Simple Font Library Overview

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 Overview* manual.

Changes Since Last Release

- A description of ERX-related files was added to “Related Files” in the “Overview” section.

Related Documentation

Refer also to the *Font File Format* manual and the *Simple Font Library Reference* 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)

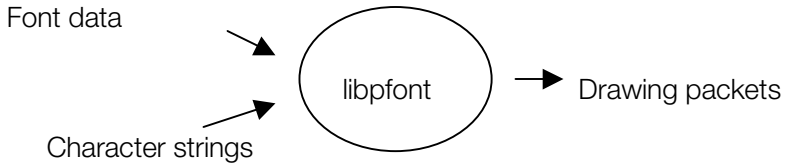
SCEE developer support is available to licensees only in the PAL television territories (including Europe and Australasia). You may obtain developer support or additional copies of this documentation by contacting the following addresses:

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)

Overview

The Simple Font Library takes font data and a text string as input, and produces drawing packets as output.

Figure 1



Font data	: Original format bitmap font
Character string	: Unicode (UTF8)
Drawing packet	: GS native format

Related Files

The following files are required to use the Simple Font Library.

Table 1

Category	Filename
Library	libpfont.a
ERX-related files	libpfont.erx
	libpfont.ilb
Header file	libpfont.h

Library Functions

The Simple Font Library provides the following primary functions.

- Display of alpha-blended bitmap fonts that are easy to read with minimal flicker, even in interlaced mode.
- Conversion of fonts for individual characters, using a conversion matrix.
- Appending of sample font data.
- Conversion of arbitrary bitmap fonts that were created using dedicated tools, enabling the fonts to be used with libpfont.
- Multilingual support based on Unicode (UTF8). When used together with libccc, Shift-JIS encoding can also be used.
- Proportional display of half-width fonts.
- Support for filters that can be defined for specific characters. For example, operations can be easily assigned to specific escape codes.

Drawing Packets

All drawing packets are created in the packet area passed from the user. As a result, once packets have been created, they do not depend on the state of the library. Note, however, that the texture cache does have state, so the order that packets are output cannot be changed.

Usage

1. Specify the number of characters to be cached and calculate the size of the required work area
   size = scePFontCalcCacheSize(num);
2. Allocate work area
   pWork = alloc_work(size);
3. Initialize work area and use the identifier that is returned for subsequent operations
   fd = scePFontInit(num, pWork);
4. Specify texture area used for font output
   scePFontSetTexMem(fd, tbp, size, cbp);
5. Attach font data that was read in
   scePFontAttachData(fd, pData);
6. Set conversion matrix
   scePFontSetScreenMatrix(fd, pMatrix);
   scePFontSetFontMatrix(fd, pMatrix);
7. Drawing event

```
{
        // Specify coordinates
        scePFontSetLocate(fd, pLocate);

        // Packet construction process
        if(1 == scePFontPuts(fd, pPacket, size, "test")){
                do{
                        kick(pPacket);
                }while(1 == scePFontPutsContinue(fd, pPacket, size));
        }
        kick(pPacket);
}
```

Sample Font Data

- Sample font data is provided with libpfont. The sample data includes both Shift-JIS (JIS-X 0208) and Latin-1 (ISO 8859-1) characters.
- Two styles of half-width English characters are provided. By default, iso0640_26.pb and jis0101_26.pb are used. Half-width Shift-JIS characters are also included. latin.pb can also be used, including all of the "Latin-1" characters. See the sample configuration file pb2pf.txt for information on setting up these fonts.

Copyright Information

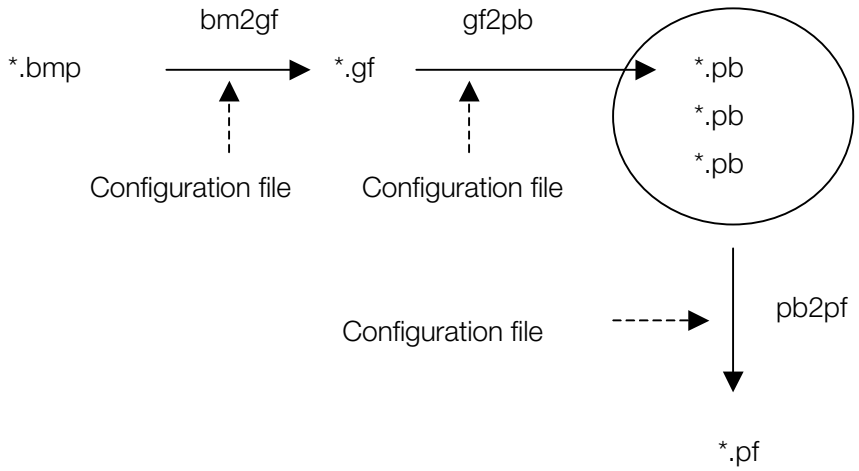
- The copyright of the sample font data is owned by Sony Computer Entertainment Inc.
- Sample font data can be incorporated in products of individual licensees, but only for use in PlayStation 2 applications. These fonts may not be modified, assigned, leased, or publicly transmitted (or converted to transmissible form) without permission from Sony Computer Entertainment Inc. Licensees are not required to include a font license with their products.
- Use of this font data and any works derived therefrom is prohibited except for use in PlayStation 2 applications.

Font Data Creation Tool

A tool is provided for adding new fonts to existing font data, and for creating original fonts.

The process involves converting a bmp file of the font to a gf file, then converting the gf file to a pb file, and finally grouping together multiple pb files into a single pf file. The pf file is read directly by libpfont.

Figure 2



The tool reads in the configuration file, and performs various types of setup. For more information, please refer to the sample configuration file.

bmp files must include a 4-bit CLUT (index). The CLUT should change colors from the background color to the character color (color of the character core) starting from the small colors.

CLUT (Index) for Font Data

In the current version of libpfont, the font's CLUT is stored statically inside the font data. To change the CLUT, you must modify the source code of the conversion tool gf2pb and reconvert the font data.

In gf2pc.c, the CLUT is defined by defFontClut.

The default CLUT is a special value that displays a black border around the font. If the black border isn't needed, set defFontCLut as shown below (it would also be a good idea to set values for nonlinear shapes so as to account for gamma).

```
{
{ 255, 255, 255,   0},
{ 255, 255, 255,   8*1-1},
{ 255, 255, 255,   8*2-1},
{ 255, 255, 255,   8*3-1},
{ 255, 255, 255,   8*4-1},
{ 255, 255, 255,   8*5-1},
{ 255, 255, 255,   8*6-1},
{ 255, 255, 255,   8*7-1},
{ 255, 255, 255,   8*8-1},
{ 255, 255, 255,   8*9-1},
{ 255, 255, 255,   8*10-1},
{ 255, 255, 255,   8*11-1},
{ 255, 255, 255,   8*12-1},
{ 255, 255, 255,   8*13-1},
{ 255, 255, 255,   8*14-1},
{ 255, 255, 255,   8*15-1},
}
```

Setting the CLUT as indicated above allows the text color to be set freely using scePFontSetColor().

On the other hand, with the default CLUT asis, the special CLUT that is used for the black border will not produce good results, because colors set using scePFontSetColor() might not be rendered well (jaggies may be visible).