

May 1998

PlayStation(R) Programmer Tool  
Runtime Library Release 4.3

Sample Directory  
from Release 4.2

-----  
Copyright(C) 1998 Sony Computer Entertainment Inc.  
All Rights Reserved.  
-----

Sample Directory Structure

-----  
The samples with "(\*)" are newly added in this release.

```
psx\sample\  
|--cd  
|   |--earth  
|   |--movie  
|   |--str3d  
|   +--tuto  
|--cmplr  
|   +--scratch  
|--ds  
|   +--tuto  
|--etc  
|   |--card  
|   |--comb  
|   |   |--tuto1  
|   |   |--tuto2  
|   |   +--tuto3  
|   |--gun  
|   |--mcrd  
|   |   |--tuto0  
|   |   |--tuto1  
|   |   +--tuto2  
|   |--mouse  
|   |--sio  
|   |   |--tuto1  
|   |   +--tuto2  
|   |--tap  
|   +--thread  
|--graphics  
|   |--basic  
|   |--bg  
|   |   |--bgsample  
|   |   +--fix32  
|   |--clutfog  
|   |--divide  
|   |   |--active  
|   |   +--clip  
|   |--dmpsx  
|   |--fballs  
|   |--gsgpu  
|   |--hmd  
|   |   |--anim  
|   |   |--basic  
|   |   |--common  
|   |   |--mime  
|   +--pdriver
```

```

|--jimen
--mesh
|   |--qmesh
|   |--rmesh
|   +--smesh
--mime
|   |--joint
|   |--vertex
|   +--vjmime
--mipmap
--misc
|   |--60frame
|   +--getode
--oden
--phong
--ppm
--rotate
|   |--arot
|   |--intrpol
|   +--mat2rot
--rotmat
--screen  (*)
--shadow
--texaddr
|   +--wave
--tmdpmd
--tmdview
|   |--lowlevel
|   |--rcube
|   |--shuttle
|   |--tmdview3
|   |--tmdview4
|   +--tmdview5
--tod
|   |--janken
|   +--todview
--trr
--tuto
--walk
+--zimen
--kanji  (*)
|   |--asc2sjis
|   |--fontdata
|   |--kanjidiv
|   |--kanjifnt
|   +--sjiscode
--math
|   +--tree
--module
|   |--cdexec
|   |--execmenu
|   |   |--anim
|   |   |--balls
|   |   +--rcube
|   |--menu  (*)
|   +--overmenu
--old
|   +--etc
|   |   |--card
|   |   |   |--doc
|   |   |   |--lib
|   |   |   |--makecard

```

```

|      |      +--max
|      +--cman
--pad
|      |      --anlgctrl
|      +--dungeon
--press
|      +--tuto
|      +--msiro
+--sound
|      |      --3deffect
|      |      --balls
|      |      --basic
|      |      --cdvol
|      |      --mutual
|      |      --simple
|      |      --stream
|      |      --tuto
+--xse

```

## Data Directory Structure

The samples with "(\*)" are newly added in this release.

```

psx\data      (*)
|
|--bgd
|--cel
|--hmd
|   +--scei
|   |   |--anim
|   |   |   |--balloon
|   |   |   |--cube0
|   |   |   |--domes
|   |   |   |   +--lw3d
|   |   |   |--domesc
|   |   |   |--interp
|   |   |   |   |--3dstudio
|   |   |   |   +--nworld
|   |   |   |--rots
|   |   |   |--snake
|   |   |   |--tri3
|   |   |   |--tri4
|   |   |   +--ufo
|   |   |--basic
|   |   |   +--dice
|   |   |--comb
|   |   |   |--ammonite
|   |   |   +--tofu
|   |   |--envmap
|   |   |   +--texture
|   |   |--equip
|   |   |   |--pp
|   |   |   +--snowman
|   |   |--mime
|   |   |   |--legs
|   |   |   |--twinp
|   |   |   +--twister
|   |   +--tuto
|   |   |   +--pyramid
|
|--pmd
|--sep
--seq

```

```
--tim
--tmd
--tod
--vab
+--vag
```

#### Sample Index (in alphabetical order)

```
-----
.\cd\earth: Earth
    Example to map a moving picture on a curved surface

.\cd\movie: movie <libcd>
    A group of samples of moving pictures with streaming
    tuto0: simple streaming program
    tuto1: free resolutions
    tuto2: on memory streaming
    tuto3: avoiding frame skip

.\cd\str3d: combination
    Sample of the combination of moving pictures with streaming
    and 3D model display
    Spreading loads of animation with DecDCTvlcSize()

.\cd\tuto: CD tutorial <libcd>
    CD-ROM step-by-step tutorial
    tuto0: simplest CD-Player (polling type)
    tuto1: simplest CD-Player (interrupt type)
    tuto2: auto repeat play among 2 points of CD-DA
    tuto3: auto repeat play using CdldataEnd
    tuto4: fast operation using CdControlF
    tuto5: auto repeat play among 2 point of CD-XA.
    tuto6: interleaved audio/data channel
    tuto7: background CD read
    tuto8: multiple-file CdRead
    tuto9: load and execute programs
    tuto10: high level CD-ROM file access
    tuto11: test CD type

.\cmplr\scratch: Scratch pad area used Sample
    Data is placed on the scratch pad area, and the difference of the
    processing speed can be seen with 3 access methods.

.\ds\tuto: DS tutorial <libds>
    CD-ROM step-by-step tutorial
    tuto0 - tuto11
        rewrite libcd tutorial for libds
    tuto12 compares the music starting time:
        playing DA with the forward seek vs. without forward seek
    tuto13 compares the music starting time:
        playing XA with the forward seek vs. without the forward
        seek.
    tuto14 compares the read time:
        continuously reading multiple files vs. reading one by one
        after seek.

.\etc\card: Memory card access sample
    Memory card file utility such as state-monitoring, formatting,
    creating.

.\etc\comb: Link cable sample
    Sample for sending and getting data on the 2 machines connected
```

to each other by a link cable.

.\etc\gun: gun controller sample  
Sample for detecting the position that the gun controller is pointed and displays the position on TV monitor.

.\etc\mcrd: Memory card sample  
tuto0 Synchronous processing: Icon generation program  
tuto1 Asynchronous processing: Sample 1  
Displays the list of files on the memory card  
tuto2 Asynchronous processing: Sample 2  
Selectively copies the Slot1 file(s) to Slot 2

.\etc\mouse: Mouse control sample  
Sample to process cursor-movements and clicking with a mouse

.\etc\sio\tutol\tutol: SIO driver sample  
Sample to connect the debugging station with PC via H3050 and echo-back input from PC.

.\etc\tap: multi tap sample

.\etc\thread: Thread sample  
Sample to process other jobs until the next VSync

.\graphics\basic: demonstrates for basic functions  
Each demo is executed from MENU.  
2d: Texture mapping on a curved surface.  
Mapping a 512x256 texture pattern on a 3-dimensional curved surface.  
When 'select' is pressed, the pattern will come to pieces.

balls: Displaying a lot of 16x16 sprites.  
The ways of using the following functions are described here.  
FntPrint KanjiFntPrint, VSyncCallback(), VSync()

diffuse: 16x16 balls or rectangle polygons are diffused from the origin in the world coordinate system.  
GTE performance demonstration.

mat: Sprite animation  
The Matchang animation is placed in the 3 dimension.

rgb24: Demo in the 24-bit mode.  
Examples with StoreImage(), LoadImage(), MoveImage() are here.

.\graphics\bg: BG drawing function sample in libgs <libgs>  
TIM\CEL\BGD files made up with Sprite Editor may be read in and displayed.  
bgsample: Background sample  
fix32: Background sample (fast)

.\graphics\clutfog: Fog sample with clut  
tuto0: A clut is generated frame by frame, and transferred to the VRAM.  
tuto1: Some cluts are placed on the VRAM, and switched according to the depth of fog.  
tuto2: Cluts are switched by DR\_MOVE.  
Applicable to drawing by libgs as well.

.\graphics\data: shared data

```

.\graphics\divide: Polygon automatic division sample <libgte>
    clip: Divide function examples to avoid texture distortion.
           In readme.txt, PCpoly function examples included.
    active: Sub-division sample with the direct mapping.
            Crack problem and z-sorting by the maximum value not
            by the center of gravity are included.

.\graphics\dmpsx: DMPSX <libgte>
    High speed sample with dmpsx

.\graphics\gsgpu: GSGPU <libgpu,libgs>
    Sample using libgs and libgte together.
    tuto0: Uses AddPrim() in libgs.
    tuto1: Draws libgs objects with DrawTag().

.\graphics\hmd: HMD format sample
    anim: animation using HMD format
           tuto0: general HMD animation
           tuto1: Realtime Motion Switch #1
           tuto2: Realtime Motion Switch #2
           tuto3: General HMD animation viewer using view point
                  animation
    animview: General HMD program
    basic: viewer using HMD format
           tuto0: general HMD viewer
           tuto1: HMD viewer using shared polygons
           tuto2: shuttle viewer using HMD format
           tuto3: HMD viewer using sub divide
    mime: MIME sample using HMD format
    pdriver: sample codes of primitive driver for HMD format

.\graphics\jimen: Undistorted texture mapping <libgte>
    Function sample for undistorted texture mapping.

.\graphics\mesh: Mesh
    qmesh: two dimensional mesh
           tuto0: sample of QMESH function...screen clip
           tuto1: sample of QMESH function...terrain data
    rmesh: round mesh
    smesh: strip mesh
           tuto0: drawing performance of SMESH functions
           tuto1: browsing SMESH function drawing mode
           tuto2: icosahedron

.\graphics\mime: MIME
    joint: Joint MIME animation of articulated model
    vertex: MIME Interactive Animation
            MIME sample program with GsDOBJ5
            Controlling 4 MIME parameters with L1,L2,R1,R2 buttons.
            The data is a simple gauraud-shaded polygon, and MIME
            processing is performed for the normal.
    vjmime: MIME animation of articulated model
           tuto0: Joint MIME animation of articulated model with
                  vertex MIME
           tuto1: Joint MIME animation of articulated model

.\graphics\mipmap: mipmap sample

.\graphics\misc: graphics miscellaneous
    60frame: difference 60 frames from 30 frames
    getode: Using Vsync with Interlace Mode

```

```

.\graphics\oden: Oden <libgs>
    Moving 3 light sources interactively, changing their colors,
    performing the real-time lighting calculation.

.\graphics\phong: phong shading <libgte>

.\graphics\ppm: undistorted mapping
    (perfect perspective mapping) <libgte>

.\graphics\rotate: rotation <libgte>
    arot: Rotation angle interpolation program
    intrpol: various kinds of interpolating about rotation
    mat2rot: Get Euler's angles from rotation matrix

.\graphics\rotmat: difference between RotMatrix and RotMatrix_gte <libgte>

.\graphics\screen: Screen
    Frame buffer viewer.
    Demo to explain the display mode and display environment
    parameters.

.\graphics\shadow: Shadow <libgte>
    The shadow dropped from a triangle is calculated in this program.
    Since the clipping is performed accurately, the shadow can be
    dropped on the steps

.\graphics\tmdpmd: TMD/PMD data viewer

.\graphics\tmdview: TMD view
    A group of samples with TMD and PMD data
    data: Directory where the data used in the samples is included.
    low level: Using low level functions
        Low level sample with GsTMDfast...() functions
        tuto0: 3 sided polygon, flat
        tuto1: 4 sided polygon, gouraud
        tuto2: eliminate a gap between polygons (with dmpsx)
        tuto3: mipmap version (with dmpsx)
    rcube: rotating cubes
        Variable effect samples for 3D
    shuttle: shuttle
        Hierarchical coordinate system sample with a space-
        shuttle model.
        The Animation such as opening/closing the hatch
        is displayed by setting the hatch and a satellite in
        the shuttle in its child coordinate.
    tmdview3: The simplest PMD data display program with GsDOBJ3
        tuto0: Simplest TMD data display program with GsDOBJ3
    tmdview4: The simplest TMD data display program with GsDOBJ2
        tuto0: simple tmdviewer using GsDOBJ2(GsSortObject4())
        tuto1: using GsSortObject4J()
        tuto2: active sub divide sample
        tuto3: sample code for split screen using GsDOBJ5
        tuto4: sample code for multiple ot and using same object with
        different handlers.
        tuto5: multiple screen coordinate sample
        tuto6: sample code of subjective move.
        tuto7: using GsSortObject4J() and using material attenuation
        in GsDOBJ2
    tmdview5: TMD data display program with GsDOBJ5
        tuto0: Simplest TMD data display program with GsDOBJ5
        tuto1: Sample of split screen
        tuto2: With modeling data some objects are displayed.

```

```

        More than one OT are used.
        tuto3: Automatic division with GsDOBJ5 attribute
        tuto4: Multiple screen coordinate system
        tuto5: Sample rewritten with GsSortObject5J
        tuto6: Sample where the viewpoint is moved subjectively

.\graphics\tod: Animation with tod
        janken: Multiple interactive tod animation
        todview: Simple animation

.\graphics\trr: TransRot...functions sample <libgte>
        Sample of TransRot...() functions to eliminate a gap between
        polygons

.\graphics\tuto: Tutorial <libgpu>
        Step-by-step tutorial
        tuto0   Displaying sprites
        tuto1   Drawing test with OT
        tuto2   Drawing a rotating polygon with GTE
        tuto3   Drawing a rotating cube
        tuto4   Drawing a cube with the light source
        tuto5   Drawing multiple 3D objects
        tuto6   Testing a 1D scrolling BG
        tuto7   Drawing a cube with the depth cueing
        tuto8   Showing a cell-type BG
        tuto9   Showing a 3D-cell-type BG
        tuto10  3D cell type BG (bird view)
        tuto11  pseudo mosaic effect
        tuto12  pseudo line scroll effect
        tuto13  multiple window operation

.\graphics\walk: An object walks on a polygon <libgte>
        Constraining an object on a polygon.
        On a object (object1) created by polygons, another object
        (object2) may move around. The object1 may take any shapes.
        The object2 changes its direction according to the direction of
        the object1's normal.

.\graphics\zimen: Terrain
        A group of programs to display the endless plane
        tuto0: Active primitive subdivision (with dmpsx)
        tuto1: Basic viewing volume clipping
        tuto2: Meshed ground pattern without height
        tuto3: Meshed infinite ground pattern
        tuto4: Meshed ground with active subdivision
        tuto5: Terrain sample with CLUT FOG (version with libgs)

.\kanji\asc2sjis: the ASCII code to the Shift-JIS code
        Converts the ASCII code to the Shift-JIS code

.\kanji\fontdata: Font data
        Size and type
            11/13/15 dots.
            non-kanji/first level/second level/vertical/half-size
            characters.
        Code conversion table

.\kanji\kanjidi: Command to extract character data
        Extracting image data from font data in character units.
        Command and viewer for extracted image data.

.\kanji\kanjifnt: Use of font data by size

```

Sample to use font data by size

```
.\KANJI\sjiscode: KANJI Code Viewer Program
    Shift-JIS codes of the built-in fonts can be displayed.

.\math\tree: Math-libaray-used sample
    Drawing a tree curve by the trigonometric function.

.\module\execmenu: EXEC sample
    BALLS, RCUBE, and ANIM are activated from the menu.
    There are 2 examples in this sample.
        * Activated by LoadExec().
        * Activated by Exec() after reading to the available
          memory.

.\module\menu: Sample Program Viewer which loads execution file.
    Sample execution files are activated from the menu.
    It is necessary that the program which can be activated from this menu
    should link "none2.obj" and be written in "menu.lst".

.\module\overmenu: Overlay sample
    BALLS, RCUBE, and ANIM are activated from the menu.
    3 BIN files are executed in order with the sound on.

.\pad\anlgctrl: Controller library sample <libpad>
    This program shows how to operate two types of actuator, switching
    the terminal type, Lock/Unlocking of terminal type switch, and the
    like. This program also displays the actuator information on the
    screen. Moreover, this program uses a new library function to
    suspend/resume communication with the controller for each PlayStation
    port.

.\pad\dungeon: Controller library sample <libpad>
    You may walk around in a dungeon with analog sticks of the controller.
    The actuators (vibrators) represent walking speed and/or impact.

.\press\tuto: Tutorial <libpress>
    MDEC step-by-step tutorial
    tuto1: simple VLC decode and MDEC on memory decompression
    tuto2: parallel execution of LoadImage() and DecDCTout()
    tuto3: simple on-memory movie operation
    tuto4: handshake using callback
    tuto5: parallel execution of LoadImage() and DecDCTout()
           using callback.
    tuto6: complete background on-memory movie decompression
    tuto7: fine tune-up for parameters

.\sound\3deffect: 3D sound using libsnd functions.
    This sample shows off 3d sound key on series.
    Best performance of 3d sound location can be heard with headphones.

.\sound\balls: combination of the sound and graphics
    Example of combination of sound and graphics.
    While pressing a button, balls bounce in the screen.
    When they hit against the wall, different sounds for each
    ball are generated.
    SEQ data is used as the background music.

.\sound\basic: Basic play 1
    SEQ/SEP data processing function sample.
    SEQ and SEP data may be played simultaneously.
    SEP data consists of 3-connected SEQ data.
```

.\sound\cdvol: SPU decoded data reading sample <libspu>  
 Music played on the CD is read as the "SPU decoded data" from the SPU in the background, and the volume is displayed with a graph (with the display of the peak level).

.\sound\mutual: Wave form data divided transfer sample <libsnd>  
 At a timing divided wave form data is read into the main memory, and transferred to the sound buffer. This process is repeated until all parts are transferred. As a result, 2 pieces of music may be played.

.\sound\simple: Basic play 2  
 Example using the SEQ data processing functions.  
 Playing SEQ endlessly.  
 On the screen the current tempo, volume, status (playing, pausing, etc.) are displayed.

.\sound\stream: SPU streaming sample program <libspu>  
 tuto1: The background is "balls".  
 tuto2: The background is "movie".

Sample using the SPU streaming library included in the basic sound library.  
 Performing playback and halt of 7-channel (14 voices) SPU streams by operating the control pad.  
 Displaying the state of SPU streaming on the screen.

.\sound\tuto: Basic usage of basic sound library <libspu>  
 tuto1: Pitch designation/key-on/key-off  
 According to the control-pad operations, a sound is played with variable pitches.  
 tuto2: Mute  
 Performing the sound generation, mute-on, mute-off  
 tuto3: SPU interrupt  
 Setting a interrupt in the middle of piano sound data.  
 When the piano sound starts, and the interrupt occurs, a sine wave is generated.  
 tuto4: Noise sound source  
 Generating a sine wave and noise by changing a pitch.  
 tuto5: Divided transfer of wave form data  
 Alternating divided transfer and sound generation after the transfer in 2 voices.  
 tuto6: Reverb  
 Generating a piano sound and designating 9 kinds of reverbs for the sound.

.\sound\xse: Auto-effect  
 Example using the sound utility functions.  
 Such effects as pitch-bend, auto-panning, auto-volume are applied to the keyed-on sounds.  
 By moving a thumb in the scroll bar, auto-panning, auto-volume, and pitch-bend can be applied.