_PARITY, 67
 - DEFAULT_PORT, 67
 - DEFAULT_STOPBITS, 67
 - get_port_string, 68
 - LINE_FEED, 68
 - POLL_DELAY, 68
 - port_conf, 69
 - RECEIVE_BUFFER, 68
 - serial_port_fd, 69
 - TRANSMIT_BUFFER, 68
- serial_port_fd
 - serial.c, 65
 - serial.h, 69
- set_clear_func
 - buffer.c, 29
 - buffer.h, 33
- set_display_func
 - buffer.c, 29
 - buffer.h, 33
- set_window_title
 - gtkterm.c, 41
- shortcut
 - macro_t, 24
- show_cursor
 - display_config_t, 22
- show_message
 - interface.c, 46

- interface.h, 48
- SIGNAL_LOAD_CONFIG
 - gtkterm.h, 44
- SIGNAL_PRINT_SECTION
 - gtkterm.h, 44
- SIGNAL_REMOVE_SECTION
 - gtkterm.h, 44
- SIGNAL_SAVE_CONFIG
 - gtkterm.h, 44
- status
 - _GtkTermWindow, 18
- stopbits
 - port_config_t, 26
- term_background
 - _GtkTermTerminal, 14
- term_conf
 - _GtkTermTerminal, 14
 - term_config.c, 70
 - term_config.h, 72
- term_config.c, 70
 - term_conf, 70
- term_config.h, 70, 72
 - DEFAULT_CHAR, 71
 - DEFAULT_DELAY, 71
 - DEFAULT_DELAY_RS485, 71
 - DEFAULT_ECHO, 71
 - DEFAULT_FONT, 71
 - DEFAULT_SCROLLBACK, 72
 - DEFAULT_VISUAL_BELL, 72
 - term_conf, 72
- term_forground
 - _GtkTermTerminal, 14
- terminal.c, 73
- terminal.h, 73, 75
 - GTKTERM_TERMINAL_TYPE, 74
- terminal_window
 - _GtkTermWindow, 18
- termios_save
 - serial.c, 66
- Text
 - interface.h, 48
- timestamp
 - display_config_t, 22
- timestamp_on
 - buffer.c, 31
 - interface.c, 46
- TIMESTAMP_SIZE
 - buffer.c, 28
- toolmenu
 - _GtkTermWindow, 18
- TRANSMIT_BUFFER
 - serial.h, 68
- unset_clear_func
 - buffer.c, 29
 - buffer.h, 33
- unset_display_func
 - buffer.c, 29
- buffer.h, 34
- view_mode
 - _GtkTermTerminal, 15
- virt_col_pos
 - buffer.c, 31
 - interface.c, 46
- visual_bell
 - display_config_t, 22
- vte_class
 - _GtkTermTerminalClass, 15
- vte_object
 - _GtkTermTerminal, 15
- waiting_for_char
 - files.h, 39
- width
 - _GtkTermWindow, 19
- write_buffer
 - buffer.c, 30
 - buffer.h, 34
- write_buffer_with_func
 - buffer.c, 30
 - buffer.h, 34
- write_func
 - buffer.c, 31