CONIO

2.1

Generated by Doxygen 1.8.2

Contents

Chapter 1

Main Page

Borland-style CONIO implementation for MinGW/Dev-C++. Homepage

Send any improvements to this library to me, I'll do new release of this devpak.

For the example of use, look at example in the `Examples\conio\conio_test.c` subdirectory of your Dev-C++ directory. It's simple:

1. Include `conio2.h`.

2. Link with `libconio.a` (add `-lconio` parameter to linker).

Functions defined already in MinGW's `conio.h`

`conio2.h` automatically includes `conio.h`. It also provides several #defines so you can use all these functions without underscores.

```
char* _cgets (char*);
int _cprintf (const char*, ...);
int _cputs (const char*);
int _cscanf (char*, ...);

int _getch (void);
int _getche (void);
int _kbhit (void);
int _putch (int);
int _ungetch (int);

int getch (void);
int getche (void);
int kbhit (void);
int putch (int);
int ungetch (int);
```

About the makefile: The makefile is written for gcc and GNU make. The "all" target passes -m32 option to the gcc. The "all64" target passes -m64 option to the gcc.

Author

Hongli Lai hongli@telekabel.nl
tkorrovi tkorrovi@altavista.net on 2002/02/26.
Andrew Westcott ajwestco@users.sourceforge.net
Michal Molhanec michal@molhanec.net

Version

2.1

```
Changes in 2.1 (2013-12-05 MM):
    clreol() fix (clear also the color attribute)
    new functions clearkeybuf(), delay(), switchbackground()
      and flashbackground()
    gettext & puttext code simplified
    64bit targets in the makefile

Changes in 2.0:
    make library aware of console window size
    make library aware of console window position in console buffer
    added support for UNICODE Win32 API
    added constream-like manipulators for C++
    implemented highvideo(), insline(), lowvideo(), movetext(), normvideo(),
      gettextinfo(), getpass(), cputsxy(), putchxy(), inittextinfo()
    fixed delline()
```

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 3

Data Structure Index

3.1   Data Structures

Here are the data structures with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 conio Namespace Reference

This namespace contain all C++ specific things.

Functions

- _Setxy setxy (int x, int y)

    setxy manipulator
- _Setclr setclr (int color)

    setclr manipulator
- _Setbk setbk (int color)

    setbk manipulator
- _Setattr setattr (int _attr)

    setattr manipulator
- _Setcrsrtype setcrsrtype (int type)

    setcrsrtype manipulator
- std::ostream & clrscr (std::ostream &o)

    clrscr manipulator
- std::ostream & clreol (std::ostream &o)

    clreol manipulator
- std::ostream & highvideo (std::ostream &o)

    highvideo manipulator
- std::ostream & lowvideo (std::ostream &o)

    lowvideo manipulator
- std::ostream & normvideo (std::ostream &o)

    normvideo manipulator
- std::ostream & delline (std::ostream &o)

    delline manipulator
- std::ostream & insline (std::ostream &o)

    insline manipulator

5.1.1 Detailed Description

This namespace contain all C++ specific things.

5.1.2 Function Documentation

5.1.2.1 std::ostream& conio::clreol (std::ostream & *o*) `[inline]`

clreol manipulator

See Also

[clreol(void)](#)

5.1.2.2 std::ostream& conio::clrscr (std::ostream & *o*) `[inline]`

clrscr manipulator

See Also

[clrscr(void)](#)

5.1.2.3 std::ostream& conio::delline (std::ostream & *o*) `[inline]`

delline manipulator

See Also

[delline(void)](#)

5.1.2.4 std::ostream& conio::highvideo (std::ostream & *o*) `[inline]`

highvideo manipulator

See Also

[highvideo(void)](#)

5.1.2.5 std::ostream& conio::insline (std::ostream & *o*) `[inline]`

insline manipulator

See Also

[insline(void)](#)

5.1.2.6 std::ostream& conio::lowvideo (std::ostream & *o*) `[inline]`

lowvideo manipulator

See Also

[lowvideo(void)](#)

5.1.2.7 **std::ostream& conio::normvideo (std::ostream & *o*)** `[inline]`

normvideo manipulator

See Also

> [normvideo(void)](#)

5.1.2.8 **_Setattr conio::setattr (int *_attr*)** `[inline]`

setattr manipulator

See Also

> [textattr](#)

5.1.2.9 **_Setbk conio::setbk (int *color*)** `[inline]`

setbk manipulator

See Also

> [textbackground](#)

5.1.2.10 **_Setclr conio::setclr (int *color*)** `[inline]`

setclr manipulator

See Also

> [textcolor](#)

5.1.2.11 **_Setcrsrtype conio::setcrsrtype (int *type*)** `[inline]`

setcrsrtype manipulator

See Also

> [_setcursortype](#)

5.1.2.12 **_Setxy conio::setxy (int *x,* int *y*)** `[inline]`

setxy manipulator

See Also

> [gotoxy](#)

Chapter 6

Data Structure Documentation

6.1 char_info Struct Reference

Structure used by gettext/puttext.

```
#include <conio2.h>
```

Data Fields

- char letter

  character value
- unsigned short attr

  attribute value

6.1.1 Detailed Description

Structure used by gettext/puttext.

See Also

_conio_gettext
puttext

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

- conio2.h

6.2 text_info Struct Reference

Structure holding information about screen.

```
#include <conio2.h>
```

Data Fields

- unsigned char curx

  cursor coordinate x
- unsigned char cury

cursor coordinate y

- unsigned short attribute

    current text attribute

- unsigned short normattr

    original value of text attribute after start of the application.

- unsigned char screenwidth

    screen width

- unsigned char screenheight

    screen height

6.2.1 Detailed Description

Structure holding information about screen.

See Also

> gettextinfo
> inittextinfo

6.2.2 Field Documentation

6.2.2.1 unsigned short text_info::normattr

original value of text attribute after start of the application.

If you don't called the `inittextinfo` on the beginning of the application, this always will be black background and light gray foreground

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

- conio2.h

Chapter 7

File Documentation

7.1 conio2.h File Reference

A conio implementation for Mingw/Dev-C++.

```
#include <conio.h>
```

Data Structures

- struct text_info

    Structure holding information about screen.
- struct char_info

    Structure used by gettext/puttext.

Macros

- #define gettext _conio_gettext

    Define alias for _conio_gettext.

- #define cgets _cgets

    This defines enables you to use all MinGW conio.h functions without underscore.

  #### Cursor types

  Predefined cursor types.

  - #define _NOCURSOR 0

      no cursor
  - #define _SOLIDCURSOR 100

      cursor filling whole cell
  - #define _NORMALCURSOR 20

      cursor filling 20 percent of cell height

Enumerations

- enum COLORS {
  BLACK, BLUE, GREEN, CYAN,
  RED, MAGENTA, BROWN, LIGHTGRAY,
  DARKGRAY, LIGHTBLUE, LIGHTGREEN, LIGHTCYAN,
  LIGHTRED, LIGHTMAGENTA, YELLOW, WHITE }

    Colors which you can use in your application.

Functions

- void gettextinfo (struct text_info ∗info)

    Returns information of the screen.
- void inittextinfo (void)

    Call this if you need real value of normattr attribute in the text_info structure.
- void clreol (void)

    Clears rest of the line from cursor position to the end of line without moving the cursor.
- void clrscr (void)

    Clears whole screen.
- void delline (void)

    Delete the current line (line on which is cursor) and then moves all lines below one line up.
- void insline (void)

    Insert blank line at the cursor position.
- void _conio_gettext (int left, int top, int right, int bottom, struct char_info ∗buf)

    Gets text from the screen.
- void puttext (int left, int top, int right, int bottom, struct char_info ∗buf)

    Puts text back to the screen.
- void movetext (int left, int top, int right, int bottom, int destleft, int desttop)

    Copies text.
- void gotoxy (int x, int y)

    Moves cursor to the specified position.
- void cputsxy (int x, int y, char ∗str)

    Puts string at the specified position.
- void putchxy (int x, int y, char ch)

    Puts char at the specified position.
- void _setcursortype (int type)

    Sets the cursor type.
- void textattr (int _attr)

    Sets attribute of text.
- void normvideo (void)

    Sets text attribute back to value it had after program start.
- void textbackground (int color)

    Sets text background color.
- void textcolor (int color)

    Sets text foreground color.
- int wherex (void)

    Reads the cursor X position.
- int wherey (void)

    Reads the cursor Y position.
- char ∗ getpass (const char ∗prompt, char ∗str)

    Reads password.
- void highvideo (void)

    Makes foreground colors light.
- void lowvideo (void)

    Makes foreground colors dark.
- void delay (int ms)

    Pauses program execution for a given time.
- void switchbackground (int color)

    Replaces background color in the whole window.
- void flashbackground (int color, int ms)

    Changes background color for a given time and then it restores it back.
- void clearkeybuf (void)

    Clears the keyboard buffer.

7.1.1 Detailed Description

A conio implementation for Mingw/Dev-C++. Written by: Hongli Lai hongli@telekabel.nl tkorrovi tkorrovi@altavista.net on 2002/02/26. Andrew Westcott ajwestco@users.sourceforge.net Michal Molhanec michal@molhanec.net

Offered for use in the public domain without any warranty.

7.1.2 Macro Definition Documentation

7.1.2.1 #define gettext _conio_gettext

Define alias for _conio_gettext.

If you want to use gettext function from some other library (e.g. GNU gettext) you have to define _CONIO_NO_GE-TTEXT_ so you won't get name conflict.

7.1.3 Enumeration Type Documentation

7.1.3.1 enum COLORS

Colors which you can use in your application.

Enumerator:

    BLACK black color

    BLUE blue color

    GREEN green color

    CYAN cyan color

    RED red color

    MAGENTA magenta color

    BROWN brown color

    LIGHTGRAY light gray color

    DARKGRAY dark gray color

    LIGHTBLUE light blue color

    LIGHTGREEN light green color

    LIGHTCYAN light cyan color

    LIGHTRED light red color

    LIGHTMAGENTA light magenta color

    YELLOW yellow color

    WHITE white color

7.1.4 Function Documentation

7.1.4.1 void _conio_gettext (int *left,* int *top,* int *right,* int *bottom,* struct char_info ∗ *buf*)

Gets text from the screen.

If you haven't defined _CONIO_NO_GETTEXT_ prior to including conio2.h you can use this function also under the gettext name.

See Also

> char_info
> puttext

Parameters

left	Left coordinate of the rectangle, inclusive, starting from 1.
top	Top coordinate of the rectangle, inclusive, starting from 1.
right	Right coordinate of the rectangle, inclusive, starting from 1.
bottom	Bottom coordinate of the rectangle, inclusive, starting from 1.
buf	You have to pass buffer of size `(right - left + 1) * (bottom - top + 1) * sizeof(char_info)`.

7.1.4.2 void _setcursortype (int *type*)

Sets the cursor type.

See Also

> cursortypes

Parameters

type	cursor type, under Win32 it is height of the cursor in percents

7.1.4.3 void clearkeybuf (void)

Clears the keyboard buffer.

To see it in effect run `conio_test` and try to press a key during the 'Flashing...' phase.

7.1.4.4 void cputsxy (int *x,* int *y,* char ∗ *str*)

Puts string at the specified position.

Parameters

x	horizontal position
y	vertical position
str	string

7.1.4.5 void delay (int *ms*)

Pauses program execution for a given time.

See Also

> switchbackground

Parameters

ms	miliseconds

7.1.4.6   void delline (void)

Delete the current line (line on which is cursor) and then moves all lines below one line up.

Lines below the line are moved one line up.

7.1.4.7   void flashbackground (int *color,* int *ms*)

Changes background color for a given time and then it restores it back.

You can use it for visual bell. Does not modify textbackground().

See Also

switchbackground
delay

Parameters

color	background color
ms	miliseconds

7.1.4.8   char∗ getpass (const char ∗ *prompt,* char ∗ *str*)

Reads password.

This function behaves like cgets.

See Also

cgets

Parameters

prompt	prompt which will be displayed to user
str	string for the password. `str[0]` have to contain length of the `str` - 3

Returns

&`str[2]`, the password will be stored in `str` beginning at `str[2]`, in `str[1]` will be length of the string without \0, at `str[2 + str[1]]` will be \0.

7.1.4.9   void gettextinfo (struct text_info ∗ *info*)

Returns information of the screen.

See Also

text_info

7.1.4.10   void gotoxy (int *x,* int *y*)

Moves cursor to the specified position.

Parameters

x	horizontal position
y	vertical position

7.1.4.11 void highvideo (void)

Makes foreground colors light.

If the current foreground color is less than `DARKGRAY` adds 8 to the its value making dark colors light.

See Also

> COLORS
> lowvideo

7.1.4.12 void inittextinfo (void)

Call this if you need real value of normattr attribute in the text_info structure.

See Also

> text_info

7.1.4.13 void insline (void)

Insert blank line at the cursor position.

Original content of the line and content of lines below moves one line down. The last line is deleted.

7.1.4.14 void lowvideo (void)

Makes foreground colors dark.

If the current foreground color is higher than `LIGHTGRAY` substracts 8 from its value making light colors dark.

See Also

> COLORS
> highvideo

7.1.4.15 void movetext (int *left,* int *top,* int *right,* int *bottom,* int *destleft,* int *desttop*)

Copies text.

Parameters

left	Left coordinate of the rectangle, inclusive, starting from 1.
top	Top coordinate of the rectangle, inclusive, starting from 1.
right	Right coordinate of the rectangle, inclusive, starting from 1.
bottom	Bottom coordinate of the rectangle, inclusive, starting from 1.
destleft	Left coordinate of the destination rectangle.
desttop	Top coordinate of the destination rectangle.

7.1.4.16 void normvideo (void)

Sets text attribute back to value it had after program start.

It uses text_info's normattr value.

See Also

> text_info

7.1.4.17 void putchxy (int *x,* int *y,* char *ch*)

Puts char at the specified position.

Parameters

x	horizontal position
y	vertical position
ch	char

7.1.4.18 void puttext (int *left,* int *top,* int *right,* int *bottom,* struct **char_info ∗ *buf*)**

Puts text back to the screen.

See Also

> char_info
> _conio_gettext

Parameters

left	Left coordinate of the rectangle, inclusive, starting from 1.
top	Top coordinate of the rectangle, inclusive, starting from 1.
right	Right coordinate of the rectangle, inclusive, starting from 1.
bottom	Bottom coordinate of the rectangle, inclusive, starting from 1.
buf	You have to pass buffer of size `(right - left + 1) * (bottom - top + 1) * sizeof(char_info)`.

7.1.4.19 void switchbackground (int *color*)

Replaces background color in the whole window.

The text however is left intact. Does not modify textbackground().

See Also

> flashbackground

Parameters

color	background color

7.1.4.20   void textattr (int _attr)

Sets attribute of text.

Parameters

_attr	new text attribute

7.1.4.21   void textbackground (int *color*)

Sets text background color.

See Also

> COLORS

Parameters

color	new background color

7.1.4.22   void textcolor (int *color*)

Sets text foreground color.

See Also

> COLORS

Parameters

color	new foreground color

7.1.4.23   int wherex (void)

Reads the cursor X position.

Returns

> cursor X position

7.1.4.24   int wherey (void)

Reads the cursor Y position.

Returns

> cursor Y position

7.2   constream File Reference

A constream implementation for Mingw/Dev-C++.

```
#include <iostream>
#include "conio2.h"
```

Namespaces

- namespace conio

    This namespace contain all C++ specific things.

Functions

- _Setxy conio::setxy (int x, int y)

    setxy manipulator
- _Setclr conio::setclr (int color)

    setclr manipulator
- _Setbk conio::setbk (int color)

    setbk manipulator
- _Setattr conio::setattr (int _attr)

    setattr manipulator
- _Setcrsrtype conio::setcrsrtype (int type)

    setcrsrtype manipulator
- std::ostream & conio::clrscr (std::ostream &o)

    clrscr manipulator
- std::ostream & conio::clreol (std::ostream &o)

    clreol manipulator
- std::ostream & conio::highvideo (std::ostream &o)

    highvideo manipulator
- std::ostream & conio::lowvideo (std::ostream &o)

    lowvideo manipulator
- std::ostream & conio::normvideo (std::ostream &o)

    normvideo manipulator
- std::ostream & conio::delline (std::ostream &o)

    delline manipulator
- std::ostream & conio::insline (std::ostream &o)

    insline manipulator

7.2.1 Detailed Description

A constream implementation for Mingw/Dev-C++.

Warning

There is not implemented constream class, only manipulators for iostream, so use them on cin/cout.

Author

Michal Molhanec michal@molhanec.net

Offered for use in the public domain without any warranty.

Index