dsnet 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 **v**

- Changes Since Last Release v
- Related Documentation v
- Typographic Conventions v
- Developer Support v

Overview **1**

Installation **2**

Common Options for the Various Clients **3**

dsreset **4**

dsedb - Debugger for EE **5**

dsidb - Debugger for IOP **7**

dsecons and dsicons **8**

dsefilesv and dsifilesv **10**

dsestart and dsistart **11**

dselist and dsilist **12**

Boot Parameters **13**

About This Manual

This is the Runtime Library Release 3.0 version of the *dsnet Overview* manual.

It describes the dsnet package, which contains a debugger, driver, manager, and other components for controlling a target (DTL-T10000).

Changes Since Last Release

- Descriptions of bit10 to bit23 and descriptions of the values of each bit were added to the EE boot parameter table in the "Boot Parameters" section. Also, the description of bit7 was changed and a description of where to refer to for EE kernel debugging features was added.
- Descriptions of the following sample settings were added to the FILEIO service sample settings table in the "Boot Parameters" section.

  % dsreset 0x00f00040 0

  % dsreset 0x000f0000 0

Related Documentation

The “dsedb/dsidb Command Reference” provides detailed information regarding dsedb and dsidb commands and options.

Note: the Developer Support Web site posts current developments regarding documentation 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:

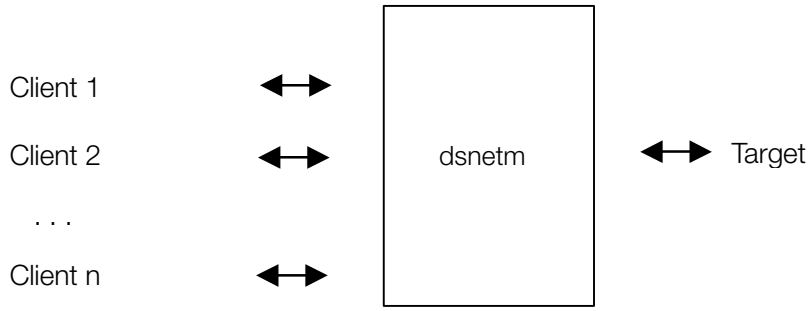
Order Information	Developer Support
Attn: Development Tools Manager Sony Computer Entertainment Europe 30 Golden Square London W1F 9LD, U.K. Tel: +44 (0) 20 7859-5000	E-mail: scee_support@ps2-pro.com Web: https://www.ps2-pro.com/ Developer Support Hotline: +44 (0) 20 7859-5777 (Call Monday through Friday,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

Overview

dsnet is a package containing a debugger, driver, manager, and other components for controlling a target (DTL-T10000/DTL-T10000H).

The dsnet package implements various types of functions using dsnetm, which manages data transfers to and from the target, and client programs such as the debugger. The following figure shows the relationship between dsnetm, client, and the target.

Figure 1



Sockets are used for exchanging data between clients and dsnetm.

Device files are used for direct data exchanges between dsnetm and the target.

Installation

If this package is installed correctly, the following files should be available under /usr/local/sce/bin.

Table 1

Command	Function
dsmcons	dsnet manager console
dsreset	reset client
dsstat	status client
dskill	kill client
dscomp	Communication port console
dsedb	EE debugger
dsecons	EE console server
dsefilesv	EE file server
dseping	EE ping client
dsestart	EE module start client
dselist	EE module list client
dsidb	IOP debugger
dsicons	IOP console server
dsifilesv	IOP file server
dsiping	IOP ping client
dsflash	flash ROM update script
dsistart	IOP module start client
dsilist	IOP module list client

Common Options for the Various Clients

The -d option, which is common to the various client programs, is explained below.

Table 2

-d Option	Host Name	Port Number
-d <host>	<host>	8510
-d <host>[/<port_offset>]	<host>	8510 + <port_offset>
-d <host>[:<port>]	<host>	<port>
-d /<port_offset>	default-host	8510 + <port_offset>
-d :<port>	default-host	<port>
-d <host>[:+<port_offset>](*)	<host>	8510 + <port_offset>
-d :+<port_offset>(*)	default-host	8510 + <port_offset>

(*) This can be specified to maintain compatibility with previous versions.

The -d option is used to specify the host name and port of the dsnetm that is the communication destination.

If the -d option is not specified, the default port (8510) on the same machine (default_host) as the one where the client program was executed will be assumed.

dsreset

dsreset is a client program for resetting the entire target.

Execution Method

dsreset [-d <host>[:<port>]]

Option

Only the common option -d is available.

dsedb - Debugger for EE

dsedb is an instruction-level debugger, not an EE source-level debugger. It supports stdio and file access from the target. For more information about how to use dsedb, please refer to the "dsedb/dsidb Command Reference."

Execution Method

dsedb [<options>] <cmd> [<arg>]...]

dsedb can also be directly executed by specifying the target program during startup such as:

dsedb run main.elf

Options

The main options of dsedb are shown below. Refer also to "dsedb Option Details" in the Command Reference.

Table 3

Option	Function
-d <host>[:<port>]	dsnetm host and port specification
-r	Reset target during startup
-nr	Do not reset target during startup
-nx	Do not execute autorun files during startup
-tex	Overwrite to enable target execution control
-ntex	Overwrite to disable target execution control
-ebootp <ebootp>	EE boot parameter initial value specification
-ibootp <ibootp>	IOP boot parameter initial value specification
-nokbd	Do not use keyboard when specifying <arg>...
-noexit	Do not exit when specifying <arg>...
-ncmv	Do not check version of manager
-p <port>	Remote console port specification

Usage Example

Normally, once dsedb is started, debugging is performed interactively as shown below.

```
$ dsedb -r                  //Starts up debugger
  ...
dsedb S> help               //Displays command help
  ...
dsedb S> pload main.elf     //Loads program
  ...
dsedb S> bp main            //Sets breakpoint at main
dsedb S> run                //Executes program
  ...
at=00010000 v0=...
```

```
  ...
dsedb S> db 0x100000      //Dumps address 0x1000000
  ...
dsedb S> step             //Performs step execution
  ...
dsedb S> cont             //Performs continuous execution
  ... ^C                  //Forcibly interrupts execution with CTRL-C
dsedb S> dr               //Displays registers
dsedb S> reset            //Resets target
dsedb S> quit             //Quits dsedb
$
```

dsidb - Debugger for IOP

dsidb is a debugger that has the IOP as its target. Although dsidb is used in almost an identical manner as dsedb, some commands are unique to dsidb and others have different specifications because of differences between the EE and IOP.

Execution Method

dsidb [<options>] <cmd> [<arg>]...]

dsecons and dsicons

dsecons and dsicons are console programs that serve as the destinations of TTYP I/O from the target program. The default handling of TTYP I/O for dsedb/dsidb is shown below.

- TTYP output (screen display)

  Display all ETTYP in units of lines. (dsedb)

  Display all ITTYP in units of lines. (dsidb)

- TTYP input (key input)

  Send key input using E0TTY when a prompt does not display while run, cont are executing. (desdb)

  Send key input using I0TTY when a prompt does not display while mstart is executing. (dsidb)

It is necessary to use dsecons and dsicons when you want a display without line units or when you want key input other than E0TTY(dsedb) or I0TTY(dsidb).

To use dsecons or dsicons and prohibit TTYP I/O handling by dsedb or dsidb, set the following line in the dsedb startup file (~/.dsedbrc or ./.dsedbrc) or the dsidb startup file (~/.dsidbrc or ./.dsidbrc).

```
option tty_mask=0
```

Execution Method

dsecons [-d <host>[:<port>]] [-e <escape>] -nokbd -ns [0-9k]...

dsicons [-d <host>[:<port>]] [-e <escape>] -nokbd -ns [0-9k]...

Options

TTYP supports the following 11 protocols each for the EE and for the IOP.

E0TTYP, E1TTYP, ..., E9TTYP, EKTTYP

I0TTYP, I1TTYP, ..., I9TTYP, IKTTYP

The argument [0-9k]... specifies which of these protocols are to be used. This argument is used as shown below.

Table 4

Specification	Protocols
dsecons 1 2 3	E1TTYP,E2TTYP,E3TTYP
dsecons 1-3 k	E1TTYP,E2TTYP,E3TTYP,EKTTYP

When this argument is specified, dsecons or dsicons receives only the specified protocols and displays the received data on the screen. When this argument isn't specified, dsecons or dsicons receives all 11 protocols and displays the received data on the screen.

The protocol that is specified first is used to send keyboard input data to the target. If this argument isn't specified, E0TTYP (dsecons) or IOTTYP (dsicons) is used to send keyboard input data.

If the -nokbd option is specified, all keyboard input other than <escape> is ignored.

If the -ns option is specified, error messages from the Remote File Access server are not displayed.

The -e option is the escape key specification. When the keyboard input is <escape>, the console program terminates. All other keyboard input is sent to the target program as standard input. The default value of

<escape> is CTRL-] (0x1d). To specify a control code for <escape>, prepend a carat "^" in front of the character. For example, specify "-e ^C".

dsefilesv and dsifilesv

dsefilesv and dsifilesv are file server programs for processing Remote File Access requests that are sent from the target program. Since dsedb and dsidb normally process Remote File Access requests, dsefilesv and dsifilesv need not be used except when you want to make the current directory during Remote File Access a fixed location that does not depend on the current directory of dsedb or dsidb.

To use dsefilesv or dsifilesv, you must inhibit Remote File Access processing by dsedb or dsidb. This can be done by setting the following line in the dsedb startup file (~/.dsedbrc or ./.dsedbrc) or the dsidb startup file (~/.dsidbrc or ./.dsidbrc).

```
option file_priority=-1
```

Execution Method

```
dsefilesv [-d <host>[:<port>]] [-i][-c <cmd>][-t <sec>]
dsifilesv [-d <host>[:<port>]] [-i][-c <cmd>][-t <sec>]
```

Unless the -i option is specified, dsefilesv or dsifilesv will operate in the background.

To terminate dsefilesv or dsifilesv, use the kill command.

Options

If the -c <cmd> option is specified, the specified shell command (<cmd>) will be executed, and the program will terminate when the following conditions are satisfied.

a. All files that were opened according to Remote File Access are closed.
b. The shell command (<cmd>) has ended.
c. The status described in a. has continued for the number of seconds (default: 1 second) that was specified according to -t <sec>.

dsestart and dsistart

dsestart starts execution of a liberx program module (ERX).

dsistart starts execution of an IOP program module (IRX).

Execution Method

dsestart [<options>] <fname> [<arg>...]

dsistart [<options>] <fname> [<arg>...]

These commands pass the specified arguments <arg>... to the specified program module <fname>, start execution of the module, and wait until a Removed or Resident termination report arrives from the IOP.

Options

Table 5

Option Name	Explanation
-d <host>[:<port>]	dsnetm host and port specification
-v	Display send/receive packet
-ncmv	Do not check version of manager
-nt	Do not handle E[0-9K]TTYP (for dsestart)
	Do not handle I[0-9K]TTYP (for dsistart)
-ns	Do not display errors from Remote File Access server
-w	Wait until LOADP connection can be made (default for dsistart)
-nw	Do not wait for LOADP connection to be made (default for dsestart)

dselist and dsilist

dselist displays a list of program modules that are loaded in memory in the liberx environment (ERX).

dsilist displays a list of program modules that are loaded in IOP memory (IRX).

Execution Method

dselist [-d <host>[:<port>]] [-v] [-l] [-ncmv]
dsilist [-d <host>[:<port>]] [-v] [-l] [-ncmv]

Options

Table 6

Option Name	Explanation
-d <host>[:<port>]	dsnetm host and port specification
-v	Display send/receive packet
-l	Display in long format
-ncmv	Do not check version of manager

Boot Parameters

The dsreset command and the dsedb/dsidb reset commands have two parameters. Both parameters are numeric values. The first numeric value is the EE boot parameter, and the second numeric value is the IOP boot parameter.

The EE boot parameter is currently defined as follows.

Table 7

Bit	Value	Explanation
bit 0	0x00000001	Workaround for the EE memory controller bug 0 Disable workaround for the EE memory controller bug 1 Enable workaround for the EE memory controller bug
bit 1	0x00000002	Startup from disk 0 Do not start up from disk 1 Start up from disk
bit 2	0x00000004	Reserved bit 0 Fixed
bit 3	0x00000008	Reserved bit 0 Fixed
bit 4	0x00000010	Component video output 0 RGB 1 Y/CrCb
bit 5	0x00000020	Memory size for EE kernel processing 0 128 MB 1 32 MB The user is not prohibited from accessing memory beyond 32 MB. When the EE kernel API LoadExecPS2 function is called, the memory space up to 32MB is cleared to 0. Also, all 128MB are cleared to zero at reset.
bit 6	0x00000040	Stack check function for each thread on the EE kernel 0 Disabled 1 Enabled When enabled, the stack is checked to see whether or not it has enough space to switch to another thread.
bit 7	0x00000080	TLB trap function 0 No TLB trap 1 Trap memory accesses above 32 MB through the TLB When the EE memory size is set to 32 MB, the TLB trap function is enabled. This allows memory accesses beyond 32 MB to be detected through the TLB.
bit 8	0x00000100	ELOADP driver control 0 Include ELOADP during liberx initialization 1 Do not use ELOADP

Bit	Value	Explanation
bit 9	0x00000200	Reserved bit 0 Fixed
bit 10	0x00000400	Reserved bit 0 Fixed
bit 11	0x00000800	Reserved bit 0 Fixed
bit 12	0x00001000	Reserved bit 0 Fixed
bit 13	0x00002000	Reserved bit 0 Fixed
bit 14	0x00004000	Reserved bit 0 Fixed
bit 15	0x00008000	Reserved bit 0 Fixed
bit 16	0x00010000	System call context check function 0 Enabled 1 Disabled
bit 17	0x00020000	System call argument check function 0 Enabled 1 Disabled
bit 18	0x00040000	Semaphore maxCount check function 0 Enabled 1 Disabled
bit 19	0x00080000	Reserved bit 0 Fixed
bit 20	0x00100000	Use of COP1(FPU) instructions in interrupt handlers 0 Can use 1 Cannot use
bit 21	0x00200000	Use of COP2(VPU0) instructions in interrupt handlers 0 Can use 1 Cannot use
bit 22	0x00400000	Reserved bit 0 Fixed
bit 23	0x00800000	Reserved bit 0 Fixed

For information on the workaround for the EE memory controller bug, see the “EE Memory Controller Bug” tech note on the developer support website. For information on starting up from disk, see the document "Disk Startup on the DTL-T10000." For information on EE kernel debugging functions, see the EE Kernel overview document.

The IOP boot parameter is currently defined as follows.

Table 8

Bit	Description
bit 0	Start EE (Initialize SIF) 1 Start EE (initialize SIF) 0 Do not start EE (do not initialize SIF)
bit 1	FILEIO Services 0 Perform FILEIO services 1 Do not perform FILEIO services
bit 2	Reserved bit 0 Fixed
bit 3	Reserved bit 0 Fixed
bit 4	TTY (printf output) 0 ON 1 OFF
bit 5	host file access 0 enabled 1 disabled
bit 6	Prohibit deci communication with the EE (use of commands such as dsedb will also not be possible)
bit 7	Memory size after IOP replacement 0 8MB 1 2MB If replacement not performed, size remains at 8 MB.
bit 8	Replacement mode 0 Development mode - Most current IOP debugging environment. - DTL-T10000 flash must match library version on disk. 1 Test mode - IOP debugger supported at level of release 1.3.4 functionality (1/19/00 release). - DTL-T10000 flash need not match library version on disk.

The FILEIO service is a service program that runs on the IOP that supports file I/O requests from the EE.

The following are some examples of typical settings.

Table 9

Sample setting	Description
% dsreset 0 0	Start EE, IOP simultaneously.
% dsreset 0 3	Start only IOP (not EE).
% dsreset 0 2	Do not perform FILEIO services.
% dsreset 2 0	Start EE, IOP simultaneously, start up from disk. Run IOP in 8 MB development mode.
% dsreset 0 80	Start EE, IOP simultaneously. Run IOP in 2 MB development mode.
% dsreset 2 100	Start EE, IOP simultaneously, start up from disk. Run IOP in 8 MB test mode.
% dsreset 2 180	Start EE, IOP simultaneously, start up from disk. Run IOP in 2 MB test mode.
% dsreset 0x00f00040 0	Enable all EE kernel debugging functions on startup
% dsreset 0x000f0000 0	Disable all EE kernel debugging functions on startup