LIBJHI-SDL

2.0

Generated by Doxygen 1.8.6

Sat Nov 8 2014 02:20:15

Contents

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

File Index

2.1 File List

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

Chapter 3

Class Documentation

3.1 JHI_Color_RGB Struct Reference

Structure of the RGB color.

```
#include <jhi_colorsSt.h>
```

Public Attributes

- Uint8 r
- Uint8 g
- Uint8 b

3.1.1 Detailed Description

Structure of the RGB color.

3.1.2 Member Data Documentation

3.1.2.1 Uint8 JHI_Color_RGB::b

Blue value to color

3.1.2.2 Uint8 JHI_Color_RGB::g

Green value to color

3.1.2.3 Uint8 JHI_Color_RGB::r

Red value to color

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

- jhi_colorsSt.h

3.2 JHI_Effect Struct Reference

Struct of the configuration of the Effect.

```
#include <jhi_sound.h>
```

Public Attributes

- Mix_Chunk * mix_chunk

3.2.1 Detailed Description

Struct of the configuration of the Effect.

3.2.2 Member Data Documentation

3.2.2.1 Mix_Chunk* JHI_Effect::mix_chunk

SDL structure of Effect.

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

- jhi_sound.h

3.3 JHI_Font Struct Reference

Structure of configuration of the Font.

```
#include <jhi_font.h>
```

Public Attributes

- TTF_Font * font
- int length

3.3.1 Detailed Description

Structure of configuration of the Font.

3.3.2 Member Data Documentation

3.3.2.1 TTF_Font* JHI_Font::font

SDL structure of Font

3.3.2.2 int JHI_Font::length

Length of the Font

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

- jhi_font.h

3.4 JHI_Image Struct Reference

Structure of configuration of the Image.

```
#include <jhi_image.h>
```

Public Attributes

- JHI_Point2d pos
- SDL_Surface ∗ sur

3.4.1 Detailed Description

Structure of configuration of the Image.

3.4.2 Member Data Documentation

3.4.2.1 JHI_Point2d JHI_Image::pos

Image's position

3.4.2.2 SDL_Surface∗ JHI_Image::sur

SDL Surface of the Window.

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

- jhi_image.h

3.5 JHI_JoystickSt Struct Reference

Joystick structure.

```
#include <jhi_joystick.h>
```

Public Attributes

- JHI_JOYSTICK_EVENT joy_event
- short int joy_index
- short int axis_index
- short int axis_value [JHI_NUMBER_AXES]
- short int button
- short int button_state

3.5.1 Detailed Description

Joystick structure.

3.5.2 Member Data Documentation

3.5.2.1 short int JHI_JoystickSt::axis_index

Joystick axis_index that was captured

3.5.2.2 short int JHI_JoystickSt::axis_value[JHI_NUMBER_AXES]

Joystick axis values

3.5.2.3 short int JHI_JoystickSt::button

Joystick value of button pressed

3.5.2.4 short int JHI_JoystickSt::button_state

Joystick buttons is pressed or no

3.5.2.5 JHI_JOYSTICK_EVENT JHI_JoystickSt::joy_event

Joystick event that was captured

3.5.2.6 short int JHI_JoystickSt::joy_index

Joystick index that was captured

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

- jhi_joystick.h

3.6 JHI_KeyboardSt Struct Reference

Keyboard structure.

```
#include <jhi_keyboard.h>
```

Public Attributes

- Uint8 key_event
- JHI_Keys key

3.6.1 Detailed Description

Keyboard structure.

3.6.2 Member Data Documentation

3.6.2.1 JHI_Keys JHI_KeyboardSt::key

Keyboard key that was pressed

3.6.2.2 Uint8 JHI_KeyboardSt::key_event

Keyboard event that was captured

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

- jhi_keyboard.h

3.7 JHI_MouseSt Struct Reference

Mouse structure.

```
#include <jhi_mouse.h>
```

Public Attributes

- int x
- int y
- JHI_MouseEvents mouse_event

3.7.1 Detailed Description

Mouse structure.

3.7.2 Member Data Documentation

3.7.2.1 JHI_MouseEvents JHI_MouseSt::mouse_event

Mouse event that was captured

3.7.2.2 int JHI_MouseSt::x

Coordinate x of the mouse

3.7.2.3 int JHI_MouseSt::y

Coordinate y of the mouse

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

- jhi_mouse.h

3.8 JHI_Music Struct Reference

Struct of configuration of the Music.

```
#include <jhi_sound.h>
```

Public Attributes

- Mix_Music * mix_music

3.8.1 Detailed Description

Struct of configuration of the Music.

3.8.2 Member Data Documentation

3.8.2.1 Mix_Music∗ JHI_Music::mix_music

SDL structure of Music.

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

- jhi_sound.h

3.9 JHI_Point2d Struct Reference

Point that contain position (x,y)

```
#include <jhi_shapes.h>
```

Public Attributes

- int x
- int y

3.9.1 Detailed Description

Point that contain position (x,y)

3.9.2 Member Data Documentation

3.9.2.1 int JHI_Point2d::x

X position

3.9.2.2 int JHI_Point2d::y

Y position

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

- jhi_shapes.h

3.10 JHI_Text Struct Reference

Structure of configuration of the Text.

```
#include <jhi_text.h>
```

Public Attributes

- JHI_Point2d pos
- SDL_Surface ∗ sur

3.10.1 Detailed Description

Structure of configuration of the Text.

3.10.2 Member Data Documentation

3.10.2.1 JHI_Point2d JHI_Text::pos

Position of text

3.10.2.2 SDL_Surface∗ JHI_Text::sur

SDL Surface of the Window

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

- jhi_text.h

3.11 JHI_Window Struct Reference

Struct of configuration of the Window.

```
#include <jhi_window.h>
```

Public Attributes

- SDL_Surface ∗ screen
- char check_quit
- SDL_Event event
- JHI_Color back_color
- JHI_MouseSt mouse [NUMBER_MAX_EVENTS]
- JHI_KeyboardSt key [NUMBER_MAX_EVENTS]
- JHI_JoystickSt joy [NUMBER_MAX_EVENTS]
- int number_of_events
- int width
- int height

3.11.1 Detailed Description

Struct of configuration of the Window.

3.11.2 Member Data Documentation

3.11.2.1 JHI_Color JHI_Window::back_color

Background color of the Window

3.11.2.2 char JHI_Window::check_quit

Flag that indicate if the Window was close

3.11.2.3 SDL_Event JHI_Window::event

SDL event struct

3.11.2.4 int JHI_Window::height

Height of the Window

3.11.2.5 JHI_JoystickSt JHI_Window::joy[NUMBER_MAX_EVENTS]

Vector to joystick status for each event captured

3.11.2.6 JHI_KeyboardSt JHI_Window::key[NUMBER_MAX_EVENTS]

Vector to keyboard status for each event captured

3.11.2.7 JHI_MouseSt JHI_Window::mouse[NUMBER_MAX_EVENTS]

Vector to mouse status for each event captured

3.11.2.8 int JHI_Window::number_of_events

Number of events captured in the iteration

3.11.2.9 SDL_Surface∗ JHI_Window::screen

SDL Surface of the Window.

3.11.2.10 int JHI_Window::width

Width of the Window

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

- jhi_window.h

Chapter 4

File Documentation

4.1 jhi_colorsSt.h File Reference

This file contains structures and enumeration of the Colors.

```
#include "SDL/SDL.h"
```

Classes

- struct JHI_Color_RGB

  Structure of the RGB color.

Enumerations

- enum JHI_Color {
  RED, **GREEN**, **BLUE**, **YELLOW**,
  BLACK, **WHITE**, **ORANGE** }

  Enumeration of the possibles colors for this lib.

4.1.1 Detailed Description

This file contains structures and enumeration of the Colors.

4.2 jhi_font.h File Reference

This file contains functions structure of the Font.

```
#include "SDL/SDL_ttf.h"
#include "jhi_shapes.h"
#include <stdio.h>
```

Classes

- struct JHI_Font

  Structure of configuration of the Font.

Functions

- void jhi_load_font (JHI_Font ∗font, const char ∗font_name, int length)

  Load the music with the font_name.
- void jhi_free_font (JHI_Font ∗font)

  Free font structure.
- int jhi_get_lenght_font (JHI_Font ∗font)

  Get the Font length.

4.2.1 Detailed Description

This file contains functions structure of the Font.

4.2.2 Function Documentation

4.2.2.1 int jhi_get_lenght_font (JHI_Font ∗ *font*)

Get the Font length.

Returns

Font length

4.2.2.2 void jhi_load_font (JHI_Font ∗ *font,* const char ∗ *font_name,* int *length*)

Load the music with the font_name.

Parameters

font	Structure of the Font to load
font_name	Name of the Font file
length	Length of the Font

4.3 jhi_image.h File Reference

This file contains functions structure of the Image.

```
#include "SDL/SDL.h"
#include "SDL/SDL_image.h"
#include "SDL/SDL_rotozoom.h"
#include "jhi_colorsSt.h"
#include "jhi_shapes.h"
#include <stdio.h>
```

Classes

- struct JHI_Image

  Structure of configuration of the Image.

Functions

- void jhi_load_image (JHI_Image ∗img, const char ∗filename)

  Load the Image with the filename.
- void jhi_load_image_with_transparent_color (JHI_Image ∗img, const char ∗filename, JHI_Color col)

  Load the Image with the filename and that will be transparent in the Image.
- void jhi_free_image (JHI_Image ∗img)

  Free image structure.
- int jhi_get_image_width (JHI_Image ∗img)

  Get the Image width.
- int jhi_get_image_height (JHI_Image ∗img)

  Get the Image height.
- void jhi_resize_image (JHI_Image ∗img, int width, int height)

  resize the image
- void jhi_draw_image (JHI_Image ∗img, JHI_Point2d point)

  Draw image in the window.
- void jhi_draw_image_with_clip (JHI_Image ∗img, JHI_Point2d point_position, JHI_Point2d point_clip, int width, int height)

  Draw clip image in the window.

4.3.1 Detailed Description

This file contains functions structure of the Image.

4.3.2 Function Documentation

4.3.2.1 void jhi_draw_image (JHI_Image ∗ *img*, JHI_Point2d *point*)

Draw image in the window.

Parameters

img	Image Structure
point	new pos of the image

4.3.2.2 void jhi_draw_image_with_clip (JHI_Image ∗ *img*, JHI_Point2d *point_position*, JHI_Point2d *point_clip*, int *width*, int *height*)

Draw clip image in the window.

Parameters

img	Image Structure
point	new pos of the clip image
width	Width of this clip
height	Height of this clip

4.3.2.3 int jhi_get_image_height (JHI_Image ∗ *img*)

Get the Image height.

Parameters

img	Image Structure

Returns

Image height

4.3.2.4 int jhi_get_image_width (JHI_Image ∗ *img*)

Get the Image width.

Parameters

img	Image Structure

Returns

Image width

4.3.2.5 void jhi_load_image (JHI_Image ∗ *img,* const char ∗ *filename*)

Load the Image with the filename.

Parameters

img	Structure of Image to load
filename	Name of the music file

4.3.2.6 void jhi_load_image_with_transparent_color (JHI_Image ∗ *img,* const char ∗ *filename,* JHI_Color *col*)

Load the Image with the filename and that will be transparent in the Image.

Parameters

img	Structure of the image to load
filename	Name of the music file
cor	Transparent color in the image

4.3.2.7 void jhi_resize_image (JHI_Image ∗ *img,* int *width,* int *height*)

resize the image

Parameters

img	Image Structure
width	the new width for image
width	the new height for image

4.4 jhi_joystick.h File Reference

This file contains structures and enumeration of keyboard.

```
#include "SDL/SDL.h"
```

Classes

- struct JHI_JoystickSt

  Joystick structure.

Macros

- #define **JHI_NUMBER_MAX_JOYSTICKS** 2
- #define **JHI_NUMBER_AXES** 2
- #define **JHI_JOY_BUTTON_PRESSED** 1
- #define **JHI_JOY_BUTTON_RELEASED** 0
- #define **JHI_MAX_AXIS_VALUE** 32767
- #define **JHI_MIN_AXIS_VALUE** -32768

Enumerations

- enum JHI_JOYSTICK_EVENT { JHI_JOY_AXIS_MOTION = 7, JHI_JOY_BUTTON_DOWN = 10, JHI_JOY_BUTTO_NUP = 11, **JHI_JOY_NOT_EVENT** }

  Enumeration of events of joystick.
- enum JHI_JOY_DIR_ {
  JOY_LEFT, **JOY_RIGHT**, **JOY_UP**, **JOY_DOWN**,
  JOY_NOT_DIR, **JOY_DIR_RELEASED** }

  Enumeration of the possibles dir that will pressed in the joystick.
- enum JHI_JOY_BUTTONS {
  JOY_BUTTON_0, **JOY_BUTTON_1**, **JOY_BUTTON_2**, **JOY_BUTTON_3**,
  JOY_BUTTON_4, **JOY_BUTTON_5**, **JOY_BUTTON_6**, **JOY_BUTTON_7**,
  JOY_BUTTON_8, **JOY_BUTTON_9**, **JOY_NOT_BUTTON** }

  Enumeration of the possibles buttons that will pressed in the joystick.

Functions

- void jhi_init_joystick (JHI_JoystickSt ∗joy)

  Initialize the joystick structure.
- int jhi_open_joystick_index (short int joy_index)

  Open Joystick with index joy_index.
- void jhi_free_joystick_index (short int joy_index)

  Free the joystick structure.
- int jhi_get_num_of_joystick ()

  Get number of joysticks.
- int jhi_is_valid_joystick_index (int index)

  Check if index is valid.
- JHI_JOY_DIR_ jhi_get_joystick_dir (JHI_JoystickSt ∗joy, int axe_index)

  Get dir of joystick.

4.4.1 Detailed Description

This file contains structures and enumeration of keyboard.

4.4.2 Enumeration Type Documentation

4.4.2.1 enum JHI_JOYSTICK_EVENT

Enumeration of events of joystick.

Enumerator

JHI_JOY_AXIS_MOTION Joystick axis motion

JHI_JOY_BUTTON_DOWN Joystick button pressed

JHI_JOY_BUTTO_NUP Joystick button released

4.4.3 Function Documentation

4.4.3.1 void jhi_free_joystick_index (short int *joy_index*)

Free the joystick structure.

Parameters

joy	joystick structure

Returns

1 success, 0 otherwise

4.4.3.2 JHI_JOY_DIR_ jhi_get_joystick_dir (JHI_JoystickSt ∗ *joy,* int *axe_index*)

Get dir of joystick.

Parameters

joy	joystick structure
axe_index	index of axe

Returns

1 ok, 0 otherwise

4.4.3.3 int jhi_get_num_of_joystick ( )

Get number of joysticks.

Returns

number of joysticks

4.4.3.4 void jhi_init_joystick (JHI_JoystickSt ∗ *joy*)

Initialize the joystick structure.

Parameters

joy	joystick structure

4.4.3.5 int jhi_is_valid_joystick_index (int *index*)

Check if index is valid.

Returns

1 ok, 0 otherwise

4.4.3.6 int jhi_open_joystick_index (short int *joy_index*)

Open Joystick with index joy_index.

Parameters

joy_index	

Returns

1 success, 0 otherwise

4.5 jhi_keyboard.h File Reference

This file contains structures and enumeration of keyboard.

```
#include "SDL/SDL.h"
```

Classes

- struct JHI_KeyboardSt

  Keyboard structure.

Enumerations

- enum JHI_Keys {
  KEY_ENTER = 13, KEY_0 = 48, KEY_1 = 49, KEY_2 = 50,
  KEY_3 = 51, KEY_4 = 52, KEY_5 = 53, KEY_6 = 54,
  KEY_7 = 55, KEY_8 = 56, KEY_9 = 57, KEY_UP = 273,
  KEY_DOWN = 274, KEY_RIGHT = 275, KEY_LEFT = 276, KEY_A = 97,
  KEY_B = 98, KEY_C = 99, KEY_D = 100, KEY_E = 101,
  KEY_F = 102, KEY_G = 103, KEY_H = 104, KEY_I = 105,
  KEY_J = 106, KEY_K = 107, KEY_L = 108, KEY_M = 109,
  KEY_N = 110, KEY_O = 111, KEY_P = 112, KEY_Q = 113,
  KEY_R = 114, KEY_S = 115, KEY_T = 116, KEY_U = 117,
  KEY_V = 118, KEY_W = 119, KEY_X = 120, KEY_Y = 121,
  KEY_Z = 122, NO_KEY = 123 }

  Enumeration of the possibles keys that will pressed in the keyboard.
- enum JHI_KeyBoardEvents { KEYBOARD_UP, KEYBOARD_DOWN, KEYBOARD_NOT_EVENT }

  Enumeration of events of keyboard.

Functions

- void jhi_init_keyboard (JHI_KeyboardSt ∗key)

  init the keyboard structure
- int jhi_is_key_arrow (JHI_Keys key)

  Check if the key is arrow key.
- JHI_Keys jhi_get_opposite_key_arrow (JHI_Keys key)

  Get the opposite dir key.

4.5.1 Detailed Description

This file contains structures and enumeration of keyboard.

4.5.2 Enumeration Type Documentation

4.5.2.1 enum JHI_KeyBoardEvents

Enumeration of events of keyboard.

Enumerator

KEYBOARD_UP	Keyboard was pressed
KEYBOARD_DOWN	Keyboard was released
KEYBOARD_NOT_EVENT	Nothing

4.5.2.2 enum JHI_Keys

Enumeration of the possibles keys that will pressed in the keyboard.

Enumerator

KEY_ENTER	Key Enter
KEY_0	Key 0
KEY_1	Key 1
KEY_2	Key 2
KEY_3	Key 3
KEY_4	Key 4
KEY_5	Key 5
KEY_6	Key 6
KEY_7	Key 7
KEY_8	Key 8
KEY_9	Key 9
KEY_UP	Key UP
KEY_DOWN	Key DOWN
KEY_RIGHT	Key RIGHT
KEY_LEFT	Key LEFT
KEY_A	Key A
KEY_B	Key B
KEY_C	Key C

KEY_D Key D

KEY_E Key E

KEY_F Key F

KEY_G Key G

KEY_H Key H

KEY_I Key I

KEY_J Key J

KEY_K Key K

KEY_L Key L

KEY_M Key M

KEY_N Key N

KEY_O Key O

KEY_P Key P

KEY_Q Key Q

KEY_R Key R

KEY_S Key S

KEY_T Key T

KEY_U Key U

KEY_V Key V

KEY_W Key W

KEY_X Key X

KEY_Y Key Y

KEY_Z Key Z

NO_KEY No Key

4.5.3 Function Documentation

4.5.3.1 JHI_Keys jhi_get_opposite_key_arrow (JHI_Keys *key*)

Get the opposite dir key.

Parameters

key	key dir

Returns

opposite dir key, NO_KEY is return in error case

4.5.3.2 void jhi_init_keyboard (JHI_KeyboardSt ∗ *key*)

init the keyboard structure

Parameters

key	keyboard that will init

4.5.3.3 int jhi_is_key_arrow (JHI_Keys *key*)

Check if the key is arrow key.

Parameters

key	key to check

Returns

1 yes, 0 no

4.6 jhi_mouse.h File Reference

This file contains structure of the mouse.

```
#include "SDL/SDL.h"
```

Classes

- struct JHI_MouseSt

  Mouse structure.

Enumerations

- enum JHI_MouseEvents { ON_CLICK_RIGHT, ON_CLICK_LEFT, MOUSE_MOTION, MOUSE_NOT_EVENT }

  Enumeration of the events of the mouse.

Functions

- void jhi_init_mouse (JHI_MouseSt *mouse)

  Initalize the mouse structure.
- int jhi_check_mouse_is_in (JHI_MouseSt mouse, int xp, int yp, int w, int h)

  Check if the mouse is in the space object.

4.6.1 Detailed Description

This file contains structure of the mouse.

4.6.2 Enumeration Type Documentation

4.6.2.1 enum JHI_MouseEvents

Enumeration of the events of the mouse.

Enumerator

ON_CLICK_RIGHT Click of right button

ON_CLICK_LEFT Click of left button

MOUSE_MOTION Mouse was moved

MOUSE_NOT_EVENT Nothing Event

4.6.3 Function Documentation

4.6.3.1 int jhi_check_mouse_is_in (JHI_MouseSt *mouse,* int *xp,* int *yp,* int *w,* int *h*)

Check if the mouse is in the space object.

Parameters

xp	x of object
yp	y of object
w	width of the object
h	height of the object

Returns

1 ok, 0 otherwise

4.6.3.2 void jhi_init_mouse (JHI_MouseSt ∗ *mouse*)

Initalize the mouse structure.

Parameters

mouse	the scructure mouse to configure

4.7 jhi_rand.h File Reference

This file contains rand auxiliary MACROS.

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

Macros

- #define RAND_INTERVAL(a, b) (rand() % ((b)-(a+1))) + (a)

  get a rand value between a,b
- #define RAND_01 ((double)(rand())/(double)(RAND_MAX))

  get a double rand value between 0,1

4.7.1 Detailed Description

This file contains rand auxiliary MACROS.

4.8 jhi_shapes.h File Reference

This file contains shapes functions of libjhi-sdl.

```
#include "jhi_window.h"
#include "jhi_colorsSt.h"
#include <math.h>
```

Classes

- struct JHI_Point2d

  Point that contain position (x,y)

Functions

- void jhi_draw_point (JHI_Point2d point, JHI_Color col)

  draw point in the window
- void jhi_draw_line (JHI_Point2d s_point, JHI_Point2d d_point, JHI_Color col)

  Drawning a line in the screen.
- void jhi_draw_rect (JHI_Point2d point, int height, int base, JHI_Color col)

  Draw a rectangle or square in the screen.
- void jhi_draw_fill_rect (JHI_Point2d point, int height, int base, JHI_Color col)

  Draw a fill rectangle or square in the screen.
- void jhi_draw_circle (JHI_Point2d center_point, float radius, JHI_Color col)

  Drawning a cirle in the screen.
- void jhi_draw_fill_circle (JHI_Point2d center_point, int radius, JHI_Color col)

  Draw a fill cirle in the screen.
- void jhi_draw_polygon (JHI_Point2d ∗points, int num_points, JHI_Color col)

  Draw a Polygon in the screen.
- int jhi_is_colid (JHI_Point2d p1, int h1, int w1, JHI_Point2d p2, int h2, int w2)

  Checks whether the objects collide.
- JHI_Point2d jhi_get_central_pos (int win_w, int win_h, int obj_w, int obj_h)

  Get the central position of object in the window.
- JHI_Point2d jhi_get_point (int x, int y)

  Get the struct JHI_Point2 by (x,y)

4.8.1 Detailed Description

This file contains shapes functions of libjhi-sdl.

4.8.2 Function Documentation

4.8.2.1 void jhi_draw_circle (**JHI_Point2d** *center_point,* float *radius,* **JHI_Color** *col*)

Drawning a cirle in the screen.

Parameters

center_point	Center point of the circle center
radius	Circle's radius
col	Circle's color

4.8.2.2 void jhi_draw_fill_circle (**JHI_Point2d** *center_point,* int *radius,* **JHI_Color** *col*)

Draw a fill cirle in the screen.

Parameters

center_point	Center point of the circle center
radius	Circle's radius
col	Circle's color

4.8.2.3 void jhi_draw_fill_rect (**JHI_Point2d** *point,* int *height,* int *base,* **JHI_Color** *col*)

Draw a fill rectangle or square in the screen.

Parameters

point	Upper left point of the rect
height	Rect's height
base	Rect's base
col	Rect's Color

4.8.2.4 void jhi_draw_line (JHI_Point2d *s_point,* JHI_Point2d *d_point,* JHI_Color *col*)

Drawning a line in the screen.

Parameters

s_point	Source point of the line
d_point	Destination point of the line
col	Line Color

4.8.2.5 void jhi_draw_point (JHI_Point2d *point,* JHI_Color *col*)

draw point in the window

Parameters

point	point position
col	point color

4.8.2.6 void jhi_draw_polygon (JHI_Point2d ∗ *points,* int *num_points,* JHI_Color *col*)

Draw a Polygon in the screen.

Parameters

points	Set of the Polygon's points
num_points	Number of points of the polygon color Color of the Polygon's lines

4.8.2.7 void jhi_draw_rect (JHI_Point2d *point,* int *height,* int *base,* JHI_Color *col*)

Draw a rectangle or square in the screen.

Parameters

point	Upper left point of the rect
height	Rect's height
base	Rect's base
col	Rect's Color

4.8.2.8 JHI_Point2d jhi_get_central_pos (int *win_w,* int *win_h,* int *obj_w,* int *obj_h*)

Get the central position of object in the window.

Parameters

win_w	Width of the window
win_h	Height of the windows
obj_w	Object's width
obj_h	Object's height

Returns

Central position for this object

4.8.2.9 JHI_Point2d jhi_get_point (int *x*, int *y*)

Get the struct JHI_Point2 by (x,y)

Parameters

x	x pos
y	y pos
point	

4.8.2.10 int jhi_is_colid (JHI_Point2d *p1*, int *h1*, int *w1*, JHI_Point2d *p2*, int *h2*, int *w2*)

Checks whether the objects collide.

Parameters

p1	Point of object 1
h1	Height of object 1
w1	Weight of object 1
p2	Point of object 2
h2	Height of object 2
w2	Weight of object 2

Returns

1 colid, 0 otherwise

4.9 jhi_sound.h File Reference

This file contains functions and structures of the Music and Effect.

```
#include "SDL/SDL.h"
#include "SDL/SDL_mixer.h"
#include <stdio.h>
```

Classes

- struct JHI_Music

  Struct of configuration of the Music.

- struct JHI_Effect

  Struct of the configuration of the Effect.

Functions

- void jhi_load_music (JHI_Music ∗music, const char ∗filename)

  Load the music with the filename.
- void jhi_play_music (JHI_Music ∗music, int loop)

  Play the music in the background.
- void jhi_stop_music ()

  Stop the current music.
- void jhi_pause_music ()

  Pause the current music.
- void jhi_free_music (JHI_Music ∗music)

  Free music structure.
- void jhi_load_effect (JHI_Effect ∗effect, const char ∗filename)

  Load the effect with the filename.
- void jhi_play_effect (JHI_Effect ∗effect, int delay)

  Play the effect.
- void jhi_free_effect (JHI_Effect ∗effect)

  Free effect structure.

4.9.1 Detailed Description

This file contains functions and structures of the Music and Effect.

4.9.2 Function Documentation

4.9.2.1 void jhi_load_effect (**JHI_Effect** ∗ *effect,* const char ∗ *filename*)

Load the effect with the filename.

Parameters

effect	Structure of effect to load
filename	Name of the effect file

4.9.2.2 void jhi_load_music (**JHI_Music** ∗ *music,* const char ∗ *filename*)

Load the music with the filename.

Parameters

music	Structure of music to load
filename	Name of music file

4.9.2.3 void jhi_play_effect (**JHI_Effect** ∗ *effect,* int *delay*)

Play the effect.

Parameters

effect	Structure of effect to play

delay	Delay to play the effect in Seconds

4.9.2.4 void jhi_play_music (**JHI_Music** ∗ *music,* int *loop*)

Play the music in the background.

Parameters

music	Structure of music to load
loop	Number of times that music will played. -1, it's infinite

4.10 jhi_text.h File Reference

This file contains functions and structures of the Text.

```
#include "jhi_font.h"
#include "jhi_colorsSt.h"
#include "jhi_shapes.h"
#include "SDL/SDL.h"
```

Classes

- struct JHI_Text

  Structure of configuration of the Text.

Functions

- void jhi_init_text (JHI_Text ∗text)

  Init the Text structure.
- void jhi_set_text (JHI_Font ∗font, JHI_Text ∗text, JHI_Color cor, const char ∗txt)

  Set the Text with the configurations.
- void jhi_free_text (JHI_Text ∗text)

  Free memory of the Text strucure.
- int jhi_get_text_width (JHI_Text ∗text)

  Get the Text width.
- int jhi_get_text_height (JHI_Text ∗text)

  Get the Text height.
- void jhi_draw_text (JHI_Text ∗text, JHI_Point2d point)

  Draw the text in the window.

4.10.1 Detailed Description

This file contains functions and structures of the Text.

4.10.2 Function Documentation

4.10.2.1 void jhi_draw_text (**JHI_Text** ∗ *text,* **JHI_Point2d** *point*)

Draw the text in the window.

Parameters

text	Text Structure
point	New text position

4.10.2.2 int jhi_get_text_height (JHI_Text ∗ *text*)

Get the Text height.

Parameters

text	Text Structure

Returns

Text height

4.10.2.3 int jhi_get_text_width (JHI_Text ∗ *text*)

Get the Text width.

Parameters

text	Text Structure

Returns

Text width

4.10.2.4 void jhi_init_text (JHI_Text ∗ *text*)

Init the Text structure.

Parameters

text	Text structure to be initialized

4.10.2.5 void jhi_set_text (JHI_Font ∗ *font,* JHI_Text ∗ *text,* JHI_Color *cor,* const char ∗ *txt*)

Set the Text with the configurations.

Parameters

font	Font of the Text
text	Text to be configured
color	Color of the Text
txt	String to the Text

4.11 jhi_timer.h File Reference

This file contains functions structure of the Timer to control number of frames per second (fps).

```
#include "SDL/SDL.h"
```

Functions

- void jhi_set_fps_timer (int fps)

  Configure the fps timer.
- void jhi_timer_start ()

  Start the count to fps.
- void jhi_wait_time ()

  Wait the time necessary to control fps.
- void jhi_delay (int seconds)

  Wait the time in second.
- void **jhi_delay_mili_seconds** (int ms)

4.11.1 Detailed Description

This file contains functions structure of the Timer to control number of frames per second (fps).

4.11.2 Function Documentation

4.11.2.1 void jhi_delay (int *seconds*)

Wait the time in second.

Parameters

second	Number of Seconds to wait

4.11.2.2 void jhi_set_fps_timer (int *fps*)

Configure the fps timer.

Parameters

fps	Frames per second to animation

4.12 jhi_window.h File Reference

This file contains functions related to Window.

```
#include "SDL/SDL_mixer.h"
#include "jhi_colorsSt.h"
#include "jhi_keyboard.h"
#include "jhi_mouse.h"
#include "jhi_joystick.h"
#include <stdio.h>
```

Classes

- struct JHI_Window

  Struct of configuration of the Window.

Macros

- #define **JHI_CLOSE** 1
- #define **JHI_NOT_CLOSE** 0
- #define **NUMBER_MAX_EVENTS** 100

Functions

- int jhi_out_window (int x, int y)

  Check if (x,y) are in valid position inside of the Window.
- void jhi_initialize_window (int width, int height, int bitperpixel, JHI_Color back_color)

  Initialize configurations of the Window.
- void jhi_print_pixel (int x, int y, JHI_Color col)

  Draw a pixel in the screen.
- void jhi_draw_object (SDL_Surface ∗object, int x, int y)

  Draw object in the window.
- void jhi_draw_object_with_clip (SDL_Surface ∗object, int x, int y, int x_clip, int y_clip, int width, int height)

  Draw a cut/clip of an image in the screen.
- void jhi_choice_window_name (const char ∗win_name)

  Set the name of the Window.
- void jhi_init_mouse_keyboard_joystick_events ()

  Initialize structs of the mouse, keyboard and joystick.
- int jhi_get_number_of_events ()

  Return the number of events captured in the iteration.
- void jhi_update ()

  Capture the events of iteration and updates the Window.
- void jhi_set_background_color (JHI_Color back_color)

  Set the background color with the color especified.
- void jhi_clean ()

  Clean the Window with the background color.
- char jhi_get_close_window ()

  Get if close the Window.
- JHI_MouseSt jhi_get_mouse_status (int i)

  Get status of the mouse input.
- JHI_KeyboardSt jhi_get_keyboard_status (int i)

  Get status of the keyboard input.
- JHI_JoystickSt jhi_get_joystick_status (int i)

  Get status of the joystick input.
- void jhi_quit_and_free ()

  Free structure of the Window.
- int jhi_get_height_window ()

  Get the height of the window.
- int jhi_get_width_window ()

  Get the height of the window.

4.12.1 Detailed Description

This file contains functions related to Window.

4.12.2 Function Documentation

4.12.2.1 void jhi_choice_window_name (const char $*$ *win_name*)

Set the name of the Window.

Parameters

win_name	Name of the Window

4.12.2.2 void jhi_draw_object (SDL_Surface ∗ *object,* int *x,* int *y*)

Draw object in the window.

Parameters

object	Object to draw

4.12.2.3 void jhi_draw_object_with_clip (SDL_Surface ∗ *object,* int *x,* int *y,* int *x_clip,* int *y_clip,* int *width,* int *height*)

Draw a cut/clip of an image in the screen.

Parameters

object	Object to draw
x	Coordinate x to clip object in the screen
y	Coordinate y to to clip object in the screen
x_clip	x position initial of clip
y_clip	y position initial of clip
width	Width of this clip
height	Height of this clip

4.12.2.4 char jhi_get_close_window ( )

Get if close the Window.

Returns

NOT_CLOSE, case this Window wasn't closed, CLOSE otherwise

4.12.2.5 int jhi_get_height_window ( )

Get the height of the window.

Returns

height of the window

4.12.2.6 JHI_JoystickSt jhi_get_joystick_status (int *i*)

Get status of the joystick input.

Parameters

i	Number of event

Returns

keyboard status

4.12.2.7 JHI_KeyboardSt jhi_get_keyboard_status (int *i*)

Get status of the keyboard input.

Parameters

i	Number of event

Returns

keyboard status

4.12.2.8 JHI_MouseSt jhi_get_mouse_status (int *i*)

Get status of the mouse input.

Parameters

i	Number of event

Returns

Mouse status

4.12.2.9 int jhi_get_number_of_events ( )

Return the number of events captured in the iteration.

Returns

Number of events captured in the iteration

4.12.2.10 int jhi_get_width_window ( )

Get the height of the window.

Returns

height of the window

4.12.2.11 void jhi_initialize_window (int *width,* int *height,* int *bitperpixel,* JHI_Color *back_color*)

Initialize configurations of the Window.

Parameters

width	Width of the Window
height	Height of the Window
bitperpixel	Number of the bits by pixel
back_color	Background color of the Window

4.12.2.12 int jhi_out_window (int *x,* int *y*)

Check if (x,y) are in valid position inside of the Window.

Parameters

x	Coordinate x
y	Coordinate y

Returns

1 if is out, 0 otherwise

4.12.2.13 void jhi_print_pixel (int *x,* int *y,* JHI_Color *col*)

Draw a pixel in the screen.

Parameters

x	Coordinate x of the pixel
y	Coordinate y of the pixel
color	Color of the pixel

4.12.2.14 void jhi_set_background_color (JHI_Color *back_color*)

Set the background color with the color especified.

Parameters

back_color	Color to fill the background

Index