gtkterm

1.99.0

Generated by Doxygen 1.9.1

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
`<ctrl><shift>L`	Clear screen
`<ctrl><shift>R`	Send file
`<ctrl><shift>Q`	Quit
`<ctrl><shift>S`	Configure port
`<ctrl><shift>V`	Paste
`<ctrl><shift>C`	Copy
`<ctrl><shift>F`	Find
`<ctrl><shift>K`	Clear Scrollback
`<ctrl><shift>A`	Select All
`<ctrl><shift>B`	Send Break
`<ctrl>B`	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	killall -USR1 gtkterm
SIGUSR2	Close Port	killall -USR2 gtkterm

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

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 meson_post_install Namespace Reference

Variables

- **install_prefix** = os.environ['MESON_INSTALL_PREFIX']
- **schemadir** = os.path.join(**install_prefix**, 'share', 'glib-2.0', 'schemas')

5.1.1 Variable Documentation

5.1.1.1 install_prefix

```
meson_post_install.install_prefix = os.environ['MESON_INSTALL_PREFIX']
```

5.1.1.2 schemadir

```
meson_post_install.schemadir = os.path.join( install_prefix, 'share', 'glib-2.0', 'schemas')
```

Chapter 6

Class Documentation

6.1 display_config_t Struct Reference

`#include <term_config.h>`

Collaboration diagram for display_config_t:

Public Attributes

- gboolean **block_cursor**
- gboolean **show_cursor**
- char **char_queue**
- gboolean **echo**

- gboolean **crlfauto**
- gboolean **timestamp**
- int **delay**
- int **rows**
- int **columns**
- int **scrollback**
- gboolean **visual_bell**
- GdkRGBA **foreground_color**
- GdkRGBA **background_color**
- PangoFontDescription * **font**

6.1.1 Member Data Documentation

6.1.1.1 background_color

```
GdkRGBA display_config_t::background_color
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

6.1.1.2 block_cursor

```
gboolean display_config_t::block_cursor
```

Referenced by **dump_configuration_to_cli()**, and **hard_default_configuration()**.

6.1.1.3 char_queue

```
char display_config_t::char_queue
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_configuration_from_file()**, and **read_command_line()**.

6.1.1.4 columns

```
int display_config_t::columns
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

6.1.1.5 crlfauto

```
gboolean display_config_t::crlfauto
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

6.1.1.6 delay

```
int display_config_t::delay
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_↩configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

6.1.1.7 echo

```
gboolean display_config_t::echo
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_↩configuration_from_file()**, and **read_command_line()**.

6.1.1.8 font

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

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_↩configuration_from_file()**, and **validate_configuration()**.

6.1.1.9 foreground_color

```
GdkRGBA display_config_t::foreground_color
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

6.1.1.10 rows

```
int display_config_t::rows
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

6.1.1.11 scrollback

```
int display_config_t::scrollback
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

6.1.1.12 show_cursor

```
gboolean display_config_t::show_cursor
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

6.1.1.13 timestamp

```
gboolean display_config_t::timestamp
```

Referenced by **dump_configuration_to_cli()**, and **hard_default_configuration()**.

6.1.1.14 visual_bell

```
gboolean display_config_t::visual_bell
```

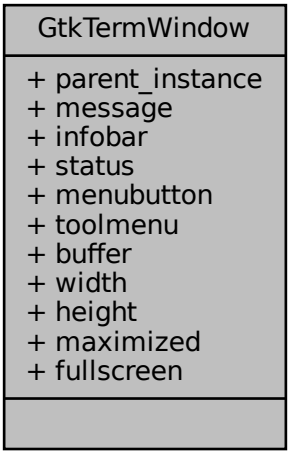
Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, and **load_configuration_from_file()**.

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

- /home/wvdakker/gtkterm_gtk4/src/ **term_config.h**

6.2 GtkTermWindow Struct Reference

Collaboration diagram for GtkTermWindow:



Public Attributes

- GtkApplicationWindow **parent_instance**
- GtkWidget * **message**
- GtkWidget * **infobar**
- GtkWidget * **status**
- GtkWidget * **menubutton**
- GMenuModel * **toolmenu**
- GtkTextBuffer * **buffer**
- int **width**
- int **height**
- gboolean **maximized**
- gboolean **fullscreen**

6.2.1 Member Data Documentation

6.2.1.1 buffer

```
GtkTextBuffer* GtkTermWindow::buffer
```

6.2.1.2 fullscreen

```
gboolean GtkTermWindow::fullscreen
```

6.2.1.3 height

```
int GtkTermWindow::height
```

6.2.1.4 infobar

```
GtkWidget* GtkTermWindow::infobar
```

6.2.1.5 maximized

```
gboolean GtkTermWindow::maximized
```

6.2.1.6 menubutton

```
GtkWidget* GtkTermWindow::menubutton
```

6.2.1.7 message

```
GtkWidget* GtkTermWindow::message
```

6.2.1.8 parent_instance

```
GtkApplicationWindow GtkTermWindow::parent_instance
```

6.2.1.9 status

```
GtkWidget* GtkTermWindow::status
```

6.2.1.10 toolmenu

```
GMenuModel* GtkTermWindow::toolmenu
```

6.2.1.11 width

```
int GtkTermWindow::width
```

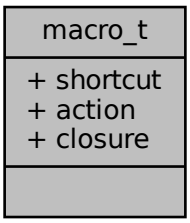
The documentation for this struct was generated from the following file:

- /home/wvdakker/gtkterm_gtk4/src/ **gtkterm.c**

6.3 macro_t Struct Reference

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

Collaboration diagram for macro_t:



Public Attributes

- char * **shortcut**
- char * **action**
- GClosure * **closure**

6.3.1 Member Data Documentation

6.3.1.1 action

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

Referenced by **create_shortcuts()**.

6.3.1.2 closure

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

6.3.1.3 shortcut

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

Referenced by **create_shortcuts()**.

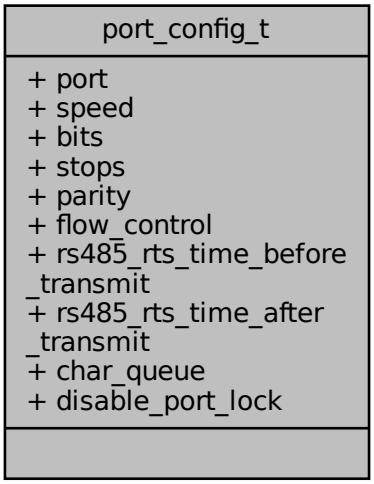
The documentation for this struct was generated from the following file:

- /home/wvdakker/gtkterm_gtk4/src/ **macros.h**

6.4 port_config_t Struct Reference

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

Collaboration diagram for port_config_t:



Public Attributes

- char **port** [256]
- long int **speed**
- int **bits**
- int **stops**
- int **parity**
- int **flow_control**
- int **rs485_rts_time_before_transmit**
- int **rs485_rts_time_after_transmit**
- char **char_queue**
- gboolean **disable_port_lock**

6.4.1 Member Data Documentation

6.4.1.1 bits

```
int port_config_t::bits
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **get_port_string()**, **hard_default_↩ configuration()**, **load_configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

6.4.1.2 char_queue

```
char port_config_t::char_queue
```

6.4.1.3 disable_port_lock

```
gboolean port_config_t::disable_port_lock
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_↩ configuration_from_file()**, and **read_command_line()**.

6.4.1.4 flow_control

```
int port_config_t::flow_control
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_↩ configuration_from_file()**, and **read_command_line()**.

6.4.1.5 parity

```
int port_config_t::parity
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **get_port_string()**, **hard_default_↩ configuration()**, **load_configuration_from_file()**, and **read_command_line()**.

6.4.1.6 port

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

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **get_port_string()**, **hard_default_↩ configuration()**, **load_configuration_from_file()**, and **read_command_line()**.

6.4.1.7 rs485_rts_time_after_transmit

```
int port_config_t::rs485_rts_time_after_transmit
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_↩ configuration_from_file()**, and **read_command_line()**.

6.4.1.8 rs485_rts_time_before_transmit

```
int port_config_t::rs485_rts_time_before_transmit
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_↩ configuration_from_file()**, and **read_command_line()**.

6.4.1.9 speed

```
long int port_config_t::speed
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **get_port_string()**, **hard_default_↩ configuration()**, **load_configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

6.4.1.10 stops

```
int port_config_t::stops
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **get_port_string()**, **hard_default_↩ configuration()**, **load_configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

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

- /home/wvdakker/gtkterm_gtk4/src/ **serial.h**

Chapter 7

File Documentation

7.1 /home/wvdakker/gtkterm_gtk4/meson_post_install.py File Reference

Namespaces

- **meson_post_install**

Variables

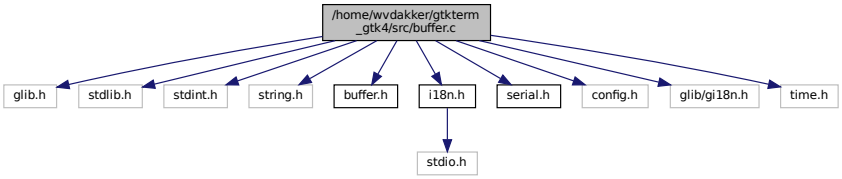
- **meson_post_install.install_prefix** = os.environ['MESON_INSTALL_PREFIX']
- **meson_post_install.schemadir** = os.path.join(install_prefix, 'share', 'glib-2.0', 'schemas')

7.2 /home/wvdakker/gtkterm_gtk4/README.md File Reference

7.3 /home/wvdakker/gtkterm_gtk4/src/buffer.c File Reference

```
#include <glib.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>
#include "buffer.h"
#include "i18n.h"
#include "serial.h"
#include <config.h>
#include <glib/gi18n.h>
#include <time.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, gboolean 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

- gboolean **timestamp_on**
- char **overlapped**
- guint **virt_col_pos**
- void(∗ **write_func**)(const char ∗, unsigned int) = NULL
- void(∗ **clear_func**)(void) = NULL

7.3.1 Macro Definition Documentation

7.3.1.1 TIMESTAMP_SIZE

```
#define TIMESTAMP_SIZE 50
```

7.3.2 Function Documentation

7.3.2.1 clear_buffer()

```
void clear_buffer (
            void  )
```

References **clear_func**.

7.3.2.2 create_buffer()

```
void create_buffer (
            void  )
```

7.3.2.3 delete_buffer()

```
void delete_buffer (
            void  )
```

7.3.2.4 insert_timestamp()

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

7.3.2.5 put_chars()

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

References **RECEIVE_BUFFER**, **timestamp_on**, and **TIMESTAMP_SIZE**.

7.3.2.6 set_clear_func()

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

References **clear_func**.

7.3.2.7 set_display_func()

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

References **write_func**.

7.3.2.8 unset_clear_func()

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

References **clear_func**.

7.3.2.9 unset_display_func()

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

References **write_func**.

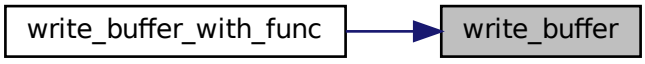
7.3.2.10 write_buffer()

```
void write_buffer (
            void  )
```

References **overlapped**, and **write_func**.

Referenced by **write_buffer_with_func()**.

Here is the caller graph for this function:

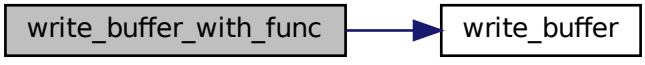


7.3.2.11 write_buffer_with_func()

```
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:



7.3.3 Variable Documentation

7.3.3.1 clear_func

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

Referenced by **clear_buffer()**, **set_clear_func()**, and **unset_clear_func()**.

7.3.3.2 overlapped

```
char overlapped
```

Referenced by **write_buffer()**.

7.3.3.3 timestamp_on

```
gboolean timestamp_on  [extern]
```

Referenced by **put_chars()**.

7.3.3.4 virt_col_pos

```
guint virt_col_pos  [extern]
```

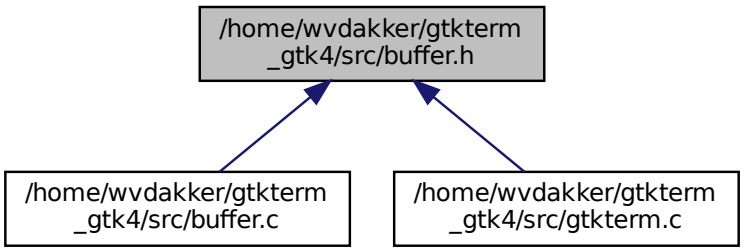
7.3.3.5 write_func

```
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()**.

7.4 /home/wvdakker/gtkterm_gtk4/src/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, gboolean)
- 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))

7.4.1 Macro Definition Documentation

7.4.1.1 BUFFER_SIZE

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

7.4.2 Function Documentation

7.4.2.1 clear_buffer()

```
void clear_buffer (
            void  )
```

References **clear_func**.

7.4.2.2 create_buffer()

```
void create_buffer (
            void  )
```

7.4.2.3 delete_buffer()

```
void delete_buffer (
            void  )
```

7.4.2.4 put_chars()

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

References **RECEIVE_BUFFER**, **timestamp_on**, and **TIMESTAMP_SIZE**.

7.4.2.5 set_clear_func()

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

References **clear_func**.

7.4.2.6 set_display_func()

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

7.4.2.7 unset_clear_func()

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

References **clear_func**.

7.4.2.8 unset_display_func()

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

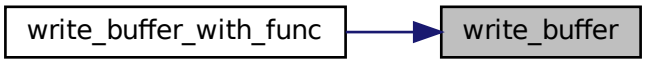
7.4.2.9 write_buffer()

```
void write_buffer (
            void  )
```

References **overlapped**, and **write_func**.

Referenced by **write_buffer_with_func()**.

Here is the caller graph for this function:



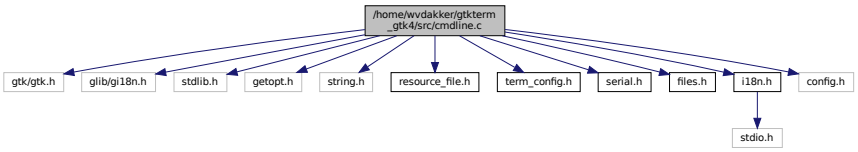
7.4.2.10 write_buffer_with_func()

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

7.5 /home/wvdakker/gtkterm_gtk4/src/cmdline.c File Reference

```
#include <gtk/gtk.h>
#include <glib/gi18n.h>
#include <stdlib.h>
#include <getopt.h>
#include <string.h>
#include "resource_file.h"
#include "term_config.h"
#include "serial.h"
#include "files.h"
#include "i18n.h"
#include <config.h>
```

Include dependency graph for cmdline.c:



Functions

- void **display_help** (void)
- int **read_command_line** (int argc, char ∗∗argv, char ∗configuration_to_read)

Variables

- struct configuration_port **config**

7.5.1 Function Documentation

7.5.1.1 display_help()

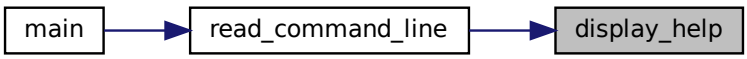
```
void display_help (
            void  )
```

References **i18n_printf()**.

Referenced by **read_command_line()**.

Here is the call graph for this function:

Here is the caller graph for this function:



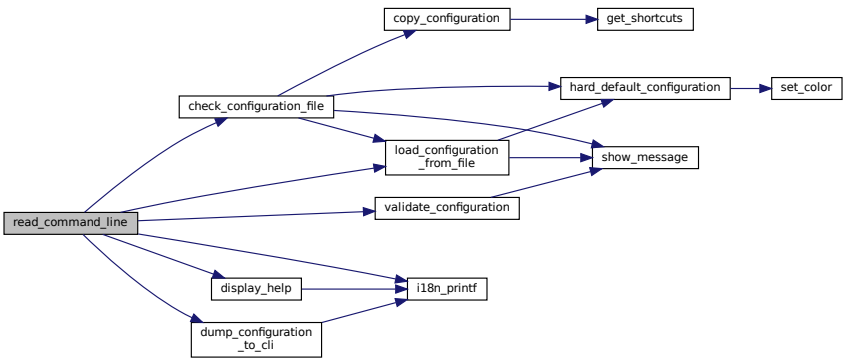
7.5.1.2 read_command_line()

```
int read_command_line (
            int argc,
            char ** argv,
            char * configuration_to_read )
```

References **port_config_t::bits**, **display_config_t::char_queue**, **check_configuration_file()**, **default_filename**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_help()**, **dump_configuration_to_cli()**, **display_config_t::echo**, **port_config_t::flow_control**, **i18n_printf()**, **load_configuration_from_file()**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, and **validate_configuration()**.

Referenced by **main()**.

Here is the call graph for this function:



Here is the caller graph for this function:

7.5.2 Variable Documentation

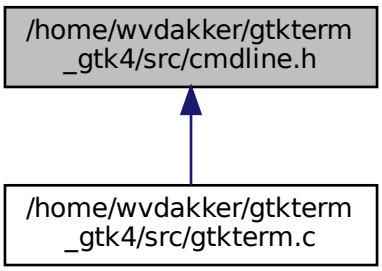
7.5.2.1 config

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

Referenced by **check_configuration_file()**, and **save_configuration_to_file()**.

7.6 /home/wvdakker/gtkterm_gtk4/src/cmdline.h File Reference

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



Functions

- int **read_command_line** (int, char **)

7.6.1 Function Documentation

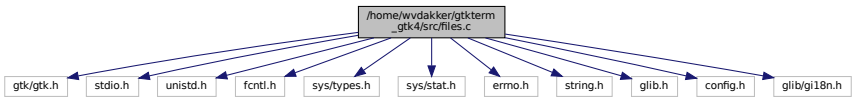
7.6.1.1 read_command_line()

```
int read_command_line (
            int ,
            char **  )
```

7.7 /home/wvdakker/gtkterm_gtk4/src/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

7.7.1 Variable Documentation

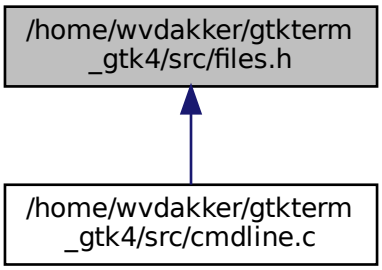
7.7.1.1 default_filename

```
char* default_filename = NULL
```

Referenced by **read_command_line()**.

7.8 /home/wvdakker/gtkterm_gtk4/src/files.h File Reference

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



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**

7.8.1 Function Documentation

7.8.1.1 add_input()

```
void add_input (
            void  )
```

7.8.1.2 save_raw_file()

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

7.8.1.3 send_raw_file()

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

7.8.2 Variable Documentation

7.8.2.1 default_filename

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

Referenced by **read_command_line()**.

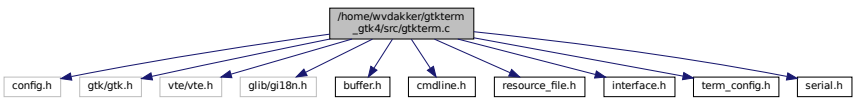
7.8.2.2 waiting_for_char

```
gboolean waiting_for_char  [extern]
```

7.9 /home/wvdakker/gtkterm_gtk4/src/gtkterm.c File Reference

```
#include "config.h"
#include <gtk/gtk.h>
#include <vte/vte.h>
#include <glib/gi18n.h>
#include "buffer.h"
#include "cmdline.h"
#include "resource_file.h"
#include "interface.h"
#include "term_config.h"
#include "serial.h"
```

Include dependency graph for gtkterm.c:



Classes

- struct **GtkTermWindow**

Typedefs

- typedef GtkApplication **GtkTerm**
- typedef GtkApplicationClass **GtkTermClass**
- typedef GtkApplicationWindowClass **GtkTermWindowClass**

Functions

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

7.9.1 Typedef Documentation

7.9.1.1 GtkTerm

```
typedef GtkApplication GtkTerm
```

7.9.1.2 GtkTermClass

```
typedef GtkApplicationClass GtkTermClass
```

7.9.1.3 GtkTermWindowClass

```
typedef GtkApplicationWindowClass GtkTermWindowClass
```

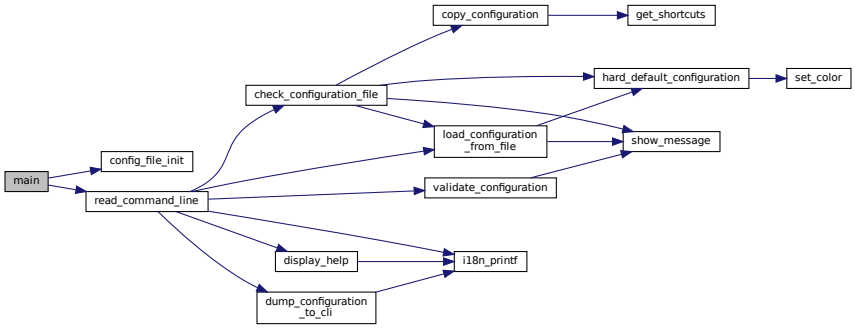
7.9.2 Function Documentation

7.9.2.1 main()

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

References **config_file_init()**, and **read_command_line()**.

Here is the call graph for this function:



7.9.2.2 set_window_title()

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

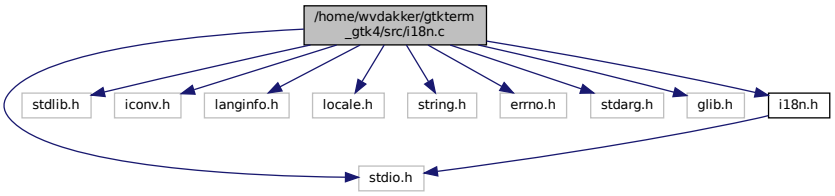
References **get_port_string()**.

Here is the call graph for this function:

7.10 /home/wvdakker/gtkterm_gtk4/src/i18n.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <iconv.h>
#include <langinfo.h>
#include <locale.h>
#include <string.h>
#include <errno.h>
#include <stdarg.h>
#include <glib.h>
#include "i18n.h"
```

Include dependency graph for i18n.c:



Functions

- int **i18n_printf** (const char *format,...)
- int **i18n_fprintf** (FILE *stream, const char *format,...)
- void **i18n_perror** (const char *s)
- char * **strerror_utf8** (int errornum)

7.10.1 Function Documentation

7.10.1.1 i18n_fprintf()

```
int i18n_fprintf (
            FILE * stream,
            const char * format,
             ...  )
```

7.10.1.2 i18n_perror()

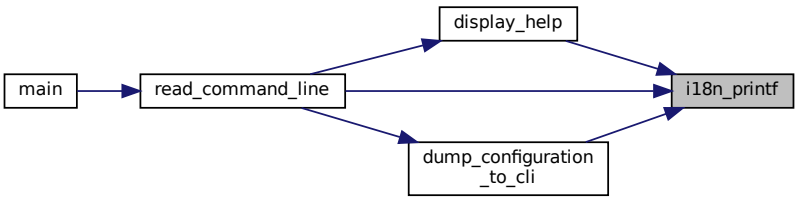
```
void i18n_perror (
            const char * s )
```

7.10.1.3 i18n_printf()

```
int i18n_printf (
            const char * format,
             ...  )
```

Referenced by **display_help()**, **dump_configuration_to_cli()**, and **read_command_line()**.

Here is the caller graph for this function:



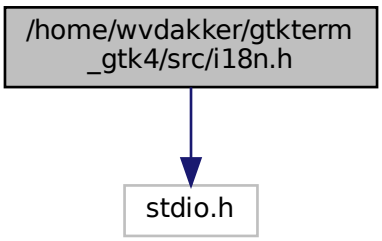
7.10.1.4 strerror_utf8()

```
char* strerror_utf8 (
            int errornum )
```

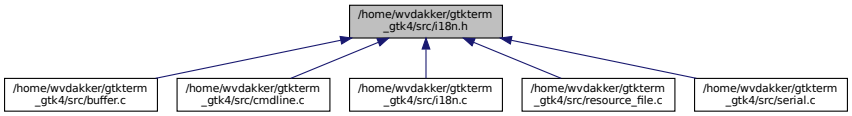
7.11 /home/wvdakker/gtkterm_gtk4/src/i18n.h File Reference

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

Include dependency graph for i18n.h:



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



Macros

- #define **I18N_H**

Functions

- int **i18n_printf** (const char *,...)
- int **i18n_fprintf** (FILE *, const char *,...)
- void **i18n_perror** (const char *)
- char * **strerror_utf8** (int)

7.11.1 Macro Definition Documentation

7.11.1.1 I18N_H

```
#define I18N_H
```

7.11.2 Function Documentation

7.11.2.1 i18n_fprintf()

```
int i18n_fprintf (
            FILE * stream,
            const char * format,
             ...  )
```

7.11.2.2 i18n_perror()

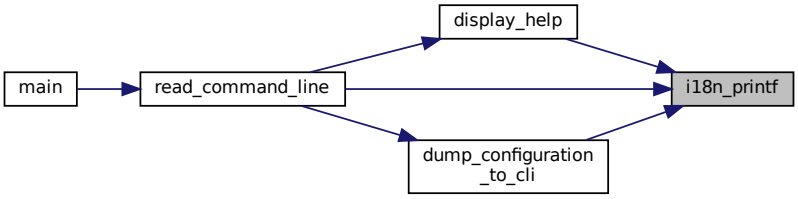
```
void i18n_perror (
            const char * s )
```

7.11.2.3 i18n_printf()

```
int i18n_printf (
            const char * format,
             ...  )
```

Referenced by **display_help()**, **dump_configuration_to_cli()**, and **read_command_line()**.

Here is the caller graph for this function:



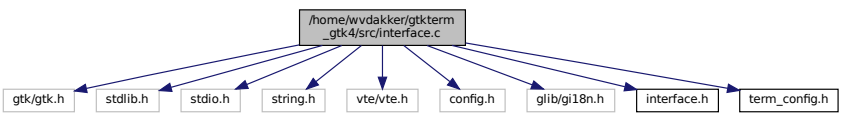
7.11.2.4 strerror_utf8()

```
char* strerror_utf8 (
            int errornum )
```

7.12 /home/wvdakker/gtkterm_gtk4/src/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

- gboolean **timestamp_on** = 0
- struct configuration_port **config**
- int **virt_col_pos** = 0
- GtkWidget ∗ **display** = NULL

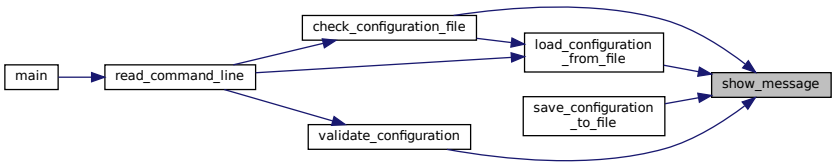
7.12.1 Function Documentation

7.12.1.1 show_message()

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

Referenced by **check_configuration_file()**, **load_configuration_from_file()**, **save_configuration_to_file()**, and **validate_configuration()**.

Here is the caller graph for this function:



7.12.2 Variable Documentation

7.12.2.1 config

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

7.12.2.2 display

```
GtkWidget* display = NULL
```

7.12.2.3 timestamp_on

```
gboolean timestamp_on = 0
```

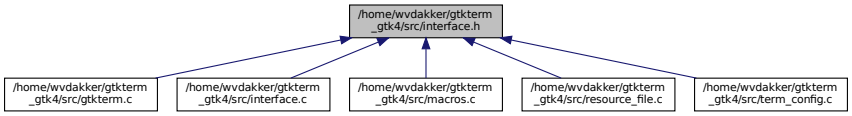
Referenced by **put_chars()**.

7.12.2.4 virt_col_pos

```
int virt_col_pos = 0
```

7.13 /home/wvdakker/gtkterm_gtk4/src/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**

7.13.1 Macro Definition Documentation

7.13.1.1 ASCII_VIEW

```
#define ASCII_VIEW 0
```

7.13.1.2 HEXADECIMAL_VIEW

```
#define HEXADECIMAL_VIEW 1
```

7.13.1.3 MSG_ERR

```
#define MSG_ERR 1
```

7.13.1.4 MSG_WRN

```
#define MSG_WRN 0
```

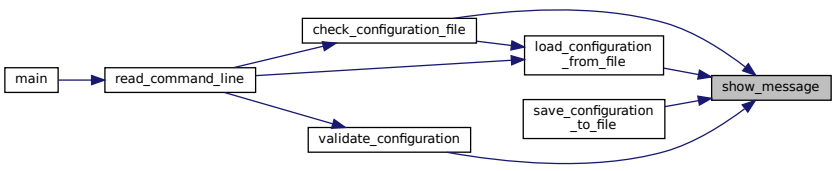
7.13.2 Function Documentation

7.13.2.1 show_message()

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

Referenced by **check_configuration_file()**, **load_configuration_from_file()**, **save_configuration_to_file()**, and **validate_configuration()**.

Here is the caller graph for this function:



7.13.3 Variable Documentation

7.13.3.1 display

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

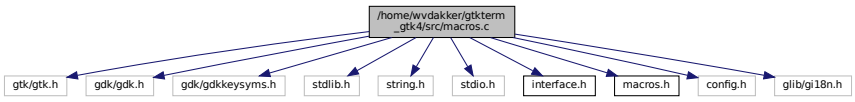
7.13.3.2 Text

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

7.14 /home/wvdakker/gtkterm_gtk4/src/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** }

Functions

- **macro_t** ∗ **get_shortcuts** (int ∗size)
- void **create_shortcuts** (**macro_t** ∗macro, int size)
- void **remove_shortcuts** (void)

Variables

- **macro_t** ∗ **macros** = NULL

7.14.1 Enumeration Type Documentation

7.14.1.1 anonymous enum

```
anonymous enum
```

Enumerator

COLUMN_SHORTCUT	
COLUMN_ACTION	
NUM_COLUMNS	

7.14.2 Function Documentation

7.14.2.1 create_shortcuts()

```
void create_shortcuts (
            macro_t * macro,
           int size )
```

References **macro_t::action**, **macros**, and **macro_t::shortcut**.

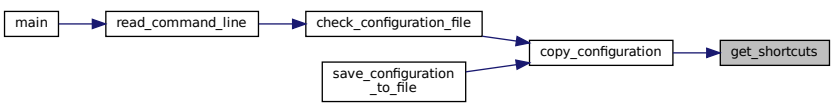
7.14.2.2 get_shortcuts()

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

References **macros**.

Referenced by **copy_configuration()**.

Here is the caller graph for this function:



7.14.2.3 remove_shortcuts()

```
void remove_shortcuts (
            void  )
```

References **macros**.

7.14.3 Variable Documentation

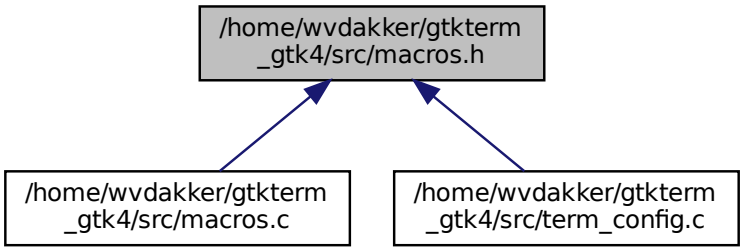
7.14.3.1 macros

```
macro_t* macros = NULL
```

Referenced by **copy_configuration()**, **create_shortcuts()**, **get_shortcuts()**, and **remove_shortcuts()**.

7.15 /home/wvdakker/gtkterm_gtk4/src/macros.h File Reference

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



Classes

- struct **macro_t**

Functions

- void **remove_shortcuts** (void)
- void **add_shortcuts** (void)
- void **create_shortcuts** (**macro_t** *, gint)
- **macro_t** * **get_shortcuts** (gint *)

7.15.1 Function Documentation

7.15.1.1 add_shortcuts()

```
void add_shortcuts (
            void  )
```

7.15.1.2 create_shortcuts()

```
void create_shortcuts (
             macro_t * ,
            gint  )
```

7.15.1.3 get_shortcuts()

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

7.15.1.4 remove_shortcuts()

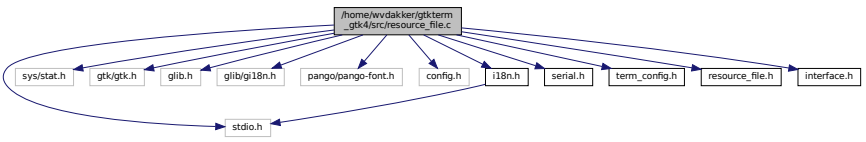
```
void remove_shortcuts (
            void  )
```

References **macros**.

7.16 /home/wvdakker/gtkterm_gtk4/src/resource_file.c File Reference

```
#include <stdio.h>
#include <sys/stat.h>
#include <gtk/gtk.h>
#include <glib.h>
#include <glib/gi18n.h>
#include <pango/pango-font.h>
#include <config.h>
#include "i18n.h"
#include "serial.h"
#include "term_config.h"
#include "resource_file.h"
#include "interface.h"
```

Include dependency graph for resource_file.c:



Macros

- #define **CONFIGURATION_FILENAME** ".gtktermrc"

Enumerations

- enum {
  CONF_ITEM_PORT , **CONF_ITEM_SPEED** , **CONF_ITEM_BITS** , **CONF_ITEM_STOPBITS** ,
  CONF_ITEM_PARITY , **CONF_ITEM_FLOW_CONTROL** , **CONF_ITEM_WAIT_DELAY** , **CONF_ITEM_↩**
  WAIT_CHAR ,
  CONF_ITEM_RS485_RTS_TIME_BEFORE_TX , **CONF_ITEM_RS485_RTS_TIME_AFTER_TX** , **CONF↩**
  _ITEM_ECHO , **CONF_ITEM_CRLF_AUTO** ,
  CONF_ITEM_DISABLE_PORT_LOCK , **CONF_ITEM_FONT** , **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 }

Functions

- void **config_file_init** (void)
- void **dump_configuration_to_cli** (char ∗section)
- void **save_configuration_to_file** (GKeyFile ∗ **config**, const char ∗section)
- int **load_configuration_from_file** (const char ∗section)
- int **check_configuration_file** (void)
- void **copy_configuration** (GKeyFile ∗configrc, const char ∗section)
- int **remove_section** (char ∗cfg_file, char ∗section)
- void **hard_default_configuration** (void)
- void **validate_configuration** (void)
- void **set_color** (GdkRGBA ∗color, float R, float G, float B, float A)

Variables

- GFile ∗ **config_file**
- char **ConfigurationItem** [ ][32]

7.16.1 Macro Definition Documentation

7.16.1.1 CONFIGURATION_FILENAME

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

7.16.2 Enumeration Type Documentation

7.16.2.1 anonymous enum

```
anonymous enum
```

Enumerator

CONF_ITEM_PORT	
CONF_ITEM_SPEED	
CONF_ITEM_BITS	
CONF_ITEM_STOPBITS	
CONF_ITEM_PARITY	
CONF_ITEM_FLOW_CONTROL	
CONF_ITEM_WAIT_DELAY	
CONF_ITEM_WAIT_CHAR	
CONF_ITEM_RS485_RTS_TIME_BEFORE_TX	
CONF_ITEM_RS485_RTS_TIME_AFTER_TX	
CONF_ITEM_ECHO	
CONF_ITEM_CRLF_AUTO	
CONF_ITEM_DISABLE_PORT_LOCK	
CONF_ITEM_FONT	
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	

7.16.3 Function Documentation

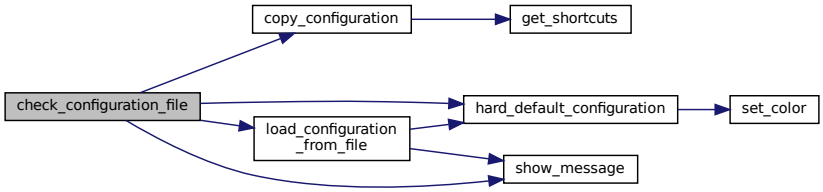
7.16.3.1 check_configuration_file()

```
int check_configuration_file (
            void  )
```

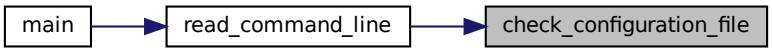
References **config**, **config_file**, **copy_configuration()**, **hard_default_configuration()**, **load_configuration_from_file()**, **MSG_WRN**, and **show_message()**.

Referenced by **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



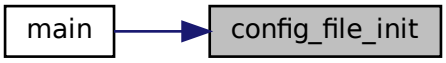
7.16.3.2 config_file_init()

```
void config_file_init (
            void  )
```

References **config_file**, and **CONFIGURATION_FILENAME**.

Referenced by **main()**.

Here is the caller graph for this function:



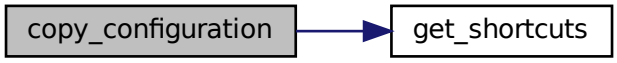
7.16.3.3 copy_configuration()

```
void copy_configuration (
            GKeyFile * configrc,
            const char * section )
```

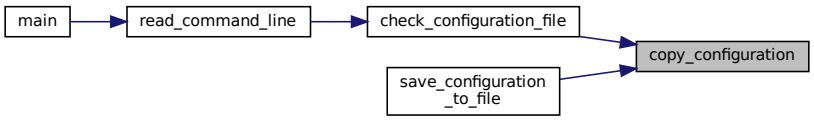
References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::char_queue**, **display_config_t::columns**, **CONF_ITEM_BITS**, **CONF_ITEM_CRLF_AUTO**, **CONF_ITEM_DISABLE_PORT_LOCK**, **CONF_ITEM_ECHO**, **CONF_ITEM_FLOW_CONTROL**, **CONF_ITEM_FONT**, **CONF_ITEM_PARITY**, **CONF_ITEM_PORT**, **CONF_ITEM_RS485_RTS_TIME_AFTER_TX**, **CONF_ITEM_RS485_RTS_TIME_BEFORE_TX**, **CONF_ITEM_SPEED**, **CONF_ITEM_STOPBITS**, **CONF_ITEM_TERM_BACKGROUND_ALPHA**, **CONF_ITEM_TERM_BACKGROUND_BLUE**, **CONF_ITEM_TERM_BACKGROUND_GREEN**, **CONF_ITEM_TERM_BACKGROUND_RED**, **CONF_ITEM_TERM_COLS**, **CONF_ITEM_TERM_FOREGROUND_ALPHA**, **CONF_ITEM_TERM_FOREGROUND_BLUE**, **CONF_ITEM_TERM_FOREGROUND_GREEN**, **CONF_ITEM_TERM_FOREGROUND_RED**, **CONF_ITEM_TERM_ROWS**, **CONF_ITEM_TERM_SCROLLBACK**, **CONF_ITEM_TERM_SHOW_CURSOR**, **CONF_ITEM_TERM_VISUAL_BELL**, **CONF_ITEM_WAIT_CHAR**, **CONF_ITEM_WAIT_DELAY**, **ConfigurationItem**, **display_config_t::crlfauto**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**, **port_config_t::flow_control**, **display_config_t::font**, **display_config_t::foreground_color**, **get_shortcuts()**, **macros**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **display_config_t::show_cursor**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, and **display_config_t::visual_bell**.

Referenced by **check_configuration_file()**, and **save_configuration_to_file()**.

Here is the call graph for this function:



Here is the caller graph for this function:



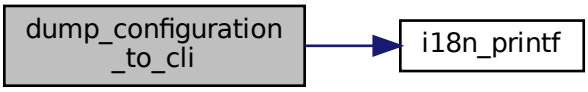
7.16.3.4 dump_configuration_to_cli()

```
void dump_configuration_to_cli (
            char * section )
```

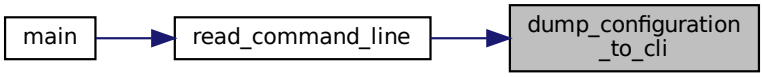
References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::block_cursor**, **display_config_t::char_queue**, **display_config_t::columns**, **display_config_t::crlfauto**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**, **port_config_t::flow_control**, **display_config_t::font**, **display_config_t::foreground_color**, **i18n_printf()**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **display_config_t::show_cursor**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, **display_config_t::timestamp**, and **display_config_t::visual_bell**.

Referenced by **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



7.16.3.5 hard_default_configuration()

```
void hard_default_configuration (
            void  )
```

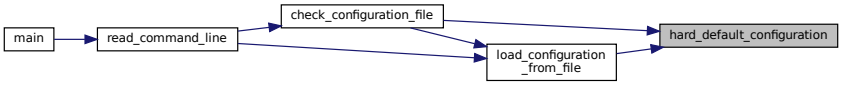
References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::block_cursor**, **display_config_t::char_queue**, **display_config_t::columns**, **display_config_t::crlfauto**, **DEFAULT_BITS**, **DEFAULT_CHAR**, **DEFAULT_DELAY**, **DEFAULT_DELAY_RS485**, **DEFAULT_ECHO**, **DEFAULT_FLOW**, **DEFAULT_FONT**, **DEFAULT_PARITY**, **DEFAULT_PORT**, **DEFAULT_SCROLLBACK**, **DEFAULT_SPEED**, **DEFAULT_STOP**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**,

port_config_t::flow_control, **display_config_t::font**, **display_config_t::foreground_color**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **set_color()**, **display_config_t::show_cursor**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, **display_config_t::timestamp**, and **display_config_t::visual_bell**.

Referenced by **check_configuration_file()**, and **load_configuration_from_file()**.

Here is the call graph for this function:

Here is the caller graph for this function:



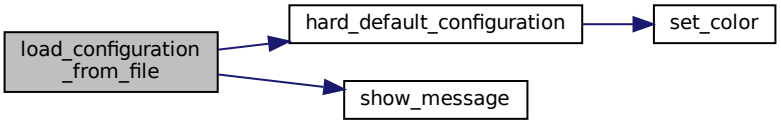
7.16.3.6 load_configuration_from_file()

```
int load_configuration_from_file (
            const char * section )
```

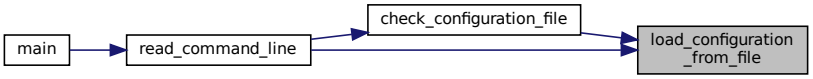
References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::char_queue**, **display_config_t::columns**, **CONF_ITEM_BITS**, **CONF_ITEM_CRLF_AUTO**, **CONF_ITEM_DISABLE_PORT_LOCK**, **CONF_ITEM_ECHO**, **CONF_ITEM_FLOW_CONTROL**, **CONF_ITEM_FONT**, **CONF_ITEM_PARITY**, **CONF_ITEM_PORT**, **CONF_ITEM_RS485_RTS_TIME_AFTER_TX**, **CONF_ITEM_RS485_RTS_TIME_BEFORE_TX**, **CONF_ITEM_SPEED**, **CONF_ITEM_STOPBITS**, **CONF_ITEM_TERM_BACKGROUND_ALPHA**, **CONF_ITEM_TERM_BACKGROUND_BLUE**, **CONF_ITEM_TERM_BACKGROUND_GREEN**, **CONF_ITEM_TERM_BACKGROUND_RED**, **CONF_ITEM_TERM_COLS**, **CONF_ITEM_TERM_FOREGROUND_ALPHA**, **CONF_ITEM_TERM_FOREGROUND_BLUE**, **CONF_ITEM_TERM_FOREGROUND_GREEN**, **CONF_ITEM_TERM_FOREGROUND_RED**, **CONF_ITEM_TERM_ROWS**, **CONF_ITEM_TERM_SCROLLBACK**, **CONF_ITEM_TERM_SHOW_CURSOR**, **CONF_ITEM_TERM_VISUAL_BELL**, **CONF_ITEM_WAIT_CHAR**, **CONF_ITEM_WAIT_DELAY**, **config_file**, **ConfigurationItem**, **display_config_t::crlfauto**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**, **port_config_t::flow_control**, **display_config_t::font**, **display_config_t::foreground_color**, **hard_default_configuration()**, **MSG_ERR**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **display_config_t::show_cursor**, **show_message()**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, and **display_config_t::visual_bell**.

Referenced by **check_configuration_file()**, and **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



7.16.3.7 remove_section()

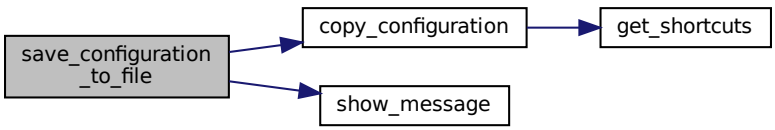
```
int remove_section (
            char * cfg_file,
            char * section )
```

7.16.3.8 save_configuration_to_file()

```
void save_configuration_to_file (
            GKeyFile * config,
            const char * section )
```

References **config**, **config_file**, **copy_configuration()**, **MSG_WRN**, and **show_message()**.

Here is the call graph for this function:

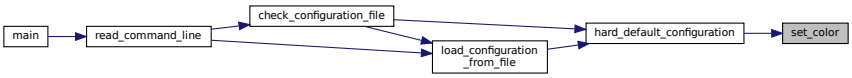


7.16.3.9 set_color()

```
void set_color (
            GdkRGBA * color,
            float R,
            float G,
            float B,
            float A )
```

Referenced by **hard_default_configuration()**.

Here is the caller graph for this function:



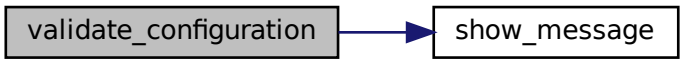
7.16.3.10 validate_configuration()

```
void validate_configuration (
            void  )
```

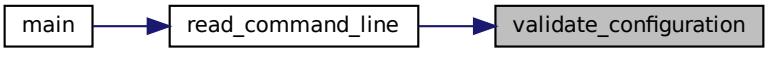
References **port_config_t::bits**, **DEFAULT_BITS**, **DEFAULT_DELAY**, **DEFAULT_FONT**, **DEFAULT_STOP**, **display_config_t::delay**, **display_config_t::font**, **MSG_ERR**, **port_conf**, **show_message()**, **port_config_t::speed**, **port_config_t::stops**, and **term_conf**.

Referenced by **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



7.16.4 Variable Documentation

7.16.4.1 config_file

```
GFile* config_file
```

Referenced by **check_configuration_file()**, **config_file_init()**, **load_configuration_from_file()**, and **save_configuration_to_file()**.

7.16.4.2 ConfigurationItem

```
char ConfigurationItem[][32]
```

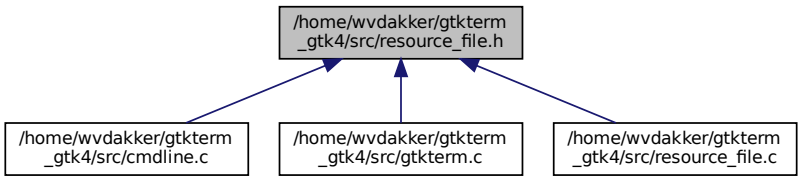
Initial value:

```
= {
        "port",
        "speed",
        "bits",
        "stopbits",
        "parity",
        "flow_control",
        "wait_delay",
        "wait_char",
        "rs485_rts_time_before_tx",
        "rs485_rts_time_after_tx",
        "echo",
        "crlfauto",
        "disable_port_lock",
        "font",
        "term_show_cursor",
        "term_rows",
        "term_columns",
        "term_scrollback",
        "term_visual_bell",
        "term_foreground_red",
        "term_foreground_green",
        "term_foreground_blue",
        "term_foreground_alpha",
        "term_background_red",
        "term_background_green",
        "term_background_blue",
        "term_background_alpha"
}
```

Referenced by **copy_configuration()**, and **load_configuration_from_file()**.

7.17 /home/wvdakker/gtkterm_gtk4/src/resource_file.h File Reference

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



Functions

- void **config_file_init** (void)
- void **save_configuration_to_file** (GKeyFile *, const char *)
- int **load_configuration_from_file** (const char *)
- int **check_configuration_file** ()
- void **dump_configuration_to_cli** (char *)
- void **hard_default_configuration** (void)
- void **validate_configuration** (void)
- void **copy_configuration** (GKeyFile *, const char *)
- int **remove_section** (char *cfg_file, char *section)
- void **set_color** (GdkRGBA *color, float, float, float, float)

Variables

- GFile * **config_file**

7.17.1 Function Documentation

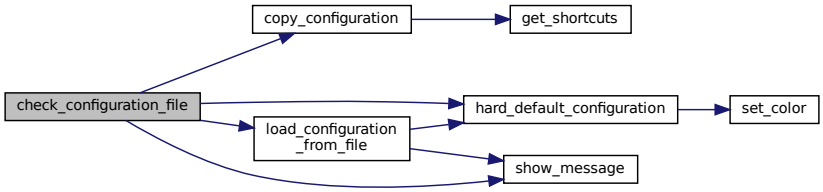
7.17.1.1 check_configuration_file()

```
int check_configuration_file ( )
```

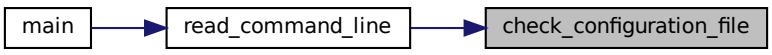
References **config**, **config_file**, **copy_configuration()**, **hard_default_configuration()**, **load_configuration_from_file()**, **MSG_WRN**, and **show_message()**.

Referenced by **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



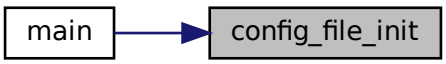
7.17.1.2 config_file_init()

```
void config_file_init (
            void  )
```

References **config_file**, and **CONFIGURATION_FILENAME**.

Referenced by **main()**.

Here is the caller graph for this function:



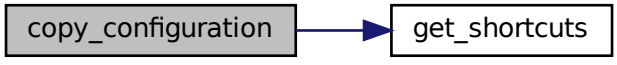
7.17.1.3 copy_configuration()

```
void copy_configuration (
            GKeyFile * configrc,
            const char * section )
```

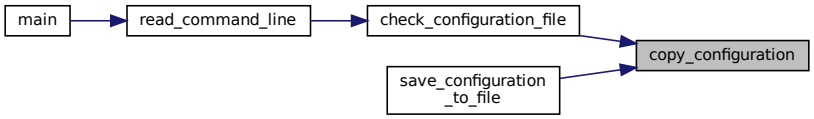
References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::char_queue**, **display_config_t::columns**, **CONF_ITEM_BITS**, **CONF_ITEM_CRLF_AUTO**, **CONF_ITEM_DISABLE↩ _PORT_LOCK**, **CONF_ITEM_ECHO**, **CONF_ITEM_FLOW_CONTROL**, **CONF_ITEM_FONT**, **CONF↩ _ITEM_PARITY**, **CONF_ITEM_PORT**, **CONF_ITEM_RS485_RTS_TIME_AFTER_TX**, **CONF_ITEM↩ _RS485_RTS_TIME_BEFORE_TX**, **CONF_ITEM_SPEED**, **CONF_ITEM_STOPBITS**, **CONF_ITEM_↩ TERM_BACKGROUND_ALPHA**, **CONF_ITEM_TERM_BACKGROUND_BLUE**, **CONF_ITEM_TERM_↩ BACKGROUND_GREEN**, **CONF_ITEM_TERM_BACKGROUND_RED**, **CONF_ITEM_TERM_COLS**, **CONF_↩ ITEM_TERM_FOREGROUND_ALPHA**, **CONF_ITEM_TERM_FOREGROUND_BLUE**, **CONF_ITEM_TERM_↩ FOREGROUND_GREEN**, **CONF_ITEM_TERM_FOREGROUND_RED**, **CONF_ITEM_TERM_ROWS**, **CONF↩ _ITEM_TERM_SCROLLBACK**, **CONF_ITEM_TERM_SHOW_CURSOR**, **CONF_ITEM_TERM_VISUAL_BELL**, **CONF_ITEM_WAIT_CHAR**, **CONF_ITEM_WAIT_DELAY**, **ConfigurationItem**, **display_config_t::crlfauto**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**, **port_config_t↩ ::flow_control**, **display_config_t::font**, **display_config_t::foreground_color**, **get_shortcuts()**, **macros**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485↩ _rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **display_config_t::show_cursor**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, and **display_↩ config_t::visual_bell**.

Referenced by **check_configuration_file()**, and **save_configuration_to_file()**.

Here is the call graph for this function:



Here is the caller graph for this function:



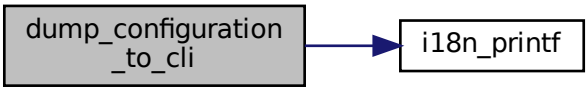
7.17.1.4 dump_configuration_to_cli()

```
void dump_configuration_to_cli (
            char * section )
```

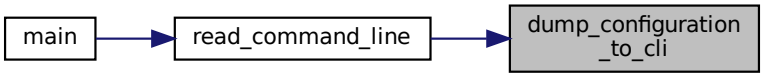
References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::block_cursor**, **display_config_t::char_queue**, **display_config_t::columns**, **display_config_t::crlfauto**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**, **port_config_t::flow_control**, **display_config_t::font**, **display_config_t::foreground_color**, **i18n_printf()**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **display_config_t::show_cursor**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, **display_config_t::timestamp**, and **display_config_t::visual_bell**.

Referenced by **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



7.17.1.5 hard_default_configuration()

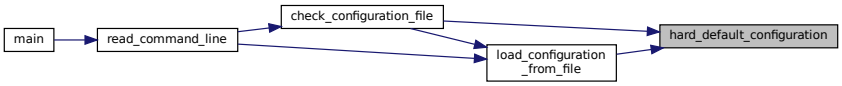
```
void hard_default_configuration (
            void  )
```

References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::block_cursor**, **display_config_t::char_queue**, **display_config_t::columns**, **display_config_t::crlfauto**, **DEFAULT_BITS**, **DEFAULT_CHAR**, **DEFAULT_DELAY**, **DEFAULT_DELAY_RS485**, **DEFAULT_ECHO**, **DEFAULT_FLOW**, **DEFAULT_FONT**, **DEFAULT_PARITY**, **DEFAULT_PORT**, **DEFAULT_SCROLLBACK**, **DEFAULT_SPEED**, **DEFAULT_STOP**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**, **port_config_t::flow_control**, **display_config_t::font**, **display_config_t::foreground_color**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **set_color()**, **display_config_t::show_cursor**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, **display_config_t::timestamp**, and **display_config_t::visual_bell**.

Referenced by **check_configuration_file()**, and **load_configuration_from_file()**.

Here is the call graph for this function:

Here is the caller graph for this function:



7.17.1.6 load_configuration_from_file()

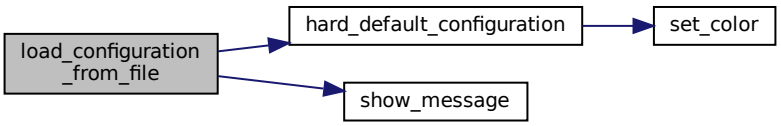
```
int load_configuration_from_file (
            const char * section )
```

References **display_config_t::background_color**, **port_config_t::bits**, **display_config_t::char_queue**, **display_config_t::columns**, **CONF_ITEM_BITS**, **CONF_ITEM_CRLF_AUTO**, **CONF_ITEM_DISABLE_PORT_LOCK**, **CONF_ITEM_ECHO**, **CONF_ITEM_FLOW_CONTROL**, **CONF_ITEM_FONT**, **CONF_ITEM_PARITY**, **CONF_ITEM_PORT**, **CONF_ITEM_RS485_RTS_TIME_AFTER_TX**, **CONF_ITEM_RS485_RTS_TIME_BEFORE_TX**, **CONF_ITEM_SPEED**, **CONF_ITEM_STOPBITS**, **CONF_ITEM_TERM_BACKGROUND_ALPHA**, **CONF_ITEM_TERM_BACKGROUND_BLUE**, **CONF_ITEM_TERM_BACKGROUND_GREEN**, **CONF_ITEM_TERM_BACKGROUND_RED**, **CONF_ITEM_TERM_COLS**, **CONF_ITEM_TERM_FOREGROUND_ALPHA**, **CONF_ITEM_TERM_FOREGROUND_BLUE**, **CONF_ITEM_TERM_FOREGROUND_GREEN**, **CONF_ITEM_TERM_FOREGROUND_RED**, **CONF_ITEM_TERM_ROWS**, **CONF_ITEM_TERM_SCROLLBACK**, **CONF_ITEM_TERM_SHOW_CURSOR**, **CONF_ITEM_TERM_VISUAL_BELL**,

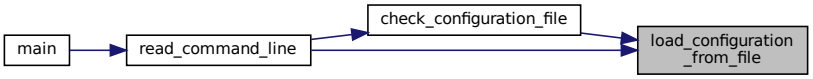
CONF_ITEM_WAIT_CHAR, **CONF_ITEM_WAIT_DELAY**, **config_file**, **ConfigurationItem**, **display_config_t::crlfauto**, **display_config_t::delay**, **port_config_t::disable_port_lock**, **display_config_t::echo**, **port_config_t::flow_control**, **display_config_t::font**, **display_config_t::foreground_color**, **hard_default_configuration()**, **MSG_ERR**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **display_config_t::rows**, **port_config_t::rs485_rts_time_after_transmit**, **port_config_t::rs485_rts_time_before_transmit**, **display_config_t::scrollback**, **display_config_t::show_cursor**, **show_message()**, **port_config_t::speed**, **port_config_t::stops**, **term_conf**, and **display_config_t::visual_bell**.

Referenced by **check_configuration_file()**, and **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



7.17.1.7 remove_section()

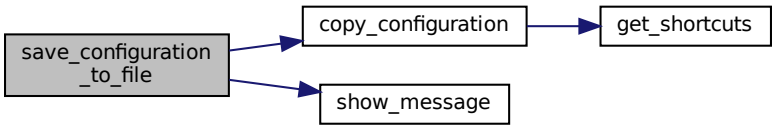
```
int remove_section (
            char * cfg_file,
            char * section )
```

7.17.1.8 save_configuration_to_file()

```
void save_configuration_to_file (
            GKeyFile * config,
            const char * section )
```

References **config**, **config_file**, **copy_configuration()**, **MSG_WRN**, and **show_message()**.

Here is the call graph for this function:

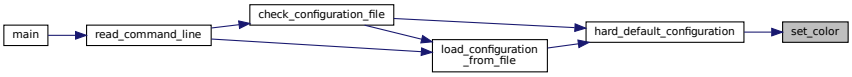


7.17.1.9 set_color()

```
void set_color (
            GdkRGBA * color,
            float R,
            float G,
            float B,
            float A )
```

Referenced by **hard_default_configuration()**.

Here is the caller graph for this function:



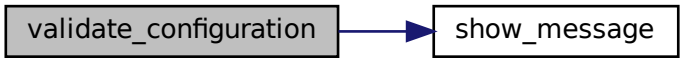
7.17.1.10 validate_configuration()

```
void validate_configuration (
            void  )
```

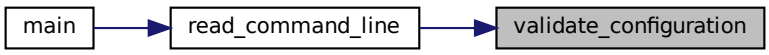
References **port_config_t::bits**, **DEFAULT_BITS**, **DEFAULT_DELAY**, **DEFAULT_FONT**, **DEFAULT_STOP**, **display_config_t::delay**, **display_config_t::font**, **MSG_ERR**, **port_conf**, **show_message()**, **port_config_t::speed**, **port_config_t::stops**, and **term_conf**.

Referenced by **read_command_line()**.

Here is the call graph for this function:



Here is the caller graph for this function:



7.17.2 Variable Documentation

7.17.2.1 config_file

```
GFile* config_file  [extern]
```

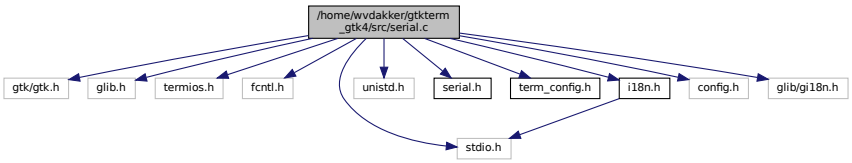
Referenced by **check_configuration_file()**, **config_file_init()**, **load_configuration_from_file()**, and **save_↩ configuration_to_file()**.

7.18 /home/wvdakker/gtkterm_gtk4/src/serial.c File Reference

```
#include <gtk/gtk.h>
#include <glib.h>
#include <termios.h>
#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include "serial.h"
#include "term_config.h"
#include "i18n.h"
#include <config.h>
```

```
#include <glib/gi18n.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

7.18.1 Function Documentation

7.18.1.1 get_port_string()

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

References **port_config_t::bits**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **serial_port_fd**, **port_config_t::speed**, and **port_config_t::stops**.

Referenced by **set_window_title()**.

Here is the caller graph for this function:

7.18.2 Variable Documentation

7.18.2.1 port_conf

```
port_config_t port_conf
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **get_port_string()**, **hard_default_configuration()**, **load_configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

7.18.2.2 serial_port_fd

```
int serial_port_fd = -1
```

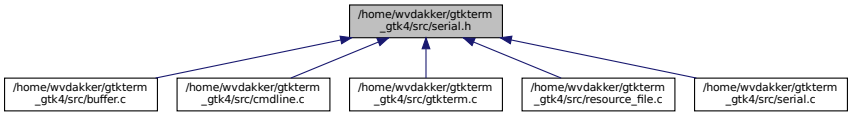
Referenced by **get_port_string()**.

7.18.2.3 termios_save

```
struct termios termios_save
```

7.19 /home/wvdakker/gtkterm_gtk4/src/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_SPEED** 115200
- #define **DEFAULT_PARITY** 0
- #define **DEFAULT_BITS** 8
- #define **DEFAULT_STOP** 1
- #define **DEFAULT_FLOW** 0
- #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**

7.19.1 Macro Definition Documentation

7.19.1.1 DEFAULT_BITS

```
#define DEFAULT_BITS 8
```

7.19.1.2 DEFAULT_FLOW

```
#define DEFAULT_FLOW 0
```

7.19.1.3 DEFAULT_PARITY

```
#define DEFAULT_PARITY 0
```

7.19.1.4 DEFAULT_PORT

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

7.19.1.5 DEFAULT_SPEED

```
#define DEFAULT_SPEED 115200
```

7.19.1.6 DEFAULT_STOP

```
#define DEFAULT_STOP 1
```

7.19.1.7 LINE_FEED

```
#define LINE_FEED 0x0A
```

7.19.1.8 POLL_DELAY

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

7.19.1.9 RECEIVE_BUFFER

```
#define RECEIVE_BUFFER 8192
```

7.19.1.10 TRANSMIT_BUFFER

```
#define TRANSMIT_BUFFER 4096
```

7.19.2 Function Documentation

7.19.2.1 get_port_string()

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

References **port_config_t::bits**, **port_config_t::parity**, **port_config_t::port**, **port_conf**, **serial_port_fd**, **port_config_t::speed**, and **port_config_t::stops**.

Referenced by **set_window_title()**.

Here is the caller graph for this function:

7.19.3 Variable Documentation

7.19.3.1 port_conf

```
port_config_t port_conf  [extern]
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **get_port_string()**, **hard_default_configuration()**, **load_configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

7.19.3.2 serial_port_fd

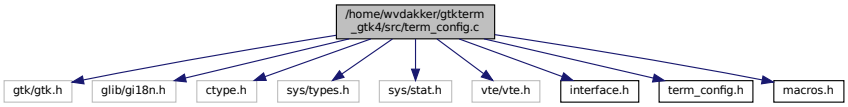
```
int serial_port_fd  [extern]
```

Referenced by **get_port_string()**.

7.20 /home/wvdakker/gtkterm_gtk4/src/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:



Macros

- #define **CONFIGURATION_FILENAME** ".gtktermrc"

Variables

- **display_config_t term_conf**

7.20.1 Macro Definition Documentation

7.20.1.1 CONFIGURATION_FILENAME

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

7.20.2 Variable Documentation

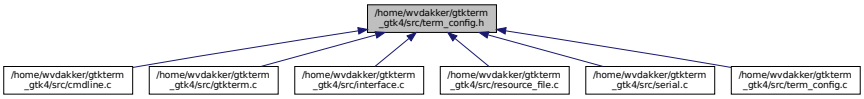
7.20.2.1 term_conf

```
display_config_t term_conf
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

7.21 /home/wvdakker/gtkterm_gtk4/src/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

Variables

- **display_config_t term_conf**

7.21.1 Macro Definition Documentation

7.21.1.1 DEFAULT_CHAR

```
#define DEFAULT_CHAR -1
```

7.21.1.2 DEFAULT_DELAY

```
#define DEFAULT_DELAY 0
```

7.21.1.3 DEFAULT_DELAY_RS485

```
#define DEFAULT_DELAY_RS485 30
```

7.21.1.4 DEFAULT_ECHO

```
#define DEFAULT_ECHO FALSE
```

7.21.1.5 DEFAULT_FONT

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

7.21.1.6 DEFAULT_SCROLLBACK

```
#define DEFAULT_SCROLLBACK 10000
```

7.21.2 Variable Documentation

7.21.2.1 term_conf

```
display_config_t term_conf  [extern]
```

Referenced by **copy_configuration()**, **dump_configuration_to_cli()**, **hard_default_configuration()**, **load_configuration_from_file()**, **read_command_line()**, and **validate_configuration()**.

Index