Simple Font Library Reference

Publication date: December 2003

Sony Computer Entertainment Inc.
1-1, Akasaka 7-chome, Minato-ku
Tokyo 107-0052, Japan

Sony Computer Entertainment America
919 E. Hillsdale Blvd.
Foster City, CA 94404, U.S.A.

Sony Computer Entertainment Europe
30 Golden Square
London W1F 9LD, U.K.

Table of Contents

About This Manual

This manual is the PS2 Programmer Tool Runtime Library libpfont, Version 1.2 version of the *Simple Font Library Reference* manual.

Changes Since Last Release

- A description of the scePFontGetErxEntries() function was added.

Related Documentation

Refer also to the *Font File Format* manual and the *Simple Font Library Overview* manual.

Note: the Developer Support Web site posts current developments regarding the Libraries and also provides notice of future documentation releases and upgrades.

Typographic Conventions

Certain Typographic Conventions are used throughout this manual to clarify the meaning of the text:

Convention	Meaning
`courier`	Indicates literal program code.
italic	Indicates names of arguments and structure members (in structure/function definitions only).
medium bold	Indicates data types and structure/function names (in structure/function definitions only).
blue	Indicates a hyperlink.

Developer Support

Sony Computer Entertainment America (SCEA)

SCEA developer support is available to licensees in North America only. You may obtain developer support or additional copies of this documentation by contacting the following addresses:

Order Information	Developer Support
Attn: Developer Tools Coordinator Sony Computer Entertainment America 919 East Hillsdale Blvd. Foster City, CA 94404, U.S.A. Tel: (650) 655-8000	E-mail: scea_support@ps2-pro.com Web: https://www.ps2-pro.com/ Developer Support Hotline:(650) 655-5566 (Call Monday through Friday, 8 a.m. to 5 p.m., PST/PDT)

Sony Computer Entertainment Europe (SCEE)

SCEE developer support is available to licensees only in the PAL television territories (including Europe and Australasia). You may obtain developer support or additional copies of this documentation by contacting the following addresses:

Order Information	Developer Support
Attn: Development Tools Manager Sony Computer Entertainment Europe 30 Golden Square London W1F 9LD, U.K. Tel: +44 (0) 20 7859-5000	E-mail: scee_support@ps2-pro.com Web: https://www.ps2-pro.com/ Developer Support Hotline: +44 (0) 20 7859-5777 (Call Monday through Friday, 9 a.m. to 6 p.m., GMT/BST)

Structures

scePFontBlock

Internal memory representation of font data

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontBlock{
    scePFontTag tag;                        Font tag
    scePFontBlockAttribute attr;            scePFontBlockAttribute contents
    float scale_x;                          Output size correction (X)
    float scale_y;                          Output size correction (Y)
    short image_width;                      IMAGE 1 character width (pixels)
    short image_height;                     IMAGE 1 character height (pixels)
    short max_ascent;                       Maximum ASCENT in BLOCK
    short max_descent;                      Maximum DESCENT in BLOCK
    short max_width;                        Maximum WIDTH in BLOCK
    short reserved;                         Reserved
 struct{
    int character_num;                      Character count
    int image_offset;                       Offset to start of image data
    int codeindex_num;                      codeindex count
    int codeindex_offset;                   Offset to start of codeindex data
    int codemap_num;                        codemap count
    int codemap_offset;                     Offset to start of codemap data
    int proportional_num;                   Proportional count
    int proportional_offset;                Offset to start of proportional data
    int kerning_num;                        Kerning count
    int kerning_offset;                     Offset to start of kerning data
    int clut_num;                           clut entry count
    int clut_offset;                        Offset to start of clut data
 }data;
}scePFontBlock;
```

Description

This structure is an internal memory representation of font data.

Refer to "Font File Format" for more information.

scePFontBlockAttribute

Internal memory representation of font data

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontBlockAttribute{
    u_int color_mode:3;          TEXTURE color mode
    u_int propotional:1;         Proportional format
    u_int reserved:28;           Reserved
}scePFontBlockAttribute;
```

Description

This structure is an internal memory representation of font data.

Refer to "Font File Format" for more information.

scePFontCodeIndex

Internal memory representation of font data

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontCodeIndex{
    int start_code;          Starting character code
    int end_code;            Ending character code
    int map_index;           Starting map number
    int chr_index;           Ending character number
}scePFontCodeIndex;
```

Description

This structure is an internal memory representation of font data.

Refer to "Font File Format" for more information.

scePFontControl

Function table passed to callback

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

typedef struct _scePFontControl{

int (* const *Getc***)**	Get character
(int *fd***);**	*fd* = Identifier
	Return: Character code (0: complete; -1: error)
int (* const *Ungetc***)**	Returns position of one character
(int *fd***);**	*fd* = Identifier
	Return: 0: success; -1: starting position
int (* const *Putc***)**	Output character
(int *fd***, int** *chr***);**	*fd* = Identifier
	chr = Character code
	Return: 0: success; -1: insufficient memory; -2: no glyph
int (* const *PutcX***)**	Output character (Calculate only without actually drawing)
(int *fd***, int** *chr***);**	*fd* = Identifier
	chr = Character code
	Return: 0: success; -1: insufficient memory; -2: no glyph
int (* const *Calc***)**	Calculate output rectangle for character
(int *fd***, int** *chr***, scePFontFRect*** *pRect_i***, scePFontFRect*** *pRect_o***);**	*fd* = Identifier
	chr = Character code
	pRect_i = Inside rectangle to be received (NULL: do not receive)
	pRect_o = Outside rectangle to be received (NULL: do not receive)
	Return: 0: success; -1: insufficient memory; -2: no glyph

}scePFontControl;

Description

This structure is the function table passed to the callback.

scePFontGlyph

Internal memory representation of font data

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontGlyph{
    scePFontBlock const* block;          Pointer to block
    u_long128 const* image;              Pointer to image
    scePFontPropotional const* prop;     Pointer to proportional data
    void const* kerning;                 Reserved(NULL)
}scePFontGlyph;
```

Description

This structure is an internal memory representation of font data.

Refer to "Font File Format" for more information.

scePFontInfo

Internal memory representation of font data

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontInfo{
    scePFontTag tag;                    Members
    char name[32];                      Font tag
    char comment[32];                   Font name
    short max_ascent;                   Comment
    short max_descent;                  Maximum ASCENT for entire font
    short max_width;                    Maximum DESCENT for entire font
    short reserved;                     Maximum WIDTH for entire font
    int block_num;                      Reserved
    int block_offset[0];                Number of blocks
}scePFontInfo;
```

Description

This structure is an internal memory representation of font data.

Refer to "Font File Format" for more information.

scePFontProportional

Internal memory representation of font data

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontProportional{
    short base_x;          BASE POINT X value
    short base_y;          BASE POINT Y value
    short l_bearing;       L BEARING value
    short r_bearing;       R BEARING value
    short ascent;          ASCENT value
    short descent;         DESCENT value
    short width;           WIDTH value
    short kerning;         Reserved
}scePFontPropotional;
```

Description

This structure is an internal memory representation of font data.

Refer to "Font File Format" for more information.

scePFontRect

Representation of bounding rectangle for font

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontFRect{
    float top;          Top edge
    float bottom;       Bottom edge
    float left;         Left edge
    float right;        Right edge
}scePFontFRect;
```

Description

This structure represents the bounding rectangle of a font.

scePFontTag

Font tag

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Structure

```
typedef struct _scePFontTag{
    u_int id;                    ID(0x00000000U)
    u_int version;               Version (0x00000000U)
    int size;                    Size
    int reserved;                Reserved
}scePFontTag;
```

Description

This structure is an internal memory representation of font data.

Refer to "Font File Format" for more information.

Functions

scePFontAttachData

Attach to font data

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

int scePFontAttachData(

int *fd***,**	Identifier
u_long128 const* *pData***);**	Font data (must be on a qword boundary)

Description

This function creates an attachment to font data.

Return value

On success, 0 is returned.

On error, -1 is returned.

scePFontCalcCacheSize

Calculate amount of main memory required to initialize font

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

size_t scePFontCalcCacheSize(

int *num***);** Number of characters to be cached

Description

This function calculates the amount of main memory required to initialize a font.

(fixed amount for work + number of characters * work amount per character)

Return value

Amount of memory required (in bytes) to cache the specified number of characters

scePFontCalcRect

Calculate font output region

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontCalcRect(

int *fd***,**	Identifier
char const* *str***,**	Character string to test output
scePFontFRect* *pRect_i***,**	Pointer to variable that will receive inside rectangle
scePFontFRect* *pRect_o***);**	Pointer to variable that will receive outside rectangle

Description

This function calculates the font output region in the local coordinate system.

(current status is preserved)

Return value

None

scePFontFlush

Flush character strings that are being output

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontFlush(

int *fd***);** Identifier

Description

This function flushes character strings that are being output and clears the cache.

It sets LOCATE to (0,0,0,0) and the conversion matrix to a unit line.

It also sets color to the default (1.0,1.0,1.0,1.0).

Return value

None

scePFontGetBlock

Get font block information

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

scePFontBlock const* scePFontGetBlock(

int *fd*,	Identifier
int *idx*);	Block number

Description

This function gets a pointer to the specified block from the attached font.

Return value

Returns a pointer to the font block information.

Returns NULL on error.

scePFontGetColor

Get drawing color

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontGetColor(

int *fd***,**	Identifier
sceVu0FVECTOR* *pColor***);**	Pointer to variable that will receive the color

Description

This function gets the current drawing color.

Return value

None

scePFontGetErxEntries

Get library entries

Library	*Introduced*	*Documentation last modified*
libpfont	1.2	December 18, 2003

Syntax

void *scePFontGetErxEntries(void);

Description

This function gets a pointer to the SceErxLibraryHeader structure which is needed to register the libpfont library in liberx.

If the SceErxLibraryHeader structure returned by this function is passed to the sceErxRegisterLibraryEntries() function of liberx, the statically linked libpfont library can be called from an ERX module (can be dynamically linked).

When this function is linked, the entire implementation of libpfont will be linked via the library entry table of liberx.

In some cases, this may also increase the size of the program.

Return value

The entry library for libpfont is returned.

See also

sceErxInit()

scePFontGetFontInfo

Get font information

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

scePFontInfo const* scePFontGetFontInfo(

int *fd*); Identifier

Description

This function gets a pointer to the font information from the attached font.

Return value

Returns a pointer to the font information.

Returns NULL on error.

scePFontGetFontMatrix

Get conversion matrix

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontGetFontMatrix(

int *fd***,**	Identifier
sceVu0FMATRIX* *pMatrix***);**	Address of variable that will receive conversion matrix

Description

This function gets the current conversion matrix.

(center is base_x+width/2,base_y)

Return value

None

scePFontGetGlyph

Get glyph for specified character code

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

int scePFontGetGlyph(

int *fd***,**	Identifier
int *chr***,**	Character code
scePFontGlyph* *pGlyph***);**	Pointer to variable for receiving glyph data

Description

This function gets the glyph for the indicated character code from the attached font.

Return value

On success, 0 is returned.

On error, -1 is returned.

scePFontGetLocate

Get drawing location

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontGetLocate(

int *fd***,**	Identifier
sceVu0FVECTOR* *pLocate***);**	Pointer to variable for receiving location

Description

This function gets the current drawing location.

Return value

None

scePFontGetPitch

Get character spacing (pitch)

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

float scePFontGetPitch(

int *fd***);** Identifier

Description

This function gets the current character spacing (pitch).

Return value

Returns the character spacing (pitch).

scePFontGetScreenMatrix

Get coordinate conversion matrix

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontGetScreenMatrix(

int *fd***,**	Identifier
sceVu0FMATRIX* *pMatrix***);**	Address of variable for receiving coordinate conversion matrix

Description

This function gets the current coordinate conversion matrix.

Return value

None

scePFontGetWidth

Get character width for calculation

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

short scePFontGetWidth(

int *fd***);** Identifier

Description

This function gets the character width for the current calculation.

Return value

Returns the character width.

scePFontInit

Initialize font

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

int scePFontInit(

int *num***,**	Number of characters to be cached (must be at least 1)
u_long128* *pWork***);**	Working memory for cache (128-bit aligned)

Description

This function initializes a font.

Always allocate and pass enough memory as calculated by scePFontCalcCacheSize.

This memory should be maintained internally until scePFontRelease() is called.

Return value

Font identifier (negative numbers represent errors)

scePFontPutc

Output one character

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

int scePFontPutc(

int *fd***,**	Identifier
sceVif1Packet* *pPacket***,**	Initialized packet
int *size***,**	Maximum packet size (qwc)
int *chr***);**	Output character (UCS2)

Description

This function outputs one character.

The packet is attached and closed according to cnt.

Draws with alpha on.

Uses context 1.

ALPHA_0 and TEST_0 should be set externally.

Return value

On success, 0 is returned

On error, -1 is returned (no space in packet)

scePFontPuts

Output character string

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

```
int scePFontPuts(
 int fd,                          Identifier
 sceVif1Packet* pPacket,          Initialized packet
 int size,                        Maximum packet size (qwc)
 char const* str);                Character string
```

Description

This function packetizes a character string and outputs it.

Return value

Table 1

Value	Result
0	Completed
1	Continuation
-1	Packet too small

scePFontPutsContinue

Output continuation of character string

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

int scePFontPutsContinue(

int *fd***,**	Identifier
sceVif1Packet* *pPacket***,**	Initialized packet
int *size***);**	Maximum packet size (qwc)

Description

This function outputs a continuation of a character string that could not be output using scePFontPuts().

A return value of 1 from scePFontPuts() indicates that the packet was full and could not be displayed, in which case this function should be called.

Return value

Table 2

Value	Result
0	Completed
1	Continuation
-1	Packet too small

scePFontRelease

Release font resources

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontRelease(

int *fd***);** Identifier

Description

This function releases font resources and discards any attachments.

Return value

None

scePFontSetColor

Set drawing color

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetColor(

int *fd***,**	Identifier
sceVu0FVECTOR const* *pColor***);**	Color

Description

This function sets the drawing color (multiplied with the original character color).

Return value

None

scePFontSetFilter

Set filter function

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetFilter(

int *fd***,**	Identifier
int (*_filter_**)(int** *fd***, scePFontControl*** *pCtrl***));**	Pointer to filter function Uses default if NULL is passed

Description

This function sets a filter function.

It is called by scePFontPuts(), scePFontPutsContinue(), and scePFontCalcRect().

The default filter function is shown below.

Table 3

fd	Identifier
pCtrl	Font control structure
return	0=Completed, 1=Continuation, -1=Packet too small

```
int default_filter(int fd, scePFontControl* pCtrl){
       int stat = -1;
       int c;
       while(0 < (c = pCtrl->Getc(fd))){
              if(-1 == pCtrl->Putc(fd, c)){
                     pCtrl->Ungetc(fd);
                     return stat;
              }
              stat = 1;
       };
       return 0;
}
[eof]
```

Return value

None

scePFontSetFontMatrix

Set conversion matrix

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetFontMatrix(

int *fd***,**	Identifier
sceVu0FMATRIX const* *pMatrix***);**	Pointer to conversion matrix

Description

This function sets the conversion matrix. The conversion matrix is used for scaling, rotating, and translating individual characters.

Return value

None

scePFontSetGsCtxt

Specify GS packet context

Library	*Introduced*	*Documentation last modified*
libpfont	1.1	February 26, 2003

Syntax

int scePFontSetGsCtxt(

int *fd***,**	Identifier
int *ctxt***);**	GS context 0 or 1 (default is 0)

Description

This function specifies the GS packet context.

Return value

Returns the previous value.

scePFontSetLocate

Set initial drawing location

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetLocate(

int *fd***,**	Identifier
sceVu0FVECTOR const* *pLocate***);**	Drawing location

Description

This function sets the initial drawing location.

Return value

None

scePFontSetPitch

Set character spacing (pitch)

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetPitch(

int *fd***,**	Identifier
float *pitch***);**	Pitch

Description

This function sets the character spacing (pitch).

Return value

None

scePFontSetScreenMatrix

Set coordinate conversion matrix

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetScreenMatrix(

int *fd***,**	Identifier
sceVu0FMATRIX const* *pMatrix***);**	Pointer to coordinate conversion matrix

Description

This function sets the coordinate conversion matrix (for converting local coordinates to screen coordinates).

Return value

None

scePFontSetTexMem

Set up GS memory for texture transfer

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetTexMem(

int *fd***,**	Identifier
u_int *tbp***,**	Word address of GS
u_int *size***,**	Number of words in GS memory (multiple of 2048 words)
u_int *cbp***);**	Word address for 32bitCLUT (must be able to accommodate 256 entries)

Description

This function sets up GS memory for texture transfers.

Operation is not guaranteed if the set-up is such that one character cannot be transferred.

When discarding memory information, always invalidate it using scePFontFlush().

Return value

None

scePFontSetWidth

Set character width for calculation

Library	*Introduced*	*Documentation last modified*
libpfont	1.0	June 7, 2002

Syntax

void scePFontSetWidth(

int *fd***,**	Identifier
short *width***);**	0: default >=1: fixed character width

Description

This function sets the character width for calculation.

It will not change the display width of one character. For example, if a character's width were smaller than this value, the next character would overlap its predecessor.

Return value

None