

# **NXP Graphics software library (SWIM) package code documentation version 1.0**

# Content

## 1 Symbol Reference 1

### 1.1 Functions 1

- 1.1.1 lpc\_colors\_set\_palette 1
- 1.1.2 swim\_clear\_screen 2
- 1.1.3 swim\_get\_font\_height 2
- 1.1.4 swim\_get\_horizontal\_size 3
- 1.1.5 swim\_get\_vertical\_size 3
- 1.1.6 swim\_get\_xy 4
- 1.1.7 swim\_put\_box 4
- 1.1.8 swim\_put\_char 5
- 1.1.9 swim\_put\_diamond 5
- 1.1.10 swim\_put\_image 6
- 1.1.11 swim\_put\_invert\_image 6
- 1.1.12 swim\_put\_left\_image 7
- 1.1.13 swim\_put\_line 7
- 1.1.14 swim\_put\_ltext 8
- 1.1.15 swim\_put\_newline 8
- 1.1.16 swim\_put\_pixel 9
- 1.1.17 swim\_put\_right\_image 9
- 1.1.18 swim\_put\_scale\_image 10
- 1.1.19 swim\_put\_scale\_invert\_image 10
- 1.1.20 swim\_put\_scale\_left\_image 11
- 1.1.21 swim\_put\_scale\_right\_image 11
- 1.1.22 swim\_put\_text 12
- 1.1.23 swim\_put\_text\_xy 12
- 1.1.24 swim\_put\_win\_image 13
- 1.1.25 swim\_set\_bkg\_color 13
- 1.1.26 swim\_set\_fill\_color 14
- 1.1.27 swim\_set\_font 14
- 1.1.28 swim\_set\_font\_transparency 15
- 1.1.29 swim\_set\_pen\_color 15
- 1.1.30 swim\_set\_title 16
- 1.1.31 swim\_set\_xy 16
- 1.1.32 swim\_window\_close 17
- 1.1.33 swim\_window\_open 17
- 1.1.34 swim\_window\_open\_noclear 18

### 1.2 Types 19

- 1.2.1 BOOL\_16 19
- 1.2.2 BOOL\_32 19
- 1.2.3 BOOL\_8 19

1.2.4 CHAR	20
1.2.5 COLOR_T	20
1.2.6 FONT_T	20
1.2.7 INT_16	20
1.2.8 INT_32	21
1.2.9 INT_64	21
1.2.10 INT_8	21
1.2.11 PFI	21
1.2.12 PFV	21
1.2.13 STATUS	22
1.2.14 SWIM_ROTATION_T	22
1.2.15 SWIM_WINDOW_T	22
1.2.16 UNS_16	23
1.2.17 UNS_32	23
1.2.18 UNS_64	24
1.2.19 UNS_8	24

### **1.3 Variables 24**

1.3.1 font_helvr10	24
1.3.2 font_rom8x16	24
1.3.3 font_rom8x8	25
1.3.4 font_winfreesys14x16	25
1.3.5 font_x5x7	25
1.3.6 font_x6x13	25
1.3.7 helvr10_bits	26
1.3.8 helvR10_width	27
1.3.9 rom8x16_bits	28
1.3.10 rom8x16_width	33
1.3.11 rom8x8_bits	33
1.3.12 rom8x8_width	36
1.3.13 winfreesystem14x16_bits	37
1.3.14 winfreesystem14x16_width	41
1.3.15 x5x7_bits	42
1.3.16 x5x7_width	43
1.3.17 x6x13_bits	43
1.3.18 x6x13_width	46

### **1.4 Macros 46**

1.4.1 _BIT	46
1.4.2 _BITMASK	46
1.4.3 _ERROR	47
1.4.4 _NO_ERROR	47
1.4.5 _SBF	47
1.4.6 BLACK	47

1.4.7 BLUE	47
1.4.8 BLUE_COLORS	48
1.4.9 BLUEMASK	48
1.4.10 BLUESHIFT	48
1.4.11 COLORS_DEF	48
1.4.12 CYAN	49
1.4.13 DARKGRAY	49
1.4.14 EXTERN	49
1.4.15 FALSE	49
1.4.16 GREEN	49
1.4.17 GREEN_COLORS	50
1.4.18 GREENMASK	50
1.4.19 GREENSHIFT	50
1.4.20 LIGHTBLUE	50
1.4.21 LIGHTCYAN	51
1.4.22 LIGHTGRAY	51
1.4.23 LIGHTGREEN	51
1.4.24 LIGHTMAGENTA	51
1.4.25 LIGHTRED	51
1.4.26 LIGHTYELLOW	52
1.4.27 LPC_BAD_CLK	52
1.4.28 LPC_BAD_HANDLE	52
1.4.29 LPC_BAD_PARAMS	52
1.4.30 LPC_CANT_START	53
1.4.31 LPC_CANT_STOP	53
1.4.32 LPC_COLOR_TYPES_H	53
1.4.33 LPC_DEV_UNKNOWN	53
1.4.34 LPC_FONTS_H	53
1.4.35 LPC_HEVR10_FONT_H	54
1.4.36 LPC_IN_USE	54
1.4.37 LPC_NOT_OPEN	54
1.4.38 LPC_NOT_SUPPORTED	54
1.4.39 LPC_PIN_CONFLICT	55
1.4.40 LPC_ROM8X16_FONT_H	55
1.4.41 LPC_ROM8X8_FONT_H	55
1.4.42 LPC_SWIM_FONT_H	55
1.4.43 LPC_SWIM_H	55
1.4.44 LPC_SWIM_IMAGE_H	56
1.4.45 LPC_TYPES_H	56
1.4.46 LPC_WINFREESYS_14X16_FONT_H	56
1.4.47 LPC_X5X7_FONT_H	56
1.4.48 LPC_X6X13_FONT_H	57

1.4.49 MAGENTA	57
1.4.50 NELEMENTS	57
1.4.51 NULL	57
1.4.52 NUM_COLORS	57
1.4.53 RED	58
1.4.54 RED_COLORS	58
1.4.55 REDMASK	58
1.4.56 REDSHIFT	58
1.4.57 SMA_BAD_CLK	59
1.4.58 SMA_BAD_HANDLE	59
1.4.59 SMA_BAD_PARAMS	59
1.4.60 SMA_CANT_START	59
1.4.61 SMA_CANT_STOP	59
1.4.62 SMA_DEV_UNKNOWN	60
1.4.63 SMA_IN_USE	60
1.4.64 SMA_NOT_OPEN	60
1.4.65 SMA_NOT_SUPPORTED	60
1.4.66 SMA_PIN_CONFLICT	61
1.4.67 STATIC	61
1.4.68 SUCCESS	61
1.4.69 TRUE	61
1.4.70 WHITE	61
1.4.71 YELLOW	62

## **1.5 Files 62**

1.5.1 lpc_colors.c	62
1.5.2 lpc_colors.h	63
1.5.3 lpc_fonts.c	64
1.5.4 lpc_fonts.h	64
1.5.5 lpc_helvr10.c	65
1.5.6 lpc_helvr10.h	65
1.5.7 lpc_rom8x16.c	66
1.5.8 lpc_rom8x16.h	66
1.5.9 lpc_rom8x8.c	67
1.5.10 lpc_rom8x8.h	67
1.5.11 lpc_swim.c	67
1.5.12 lpc_swim.h	68
1.5.13 lpc_swim_font.c	70
1.5.14 lpc_swim_font.h	70
1.5.15 lpc_swim_image.c	72
1.5.16 lpc_swim_image.h	72
1.5.17 lpc_types.h	73
1.5.18 lpc_winfreesystem14x16.c	75

1.5.19 lpc\_winfreesystem14x16.h 75

1.5.20 lpc\_x5x7.c 76

1.5.21 lpc\_x5x7.h 76

1.5.22 lpc\_x6x13.c 77

1.5.23 lpc\_x6x13.h 77

## **2 Index 79**

# NXP Graphics software library (SWIM) package code documentation version 1.0

## 1 Symbol Reference

---

### 1.1 Functions

---

#### 1.1.1 lpc\_colors\_set\_palette

```
void lpc_colors_set_palette(UNS_16 * palette_table);
```

##### File

lpc\_colors.h (see page 63)

##### Parameters

Parameters	Description
UNS_16 * palette_table	Pointer of where to put the 256 8-bit to 16-bit palette conversion entries.
Outputs	None

##### Returns

Nothing

Notes: If compiled in 16-bit color mode, this will be a NULL (see page 57) function. Select the appropriate define in this function for 555 or 565 color mode displays when using an 256 color frame buffer.

##### Description

Generate a palette table (only in 8-bit mode). If compiled in 16-bit color mode, this will be a NULL (see page 57) function.

Function: lpc\_colors\_set\_palette

Purpose: Generate a palette table (only in 8-bit mode).

Processing: Depending on the target LCD color mapping (either 555 or 565), a palette table will be generated to convert colors stored in 233 format to either 555 or 565 format through a lookup table.

---

## 1.1.2 swim\_clear\_screen

```
void swim_clear_screen(SWIM_WINDOW_T * win, COLOR_T colr);
```

### File

lpc\_swim.h (see page 68)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
COLOR_T colr	Color to place in the window
Outputs	None

### Returns

Nothing

Notes: None

### Description

Fills the draw area of the display with the selected color

Function: swim\_clear\_screen

Purpose: Fills the draw area of the display with the selected color

Processing: Loop through all virtual window (draw area) locations and updates them with the passed color value.

---

## 1.1.3 swim\_get\_font\_height

```
INT_16 swim_get_font_height(SWIM_WINDOW_T * win);
```

### File

lpc\_swim\_font.h (see page 70)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
Outputs	None

### Returns

The height of the active font in pixels.

Notes: None

### Description

Returns the active font's height in pixels

Function: swim\_get\_font\_height

Purpose: Returns the active font's height in pixels

Processing: See function.



---

## 1.1.4 swim\_get\_horizontal\_size

```
INT_32 swim_get_horizontal_size(SWIM_WINDOW_T * win);
```

### File

lpc\_swim.h (see page 68)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
Outputs	None

### Returns

The virtual window horizontal size

Notes: None

### Description

Get the virtual window horizontal size

Function: swim\_get\_horizontal\_size

Purpose: Get the virtual window horizontal size

Processing: For the passed window ID, return the x size of the window.

---

## 1.1.5 swim\_get\_vertical\_size

```
INT_32 swim_get_vertical_size(SWIM_WINDOW_T * win);
```

### File

lpc\_swim.h (see page 68)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
Outputs	None

### Returns

The virtual window horizontal size

Notes: None

### Description

Get the virtual window vertical size

Function: swim\_get\_vertical\_size

Purpose: Get the virtual window vertical size

Processing: For the passed window ID, return the x size of the window.

---

## 1.1.6 swim\_get\_xy

```
void swim_get_xy(SWIM_WINDOW_T * win, INT_32 * x, INT_32 * y);
```

### File

lpc\_swim\_font.h (see page 70)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
INT_32 * x	Address of where to return virtual X value
INT_32 * y	Address of where to return virtual X value
Outputs	None

### Returns

Nothing

Notes: X, Y coords are in virtual pixels!

### Description

Returns the X, Y pixel coordinates for the next text operation

Function: swim\_get\_xy

Purpose: Returns the X, Y pixel coordinates for the next text operation

Processing: The logical X and Y positions are computed by subtracting the physical text position values by the physical minimum window limits.

---

## 1.1.7 swim\_put\_box

```
void swim_put_box(SWIM_WINDOW_T * win, INT_32 x1, INT_32 y1, INT_32 x2, INT_32 y2);
```

### File

lpc\_swim.h (see page 68)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
INT_32 x1	Virtual left position of box
INT_32 y1	Virtual upper position of box
INT_32 x2	Virtual right position of box
INT_32 y2	Virtual lower position of box
Outputs	None

### Returns

Nothing

Notes: None

### Description

Place a box with corners (X1, Y1) and (X2, Y2). Use pen color for edges and fill color for center

Function: swim\_put\_box

Purpose: Place a box with corners (X1, Y1) and (X2, Y2)

Processing: See function.

---

## 1.1.8 swim\_put\_char

```
void swim_put_char(SWIM_WINDOW_T * win, const CHAR textchar);
```

### File

lpc\_swim\_font.h (see page 70)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const CHAR textchar	Text string to output in window
Outputs	None

### Returns

Nothing

Notes: None

### Description

Puts a single character to the window

Function: swim\_put\_char

Purpose: Puts a character in the window.

Processing: See function.

---

## 1.1.9 swim\_put\_diamond

```
void swim_put_diamond(SWIM_WINDOW_T * win, INT_32 x, INT_32 y, INT_32 rx, INT_32 ry);
```

### File

lpc\_swim.h (see page 68)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
INT_32 x	Virtual X position of the diamond
INT_32 y	Virtual Y position of the diamond
INT_32 rx	Radius for horizontal
INT_32 ry	Radius for vertical
Outputs	None

### Returns

Nothing

Notes: This function supports clipping.

### Description

Draw a diamond in the virtual window

Function: swim\_put\_diamond

Purpose: Purpose: Draw a diamond in the virtual window

Processing: See function.

---

## 1.1.10 swim\_put\_image

```
void swim_put_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize, INT_32 ysize);
```

### File

lpc\_swim\_image.h (see page 72)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	Pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
Outputs	None

### Returns

Nothing

Notes: Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

### Description

Puts a raw image into a window

Function: swim\_put\_image

Purpose: Puts an raw image in a window unscaled, clips off edges

Processing: See function.

---

## 1.1.11 swim\_put\_invert\_image

```
void swim_put_invert_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize, INT_32 ysize);
```

### File

lpc\_swim\_image.h (see page 72)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	Pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
Outputs	None

### Returns

Nothing

Notes: Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

### Description

Puts a raw image into a window inverted

Function: swim\_put\_invert\_image

Purpose: Puts an raw image in a window unscaled, inverted, with clipped edges.

Processing: See function.

---

## 1.1.12 swim\_put\_left\_image

```
void swim_put_left_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize, INT_32 ysize);
```

### File

lpc\_swim\_image.h (see page 72)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	Pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
Outputs	None

### Returns

Nothing

Notes: Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

### Description

Puts a raw image into a window rotated left

Function: swim\_put\_left\_image

Purpose: Puts an raw image in a window unscaled, rotated left, with clipped edges.

Processing: See function.

---

## 1.1.13 swim\_put\_line

```
void swim_put_line(SWIM_WINDOW_T * win, INT_32 x1, INT_32 y1, INT_32 x2, INT_32 y2);
```

### File

lpc\_swim.h (see page 68)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
INT_32 x1	Virtual X position of X line start
INT_32 y1	Virtual Y position of Y line start
INT_32 x2	Virtual X position of X line end
INT_32 y2	Virtual Y position of Y line end
Outputs	None

### Returns

Nothing

Notes: This function supports clipping.

### Description

Draw a line in the virtual window

Function: swim\_put\_line

Purpose: Draw a line in the virtual window with clipping.

Processing: See function.

---

## 1.1.14 swim\_put\_ltext

```
void swim_put_ltext(SWIM_WINDOW_T * win, const CHAR * text);
```

### File

lpc\_swim\_font.h (see page 70)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const CHAR * text	Text string to output in window
Outputs	None

### Returns

Nothing

Notes: None

### Description

Puts a null-terminated string of text in a window, but will move an entire word to the next line if it will not fit on the present line

Function: swim\_put\_ltext

Purpose: Puts a string of text in a window, but will adjust the position of a word if the word length exceeds the edge of the display.

Processing: While the string has data in it, check for the newline character. If it exists, output a newline. If the string data is inside the font character table, output the first word in the string (with support for generating a newline if the word will exceed the window edge). Continue until all words/characters are output.

---

## 1.1.15 swim\_put\_newline

```
void swim_put_newline(SWIM_WINDOW_T * win);
```

### File

lpc\_swim\_font.h (see page 70)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
Outputs	None

### Returns

Nothing

Notes: None

### Description

Puts a newline in the window

Function: swim\_put\_newline

**Purpose:** Performs a newline in a window

**Processing:** Set the text pointer for the next text character operation to the beginning of the following line. If the following line exceeds the window size, perform a line scroll.

## 1.1.16 swim\_put\_pixel

```
void swim_put_pixel(SWIM_WINDOW_T * win, INT_32 x1, INT_32 y1);
```

### File

lpc\_swim.h (see page 68)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
INT_32 x1	Virtual X position of pixel
INT_32 y1	Virtual Y position of pixel
Outputs	None

### Returns

Nothing

**Notes:** The pixel will not be displayed if the pixel exceeds the window virtual size. Pixel positions below 0 should not be used with this function.

### Description

Puts a pixel at (X, Y) in the pen color

Function: swim\_put\_pixel

**Purpose:** Puts a pixel at the virtual X, Y coordinate in the window

**Processing:** Convert the virtual pixel position to a physical position. If the pixel is inside the window draw area, update the pixel on the display.

## 1.1.17 swim\_put\_right\_image

```
void swim_put_right_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize, INT_32 ysize);
```

### File

lpc\_swim\_image.h (see page 72)

### Parameters

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	Pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
Outputs	None

### Returns

Nothing

**Notes:** Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

**Description**

Puts a raw image into a window rotated right

Function: swim\_put\_right\_image

Purpose: Puts an raw image in a window unscaled, rotated right, with clipped edges.

Processing: See function.

---

## 1.1.18 swim\_put\_scale\_image

```
void swim_put_scale_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize, INT_32 ysize);
```

**File**

lpc\_swim\_image.h (see page 72)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
Outputs	None

**Returns**

Nothing

Notes: Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

**Description**

Puts and scales a raw image into a window

Function: swim\_put\_scale\_image

Purpose: Puts an raw image in a window scaled.

Processing: See function.

---

## 1.1.19 swim\_put\_scale\_invert\_image

```
void swim_put_scale_invert_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize, INT_32 ysize);
```

**File**

lpc\_swim\_image.h (see page 72)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
Outputs	None



**Returns**

Nothing

Notes: Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

**Description**

Puts and scales a raw image into a window inverted

Function: swim\_put\_scale\_invert\_image

Purpose: Puts an raw image in a window scaled and inverted.

Processing: See function.

## 1.1.20 swim\_put\_scale\_left\_image

```
void swim_put_scale_left_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize,
INT_32 ysize);
```

**File**

lpc\_swim\_image.h (see page 72)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
Outputs	None

**Returns**

Nothing

Notes: Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

**Description**

Puts and scales a raw image into a window rotated left

Function: swim\_put\_scale\_left\_image

Purpose: Puts an raw image in a window scaled and rotated left.

Processing: See function.

## 1.1.21 swim\_put\_scale\_right\_image

```
void swim_put_scale_right_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize,
INT_32 ysize);
```

**File**

lpc\_swim\_image.h (see page 72)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels

INT_32 ysize	Size of the image in vertical pixels
Outputs	None

**Returns**

Nothing

Notes: Pixels should be organized in the image from left to right, top to bottom. (BMP images are not stored like this.)

**Description**

Puts and scales a raw image into a window rotated right

Function: swim\_put\_scale\_right\_image

Purpose: Puts an raw image in a window scaled and rotated right.

Processing: See function.

## 1.1.22 swim\_put\_text

```
void swim_put_text(SWIM_WINDOW_T * win, const CHAR * text);
```

**File**

lpc\_swim\_font.h (see page 70)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const CHAR * text	Text string to output in window
Outputs	None

**Returns**

Nothing

Notes: None

**Description**

Puts a null-terminated string of text in a window

Function: swim\_put\_text

Purpose: Puts a string of text in a window

Processing: Each character will be routed to the swim\_put\_char (see page 5) function until a string terminator is reached. For newline characters, a newline will occur instead of a character output.

## 1.1.23 swim\_put\_text\_xy

```
void swim_put_text_xy(SWIM_WINDOW_T * win, const CHAR * text, INT_32 x, INT_32 y);
```

**File**

lpc\_swim\_font.h (see page 70)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const CHAR * text	Text string to output in window
INT_32 x	Virtual X position of start of text

INT_32 y	Virtual Y position of start of text
Outputs	None

**Returns**

Nothing

Notes: X, Y coords are in virtual pixels!

**Description**

Put a text message at an X, Y pixel coordinate in the window

Function: swim\_put\_text\_xy

Purpose: Put text at x, y (char) position on screen

Processing: Set the virtual (upper left) text position in the window and render the text string at this position.

## 1.1.24 swim\_put\_win\_image

```
void swim_put_win_image(SWIM_WINDOW_T * win, const COLOR_T * image, INT_32 xsize, INT_32 ysize, INT_32 scale, SWIM_ROTATION_T rtype);
```

**File**

lpc\_swim\_image.h (see page 72)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const COLOR_T * image	pointer to image data, must be in display color format
INT_32 xsize	Size of the image in horizontal pixels
INT_32 ysize	Size of the image in vertical pixels
INT_32 scale	If set, the picture will be scaled to the window size If not set, the picture will be clipped
SWIM_ROTATION_T rtype	Rotation type flag, either Norotation, Left, Right, or Invert
Outputs	None

**Returns**

Nothing

Notes: None

**Description**

One API for all the functions

Function: swim\_put\_win\_image

Purpose: This function simply provides a single API for all the image functions.

Processing: See function.

## 1.1.25 swim\_set\_bkg\_color

```
void swim_set_bkg_color(SWIM_WINDOW_T * win, COLOR_T bkg_color);
```

**File**

lpc\_swim.h (see page 68)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
Outputs	None
tbkg_color	New background color

**Returns**

Nothing

Notes: None

**Description**

Set background color

Function: swim\_set\_bkg\_color

Purpose: Sets the color used for backgrounds

Processing: For the passed window ID, update to the passed background color.

---

## 1.1.26 swim\_set\_fill\_color

```
void swim_set_fill_color(SWIM_WINDOW_T * win, COLOR_T fill_color);
```

**File**

lpc\_swim.h (see page 68)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
COLOR_T fill_color	New fill color
Outputs	None

**Returns**

Nothing

Notes: None

**Description**

Set fill color (used for boxes and circles)

Function: swim\_set\_fill\_color

Purpose: Sets the fill color

Processing: For the passed window ID, update to the passed fill color.

---

## 1.1.27 swim\_set\_font

```
void swim_set_font(SWIM_WINDOW_T * win, FONT_T * font);
```

**File**

lpc\_swim\_font.h (see page 70)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
FONT_T * font	Pointer to the selected font data structure
Outputs	None

**Returns**

Nothing

Notes: None

**Description**

Select the active font

Function: swim\_set\_font

Purpose: Sets the active font

Processing: Switch to the selected font by setting the font structure pointer in the windows structure based on the passed enumeration. If the next character output in the new font will exceed the window limit, perform a window text scroll.

---

## 1.1.28 swim\_set\_font\_transparency

```
void swim_set_font_transparency(SWIM_WINDOW_T * win, INT_32 trans);
```

**File**

lpc\_swim\_font.h (see page 70)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
INT_32 trans	When not 0, the font backgrounds will not be drawn
Outputs	None

**Returns**

Nothing

Notes: None

**Description**

Enables and disables font backgrounds

Function: swim\_set\_font\_transparency

Purpose: Enables and disables font backgrounds. When set, the font background will not be drawn in the background color (useful for painting text over pictures).

Processing: See function.

---

## 1.1.29 swim\_set\_pen\_color

```
void swim_set_pen_color(SWIM_WINDOW_T * win, COLOR_T pen_color);
```

**File**

lpc\_swim.h (see page 68)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
COLOR_T pen_color	New pen color
Outputs	None

**Returns**

Nothing

Notes: None

**Description**

Set the pen color

Function: swim\_set\_pen\_color

Purpose: Sets the pen color

Processing: For the passed window ID, update to the passed pen color.

---

## 1.1.30 swim\_set\_title

```
void swim_set_title(SWIM_WINDOW_T * win, const CHAR * title, COLOR_T ttlbkcolor);
```

**File**

lpc\_swim\_font.h (see page 70)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
const CHAR * title	title string to use for window
COLOR_T ttlbkcolor	Background color in title area
Outputs	None

**Returns**

Nothing

Notes: Do not call this function more than once for a window or problems may occur.

**Description**

Create a title bar

Function: swim\_set\_title

Purpose: Creates a title bar in the window and adjusts the client area to be outside the title bar area.

Processing: See function.

---

## 1.1.31 swim\_set\_xy

```
void swim_set_xy(SWIM_WINDOW_T * win, INT_32 x, INT_32 y);
```

**File**

lpc\_swim\_font.h (see page 70)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
INT_32 x	Virtual X position of start of text
INT_32 y	Virtual Y position of start of text
Outputs	None

**Returns**

Nothing

Notes: X, Y coords are in virtual pixels!

**Description**

Sets the X, Y pixel coordinates for the next text operation

Function: swim\_set\_xy

Purpose: Sets the X, Y pixel coordinates for the next text operation

Processing: Update the X, Y text position pointers, limiting the position to the window dimensions.

---

## 1.1.32 swim\_window\_close

```
void swim_window_close(SWIM_WINDOW_T * win);
```

**File**

lpc\_swim.h (see page 68)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Window identifier
Outputs	None

**Returns**

Nothing

Notes: This is a defunct function and is not needed.

**Description**

Destroy a window

Function: swim\_window\_close

Purpose: Reallocates a window for use

Processing: For the passed window ID, clear the window used flag.

---

## 1.1.33 swim\_window\_open

```
BOOL_32 swim_window_open(SWIM_WINDOW_T * win, INT_32 xsize, INT_32 ysize, COLOR_T * fbaddr,  
INT_32 xwin_min, INT_32 ywin_min, INT_32 xwin_max, INT_32 ywin_max, INT_32 border_width,  
COLOR_T pcolor, COLOR_T bkcolor, COLOR_T fcolor);
```

**File**

lpc\_swim.h (see page 68)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Preallocated windows structure to fill
INT_32 xsize	Physical horizontal dimension of the display
INT_32 ysize	Physical vertical dimension of the display
COLOR_T * fbaddr	Address of the display's frame buffer
INT_32 xwin_min	Physical window left coordinate
INT_32 ywin_min	Physical window top coordinate
INT_32 xwin_max	Physical window right coordinate
INT_32 ywin_max	Physical window bottom coordinate
INT_32 border_width	Width of the window border in pixels
COLOR_T pcolor	Pen color
COLOR_T bkcolor	Background color
COLOR_T fcolor	Fill color
Outputs	None

**Returns**

TRUE if the window was initialized correctly, otherwise FALSE

Notes: This function must be called prior to any other window function

**Description**

Initialize a window

Function: swim\_window\_open

Purpose: Initializes a window and the default values for the window

Processing: See function.

## 1.1.34 swim\_window\_open\_noclear

```
BOOL_32 swim_window_open_noclear(SWIM_WINDOW_T * win, INT_32 xsize, INT_32 ysize, COLOR_T *
fbaddr, INT_32 xwin_min, INT_32 ywin_min, INT_32 xwin_max, INT_32 ywin_max, INT_32
border_width, COLOR_T pcolor, COLOR_T bkcolor, COLOR_T fcolor);
```

**File**

lpc\_swim.h (see page 68)

**Parameters**

Parameters	Description
SWIM_WINDOW_T * win	Preallocated windows structure to fill
INT_32 xsize	Physical horizontal dimension of the display
INT_32 ysize	Physical vertical dimension of the display
COLOR_T * fbaddr	Address of the display's frame buffer
INT_32 xwin_min	Physical window left coordinate
INT_32 ywin_min	Physical window top coordinate
INT_32 xwin_max	Physical window right coordinate
INT_32 ywin_max	Physical window bottom coordinate
INT_32 border_width	Width of the window border in pixels
COLOR_T pcolor	Pen color
COLOR_T bkcolor	Background color
COLOR_T fcolor	Fill color
Outputs	None

**Returns**

TRUE if the window was initialized correctly, otherwise FALSE



Notes: This function must be called prior to any other window function

**Description**

Initialize a window without clearing it

Function: `swim_window_open_noclear`

Purpose: Initializes a window and the default values for the window

Processing: See function.

---

## 1.2 Types

---

### 1.2.1 BOOL\_16

```
typedef INT_16 BOOL_16;
```

**File**

`lpc_types.h` ([see page 73](#))

**Description**

16 bit boolean type

---

### 1.2.2 BOOL\_32

```
typedef INT_32 BOOL_32;
```

**File**

`lpc_types.h` ([see page 73](#))

**Description**

32 bit boolean type

---

### 1.2.3 BOOL\_8

```
typedef INT_8 BOOL_8;
```

**File**

`lpc_types.h` ([see page 73](#))

**Description**

8 bit boolean type

---

## 1.2.4 CHAR

```
typedef char CHAR;
```

### File

lpc\_types.h (see page 73)

### Description

SMA type for character type

---

## 1.2.5 COLOR\_T

```
typedef UNS_8 COLOR_T;
```

### File

lpc\_colors.h (see page 63)

### Description

Color type is a 8-bit value

---

## 1.2.6 FONT\_T

```
typedef struct {  
    INT_16 font_height;  
    UNS_8 first_char;  
    UNS_8 last_char;  
    UNS_16 * font_table;  
    UNS_8 * font_width_table;  
} FONT_T;
```

### File

lpc\_fonts.h (see page 64)

### Description

Font data structure

---

## 1.2.7 INT\_16

```
typedef signed short INT_16;
```

### File

lpc\_types.h (see page 73)

### Description

SMA type for 16 bit signed value

---

---

## 1.2.8 INT\_32

```
typedef signed int INT_32;
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

SMA type for 32 bit signed value

---

## 1.2.9 INT\_64

```
typedef long long INT_64;
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

SMA type for 64 bit signed value

---

## 1.2.10 INT\_8

```
typedef signed char INT_8;
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

SMA type for 8 bit signed value

---

## 1.2.11 PFI

```
typedef INT_32 (* PFI)();
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

Pointer to Function returning INT\_32 ([see page 21](#)) (any number of parameters)

---

## 1.2.12 PFV

```
typedef void (* PFV)();
```

---

**File**

lpc\_types.h (see page 73)

**Description**

Pointer to Function returning Void (any number of parameters)

---

## 1.2.13 STATUS

```
typedef INT_32 STATUS;
```

**File**

lpc\_types.h (see page 73)

**Description**

Status type

---

## 1.2.14 SWIM\_ROTATION\_T

```
typedef enum {  
    NOROTATION,  
    RIGHT,  
    INVERT,  
    LEFT  
} SWIM_ROTATION_T;
```

**File**

lpc\_swim\_image.h (see page 72)

**Description**

Image rotation tags

---

## 1.2.15 SWIM\_WINDOW\_T

```
typedef struct {  
    INT_32 xpsize;  
    INT_32 ypsize;  
    INT_32 xpmin;  
    INT_32 ypmin;  
    INT_32 xpmax;  
    INT_32 ypmax;  
    INT_32 bdsizes;  
    INT_32 xvsize;  
    INT_32 yvsize;  
    INT_32 xpvmin;  
    INT_32 ypvmin;  
    INT_32 xpvmax;  
    INT_32 ypvmax;  
    INT_32 xvpos;  
    INT_32 yvpos;  
    COLOR_T pen;  
    COLOR_T bkg;  
    COLOR_T fill;  
    FONT_T * font;  
    INT_32 tfont;  
    COLOR_T * fb;
```

```

    INT_32 winused;
    BOOL_32 tfonts;
} SWIM_WINDOW_T;

```

**File**

lpc\_swim.h (see page 68)

**Members**

Members	Description
INT_32 xpsize;	Physical (absolute) horizontal screen size
INT_32 ypsize;	Physical (absolute) vertical screen size
INT_32 xpmmin;	Physical left edge of window
INT_32 ypmmin;	Physical top edge of window
INT_32 xpmmax;	Physical right edge of window
INT_32 ypmmax;	Physical bottom edge of window
INT_32 bdsiz;	Size of window frame in pixels
INT_32 xvsize;	Virtual horizontal window size
INT_32 yvsize;	Virtual vertical window size
INT_32 xpvmin;	Physical left edge of draw window
INT_32 ypvmin;	Physical top edge of draw window
INT_32 xpvmax;	Physical right edge of draw window
INT_32 ypvmax;	Physical bottom edge of draw window
INT_32 xvpos;	Next virtual 'x' position of output
INT_32 yvpos;	Next virtual 'y' position of output
COLOR_T pen;	Pen/text color
COLOR_T bkg;	Window/text background color
COLOR_T fill;	Fill/border color
FONT_T * font;	Selected font structure
INT_32 tfont;	Transparent font background flag when true
COLOR_T * fb;	Frame buffer address for the physical display
INT_32 winused;	Window used flag
BOOL_32 tfonts;	Transparent font background flag

**Description**

Structure is used to store information about a specific window

---

## 1.2.16 UNS\_16

```
typedef unsigned short UNS_16;
```

**File**

lpc\_types.h (see page 73)

**Description**

SMA type for 16 bit unsigned value

---

## 1.2.17 UNS\_32

```
typedef unsigned int UNS_32;
```

**File**

lpc\_types.h (see page 73)

**Description**

SMA type for 32 bit unsigned value

---

## 1.2.18 UNS\_64

```
typedef unsigned long long UNS_64;
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

SMA type for 64 bit unsigned value

---

## 1.2.19 UNS\_8

```
typedef unsigned char UNS_8;
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

SMA type for 8 bit unsigned value

---

# 1.3 Variables

---

## 1.3.1 font\_helvr10

```
const FONT_T font_helvr10;
```

**File**

lpc\_helvr10.c ([↗](#) see page 65)

**Description**

Externally available font information structure

---

## 1.3.2 font\_rom8x16

```
const FONT_T font_rom8x16;
```

**File**

lpc\_rom8x16.c ([↗](#) see page 66)

---

**Description**

Externally available font information structure

---

## 1.3.3 font\_rom8x8

```
const FONT_T font_rom8x8;
```

**File**

lpc\_rom8x8.c (see page 67)

**Description**

Externally available font information structure

---

## 1.3.4 font\_winfreesys14x16

```
const FONT_T font_winfreesys14x16;
```

**File**

lpc\_winfreesystem14x16.c (see page 75)

**Description**

Externally available font information structure

---

## 1.3.5 font\_x5x7

```
const FONT_T font_x5x7;
```

**File**

lpc\_x5x7.c (see page 76)

**Description**

Externally available font information structure

---

## 1.3.6 font\_x6x13

```
const FONT_T font_x6x13;
```

**File**

lpc\_x6x13.c (see page 77)

**Description**

Externally available font information structure

---

## 1.3.7 helvr10\_bits

```
static UNS_16 helvr10_bits[] = { 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x4000, 0x4000, 0x4000, 0x4000,
0x4000, 0x4000, 0x0000, 0x4000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x5000, 0x5000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x2800,
0x2800, 0x7c00, 0x2800, 0xf800, 0x5000, 0x5000, 0x0000, 0x0000, 0x0000, 0x0000, 0x2000,
0x7000, 0xa800, 0xa000, 0x7000, 0x2800, 0xa800, 0x7000, 0x2000, 0x0000, 0x0000, 0x0000,
0x6400, 0x9400, 0x6800, 0x0800, 0x1000, 0x1600, 0x2900, 0x2600, 0x0000, 0x0000, 0x0000,
0x0000, 0x1000, 0x2800, 0x2800, 0x3000, 0x5200, 0x4c00, 0x4c00, 0x3200, 0x0000, 0x0000,
0x0000, 0x0000, 0x2000, 0x2000, 0x4000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x2000, 0x4000, 0x4000, 0x8000, 0x8000, 0x8000, 0x8000, 0x4000,
0x4000, 0x2000, 0x0000, 0x0000, 0x4000, 0x2000, 0x2000, 0x1000, 0x1000, 0x1000, 0x1000,
0x2000, 0x2000, 0x4000, 0x0000, 0x0000, 0xa000, 0x4000, 0xa000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x2000, 0x2000, 0xf800,
0x2000, 0x2000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x4000, 0x4000, 0x8000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x7c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x2000,
0x2000, 0x4000, 0x4000, 0x4000, 0x4000, 0x8000, 0x8000, 0x8000, 0x0000, 0x0000, 0x0000,
0x7000, 0x8800, 0x8800, 0x8800, 0x8800, 0x8800, 0x8800, 0x7000, 0x0000, 0x0000, 0x0000,
0x0000, 0x2000, 0x6000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x0000, 0x0000,
0x0000, 0x0000, 0x7000, 0x8800, 0x0800, 0x0800, 0x3000, 0x4000, 0x8000, 0xf800, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x8800, 0x0800, 0x0800, 0x3000, 0x0800, 0x0800, 0x8800, 0x7000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1000, 0x3000, 0x5000, 0x5000, 0x9000, 0xf800, 0x1000,
0x1000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf800, 0x8000, 0x8000, 0xf000, 0x0800, 0x0800,
0x8800, 0x7000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7000, 0x8800, 0x8000, 0xb000, 0xc800,
0x8800, 0x8800, 0x7000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf800, 0x0800, 0x1000, 0x1000,
0x2000, 0x2000, 0x4000, 0x4000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7000, 0x8800, 0x8800,
0x7000, 0x8800, 0x8800, 0x8800, 0x7000, 0x0000, 0x0000, 0x0000, 0x7000, 0x8800,
0x8800, 0x9800, 0x6800, 0x0800, 0x8800, 0x7000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x4000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x4000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x4000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x4000, 0x8000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1000, 0x2000, 0x4000, 0x1000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf000, 0x0000, 0xf000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x4000, 0x2000, 0x1000, 0x2000, 0x4000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x3000, 0x4800, 0x0800, 0x1000, 0x2000, 0x2000, 0x0000,
0x2000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1f00, 0x2080, 0x4d40, 0x9240, 0xa240, 0xa480,
0xa480, 0x9b00, 0x4000, 0x3e00, 0x0000, 0x0000, 0x1000, 0x1000, 0x2800, 0x2800, 0x4400,
0x7c00, 0x8200, 0x8200, 0x0000, 0x0000, 0x0000, 0x0000, 0x7800, 0x4400, 0x4400, 0x7800,
0x4400, 0x4400, 0x4400, 0x7800, 0x0000, 0x0000, 0x0000, 0x0000, 0x3c00, 0x4200, 0x4000,
0x4000, 0x4000, 0x4000, 0x4200, 0x3c00, 0x0000, 0x0000, 0x0000, 0x7800, 0x4400,
0x4200, 0x4200, 0x4200, 0x4200, 0x4400, 0x7800, 0x0000, 0x0000, 0x0000, 0x7c00,
0x4000, 0x4000, 0x7c00, 0x4000, 0x4000, 0x4000, 0x7c00, 0x0000, 0x0000, 0x0000, 0x0000,
0x7c00, 0x4000, 0x4000, 0x7800, 0x4000, 0x4000, 0x4000, 0x4000, 0x0000, 0x0000, 0x0000,
0x0000, 0x3c00, 0x4200, 0x4000, 0x4000, 0x4000, 0x4600, 0x4200, 0x4600, 0x3a00, 0x0000,
0x0000, 0x0000, 0x4200, 0x4200, 0x4200, 0x7e00, 0x4200, 0x4200, 0x4200, 0x4200, 0x4200,
0x0000, 0x0000, 0x0000, 0x0000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1000, 0x1000, 0x1000, 0x1000, 0x1000, 0x1000, 0x9000,
0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x4400, 0x4800, 0x5000, 0x7000, 0x4800, 0x4800,
0x4400, 0x4400, 0x0000, 0x0000, 0x0000, 0x0000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000,
0x4000, 0x4000, 0x7800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x4100, 0x6300, 0x6300, 0x5500,
0x5500, 0x4900, 0x4900, 0x4900, 0x0000, 0x0000, 0x0000, 0x0000, 0x6200, 0x6200, 0x5200,
0x5200, 0x4a00, 0x4a00, 0x4600, 0x4600, 0x0000, 0x0000, 0x0000, 0x0000, 0x3c00, 0x4200,
0x4200, 0x4200, 0x4200, 0x4200, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x7800,
0x4400, 0x4400, 0x7800, 0x4000, 0x4000, 0x4000, 0x4000, 0x0000, 0x0000, 0x0000,
0x3c00, 0x4200, 0x4200, 0x4200, 0x4200, 0x4a00, 0x4600, 0x3e00, 0x0100, 0x0000, 0x0000,
0x0000, 0x7800, 0x4400, 0x4400, 0x7800, 0x4400, 0x4400, 0x4400, 0x4400, 0x0000, 0x0000,
0x0000, 0x0000, 0x3800, 0x4400, 0x4000, 0x3800, 0x0400, 0x4400, 0x4400, 0x3800, 0x0000,
0x0000, 0x0000, 0x0000, 0xf800, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000,
0x0000, 0x0000, 0x0000, 0x0000, 0x4200, 0x4200, 0x4200, 0x4200, 0x4200, 0x4200, 0x4200,
0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x8200, 0x8200, 0x4400, 0x4400, 0x2800,
0x2800, 0x1000, 0x0000, 0x0000, 0x0000, 0x0000, 0x8880, 0x8880, 0x4900, 0x4900, 0x5500,
0x2200, 0x2200, 0x2200, 0x0000, 0x0000, 0x0000, 0x0000, 0x4400, 0x4400, 0x2800, 0x1000,
0x2800, 0x2800, 0x4400, 0x4400, 0x0000, 0x0000, 0x0000, 0x0000, 0x8200, 0x4400, 0x4400,
0x2800, 0x2800, 0x1000, 0x1000, 0x1000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7c00, 0x0400,
0x0800, 0x1000, 0x1000, 0x2000, 0x4000, 0x7c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x6000,
```



```

0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x6000, 0x0000, 0x0000,
0x8000, 0x8000, 0x4000, 0x4000, 0x4000, 0x4000, 0x2000, 0x2000, 0x0000, 0x0000, 0x0000,
0x0000, 0xc000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0xc000,
0x0000, 0x0000, 0x2000, 0x2000, 0x5000, 0x5000, 0x8800, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0xfc00, 0x0000, 0x0000, 0x4000, 0x8000, 0x8000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xe000, 0x1000, 0x7000, 0x9000,
0x9000, 0x6800, 0x0000, 0x0000, 0x0000, 0x0000, 0x8000, 0x8000, 0xb000, 0xc800, 0x8800,
0x8800, 0xc800, 0xb000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6000, 0x9000,
0x8000, 0x8000, 0x9000, 0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0800, 0x0800, 0x6800,
0x9800, 0x8800, 0x8800, 0x9800, 0x6800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x6000, 0x9000, 0xf000, 0x8000, 0x9000, 0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3000,
0x4000, 0xe000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x6800, 0x9800, 0x8800, 0x8800, 0x9800, 0x6800, 0x0800, 0x7000, 0x0000,
0x0000, 0x8000, 0x8000, 0xb000, 0xc800, 0x8800, 0x8800, 0x8800, 0x8800, 0x0000, 0x0000,
0x0000, 0x0000, 0x8000, 0x0000, 0x8000, 0x8000, 0x8000, 0x8000, 0x8000, 0x8000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x8000, 0x8000, 0x8000, 0x8000, 0x8000, 0x8000, 0x8000,
0x8000, 0x8000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xec00, 0x9200, 0x9200,
0x9200, 0x9200, 0x9200, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xb000, 0xc800,
0x8800, 0x8800, 0x8800, 0x8800, 0x8800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7000,
0x8800, 0x8800, 0x8800, 0x8800, 0x7000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0xb000, 0xc800, 0x8800, 0x8800, 0xc800, 0xb000, 0x8000, 0x8000, 0x0000, 0x0000, 0x0000,
0x0000, 0x6800, 0x9800, 0x8800, 0x8800, 0x9800, 0x6800, 0x0800, 0x0800, 0x0000, 0x0000,
0x0000, 0x0000, 0xa000, 0xc000, 0x8000, 0x8000, 0x8000, 0x8000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x6000, 0x9000, 0x6000, 0x1000, 0x9000, 0x6000, 0x0000, 0x0000,
0x0000, 0x0000, 0x4000, 0x4000, 0xe000, 0x4000, 0x4000, 0x4000, 0x4000, 0x6000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x9000, 0x9000, 0x9000, 0x9000, 0x9000, 0x7000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x8800, 0x8800, 0x5000, 0x5000, 0x2000,
0x2000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x9200, 0x9200, 0x5400, 0x5400,
0x2800, 0x2800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x8800, 0x5000, 0x2000,
0x5000, 0x8800, 0x8800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x9000, 0x9000,
0xa000, 0xa000, 0x6000, 0x4000, 0x4000, 0x8000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf000,
0x1000, 0x2000, 0x4000, 0x8000, 0xf000, 0x0000, 0x0000, 0x0000, 0x0000, 0x2000, 0x4000,
0x4000, 0x4000, 0x8000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x4000, 0x0000, 0x0000,
0x8000, 0x4000, 0x4000, 0x4000, 0x2000, 0x4000, 0x4000, 0x4000, 0x4000, 0x8000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x6400, 0x9800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, };
```

**File**

lpc\_helvr10.c (see page 65)

**Description**

Font character bitmap data.

## 1.3.8 helvR10\_width

```

static UNS_8 helvR10_width[] = { 3, 3, 4, 6, 6, 9, 8, 3, 4, 4, 4, 6, 3, 7, 3, 3, 6, 6, 6,
6, 6, 6, 6, 6, 6, 6, 3, 3, 6, 5, 6, 6, 11, 7, 7, 8, 8, 7, 6, 8, 8, 3, 5, 7, 6, 9, 8, 8, 7,
8, 7, 7, 5, 8, 7, 9, 7, 7, 7, 3, 3, 6, 6, 3, 5, 6, 5, 6, 5, 4, 6, 6, 2, 2, 5, 2, 8, 6,
6, 6, 6, 4, 5, 4, 5, 6, 8, 6, 5, 3, 3, 3, 7, };
```

**File**

lpc\_helvr10.c (see page 65)

**Description**

Character width data.

## 1.3.9 rom8x16\_bits

```
static UNS_16 rom8x16_bits[] = { 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x7e00, 0x8100, 0xa500, 0x8100, 0x8100, 0xbd00, 0x9900, 0x8100, 0x8100, 0x7e00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7c00, 0xfe00, 0xfe00, 0xd600, 0xfe00, 0xfe00,
0xba00, 0xc600, 0xfe00, 0x7c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x6c00, 0xee00, 0xfe00, 0xfe00, 0xfe00, 0xfe00, 0x7c00, 0x3800, 0x1000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1000, 0x3800, 0x7c00, 0xfe00, 0x7c00, 0x3800,
0x1000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1000,
0x3800, 0x3800, 0x1000, 0x6c00, 0xee00, 0x6c00, 0x1000, 0x3800, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x1000, 0x3800, 0x7c00, 0x7c00, 0xfe00, 0xfe00, 0xfe00, 0x6c00,
0x1000, 0x3800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x1800, 0x3c00, 0x3c00, 0x3c00, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0xff00, 0xff00, 0xff00, 0xff00, 0xff00, 0xe700, 0xc300, 0xc300, 0xc300, 0xe700, 0xff00,
0xff00, 0xff00, 0xff00, 0xff00, 0xff00, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3c00,
0x6600, 0x6600, 0x6600, 0x3c00, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xff00,
0xff00, 0xff00, 0xff00, 0xe700, 0xc300, 0x9900, 0x9900, 0x9900, 0xc300, 0xe700, 0xff00,
0xff00, 0xff00, 0xff00, 0xff00, 0xff00, 0x0000, 0x0000, 0x0000, 0x0000, 0x3600, 0x7800,
0xcc00, 0xcc00, 0xcc00, 0xcc00, 0x7800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x3c00, 0x6600, 0x6600, 0x6600, 0x3c00, 0x1800, 0x7e00, 0x1800, 0x1800, 0x1800, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1e00, 0x1a00, 0x1e00, 0x1800, 0x1800, 0x1800,
0x1800, 0x7800, 0xf800, 0x7000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3e00,
0x3600, 0x3e00, 0x3600, 0x3600, 0x7600, 0xf600, 0x6600, 0x0e00, 0x1e00, 0x0c00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0xdb00, 0x7e00, 0x3c00, 0x6600, 0x6600, 0x3c00,
0x7e00, 0xdb00, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x8000,
0xe000, 0xf000, 0xfc00, 0xfe00, 0xfc00, 0xf000, 0xe000, 0x8000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0200, 0x0e00, 0x3e00, 0x7e00, 0xfe00, 0x7e00, 0x3e00,
0x0e00, 0x0200, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3c00, 0x7e00,
0x1800, 0x1800, 0x1800, 0x1800, 0x7e00, 0x3c00, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x0000,
0x6600, 0x6600, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7f00, 0xdb00, 0xdb00,
0xdb00, 0xdb00, 0x7b00, 0x1b00, 0x1b00, 0x1b00, 0x1b00, 0x1b00, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x7c00, 0xc600, 0xc600, 0xc600, 0x6000, 0x7c00, 0xf600, 0xde00, 0x7c00,
0x0c00, 0xc600, 0xc600, 0x7c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0xfe00, 0xfe00, 0xfe00, 0xfe00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x1800, 0x3c00, 0x7e00, 0x1800, 0x1800, 0x1800, 0x7e00, 0x3c00, 0x1800, 0x7e00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3c00, 0x7e00, 0x1800, 0x1800, 0x1800,
0x1800, 0x1800, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800,
0x1800, 0x1800, 0x1800, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x3000, 0x7000, 0xfe00, 0x7000, 0x3000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xc000, 0xc000, 0xc000, 0xfe00,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x2400, 0x6600, 0xff00, 0x6600, 0x2400, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x1000, 0x3800, 0x3800, 0x3800, 0x7c00, 0x7c00, 0xfe00, 0xfe00,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7c00,
0x7c00, 0x3800, 0x3800, 0x1000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3c00, 0x3c00, 0x3c00, 0x3c00,
0x3600, 0x3600, 0x3600, 0x1400, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6c00, 0x6c00, 0x6c00, 0xfe00, 0x6c00, 0x6c00,
0xfe00, 0x6c00, 0x6c00, 0x6c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800,
0x1800, 0x7c00, 0xc600, 0xc000, 0x7800, 0x3c00, 0x0600, 0xc600, 0x7c00, 0x1800, 0x1800,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6200, 0x6600, 0x0c00, 0x1800,
0x3000, 0x6600, 0xc600, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3800, 0x6c00,
0x3800, 0x3000, 0x7600, 0x7e00, 0xcc00, 0xcc00, 0xcc00, 0x7600, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0c00, 0x0c00, 0x0c00, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0c00, 0x1800, 0x3000,
0x3000, 0x3000, 0x3000, 0x3000, 0x1800, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00,
0x0000, 0x0000, 0x3000, 0x1800, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x1800,
0x3000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6c00,
0x3800, 0xfe00, 0x3800, 0x6c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800, 0x7e00, 0x1800, 0x1800, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
```

29

30

31

32

```
0x1800, 0x3000, 0x0000, 0x0000, 0x7e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0c00, 0x1800, 0x3000, 0x6000, 0x3000, 0x1800, 0x0c00, 0x0000, 0x0000, 0x7e00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0c00, 0x1e00, 0x1a00, 0x1800,
0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800,
0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800, 0x0000, 0x7e00, 0x0000,
0x1800, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x7600, 0xdc00, 0x0000, 0x7600, 0xdc00, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x7800, 0xcc00, 0xcc00, 0x7800, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x1800, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1f00, 0x1800, 0x1800,
0x1800, 0x1800, 0xd800, 0xd800, 0x7800, 0x3800, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0xd800, 0x6c00, 0x6c00, 0x6c00, 0x6c00, 0x6c00, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7000, 0xd800, 0x1800, 0x3000, 0x6000,
0xf800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x7e00, 0x7e00, 0x7e00, 0x7e00, 0x7e00, 0x7e00, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, };
```

## File

lpc\_rom8x16.c (see page 66)

### Description

This is variable rom8x16\_bits.

### 1.3.10 rom8x16 width

[illegible]

## File

lpc\_rom8x16.c (see page 66)

### Description

Character width data.

### 1.3.11 rom8x8 bits

```
static UNS_16 rom8x8_bits[] = { 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x7e00, 0x8100, 0xa500, 0x8100, 0xbd00, 0x9900, 0x8100, 0x7e00, 0x7c00, 0xfe00,
0xd600, 0xba00, 0xc600, 0xfe00, 0x7c00, 0x0000, 0xc600, 0xee00, 0xfe00, 0xfe00, 0x7c00,
0x3800, 0x1000, 0x0000, 0x1000, 0x3800, 0x7c00, 0xfe00, 0x7c00, 0x3800, 0x1000, 0x0000,
0x1000, 0x3800, 0x1000, 0xee00, 0xee00, 0x1000, 0x3800, 0x0000, 0x3800, 0x7c00, 0xfe00
```

0xfe00, 0x6c00, 0x1000, 0x3800, 0x0000, 0x0000, 0x1800, 0x3c00, 0x7e00, 0x3c00, 0x1800,  
0x0000, 0x0000, 0xff00, 0xe700, 0xc300, 0x8100, 0xc300, 0xe700, 0xff00, 0xff00, 0x0000,  
0x1800, 0x3c00, 0x6600, 0x6600, 0x3c00, 0x1800, 0x0000, 0xff00, 0xe700, 0xc300, 0x9900,  
0x9900, 0xc300, 0xe700, 0xff00, 0x1e00, 0x0e00, 0x1e00, 0x3600, 0x7800, 0xcc00, 0xcc00,  
0x7800, 0x7e00, 0xc300, 0xc300, 0x7e00, 0x1800, 0x7e00, 0x1800, 0x1800, 0x1e00, 0x1a00,  
0x1e00, 0x1800, 0x1800, 0x7000, 0xf000, 0x6000, 0x3e00, 0x3e00, 0x3600, 0x3600, 0xf600,  
0x6600, 0x1e00, 0x0c00, 0xdb00, 0x3c00, 0x6600, 0xe700, 0x6600, 0x3c00, 0xdb00, 0x0000,  
0x8000, 0xc000, 0xf000, 0xf800, 0xf000, 0xc000, 0x8000, 0x0000, 0x0200, 0x0600, 0x1e00,  
0x3e00, 0x1e00, 0x0600, 0x0200, 0x0000, 0x1800, 0x3c00, 0x7e00, 0x1800, 0x7e00, 0x3c00,  
0x1800, 0x0000, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x0000, 0x6600, 0x0000, 0x7f00,  
0xdb00, 0x7b00, 0x3b00, 0x1b00, 0x1b00, 0x1b00, 0x0000, 0x3c00, 0x6600, 0x3800, 0x6c00,  
0x6c00, 0x3800, 0xcc00, 0x7800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xfe00, 0xfe00,  
0x0000, 0x1800, 0x3c00, 0x7e00, 0x1800, 0x7e00, 0x3c00, 0x1800, 0x7e00, 0x1800, 0x3c00,  
0x7e00, 0x1800, 0x1800, 0x1800, 0x1800, 0x0000, 0x1800, 0x1800, 0x1800, 0x1800, 0x7e00,  
0x3c00, 0x1800, 0x0000, 0x0000, 0x1800, 0x1c00, 0xfe00, 0x1c00, 0x1800, 0x0000, 0x0000,  
0x0000, 0x3000, 0x7000, 0xfe00, 0x7000, 0x3000, 0x0000, 0x0000, 0x0000, 0x0000, 0xc000,  
0xc000, 0xc000, 0xfe00, 0x0000, 0x0000, 0x0000, 0x2400, 0x6600, 0xff00, 0x6600, 0x2400,  
0x0000, 0x0000, 0x0000, 0x1000, 0x3800, 0x7c00, 0x7c00, 0xfe00, 0x0000, 0x0000, 0x0000,  
0xfe00, 0x7c00, 0x7c00, 0x3800, 0x1000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,  
0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3c00, 0x3c00, 0x1800, 0x1800, 0x0000, 0x1800,  
0x0000, 0x6c00, 0x6c00, 0x6c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6c00, 0x6c00,  
0xfe00, 0x6c00, 0xfe00, 0x6c00, 0x6c00, 0x0000, 0x1800, 0x7e00, 0xc000, 0x7c00, 0x0600,  
0xfc00, 0x1800, 0x0000, 0x0000, 0xc600, 0x0c00, 0x1800, 0x3000, 0x6000, 0xc600, 0x0000,  
0x3800, 0x6c00, 0x3800, 0x7600, 0xcc00, 0xcc00, 0x7600, 0x0000, 0x1800, 0x1800, 0x3000,  
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3000, 0x6000, 0x6000, 0x6000, 0x3000,  
0x1800, 0x0000, 0x6000, 0x3000, 0x1800, 0x1800, 0x3000, 0x6000, 0x0000, 0x0000, 0x0000,  
0xee00, 0x7c00, 0xfe00, 0x7c00, 0xee00, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800, 0x7e00,  
0x1800, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800, 0x3000,  
0x0000, 0x0000, 0x0000, 0x0000, 0xfe00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,  
0x0000, 0x0000, 0x0000, 0x3800, 0x3800, 0x0000, 0x0600, 0x0c00, 0x1800, 0x3000, 0x6000,  
0xc000, 0x8000, 0x0000, 0x7c00, 0xc600, 0xce00, 0xde00, 0xf600, 0xe600, 0x7c00, 0x0000,  
0x1800, 0x7800, 0x1800, 0x1800, 0x1800, 0x1800, 0x7e00, 0x0000, 0x7c00, 0xc600, 0x0c00,  
0x1800, 0x3000, 0x6600, 0xfe00, 0x0000, 0x7c00, 0xc600, 0x0600, 0x3c00, 0x0600, 0xc600,  
0x7c00, 0x0000, 0x0c00, 0x1c00, 0x3c00, 0x6c00, 0xfe00, 0x0c00, 0x0c00, 0x0000, 0xfe00,  
0xc000, 0xfc00, 0x0600, 0x0600, 0xc600, 0x7c00, 0x0000, 0x7c00, 0xc600, 0xc000, 0xfc00,  
0xc600, 0xc600, 0x7c00, 0x0000, 0xfe00, 0xc600, 0x0c00, 0x0c00, 0x1800, 0x1800, 0x1800,  
0x0000, 0x7c00, 0xc600, 0xc600, 0x7c00, 0x0000, 0x0000, 0x1c00, 0x1c00, 0x0000, 0x0000,  
0x1c00, 0x1c00, 0x0000, 0x0000, 0x1800, 0x1800, 0x0000, 0x0000, 0x1800, 0x1800, 0x3000,  
0x0c00, 0x1800, 0x3000, 0x6000, 0x3000, 0x1800, 0x0c00, 0x0000, 0x0000, 0x0000, 0xfe00,  
0x0000, 0x0000, 0x0000, 0xfe00, 0x0000, 0x0000, 0x6000, 0x3000, 0x1800, 0x0c00, 0x1800, 0x3000,  
0x6000, 0x0000, 0x7c00, 0xc600, 0x0600, 0x0c00, 0x1800, 0x0000, 0x1800, 0x0000, 0x7c00,  
0xc600, 0xc600, 0xde00, 0xdc00, 0xc000, 0x7e00, 0x0000, 0x3800, 0x6c00, 0xc600, 0xc600,  
0xfe00, 0xc600, 0xc600, 0x0000, 0xfc00, 0x6600, 0x6600, 0x7c00, 0x6600, 0x6600, 0xfc00,  
0x0000, 0x3c00, 0x6600, 0xc000, 0xc000, 0xc000, 0x6600, 0x3c00, 0x0000, 0xf800, 0x6c00,  
0x6600, 0x6600, 0x6600, 0x6600, 0xf800, 0x0000, 0xfe00, 0xc200, 0xc000, 0xf800, 0xc000,  
0xc200, 0xfe00, 0x0000, 0xfe00, 0x6200, 0x6000, 0x7c00, 0x6000, 0x6000, 0xf000, 0x0000,  
0x7c00, 0xc600, 0xc000, 0xc000, 0xde00, 0xc600, 0x7c00, 0x0000, 0xc600, 0xc600, 0xc600,  
0xfe00, 0xc600, 0xc600, 0xc600, 0x0000, 0x3c00, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800,  
0x3c00, 0x0000, 0x3c00, 0x1800, 0x1800, 0x1800, 0xd800, 0xd800, 0x0000, 0xf000, 0x6000, 0x6000,  
0xcc00, 0xd800, 0xf000, 0xd800, 0xcc00, 0xc600, 0x0000, 0xf000, 0x6000, 0x6000, 0x6000,  
0x6000, 0x6200, 0xfe00, 0x0000, 0xc600, 0xee00, 0xfe00, 0xd600, 0xd600, 0xc600, 0xc600,  
0x0000, 0xc600, 0xe600, 0xe600, 0xf600, 0xde00, 0xce00, 0xc600, 0x0000, 0x7c00, 0xc600,  
0xc600, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0000, 0xfc00, 0x6600, 0x6600, 0x7c00, 0x6000,  
0x6000, 0xf000, 0x0000, 0x0000, 0x7c00, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0600, 0xc600,  
0x7c00, 0x0600, 0xc600, 0x7c00, 0x0000, 0x7e00, 0x5a00, 0x1800, 0x1800, 0x1800, 0x1800,  
0x3c00, 0x0000, 0xc600, 0xc600, 0xc600, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0000, 0xc600,  
0xc600, 0xc600, 0xc600, 0x6c00, 0x3800, 0x1000, 0x0000, 0xc600, 0xc600, 0xd600, 0xd600,  
0xfe00, 0xee00, 0xc600, 0x0000, 0xc600, 0x6c00, 0x3800, 0x3800, 0x6c00, 0xc600, 0xc600,  
0x0000, 0x6600, 0x6600, 0x6600, 0x3c00, 0x1800, 0x1800, 0x3c00, 0x0000, 0xfe00, 0x8600,  
0x0c00, 0x1800, 0x3000, 0x6200, 0xfe00, 0x0000, 0x7c00, 0x6000, 0x6000, 0x6000, 0x6000,  
0x6000, 0x7c00, 0x0000, 0xc000, 0x6000, 0x3000, 0x1800, 0x0c00, 0x0600, 0x0200, 0x0000,  
0x7c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x7c00, 0x0000, 0x1000, 0x3800, 0x6c00,  
0xc600, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,  
0x0000, 0xff00, 0x3000, 0x3000, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,  
0x0000, 0x7800, 0x0c00, 0x7c00, 0xcc00, 0x7e00, 0x0000, 0xe000, 0x6000, 0x7c00, 0x6600,  
0x6600, 0x6600, 0xfc00, 0x0000, 0x0000, 0x0000, 0x7c00, 0xc600, 0xc000, 0xc600, 0x7c00,  
0x0000, 0x1c00, 0x0c00, 0x7c00, 0xcc00, 0xcc00, 0xc000, 0x7e00, 0x0000, 0x0000, 0x0000,  
0x7c00, 0xc600, 0xfe00, 0xc000, 0x7c00, 0x0000, 0x1c00, 0x3600, 0x3000, 0xfc00, 0x3000,  
0x3000, 0x7800, 0x0000, 0x0000, 0x0000, 0x7600, 0xce00, 0xc600, 0x7e00, 0x0600, 0x7c00,  
0xe000, 0x6000, 0x7c00, 0x6600, 0x6600, 0x6600, 0x6600, 0xe600, 0x0000, 0x1800, 0x0000, 0x3800,



0x1800, 0x1800, 0x1800, 0x3c00, 0x0000, 0x0c00, 0x0000, 0x1c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00,  
0x0c00, 0x7800, 0xe000, 0x6000, 0x6600, 0x6c00, 0x7800, 0x6c00, 0xe600, 0x0000, 0x1800, 0x1800,  
0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1c00, 0x0000, 0x0000, 0x0000, 0x6c00, 0xfe00,  
0xd600, 0xd600, 0xc600, 0x0000, 0x0000, 0x0000, 0xdc00, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600,  
0x0000, 0x0000, 0x0000, 0x7c00, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0000, 0x0000, 0x0000,  
0xdc00, 0x6600, 0x6600, 0x7c00, 0x6000, 0xf000, 0x0000, 0x0000, 0x7600, 0xcc00, 0xcc00,  
0x7c00, 0x0c00, 0x1e00, 0x0000, 0x0000, 0xdc00, 0x6600, 0x6000, 0x6000, 0xf000, 0x0000, 0x0000,  
0x0000, 0x0000, 0x7c00, 0xc000, 0xc000, 0x7c00, 0x0600, 0x7c00, 0x0000, 0x3000, 0x3000, 0xfc00,  
0x3000, 0x3000, 0x3600, 0x1c00, 0x0000, 0x0000, 0x0000, 0xcc00, 0xcc00, 0xcc00, 0xcc00, 0xcc00,  
0x7600, 0x0000, 0x0000, 0x0000, 0xc600, 0xc600, 0x6c00, 0x3800, 0x1000, 0x0000, 0x0000, 0x0000,  
0x0000, 0xc600, 0xc600, 0xd600, 0xfe00, 0x6c00, 0x0000, 0x0000, 0x0000, 0xc600, 0x6c00, 0x6c00,  
0x3800, 0x6c00, 0xc600, 0x0000, 0x0000, 0x0000, 0xc600, 0xc600, 0xc600, 0xc600, 0x7600, 0x0600,  
0x7c00, 0x0000, 0x0000, 0xfc00, 0x9800, 0x3000, 0x6400, 0xfc00, 0x0000, 0x0e00, 0x1800, 0x1800,  
0x7000, 0x1800, 0x1800, 0x0e00, 0x0000, 0x1800, 0x1800, 0x1800, 0x0000, 0x1800, 0x1800,  
0x1800, 0x1800, 0x0000, 0x7000, 0x1800, 0x1800, 0x0e00, 0x1800, 0x1800, 0x7000, 0x0000,  
0x7600, 0xdc00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1000, 0x3800, 0x3800,  
0x3800, 0x6c00, 0x6c00, 0xfe00, 0x0000, 0x3c00, 0x6600, 0xc000, 0x6600, 0x3c00, 0x1800, 0x1800,  
0xcc00, 0x7800, 0x0000, 0xc600, 0x0000, 0xc600, 0xc600, 0xc600, 0xc600, 0x7600, 0x0000, 0x0e00,  
0x0000, 0x7c00, 0xc600, 0xfe00, 0xc000, 0x7c00, 0x0000, 0x7c00, 0xc600, 0x7800, 0x0c00, 0x7c00,  
0xc600, 0x7e00, 0x0000, 0xc600, 0x0000, 0x7800, 0x0c00, 0x7c00, 0xc600, 0xc600, 0x7e00,  
0x0000, 0xe000, 0x0000, 0x7800, 0x0c00, 0x7c00, 0xc600, 0x7e00, 0x0000, 0x3800, 0x3800, 0x3800,  
0x7800, 0x0c00, 0x7c00, 0xc600, 0xc600, 0x7e00, 0x0000, 0x0000, 0x0000, 0x7c00, 0xc600, 0x7c00,  
0x1800, 0x6c00, 0x3800, 0x7c00, 0xc600, 0x7c00, 0xc600, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0000,  
0xc600, 0x0000, 0x7c00, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0000, 0xe000, 0x0000, 0x7c00, 0xc600,  
0xc600, 0xc600, 0xc000, 0x7c00, 0x0000, 0x6600, 0x0000, 0x3800, 0x1800, 0x1800, 0x1800,  
0x3c00, 0x0000, 0x7c00, 0xc600, 0x3800, 0x1800, 0x1800, 0x3c00, 0x0000, 0xc600, 0x3800, 0x6c00, 0xc600,  
0xfe00, 0xc600, 0xc600, 0x0000, 0x3800, 0x3800, 0x0000, 0x7c00, 0xc600, 0xc600, 0xc600, 0xc600,  
0x0000, 0x0e00, 0x0000, 0xc600, 0xc000, 0xf800, 0xc000, 0xc600, 0x0000, 0x0000, 0x0000,  
0x6c00, 0x9a00, 0x7e00, 0xd800, 0x6e00, 0x0000, 0x7e00, 0xd800, 0xd800, 0xc600, 0xc600, 0xc600,  
0xd800, 0xc600, 0x0000, 0x7c00, 0xc600, 0x0000, 0x0000, 0x7c00, 0xc600, 0xc600, 0x7c00, 0x0000,  
0x0000, 0xc600, 0x0000, 0x7c00, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0000, 0x0000, 0xc600, 0xc600,  
0x7c00, 0xc600, 0xc600, 0x7c00, 0x0000, 0x7c00, 0xc600, 0x0000, 0xc600, 0xc600, 0xc600, 0xc600,  
0x7600, 0x0000, 0x0000, 0xe000, 0x0000, 0xc600, 0xc600, 0xc600, 0xc600, 0x7600, 0x0000, 0x0000,  
0xc600, 0x0000, 0xc600, 0xc600, 0xc600, 0x7600, 0x0600, 0x7c00, 0xc600, 0x3800, 0x6c00, 0xc600,  
0xc600, 0x0600, 0x3800, 0x0000, 0xc600, 0xc600, 0xc600, 0xc600, 0xc600, 0xc600, 0x7c00, 0x0000,  
0x0000, 0x0000, 0x1800, 0x7e00, 0xd800, 0xd800, 0x7e00, 0x1800, 0x0000, 0x3800, 0x6c00, 0x6c00,  
0x6000, 0xf000, 0x6600, 0xf600, 0x6c00, 0x0000, 0xc300, 0x6600, 0x3c00, 0x7e00, 0x1800, 0x1800,  
0x3c00, 0x1800, 0x0000, 0xfc00, 0xc600, 0xfc00, 0xcc00, 0xde00, 0xcc00, 0xc600, 0x0000, 0x0000,  
0x0c00, 0x1e00, 0x1800, 0x7e00, 0x1800, 0x1800, 0xd800, 0x7000, 0x0e00, 0x0000, 0x7800, 0x7800,  
0x0c00, 0x7c00, 0xc600, 0x7e00, 0x0000, 0x1c00, 0x0000, 0x3800, 0x1800, 0x1800, 0x1800, 0x1800,  
0x3c00, 0x0000, 0x0000, 0x0e00, 0x0000, 0x7c00, 0xc600, 0xc600, 0x7c00, 0x0000, 0x0000, 0x0000,  
0x0e00, 0x0000, 0xcc00, 0xcc00, 0xdc00, 0x7600, 0x0000, 0x0000, 0xfc00, 0x0000, 0xbc00, 0x6600,  
0x6600, 0x6600, 0xe600, 0x0000, 0xc600, 0x0000, 0xc600, 0xc600, 0xc600, 0xc600, 0xc600, 0xc600,  
0x0000, 0x3800, 0x6c00, 0x3e00, 0x0000, 0x7e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x7c00, 0xc600,  
0x7c00, 0x0000, 0x7c00, 0x0000, 0x0000, 0x0000, 0x1800, 0x0000, 0x1800, 0x3000, 0x3000, 0x6000,  
0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7c00, 0x6000, 0x6000, 0x0000, 0x0000,  
0x0000, 0x0000, 0x0000, 0x7c00, 0x0c00, 0x0c00, 0x0000, 0x0000, 0xc000, 0xc600, 0xc600, 0xd800,  
0x3000, 0x7c00, 0x3600, 0x0c00, 0x3e00, 0xc000, 0xc600, 0xd800, 0x3000, 0x6c00, 0x3c00, 0x0000,  
0x7e00, 0x0c00, 0x1800, 0x0000, 0x1800, 0x1800, 0x3c00, 0x3c00, 0x1800, 0x0000, 0x0000, 0x0000,  
0x3600, 0x6c00, 0xd800, 0x6c00, 0x3600, 0x0000, 0x0000, 0x0000, 0x0000, 0xd800, 0x6c00, 0x3600,  
0x6c00, 0xd800, 0x0000, 0x0000, 0x2200, 0x8800, 0x2200, 0x8800, 0x2200, 0x8800, 0x2200, 0x8800,  
0x8800, 0x5500, 0xaa00, 0x5500, 0xaa00, 0x5500, 0xaa00, 0x5500, 0xaa00, 0xdd00, 0x7700, 0x7700,  
0xdd00, 0x7700, 0xdd00, 0x7700, 0xdd00, 0x7700, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800,  
0x1800, 0x1800, 0xf800, 0x1800, 0xf800, 0x1800, 0x1800, 0x1800, 0x1800, 0x3600, 0x3600, 0x3600,  
0x3600, 0xf600, 0x3600, 0x3600, 0x3600, 0x0000, 0x0000, 0x0000, 0x0000, 0xc600, 0x3600, 0x3600,  
0x3600, 0x3600, 0x0000, 0x0000, 0xf800, 0x1800, 0xf800, 0x1800, 0x1800, 0x1800, 0x3600, 0x3600,  
0x3600, 0xf600, 0x0600, 0xf600, 0x3600, 0x3600, 0x3600, 0x3600, 0x3600, 0x3600, 0x3600, 0x3600,  
0x3600, 0x3600, 0x3600, 0x3600, 0x0600, 0xc600, 0x0000, 0x0000, 0x0000, 0x0000, 0x3600, 0x3600,  
0x3600, 0x3600, 0xc600, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800, 0xf800, 0x1800, 0xf800,  
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf800, 0x1800, 0x1800, 0x1800, 0x1800,  
0x1800, 0x1800, 0x1800, 0x1800, 0x1f00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800,  
0x1800, 0xff00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xff00, 0x1800, 0x1800,  
0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1f00, 0x1800, 0x1800, 0x1800, 0x0000, 0x0000,  
0x0000, 0x0000, 0x0000, 0xff00, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800, 0x1800, 0x1800,  
0xff00, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1f00, 0x1800, 0x1f00, 0x1800, 0x1800, 0x1800,  
0x1800, 0x3600, 0x3600, 0x3600, 0x3600, 0x3700, 0x3600, 0x3600, 0x3600, 0x3600, 0x3600, 0x3600,  
0x3700, 0x3000, 0x3f00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3f00, 0x3000, 0x3700, 0x3700,  
0x3600, 0x3600, 0x3600, 0x3600, 0x3600, 0xf700, 0x0000, 0xff00, 0x0000, 0x0000, 0x0000, 0x0000,  
0x0000, 0x0000, 0xff00, 0x0000, 0xf700, 0x3600, 0x3600, 0x3600, 0x3600, 0x3600, 0x3700,

[illegible]

## File

lpc\_rom8x8.c (see page 67)

### Description

This is variable rom8x8 bits.

### 1.3.12 rom8x8\_width

[illegible]

**File**

lpc\_rom8x8.c (see page 67)

**Description**

Character width data.

## 1.3.13 winfreesystem14x16\_bits

```
static UNS_16 winfreesystem14x16_bits[] = { 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x0000, 0x6000,
0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x3600, 0x3600, 0x7f00, 0x7f00, 0x3600, 0x3600, 0x6c00, 0x6c00, 0xfe00, 0xfe00, 0x6c00,
0x6c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x7e00, 0xdb00, 0xdb00, 0xd800, 0xfc00,
0x3e00, 0x1b00, 0xdb00, 0xdb00, 0x7e00, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x7060,
0xd8c0, 0xd980, 0xdb00, 0x7600, 0x0600, 0x0dc0, 0x1b60, 0x3360, 0x6360, 0xc1c0, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1c00, 0x3600, 0x2200, 0x2200, 0x3600, 0x1c00, 0x3900,
0x6d00, 0x4700, 0x6600, 0x3f00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6000, 0x6000,
0x6000, 0x6000, 0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x3000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000,
0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0xc000, 0x6000,
0x0000, 0x0000, 0x3000, 0x3000, 0xfc00, 0x3000, 0x7800, 0x4800, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800,
0x1800, 0x1800, 0xff00, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6000, 0x6000,
0x6000, 0xc000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0xf000, 0xf000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xe000, 0xe000, 0xe000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3000, 0x3000, 0x3000, 0x7000, 0x6000,
0x6000, 0x6000, 0x6000, 0xe000, 0xc000, 0xc000, 0xc000, 0x0000, 0x0000, 0x0000, 0x3c00,
0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x3c00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x7800, 0x1800, 0x1800, 0x1800, 0x1800, 0x1800,
0x1800, 0x1800, 0x1800, 0x1800, 0x0000, 0x0000, 0x0000, 0x0000, 0x3c00, 0x6600,
0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600,
0x6600, 0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6600, 0x6600,
0x6600, 0x6600, 0x7e00, 0x7e00, 0x0600, 0x0600, 0x0600, 0x0600, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x7e00, 0x6000, 0x6000, 0x6000, 0x7c00, 0x6600, 0x0600, 0x0600, 0x6600,
0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3c00, 0x6600, 0x6600, 0x6600,
0x6000, 0x7c00, 0x6600, 0x6600, 0x6600, 0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x7e00, 0x0600, 0x0600, 0x0c00, 0x0c00, 0x7e00, 0x1800, 0x1800, 0x3000, 0x3000,
0x3000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3c00, 0x6600, 0x6600, 0x6600, 0x6600,
0x3c00, 0x6600, 0x6600, 0x6600, 0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x3c00, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6000, 0x6000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x6000, 0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x6000, 0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6000, 0x6000, 0xc000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0600, 0x0600, 0x1800, 0x3000, 0x6000,
0x3000, 0x1800, 0x0c00, 0x0600, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x7e00, 0x7e00, 0x0000, 0x7e00, 0x7e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x6000, 0x3000, 0x1800, 0x0c00, 0x0600, 0x0c00,
0x1800, 0x3000, 0x6000, 0x0c00, 0x0600, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0780, 0x1ce0, 0x3870, 0x3330, 0x6798, 0x66d8, 0x6cd8, 0x6cd8, 0x6d98, 0x6798,
0x32f0, 0x3000, 0x1c70, 0x07c0, 0x0000, 0x0000, 0x0000, 0x1800, 0x1800, 0x3c00, 0x3c00,
0x2400, 0x6600, 0x7e00, 0x7e00, 0xe700, 0xc300, 0xc300, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x7f00, 0x6180, 0x6180, 0x6180, 0x6180, 0x7f00, 0x6180, 0x6180, 0x6180, 0x6180,
0x7f00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1e00, 0x3300, 0x6100, 0x6100, 0x6000,
0x6000, 0x6000, 0x6100, 0x6100, 0x3300, 0x1e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x7e00, 0x6300, 0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x6300, 0x7e00,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7f00, 0x6000, 0x6000, 0x6000, 0x6000, 0x7f00,
0x6000, 0x6000, 0x6000, 0x6000, 0x7f00, 0x0000, 0x0000, 0x0000, 0x0000, 0x7f00,
0x6000, 0x6000, 0x6000, 0x6000, 0x7f00, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x0000,
```

38

39

40

```

0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x3300, 0x1e00, 0x0000, 0x0000, 0x0000, 0x3300,
0x3300, 0x0000, 0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x6180, 0x3300,
0x1e00, 0x0000, 0x0000, 0x0000, 0x0e00, 0x0c00, 0x1800, 0xc0c0, 0xc0c0, 0x6180, 0x3300,
0x1e00, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x0c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x6000, 0x6000, 0x7e00, 0x6300, 0x6300, 0x6300, 0x6300, 0x7e00, 0x6000, 0x6000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3c00, 0x6600, 0x6600, 0x6600, 0x6c00,
0x6600, 0x6600, 0x6600, 0x6600, 0x6c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x3800,
0x1800, 0x0c00, 0x0000, 0x3c00, 0x6600, 0x1e00, 0x3600, 0x6600, 0x6600, 0x3e00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x1c00, 0x1800, 0x3000, 0x0000, 0x3c00, 0x6600, 0x1e00,
0x3600, 0x6600, 0x6600, 0x3e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3c00,
0x6600, 0x0000, 0x3c00, 0x6600, 0x1e00, 0x3600, 0x6600, 0x3e00, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x7a00, 0x5e00, 0x0000, 0x3c00, 0x6600, 0x1e00, 0x3600,
0x6600, 0x6600, 0x3e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6600, 0x6600,
0x0000, 0x3c00, 0x6600, 0x1e00, 0x3600, 0x6600, 0x6600, 0x3e00, 0x0000, 0x0000,
0x0000, 0x0000, 0x1800, 0x3c00, 0x1800, 0x0000, 0x3c00, 0x6600, 0x1e00, 0x3600, 0x6600,
0x6600, 0x3e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x3fc0, 0x6660, 0x1fe0, 0x3600, 0x6600, 0x6660, 0x3fc0, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3c00, 0x6600, 0x6000, 0x6000, 0x6000, 0x6600,
0x3c00, 0x1800, 0x0c00, 0x3800, 0x0000, 0x0000, 0x3800, 0x1800, 0x0c00, 0x0000, 0x3c00,
0x6600, 0x7e00, 0x6000, 0x6000, 0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x1c00, 0x1800, 0x3000, 0x0000, 0x3c00, 0x6600, 0x7e00, 0x6000, 0x6000, 0x6600, 0x3c00,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1800, 0x3c00, 0x6600, 0x0000, 0x3c00, 0x6600,
0x7e00, 0x6000, 0x6000, 0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x6600, 0x6600, 0x0000, 0x3c00, 0x6600, 0x7e00, 0x6000, 0x6000, 0x6600, 0x3c00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0xe000, 0x6000, 0x3000, 0x0000, 0x6000, 0x6000, 0x6000,
0x6000, 0x6000, 0x6000, 0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7000, 0x6000,
0xc000, 0x0000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x6000, 0xf000, 0x0000, 0x0000, 0x6000, 0x6000, 0x6000, 0x6000,
0x6000, 0x6000, 0x6000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x9000, 0x9000,
0x0000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x6000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x7600, 0x6c00, 0x0c00, 0x3e00, 0x6600, 0x6600, 0x6600,
0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7a00, 0x5e00, 0x0000,
0x7c00, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x3800, 0x1800, 0x0c00, 0x0000, 0x3c00, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600,
0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1c00, 0x1800, 0x3000, 0x0000, 0x3c00,
0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000,
0x1800, 0x3c00, 0x6600, 0x0000, 0x3c00, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x3c00,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x7a00, 0x5e00, 0x0000, 0x3c00, 0x6600,
0x6600, 0x6600, 0x6600, 0x6600, 0x3c00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x6600, 0x6600, 0x6600, 0x6600, 0x3c00, 0x6600, 0x6600, 0x6600, 0x6600, 0x3c00, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x3000, 0x3000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x3e00, 0x6e00, 0x6e00, 0x6600, 0x7600, 0x7600, 0x7c00, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x3800, 0x1800, 0x0c00, 0x0000, 0x6600, 0x6600, 0x6600, 0x6600,
0x6600, 0x6600, 0x3e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x1c00, 0x1800, 0x3000,
0x0000, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x3e00, 0x0000, 0x0000,
0x0000, 0x0000, 0x1800, 0x3c00, 0x6600, 0x0000, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600,
0x6600, 0x3e00, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6600, 0x6600, 0x0000,
0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x6600, 0x7c00, 0x6000, 0x6000, 0x6000, 0x0000, 0x0000,
0x0000, 0x6600, 0x6600, 0x6600, 0x0000, 0xc300, 0xc300, 0x6600, 0x6600, 0x3c00, 0x3c00, 0x1800,
0x1800, 0x3000, 0x6000, };

```

**File**

lpc\_winfreesystem14x16.c (see page 75)

**Description**

This is variable winfreesystem14x16\_bits.

## 1.3.14 winfreesystem14x16\_width

```

static UNS_8 winfreesystem14x16_width[] = { 4, 4, 6, 8, 8, 11, 9, 4, 4, 4, 6, 8, 4, 4, 4,
4, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 4, 4, 8, 8, 8, 8, 14, 8, 10, 9, 10, 9, 8, 10, 10, 4, 7, 9,
8, 12, 10, 10, 9, 10, 10, 9, 8, 10, 8, 14, 9, 10, 9, 4, 4, 4, 5, 8, 5, 8, 8, 7, 8, 8, 4, 8,

```

```
8, 4, 4, 7, 4, 12, 8, 8, 8, 8, 5, 8, 4, 8, 8, 10, 8, 8, 8, 5, 4, 5, 5, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
8, 8, 4, 8, 5, 10, 5, 7, 8, 4, 10, 8, 5, 8, 4, 4, 5, 8, 7, 4, 5, 4, 5, 7, 11, 11, 11, 8, 8,
8, 8, 8, 8, 8, 13, 9, 9, 9, 9, 9, 4, 4, 4, 4, 10, 10, 10, 10, 10, 10, 10, 8, 10, 10, 10,
10, 10, 10, 9, 8, 8, 8, 8, 8, 8, 8, 12, 7, 8, 8, 8, 8, 4, 4, 4, 4, 8, 8, 8, 8, 8, 8, 8, 6,
8, 8, 8, 8, 8, 8, 8, 8, };
```

#### File

lpc\_winfreesystem14x16.c (see page 75)

#### Description

Character width data.

## 1.3.15 x5x7\_bits

```
static UNS_16 x5x7_bits[] = { 0xf000, 0xf000, 0xf000, 0xf000, 0xf000, 0xf000, 0x0000,
0x0000, 0x2000, 0x7000, 0xf800, 0x7000, 0x2000, 0x0000, 0x5000, 0xa000, 0x5000, 0xa000,
0x5000, 0xa000, 0x0000, 0xa000, 0xe000, 0xa000, 0xa000, 0x7000, 0x2000, 0x2000, 0xc000,
0x8000, 0xc000, 0xb000, 0x2000, 0x3000, 0x2000, 0xc000, 0x8000, 0xc000, 0x6000, 0x5000,
0x6000, 0x5000, 0x8000, 0x8000, 0xc000, 0x3000, 0x2000, 0x3000, 0x2000, 0x2000, 0x5000,
0x2000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x2000, 0x7000, 0x2000, 0x0000, 0x7000, 0x0000,
0x0000, 0x9000, 0xd000, 0xb000, 0x9000, 0x2000, 0x2000, 0x3000, 0xa000, 0xa000, 0xa000,
0x4000, 0xf7000, 0x2000, 0x0000, 0x0000, 0x2000, 0x2000, 0x2000, 0xe000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0xe000, 0x2000, 0x2000, 0x2000, 0x0000, 0x0000, 0x0000, 0x3800,
0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x3800, 0x0000, 0x0000, 0x0000, 0x2000,
0x2000, 0x2000, 0xf800, 0x2000, 0x2000, 0x2000, 0x0000, 0xf800, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0xf800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0xf800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf800, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0xf800, 0x0000, 0x2000, 0x2000, 0x2000, 0x3800, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0xe000, 0x2000, 0x2000, 0x2000,
0x2000, 0x2000, 0x2000, 0xf800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf800,
0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0xe000, 0x2000,
0x2000, 0x2000, 0x2000, 0x2000, 0xf800, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xf800,
0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x1000,
0x2000, 0x4000, 0x2000, 0x1000, 0x7000, 0x0000, 0x4000, 0x2000, 0x1000, 0x2000, 0x4000,
0x7000, 0x0000, 0x0000, 0x0000, 0x7000, 0x5000, 0x5000, 0x5000, 0x0000, 0x0000, 0x1000,
0x7000, 0x2000, 0x7000, 0x4000, 0x0000, 0x0000, 0x3000, 0x4000, 0xe000, 0x4000, 0xb000,
0x0000, 0x0000, 0x0000, 0x0000, 0x2000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x0000, 0x0000, 0x0000, 0x0000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x0000,
0x5000, 0x5000, 0x5000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x5000, 0xf800, 0x5000,
0xf800, 0x5000, 0x0000, 0x0000, 0x7000, 0xa000, 0x7000, 0x2800, 0x7000, 0x0000, 0x8000,
0x9000, 0x2000, 0x4000, 0x9000, 0x1000, 0x0000, 0x0000, 0x4000, 0xa000, 0x4000, 0xa000,
0x5000, 0x0000, 0x6000, 0x4000, 0x8000, 0x0000, 0x0000, 0x0000, 0x0000, 0x2000, 0x4000,
0x4000, 0x4000, 0x4000, 0x2000, 0x0000, 0x4000, 0x2000, 0x2000, 0x2000, 0x2000, 0x2000, 0x4000,
0x0000, 0x0000, 0xa000, 0x4000, 0xe000, 0x4000, 0xa000, 0x0000, 0x0000, 0x2000, 0x2000,
0xf800, 0x2000, 0x2000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x6000, 0x4000, 0x8000,
0x0000, 0x0000, 0x0000, 0xf000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
0x6000, 0x6000, 0x0000, 0x0000, 0x1000, 0x2000, 0x4000, 0x8000, 0x0000, 0x0000, 0x4000,
0xa000, 0xa000, 0xa000, 0xa000, 0x4000, 0x0000, 0x4000, 0x0000, 0xc000, 0x4000, 0x4000, 0x4000,
0xe000, 0x0000, 0x6000, 0x9000, 0x1000, 0x2000, 0x4000, 0xf000, 0x0000, 0xf000, 0x1000,
0x6000, 0x1000, 0x9000, 0x6000, 0x0000, 0x2000, 0x6000, 0xa000, 0xf000, 0x2000, 0x2000,
0x0000, 0xf000, 0x8000, 0xe000, 0x1000, 0x9000, 0x6000, 0x0000, 0x6000, 0x8000, 0xe000,
0x9000, 0xf000, 0x6000, 0x0000, 0xf000, 0x0000, 0xf000, 0x0000, 0x4000, 0x0000, 0x4000,
0x6000, 0x9000, 0x8000, 0x8000, 0x9000, 0xe000, 0x9000, 0xe000, 0x9000, 0x9000, 0xe000, 0x0000,
0x6000, 0x9000, 0x8000, 0x8000, 0x9000, 0x6000, 0x0000, 0xe000, 0x9000, 0x9000, 0x9000,
0x9000, 0xe000, 0x0000, 0xf000, 0x8000, 0xe000, 0x8000, 0xf000, 0x8000, 0xf000, 0x0000, 0xf000,
0x8000, 0xe000, 0x8000, 0x8000, 0x8000, 0x0000, 0x6000, 0x9000, 0x8000, 0xb000, 0x9000,
0x7000, 0x0000, 0x9000, 0x9000, 0xf000, 0x9000, 0x9000, 0x9000, 0x0000, 0xe000, 0x4000,
0x4000, 0x4000, 0x4000, 0xe000, 0x0000, 0x1000, 0x1000, 0x1000, 0x1000, 0x9000, 0x6000,
0x0000, 0x9000, 0xa000, 0xc000, 0xc000, 0xa000, 0x9000, 0x0000, 0x8000, 0x8000, 0x8000,
0x8000, 0x8000, 0xf000, 0x0000, 0x9000, 0xf000, 0xf000, 0x9000, 0x9000, 0x9000, 0x0000,
0x9000, 0xd000, 0xd000, 0xb000, 0xb000, 0x9000, 0x0000, 0x6000, 0x9000, 0x9000, 0x9000,
```





44

45



---

## 1.4.3 \_ERROR

```
#define _ERROR (INT_32)(-1)
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

ERROR macro

---

## 1.4.4 \_NO\_ERROR

```
#define _NO_ERROR (INT_32)(0)
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

NO\_ERROR macro

---

## 1.4.5 \_SBF

```
#define _SBF(f,v) (((UNS_32)(v)) << (f))
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

Set bit field macro

---

## 1.4.6 BLACK

```
#define BLACK 0x00
```

**File**

lpc\_colors.h ([see page 63](#))

**Description**

Black color, 323 mode

---

## 1.4.7 BLUE

```
#define BLUE 0x03
```

---

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

Blue color, 323 mode

---

## 1.4.8 BLUE\_COLORS

```
#define BLUE_COLORS 0x08
```

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

Number of blue colors in 332 mode

---

## 1.4.9 BLUEMASK

```
#define BLUEMASK 0x3
```

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

Blue color mask, 323 mode

---

## 1.4.10 BLUESHIFT

```
#define BLUESHIFT 0
```

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

Blue shift value, 323 mode

---

## 1.4.11 COLORS\_DEF

```
#define COLORS_DEF 16
```

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

16-bit 565 color mode #define COLORS\_DEF 15 /\* 15-bit 555 color mode \*/ #define COLORS\_DEF 12 /\* 12-bit 444 color mode \*/

---

```
#define COLORS_DEF 8 /* 8-bit color mode
```

---

## 1.4.12 CYAN

```
#define CYAN (GREEN | BLUE)
```

### File

lpc\_colors.h ([see page 63](#))

### Description

Cyan color, 323 mode

## 1.4.13 DARKGRAY

```
#define DARKGRAY 0x25
```

### File

lpc\_colors.h ([see page 63](#))

### Description

Dark gray color, 323 mode

## 1.4.14 EXTERN

```
#define EXTERN extern
```

### File

lpc\_types.h ([see page 73](#))

### Description

External data/function define

## 1.4.15 FALSE

```
#define FALSE (0==1)
```

### File

lpc\_types.h ([see page 73](#))

### Description

FALSE macro

## 1.4.16 GREEN

```
#define GREEN 0x1C
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Green color, 323 mode

---

## 1.4.17 GREEN\_COLORS

```
#define GREEN_COLORS 0x08
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Number of green colors in 332 mode

---

## 1.4.18 GREENMASK

```
#define GREENMASK 0x1C
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Green color mask, 323 mode

---

## 1.4.19 GREENSHIFT

```
#define GREENSHIFT 2
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Green shift value, 323 mode

---

## 1.4.20 LIGHTBLUE

```
#define LIGHTBLUE 0x01
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Light blue color, 323 mode

---



---

## 1.4.21 LIGHTCYAN

```
#define LIGHTCYAN (LIGHTGREEN | LIGHTBLUE)
```

### File

lpc\_colors.h (see page 63)

### Description

Light cyan color, 323 mode

---

## 1.4.22 LIGHTGRAY

```
#define LIGHTGRAY 0x6E
```

### File

lpc\_colors.h (see page 63)

### Description

Light gray color, 323 mode

---

## 1.4.23 LIGHTGREEN

```
#define LIGHTGREEN 0x0C
```

### File

lpc\_colors.h (see page 63)

### Description

Light green color, 323 mode

---

## 1.4.24 LIGHTMAGENTA

```
#define LIGHTMAGENTA (LIGHTRED | LIGHTBLUE)
```

### File

lpc\_colors.h (see page 63)

### Description

Light magenta color, 323 mode

---

## 1.4.25 LIGHTRED

```
#define LIGHTRED 0x60
```

---

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

Light red color, 323 mode

---

## 1.4.26 LIGHTYELLOW

```
#define LIGHTYELLOW (LIGHTRED | LIGHTGREEN)
```

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

Light yellow color, 323 mode

---

## 1.4.27 LPC\_BAD\_CLK

```
#define LPC_BAD_CLK (INT_32)(-9)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Bad device clock macro

---

## 1.4.28 LPC\_BAD\_HANDLE

```
#define LPC_BAD_HANDLE (INT_32)(-8)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Bad device handle macro

---

## 1.4.29 LPC\_BAD\_PARAMS

```
#define LPC_BAD_PARAMS (INT_32)(-7)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device bad paramaters macro

---

---

## 1.4.30 LPC\_CANT\_START

```
#define LPC_CANT_START (INT_32)(-10)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device can't start macro

---

## 1.4.31 LPC\_CANT\_STOP

```
#define LPC_CANT_STOP (INT_32)(-11)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device can't stop macro

---

## 1.4.32 LPC\_COLOR\_TYPES\_H

```
#define LPC_COLOR_TYPES_H
```

**File**

lpc\_colors.h ([↗](#) see page 63)

**Description**

This is macro LPC\_COLOR\_TYPES\_H.

---

## 1.4.33 LPC\_DEV\_UNKNOWN

```
#define LPC_DEV_UNKNOWN (INT_32)(-2)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device unknown macro

---

## 1.4.34 LPC\_FONTS\_H

```
#define LPC_FONTS_H
```

---

**File**

lpc\_fonts.h ([↗](#) see page 64)

**Description**

This is macro LPC\_FONTS\_H.

---

## 1.4.35 LPC\_HEVR10\_FONT\_H

```
#define LPC_HEVR10_FONT_H
```

**File**

lpc\_helvr10.h ([↗](#) see page 65)

**Description**

This is macro LPC\_HEVR10\_FONT\_H.

---

## 1.4.36 LPC\_IN\_USE

```
#define LPC_IN_USE (INT_32)(-5)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device in use macro

---

## 1.4.37 LPC\_NOT\_OPEN

```
#define LPC_NOT_OPEN (INT_32)(-4)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device not open macro

---

## 1.4.38 LPC\_NOT\_SUPPORTED

```
#define LPC_NOT_SUPPORTED (INT_32)(-3)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device not supported macro

---

---

## 1.4.39 LPC\_PIN\_CONFLICT

```
#define LPC_PIN_CONFLICT (INT_32)(-6)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Device oin conflict macro

---

## 1.4.40 LPC\_ROM8X16\_FONT\_H

```
#define LPC_ROM8X16_FONT_H
```

**File**

lpc\_rom8x16.h ([↗](#) see page 66)

**Description**

This is macro LPC\_ROM8X16\_FONT\_H.

---

## 1.4.41 LPC\_ROM8X8\_FONT\_H

```
#define LPC_ROM8X8_FONT_H
```

**File**

lpc\_rom8x8.h ([↗](#) see page 67)

**Description**

This is macro LPC\_ROM8X8\_FONT\_H.

---

## 1.4.42 LPC\_SWIM\_FONT\_H

```
#define LPC_SWIM_FONT_H
```

**File**

lpc\_swim\_font.h ([↗](#) see page 70)

**Description**

This is macro LPC\_SWIM\_FONT\_H.

---

## 1.4.43 LPC\_SWIM\_H

```
#define LPC_SWIM_H
```

---

**File**

lpc\_swim.h ([↗](#) see page 68)

**Description**

This is macro LPC\_SWIM\_H.

---

## 1.4.44 LPC\_SWIM\_IMAGE\_H

```
#define LPC_SWIM_IMAGE_H
```

**File**

lpc\_swim\_image.h ([↗](#) see page 72)

**Description**

This is macro LPC\_SWIM\_IMAGE\_H.

---

## 1.4.45 LPC\_TYPES\_H

```
#define LPC_TYPES_H
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

This is macro LPC\_TYPES\_H.

---

## 1.4.46 LPC\_WINFREESYS\_14X16\_FONT\_H

```
#define LPC_WINFREESYS_14X16_FONT_H
```

**File**

lpc\_winfreesystem14x16.h ([↗](#) see page 75)

**Description**

This is macro LPC\_WINFREESYS\_14X16\_FONT\_H.

---

## 1.4.47 LPC\_X5X7\_FONT\_H

```
#define LPC_X5X7_FONT_H
```

**File**

lpc\_x5x7.h ([↗](#) see page 76)

**Description**

This is macro LPC\_X5X7\_FONT\_H.

---

---

## 1.4.48 LPC\_X6X13\_FONT\_H

```
#define LPC_X6X13_FONT_H
```

### File

lpc\_x6x13.h (see page 77)

### Description

This is macro LPC\_X6X13\_FONT\_H.

---

## 1.4.49 MAGENTA

```
#define MAGENTA (RED | BLUE)
```

### File

lpc\_colors.h (see page 63)

### Description

Magenta color, 323 mode

---

## 1.4.50 NELEMENTS

```
#define NELEMENTS(array) (sizeof (array) / sizeof (array[0]))
```

### File

lpc\_types.h (see page 73)

### Description

Number of elements in an array

---

## 1.4.51 NULL

```
#define NULL ((void*) 0)
```

### File

lpc\_types.h (see page 73)

### Description

NULL pointer

---

## 1.4.52 NUM\_COLORS

```
#define NUM_COLORS 256
```

---

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Number of colors in 332 mode

---

## 1.4.53 RED

```
#define RED 0xE0
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Red color, 323 mode

---

## 1.4.54 RED\_COLORS

```
#define RED_COLORS 0x08
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Number of red colors in 332 mode

---

## 1.4.55 REDMASK

```
#define REDMASK 0xE0
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Red color mask, 323 mode

---

## 1.4.56 REDSHIFT

```
#define REDSHIFT 5
```

**File**

lpc\_colors.h (🔗 see page 63)

**Description**

Red shift value, 323 mode

---



---

## 1.4.57 SMA\_BAD\_CLK

```
#define SMA_BAD_CLK LPC_BAD_CLK
```

### File

lpc\_types.h ([see page 73](#))

### Description

This is macro SMA\_BAD\_CLK.

---

## 1.4.58 SMA\_BAD\_HANDLE

```
#define SMA_BAD_HANDLE LPC_BAD_HANDLE
```

### File

lpc\_types.h ([see page 73](#))

### Description

This is macro SMA\_BAD\_HANDLE.

---

## 1.4.59 SMA\_BAD\_PARAMS

```
#define SMA_BAD_PARAMS LPC_BAD_PARAMS
```

### File

lpc\_types.h ([see page 73](#))

### Description

This is macro SMA\_BAD\_PARAMS.

---

## 1.4.60 SMA\_CANT\_START

```
#define SMA_CANT_START LPC_CANT_START
```

### File

lpc\_types.h ([see page 73](#))

### Description

This is macro SMA\_CANT\_START.

---

## 1.4.61 SMA\_CANT\_STOP

```
#define SMA_CANT_STOP LPC_CANT_STOP
```

---

**File**

lpc\_types.h ([see page 73](#))

**Description**

This is macro SMA\_CANT\_STOP.

---

## 1.4.62 SMA\_DEV\_UNKNOWN

```
#define SMA_DEV_UNKNOWN LPC_DEV_UNKNOWN
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

following are legacy defines which are OBSOLETE. DONOT USE.

---

## 1.4.63 SMA\_IN\_USE

```
#define SMA_IN_USE LPC_IN_USE
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

This is macro SMA\_IN\_USE.

---

## 1.4.64 SMA\_NOT\_OPEN

```
#define SMA_NOT_OPEN LPC_NOT_OPEN
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

This is macro SMA\_NOT\_OPEN.

---

## 1.4.65 SMA\_NOT\_SUPPORTED

```
#define SMA_NOT_SUPPORTED LPC_NOT_SUPPORTED
```

**File**

lpc\_types.h ([see page 73](#))

**Description**

This is macro SMA\_NOT\_SUPPORTED.

---

---

## 1.4.66 SMA\_PIN\_CONFLICT

```
#define SMA_PIN_CONFLICT LPC_PIN_CONFLICT
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

This is macro SMA\_PIN\_CONFLICT.

---

## 1.4.67 STATIC

```
#define STATIC static
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

Static data/function define

---

## 1.4.68 SUCCESS

```
#define SUCCESS 0
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

SUCCESS macro

---

## 1.4.69 TRUE

```
#define TRUE (!FALSE)
```

**File**

lpc\_types.h ([↗](#) see page 73)

**Description**

TRUE macro

---

## 1.4.70 WHITE

```
#define WHITE 0xFF
```

---

**File**

lpc\_colors.h (see page 63)

**Description**

White color, 323 mode

---

## 1.4.71 YELLOW

```
#define YELLOW (RED | GREEN)
```

**File**

lpc\_colors.h (see page 63)

**Description**

Yellow color, 323 mode

---

## 1.5 Files

---

### 1.5.1 lpc\_colors.c

- \$Id:: lpc\_colors.c 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Color definitions

\*

- Description:
- See the SMA\_colors.h header file for a description of this
- package.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## 1.5.2 lpc\_colors.h

- \$Id:: lpc\_colors.h 745 2008-05-13 19:59:29Z pdurgesh \$
- 
- Project: Color definitions
- 
- Description:
- This package contains functions for color mapping, color
- conversion, and common defines.
- 
- The palette table function can be configured for 555 or 565
- color.
- 
- Notes:
- Color entries are stored in BGR format, with blue mapped to the
- most significant bits of a color type.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

### Functions

Function	Description
lpc_colors_set_palette ( <a href="#">see page 1</a> )	Generate a palette table (only in 8-bit mode). If compiled in 16-bit color mode, this will be a NULL ( <a href="#">see page 57</a> ) function. Function: lpc_colors_set_palette Purpose: Generate a palette table (only in 8-bit mode). Processing: Depending on the target LCD color mapping (either 555 or 565), a palette table will be generated to convert colors stored in 233 format to either 555 or 565 format through a lookup table.

### Macros

Macro	Description
BLACK ( <a href="#">see page 47</a> )	Black color, 323 mode
BLUE ( <a href="#">see page 47</a> )	Blue color, 323 mode
BLUE_COLORS ( <a href="#">see page 48</a> )	Number of blue colors in 332 mode
BLUEMASK ( <a href="#">see page 48</a> )	Blue color mask, 323 mode
BLUESHIFT ( <a href="#">see page 48</a> )	Blue shift value, 323 mode
COLORS_DEF ( <a href="#">see page 48</a> )	16-bit 565 color mode #define COLORS_DEF 15 /* 15-bit 555 color mode */ #define COLORS_DEF 12 /* 12-bit 444 color mode */
CYAN ( <a href="#">see page 49</a> )	Cyan color, 323 mode
DARKGRAY ( <a href="#">see page 49</a> )	Dark gray color, 323 mode
GREEN ( <a href="#">see page 49</a> )	Green color, 323 mode

GREEN_COLORS (see page 50)	Number of green colors in 332 mode
GREENMASK (see page 50)	Green color mask, 323 mode
GREENSHIFT (see page 50)	Green shift value, 323 mode
LIGHTBLUE (see page 50)	Light blue color, 323 mode
LIGHTCYAN (see page 51)	Light cyan color, 323 mode
LIGHTGRAY (see page 51)	Light gray color, 323 mode
LIGHTGREEN (see page 51)	Light green color, 323 mode
LIGHTMAGENTA (see page 51)	Light magenta color, 323 mode
LIGHTRED (see page 51)	Light red color, 323 mode
LIGHTYELLOW (see page 52)	Light yellow color, 323 mode
LPC_COLOR_TYPES_H (see page 53)	This is macro LPC_COLOR_TYPES_H.
MAGENTA (see page 57)	Magenta color, 323 mode
NUM_COLORS (see page 57)	Number of colors in 332 mode
RED (see page 58)	Red color, 323 mode
RED_COLORS (see page 58)	Number of red colors in 332 mode
REDMASK (see page 58)	Red color mask, 323 mode
REDSHIFT (see page 58)	Red shift value, 323 mode
WHITE (see page 61)	White color, 323 mode
YELLOW (see page 62)	Yellow color, 323 mode

### Types

Type	Description
COLOR_T (see page 20)	Color type is a 8-bit value

## 1.5.3 lpc\_fonts.c

- \$Id:: lpc\_fonts.c 745 2008-05-13 19:59:29Z pdurgesh \$
- Project: Fonts selection
- Description:
- This package provides a common font information structure.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## 1.5.4 lpc\_fonts.h

- \$Id:: lpc\_fonts.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Fonts selection

\*

- Description:
- This package provides a common font information structure.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

#### Macros

Macro	Description
LPC_FONTS_H (see page 53)	This is macro LPC_FONTS_H.

#### Types

Type	Description
FONT_T (see page 20)	Font data structure

## 1.5.5 lpc\_helvr10.c

\$Id:: lpc\_helvr10.c 745 2008-05-13 19:59:29Z pdurgesh \$ Generated by convbdf on Tue Oct 3 00:24:24 MDT 2000. Font information:

name: -Adobe-Helvetica-Medium-R-Normal--10-100-75-75-P-56-ISO8859-1 pixel size: 10 ascent: 10 descent: 2

#### Variables

Variable	Description
font_helvr10 (see page 24)	Externally available font information structure
helvr10_bits (see page 26)	Font character bitmap data.
helvR10_width (see page 27)	Character width data.

## 1.5.6 lpc\_helvr10.h

- \$Id:: lpc\_helvr10.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Helvetica 10-point proportional font

\*

- Description:
- This package provides bit information for a font type.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## Macros

Macro	Description
LPC_HEVR10_FONT_H (see page 54)	This is macro LPC_HEVR10_FONT_H.

## 1.5.7 lpc\_rom8x16.c

\$Id:: lpc\_rom8x16.c 745 2008-05-13 19:59:29Z pdurgesh \$ Generated by convrom.exe ROM 8x16 Font bios mode 12

## Variables

Variable	Description
font_rom8x16 (see page 24)	Externally available font information structure
rom8x16_bits (see page 28)	This is variable rom8x16_bits.
rom8x16_width (see page 33)	Character width data.

## 1.5.8 lpc\_rom8x16.h

- \$Id:: lpc\_rom8x16.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: 8x16 proportional font

\*

- Description:
- This package provides bit information for a font type.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.



**Macros**

Macro	Description
LPC_ROM8X16_FONT_H (see page 55)	This is macro LPC_ROM8X16_FONT_H.

## 1.5.9 lpc\_rom8x8.c

\$Id:: lpc\_rom8x8.c 745 2008-05-13 19:59:29Z pdurgesh \$ Generated by convrom.exe ROM 8x8 Font bios mode 10

**Variables**

Variable	Description
font_rom8x8 (see page 25)	Externally available font information structure
rom8x8_bits (see page 33)	This is variable rom8x8_bits.
rom8x8_width (see page 36)	Character width data.

## 1.5.10 lpc\_rom8x8.h

• \$Id:: lpc\_rom8x8.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

• Project: 8x8 proportional font

\*

• Description:

• This package provides bit information for a font type.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

**Macros**

Macro	Description
LPC_ROM8X8_FONT_H (see page 55)	This is macro LPC_ROM8X8_FONT_H.

## 1.5.11 lpc\_swim.c

• \$Id:: lpc\_swim.c 745 2008-05-13 19:59:29Z pdurgesh \$

\*

• Project: Simple Windowing Interface Manager (SWIM)

\*

- Description:
- See the swim.h header file for a description of this package.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

---

## 1.5.12 lpc\_swim.h

- \$Id:: lpc\_swim.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Simple Windowing Interface Manager (SWIM)

\*

- Description:
- This package provides a simple windows manager that provides the
- following functions:
- Windows initialization and validity checks
- Must be in physical display space
- Color support for background, primary pen, and fill
- Simple graphics primitives (pixels, lines, boxes)
- Window deallocation

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## Functions

Function	Description
swim_clear_screen (🔗 see page 2)	Fills the draw area of the display with the selected color Function: swim_clear_screen Purpose: Fills the draw area of the display with the selected color Processing: Loop through all virtual window (draw area) locations and updates them with the passed color value.
swim_get_horizontal_size (🔗 see page 3)	Get the virtual window horizontal size Function: swim_get_horizontal_size Purpose: Get the virtual window horizontal size Processing: For the passed window ID, return the x size of the window.
swim_get_vertical_size (🔗 see page 3)	Get the virtual window vertical size Function: swim_get_vertical_size Purpose: Get the virtual window vertical size Processing: For the passed window ID, return the x size of the window.
swim_put_box (🔗 see page 4)	Place a box with corners (X1, Y1) and (X2, Y2). Use pen color for edges and fill color for center Function: swim_put_box Purpose: Place a box with corners (X1, Y1) and (X2, Y2) Processing: See function.
swim_put_diamond (🔗 see page 5)	Draw a diamond in the virtual window Function: swim_put_diamond Purpose: Purpose: Draw a diamond in the virtual window Processing: See function.
swim_put_line (🔗 see page 7)	Draw a line in the virtual window Function: swim_put_line Purpose: Draw a line in the virtual window with clipping. Processing: See function.
swim_put_pixel (🔗 see page 9)	Puts a pixel at (X, Y) in the pen color Function: swim_put_pixel Purpose: Puts a pixel at the virtual X, Y coordinate in the window Processing: Convert the virtual pixel position to a physical position. If the pixel is inside the window draw area, update the pixel on the display.
swim_set_bkg_color (🔗 see page 13)	Set background color Function: swim_set_bkg_color Purpose: Sets the color used for backgrounds Processing: For the passed window ID, update to the passed background color.
swim_set_fill_color (🔗 see page 14)	Set fill color (used for boxes and circles) Function: swim_set_fill_color Purpose: Sets the fill color Processing: For the passed window ID, update to the passed fill color.
swim_set_pen_color (🔗 see page 15)	Set the pen color Function: swim_set_pen_color Purpose: Sets the pen color Processing: For the passed window ID, update to the passed pen color.
swim_window_close (🔗 see page 17)	Destroy a window Function: swim_window_close Purpose: Reallocates a window for use Processing: For the passed window ID, clear the window used flag.
swim_window_open (🔗 see page 17)	Initialize a window Function: swim_window_open Purpose: Initializes a window and the default values for the window Processing: See function.
swim_window_open_noclear (🔗 see page 18)	Initialize a window without clearing it Function: swim_window_open_noclear Purpose: Initializes a window and the default values for the window Processing: See function.

## Macros

Macro	Description
LPC_SWIM_H (🔗 see page 55)	This is macro LPC_SWIM_H.

## Types

Type	Description
SWIM_WINDOW_T (🔗 see page 22)	Structure is used to store information about a specific window

---

## 1.5.13 lpc\_swim\_font.c

- \$Id:: lpc\_swim\_font.c 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Font management for SWIM

\*

- Description:
- See the sma\_swim\_font.h header file for a description of this
- package.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

---

## 1.5.14 lpc\_swim\_font.h

- \$Id:: lpc\_swim\_font.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Font management for SWIM

\*

- Description:
- This package provides the following font capabilities with SWIM:
- Font selection
- Text positioning
- newline and window scrolling
- Text display with multiple, selectable fonts

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without

- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## Functions

Function	Description
swim_get_font_height (see page 2)	Returns the active font's height in pixels Function: swim_get_font_height Purpose: Returns the active font's height in pixels Processing: See function.
swim_get_xy (see page 4)	Returns the X, Y pixel coordinates for the next text operation Function: swim_get_xy Purpose: Returns the X, Y pixel coordinates for the next text operation Processing: The logical X and Y positions are computed by subtracting the physical text position values by the physical minimum window limits.
swim_put_char (see page 5)	Puts a single character to the window Function: swim_put_char Purpose: Puts a character in the window. Processing: See function.
swim_put_ltext (see page 8)	Puts a null-terminated string of text in a window, but will move an entire word to the next line if it will not fit on the present line Function: swim_put_ltext Purpose: Puts a string of text in a window, but will adjust the position of a word if the word length exceeds the edge of the display. Processing: While the string has data in it, check for the newline character. If it exists, output a newline. If the string data is inside the font character table, output the first word in the string (with support for generating a newline if the word will exceed the window edge). Continue until all words/characters are output.
swim_put_newline (see page 8)	Puts a newline in the window Function: swim_put_newline Purpose: Performs a newline in a window Processing: Set the text pointer for the next text character operation to the beginning of the following line. If the following line exceeds the window size, perform a line scroll.
swim_put_text (see page 12)	Puts a null-terminated string of text in a window Function: swim_put_text Purpose: Puts a string of text in a window Processing: Each character will be routed to the swim_put_char (see page 5) function until a string terminator is reached. For newline characters, a newline will occur instead of a character output.
swim_put_text_xy (see page 12)	Put a text message at an X, Y pixel coordinate in the window Function: swim_put_text_xy Purpose: Put text at x, y (char) position on screen Processing: Set the virtual (upper left) text position in the window and render the text string at this position.
swim_set_font (see page 14)	Select the active font Function: swim_set_font Purpose: Sets the active font Processing: Switch to the selected font by setting the font structure pointer in the windows structure based on the passed enumeration. If the next character output in the new font will exceed the window limit, perform a window text scroll.
swim_set_font_transparency (see page 15)	Enables and disables font backgrounds Function: swim_set_font_transparency Purpose: Enables and disables font backgrounds. When set, the font background will not be drawn in the background color (useful for painting text over pictures). Processing: See function.
swim_set_title (see page 16)	Create a title bar Function: swim_set_title Purpose: Creates a title bar in the window and adjusts the client area to be outside the title bar area. Processing: See function.
swim_set_xy (see page 16)	Sets the X, Y pixel coordinates for the next text operation Function: swim_set_xy Purpose: Sets the X, Y pixel coordinates for the next text operation Processing: Update the X, Y text position pointers, limiting the position to the window dimensions.

## Macros

Macro	Description
LPC_SWIM_FONT_H (see page 55)	This is macro LPC_SWIM_FONT_H.

---

## 1.5.15 lpc\_swim\_image.c

- \$Id:: lpc\_swim\_image.c 745 2008-05-13 19:59:29Z pdurgesh \$
- \*
- Project: Image management for SWIM
- \*
- Description:
- See the swim.h header file for a description of this package.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

---

## 1.5.16 lpc\_swim\_image.h

- \$Id:: lpc\_swim\_image.h 745 2008-05-13 19:59:29Z pdurgesh \$
- \*
- Project: Image management for SWIM
- \*
- Description:
- This package provides the following image capabilities with SWIM:
- Display of raw image data (stored left to right, top to
- bottom)
- Stored raw images MUST be stored in the same color format as
- color\_type
- Image scaling, rotation, and clipping

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without

- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## Functions

Function	Description
swim_put_image (see page 6)	Puts a raw image into a window Function: swim_put_image Purpose: Puts an raw image in a window unscaled, clips off edges Processing: See function.
swim_put_invert_image (see page 6)	Puts a raw image into a window inverted Function: swim_put_invert_image Purpose: Puts an raw image in a window unscaled, inverted, with clipped edges. Processing: See function.
swim_put_left_image (see page 7)	Puts a raw image into a window rotated left Function: swim_put_left_image Purpose: Puts an raw image in a window unscaled, rotated left, with clipped edges. Processing: See function.
swim_put_right_image (see page 9)	Puts a raw image into a window rotated right Function: swim_put_right_image Purpose: Puts an raw image in a window unscaled, rotated right, with clipped edges. Processing: See function.
swim_put_scale_image (see page 10)	Puts and scales a raw image into a window Function: swim_put_scale_image Purpose: Puts an raw image in a window scaled. Processing: See function.
swim_put_scale_invert_image (see page 10)	Puts and scales a raw image into a window inverted Function: swim_put_scale_invert_image Purpose: Puts an raw image in a window scaled and inverted. Processing: See function.
swim_put_scale_left_image (see page 11)	Puts and scales a raw image into a window rotated left Function: swim_put_scale_left_image Purpose: Puts an raw image in a window scaled and rotated left. Processing: See function.
swim_put_scale_right_image (see page 11)	Puts and scales a raw image into a window rotated right Function: swim_put_scale_right_image Purpose: Puts an raw image in a window scaled and rotated right. Processing: See function.
swim_put_win_image (see page 13)	One API for all the functions Function: swim_put_win_image Purpose: This function simply provides a single API for all the image functions. Processing: See function.

## Macros

Macro	Description
LPC_SWIM_IMAGE_H (see page 56)	This is macro LPC_SWIM_IMAGE_H.

## Types

Type	Description
SWIM_ROTATION_T (see page 22)	Image rotation tags

# 1.5.17 lpc\_types.h

- \$Id:: lpc\_types.h 847 2008-06-27 21:23:43Z wellsk \$

\*

- Project: Common Include Files

\*

- Description:
- lpc\_types.h contains the NXP ABL typedefs for C standard types.
- It is intended to be used in ISO C conforming development environments and checks for this insofar as it is possible
- to do so.
- \*
- lpc\_types.h ensures that the name used to define types correctly
- identifies a representation size, and by direct inference the
- storage size, in bits. E.g., UNS\_32 (see page 23) identifies an unsigned
- integer type stored in 32 bits.
- \*
- It requires that the basic storage unit (char) be stored in
- 8 bits.
- \*
- No assumptions about Endianness are made or implied.
- \*
- lpc\_types.h also contains NXP ABL Global Macros:
- \_BIT (see page 46)
- \_SBF (see page 47)
- \_BITMAP
- These #defines are not strictly types, but rather Preprocessor
- Macros that have been found to be generally useful.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## Macros

Macro	Description
_BIT (see page 46)	Set bit macro
_BITMASK (see page 46)	Bitmask creation macro
_ERROR (see page 47)	ERROR macro
_NO_ERROR (see page 47)	NO_ERROR macro
_SBF (see page 47)	Set bit field macro
EXTERN (see page 49)	External data/function define
FALSE (see page 49)	FALSE macro
LPC_BAD_CLK (see page 52)	Bad device clock macro
LPC_BAD_HANDLE (see page 52)	Bad device handle macro
LPC_BAD_PARAMS (see page 52)	Device bad parameters macro
LPC_CANT_START (see page 53)	Device can't start macro
LPC_CANT_STOP (see page 53)	Device can't stop macro
LPC_DEV_UNKNOWN (see page 53)	Device unknown macro



LPC_IN_USE (↗ see page 54)	Device in use macro
LPC_NOT_OPEN (↗ see page 54)	Device not open macro
LPC_NOT_SUPPORTED (↗ see page 54)	Device not supported macro
LPC_PIN_CONFLICT (↗ see page 55)	Device oin conflict macro
LPC_TYPES_H (↗ see page 56)	This is macro LPC_TYPES_H.
NELEMENTS (↗ see page 57)	Number of elements in an array
NULL (↗ see page 57)	NULL pointer
SMA_BAD_CLK (↗ see page 59)	This is macro SMA_BAD_CLK.
SMA_BAD_HANDLE (↗ see page 59)	This is macro SMA_BAD_HANDLE.
SMA_BAD_PARAMS (↗ see page 59)	This is macro SMA_BAD_PARAMS.
SMA_CANT_START (↗ see page 59)	This is macro SMA_CANT_START.
SMA_CANT_STOP (↗ see page 59)	This is macro SMA_CANT_STOP.
SMA_DEV_UNKNOWN (↗ see page 60)	following are legacy defines which are OBSELETE. DONOT USE.
SMA_IN_USE (↗ see page 60)	This is macro SMA_IN_USE.
SMA_NOT_OPEN (↗ see page 60)	This is macro SMA_NOT_OPEN.
SMA_NOT_SUPPORTED (↗ see page 60)	This is macro SMA_NOT_SUPPORTED.
SMA_PIN_CONFLICT (↗ see page 61)	This is macro SMA_PIN_CONFLICT.
STATIC (↗ see page 61)	Static data/function define
SUCCESS (↗ see page 61)	SUCCESS macro
TRUE (↗ see page 61)	TRUE macro

## Types

Type	Description
BOOL_16 (↗ see page 19)	16 bit boolean type
BOOL_32 (↗ see page 19)	32 bit boolean type
BOOL_8 (↗ see page 19)	8 bit boolean type
CHAR (↗ see page 20)	SMA type for character type
INT_16 (↗ see page 20)	SMA type for 16 bit signed value
INT_32 (↗ see page 21)	SMA type for 32 bit signed value
INT_64 (↗ see page 21)	SMA type for 64 bit signed value
INT_8 (↗ see page 21)	SMA type for 8 bit signed value
PFI (↗ see page 21)	Pointer to Function returning INT_32 (↗ see page 21) (any number of parameters)
PFV (↗ see page 21)	Pointer to Function returning Void (any number of parameters)
STATUS (↗ see page 22)	Status type
UNS_16 (↗ see page 23)	SMA type for 16 bit unsigned value
UNS_32 (↗ see page 23)	SMA type for 32 bit unsigned value
UNS_64 (↗ see page 24)	SMA type for 64 bit unsigned value
UNS_8 (↗ see page 24)	SMA type for 8 bit unsigned value

## 1.5.18 lpc\_winfreesystem14x16.c

\$Id:: lpc\_winfreesystem14x16.c 745 2008-05-13 19:59:29Z pdurgesh \$ Generated by convfnt.exe Windows FreeSystem 14x16 Font

### Variables

Variable	Description
font_winfreesys14x16 (↗ see page 25)	Externally available font information structure
winfreesystem14x16_bits (↗ see page 37)	This is variable winfreesystem14x16_bits.
winfreesystem14x16_width (↗ see page 41)	Character width data.

## 1.5.19 lpc\_winfreesystem14x16.h

- \$Id:: lpc\_winfreesystem14x16.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Windows FreeSystem 14x16 Font

\*

- Description:
- This package provides bit information for a font type.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

#### Macros

Macro	Description
LPC_WINFREESYS_14X16_FONT_H (see page 56)	This is macro LPC_WINFREESYS_14X16_FONT_H.

## 1.5.20 lpc\_x5x7.c

\$Id:: lpc\_x5x7.c 745 2008-05-13 19:59:29Z pdurgesh \$ Generated by convbdf on Tue Oct 3 00:24:24 MDT 2000. Font information:

name: "-Misc-Fixed-Medium-R-Normal--7-70-75-75-C-50-ISO8859-1" pixel size: 7 ascent: 6 descent: 1

#### Variables

Variable	Description
font_x5x7 (see page 25)	Externally available font information structure
x5x7_bits (see page 42)	Font character bitmap data.
x5x7_width (see page 43)	Character width data.

## 1.5.21 lpc\_x5x7.h

- \$Id:: lpc\_x5x7.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Fixed 5x7 proportional font

\*

- Description:
- This package provides bit information for a font type.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the

- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

## Macros

Macro	Description
LPC_X5X7_FONT_H (see page 56)	This is macro LPC_X5X7_FONT_H.

## 1.5.22 lpc\_x6x13.c

\$Id:: lpc\_x6x13.c 745 2008-05-13 19:59:29Z pdurgesh \$ Generated by convbdf on Tue Oct 3 00:24:25 MDT 2000. Font information:

name: "-Misc-Fixed-Medium-R-SemiCondensed--13-120-75-75-C-60-ISO8859-1" pixel size: 13 ascent: 11 descent: 2

## Variables

Variable	Description
font_x6x13 (see page 25)	Externally available font information structure
x6x13_bits (see page 43)	Font character bitmap data.
x6x13_width (see page 46)	Character width data.

## 1.5.23 lpc\_x6x13.h

\$Id:: lpc\_x6x13.h 745 2008-05-13 19:59:29Z pdurgesh \$

\*

- Project: Fixed 6x13 proportional font

\*

- Description:
- This package provides bit information for a font type.

\*\*\*\*\*

- Software that is described herein is for illustrative purposes only
- which provides customers with programming information regarding the
- products. This software is supplied "AS IS" without any warranties.
- NXP Semiconductors assumes no responsibility or liability for the
- use of the software, conveys no license or title under any patent,
- copyright, or mask work right to the product. NXP Semiconductors
- reserves the right to make changes in the software without
- notification. NXP Semiconductors also make no representation or
- warranty that such application will be suitable for the specified
- use without further testing or modification.

**Macros**

Macro	Description
LPC_X6X13_FONT_H ( <a href="#">↗</a> see page 57)	This is macro LPC_X6X13_FONT_H.

# Index

—  
 \_BIT 46  
 \_BITMASK 46  
 \_ERROR 47  
 \_NO\_ERROR 47  
 \_SBF 47

## B

BLACK 47  
 BLUE 47  
 BLUE\_COLORS 48  
 BLUEMASK 48  
 BLUESHIFT 48  
 BOOL\_16 19  
 BOOL\_32 19  
 BOOL\_8 19

## C

CHAR 20  
 COLOR\_T 20  
 COLORS\_DEF 48  
 CYAN 49

## D

DARKGRAY 49

## E

EXTERN 49

## F

FALSE 49  
 font\_helvr10 24  
 font\_rom8x16 24  
 font\_rom8x8 25  
 FONT\_T 20  
 font\_winfreesys14x16 25  
 font\_x5x7 25  
 font\_x6x13 25

## G

GREEN 49  
 GREEN\_COLORS 50  
 GREENMASK 50  
 GREENSHIFT 50

## H

helvr10\_bits 26  
 helvR10\_width 27

## I

INT\_16 20  
 INT\_32 21  
 INT\_64 21  
 INT\_8 21

## L

LIGHTBLUE 50  
 LIGHTCYAN 51  
 LIGHTGRAY 51  
 LIGHTGREEN 51  
 LIGHTMAGENTA 51  
 LIGHTRED 51  
 LIGHTYELLOW 52  
 LPC\_BAD\_CLK 52  
 LPC\_BAD\_HANDLE 52  
 LPC\_BAD\_PARAMS 52  
 LPC\_CANT\_START 53  
 LPC\_CANT\_STOP 53  
 LPC\_COLOR\_TYPES\_H 53  
 lpc\_colors.c 62  
 lpc\_colors.h 63  
 lpc\_colors\_set\_palette 1  
 LPC\_DEV\_UNKNOWN 53  
 lpc\_fonts.c 64  
 lpc\_fonts.h 64  
 LPC\_FONTS\_H 53  
 lpc\_helvr10.c 65  
 lpc\_helvr10.h 65  
 LPC\_HEVR10\_FONT\_H 54

LPC\_IN\_USE 54  
 LPC\_NOT\_OPEN 54  
 LPC\_NOT\_SUPPORTED 54  
 LPC\_PIN\_CONFLICT 55  
 lpc\_rom8x16.c 66  
 lpc\_rom8x16.h 66  
 LPC\_ROM8X16\_FONT\_H 55  
 lpc\_rom8x8.c 67  
 lpc\_rom8x8.h 67  
 LPC\_ROM8X8\_FONT\_H 55  
 lpc\_swim.c 67  
 lpc\_swim.h 68  
 lpc\_swim\_font.c 70  
 lpc\_swim\_font.h 70  
 LPC\_SWIM\_FONT\_H 55  
 LPC\_SWIM\_H 55  
 lpc\_swim\_image.c 72  
 lpc\_swim\_image.h 72  
 LPC\_SWIM\_IMAGE\_H 56  
 lpc\_types.h 73  
 LPC\_TYPES\_H 56  
 LPC\_WINFREESYS\_14X16\_FONT\_H 56  
 lpc\_winfreesystem14x16.c 75  
 lpc\_winfreesystem14x16.h 75  
 lpc\_x5x7.c 76  
 lpc\_x5x7.h 76  
 LPC\_X5X7\_FONT\_H 56  
 lpc\_x6x13.c 77  
 lpc\_x6x13.h 77  
 LPC\_X6X13\_FONT\_H 57

## M

MAGENTA 57

## N

NELEMENTS 57  
 NULL 57  
 NUM\_COLORS 57

## P

PFI 21  
 PFV 21

## R

RED 58  
 RED\_COLORS 58  
 REDMASK 58  
 REDSHIFT 58  
 rom8x16\_bits 28  
 rom8x16\_width 33  
 rom8x8\_bits 33  
 rom8x8\_width 36

## S

SMA\_BAD\_CLK 59  
 SMA\_BAD\_HANDLE 59  
 SMA\_BAD\_PARAMS 59  
 SMA\_CANT\_START 59  
 SMA\_CANT\_STOP 59  
 SMA\_DEV\_UNKNOWN 60  
 SMA\_IN\_USE 60  
 SMA\_NOT\_OPEN 60  
 SMA\_NOT\_SUPPORTED 60  
 SMA\_PIN\_CONFLICT 61  
 STATIC 61  
 STATUS 22  
 SUCCESS 61  
 swim\_clear\_screen 2  
 swim\_get\_font\_height 2  
 swim\_get\_horizontal\_size 3  
 swim\_get\_vertical\_size 3  
 swim\_get\_xy 4  
 swim\_put\_box 4  
 swim\_put\_char 5  
 swim\_put\_diamond 5  
 swim\_put\_image 6  
 swim\_put\_invert\_image 6  
 swim\_put\_left\_image 7  
 swim\_put\_line 7  
 swim\_put\_ltext 8  
 swim\_put\_newline 8  
 swim\_put\_pixel 9  
 swim\_put\_right\_image 9  
 swim\_put\_scale\_image 10

swim\_put\_scale\_invert\_image 10  
swim\_put\_scale\_left\_image 11  
swim\_put\_scale\_right\_image 11  
swim\_put\_text 12  
swim\_put\_text\_xy 12  
swim\_put\_win\_image 13  
SWIM\_ROTATION\_T 22  
swim\_set\_bkg\_color 13  
swim\_set\_fill\_color 14  
swim\_set\_font 14  
swim\_set\_font\_transparency 15  
swim\_set\_pen\_color 15  
swim\_set\_title 16  
swim\_set\_xy 16  
swim\_window\_close 17  
swim\_window\_open 17  
swim\_window\_open\_noclear 18  
SWIM\_WINDOW\_T 22

## T

TRUE 61

## U

UNS\_16 23  
UNS\_32 23  
UNS\_64 24  
UNS\_8 24

## W

WHITE 61  
winfreesystem14x16\_bits 37  
winfreesystem14x16\_width 41

## X

x5x7\_bits 42  
x5x7\_width 43  
x6x13\_bits 43  
x6x13\_width 46

## Y

YELLOW 62