

**Expert Sleepers  
OSC Control Manual  
v1.0.0**

Copyright © 2009 Expert Sleepers. All rights reserved.

This manual, as well as the software described in it, is furnished under licence and may be used or copied only in accordance with the terms of such licence. The content of this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Expert Sleepers. Expert Sleepers assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# Contents

<b>OSC</b>	<b>4</b>
<b>Received OSC Commands</b>	<b>4</b>
/ping s:returnUrl s:returnPath	4
/set i:param f:value	4
/get i:param s:returnUrl s:returnPath	4
/getAll s:returnUrl s:returnPath	5
/getNumParameters s:returnUrl s:returnPath	5
/isParameterUsed i:param s:returnUrl s:returnPath	5
/getInfo i:param s:returnUrl s:returnPath	5
/registerUpdate i:param s:returnUrl s:returnPath	5
/unregisterUpdate i:param s:returnUrl s:returnPath	6
/getAllRegistered	6
/exec s:func ...	6
/call s:func s:returnUrl s:returnPath ...	6
/startNote i:note f:velocity	7
/stopNote i:note f:velocity	7
<b>Version History</b>	<b>8</b>
<b>Contact</b>	<b>9</b>
<b>Acknowledgements</b>	<b>10</b>
<b>Lua</b>	<b>10</b>
<b>oscpack</b>	<b>10</b>

# OSC

Most Expert Sleepers plug-ins can be controlled via the Open Sound Control (OSC) protocol. Refer to the user manual of the individual plug-ins for information on their particular OSC implementation.

If you're new to OSC, start by visiting [opensoundcontrol.org](http://opensoundcontrol.org).

Two settings control what port the plug-in uses to listen on for OSC commands. One is the base OSC port, set in the plug-in's preferences window. The second is the OSC Port Offset control. If the port offset is set to something other than 'Off', then the two numbers are added together and the result used as the port number. E.g. if the base port is 6000 and the port offset is 1, then the plug-in will listen on port 6001.



## Received OSC Commands

In the documentation below, OSC parameters are prefixed with a string to indicate their type, as follows:

- s - string
- i - integer
- f - float
- b - boolean

All the examples assume that the plug-in is listening at address 10.0.0.1:6001.

### **/ping s:returnUrl s:returnPath**

Responds by sending a message back to the returnUrl and returnPath with the parameters

s:hosturl s:version

E.g.

```
/ping osc.udp://10.0.0.2:7000 "/foo"
```

replies to 10.0.0.2:7000 with

```
/foo osc.udp://10.0.0.1:6001 "<plug-in name> <plug-in version>"
```

E.g.

```
/foo osc.udp://10.0.0.1:6001 "oomingmak 1.0.0"
```

### **/set i:param f:value**

Sets the value of parameter 'param' to 'value'.

### **/get i:param s:returnUrl s:returnPath**

Responds by sending a message back to the returnUrl and returnPath with the parameters

i:param f:value

where 'value' is the value of parameter 'param'. E.g.

```
/get 14 osc.udp://10.0.0.2:7000 "/foo"
```

replies to 10.0.0.2:7000 with (assuming parameter 14 has the value 64.0)

```
/foo 14 64.0
```

### **/getAll s:returnUrl s:returnPath**

Behaves exactly as if a /get message was received for every parameter.

### **/getNumParameters s:returnUrl s:returnPath**

Responds by sending a message back to the returnUrl and returnPath with the parameters

i:numParameters

where 'numParameters' is the total number of parameters defined by the plug-in. E.g.

```
/getNumParameters osc.udp://10.0.0.2:7000 "/foo"
```

replies to 10.0.0.2:7000 with (assuming the plug-in has 84 parameters)

```
/foo 84
```

Note that there can be 'gaps' in the array of parameters - see isParameterUsed below.

### **/isParameterUsed i:param s:returnUrl s:returnPath**

Responds by sending a message back to the returnUrl and returnPath with the parameters

i:param b:isUsed

where 'isUsed' is 'true' if parameter 'param' is used, and 'false' otherwise. Parameters that are not used should not be used for any other call e.g. the getInfo call below.

### **/getInfo i:param s:returnUrl s:returnPath**

Responds by sending a message back to the returnUrl and returnPath with the parameters

i:param f:minValue f:maxValue f:defaultValue s:name i:unit

where 'minValue' and 'maxValue' are the minimum and maximum values that parameter 'param' can take, 'defaultValue' is the default value of the parameter, 'name' is the name of the parameter, and 'unit' is a value that indicates the unit of the parameter (e.g. Hz, db, seconds). The unit is one of the values defined by Apple's Audio Unit specification.

### **/registerUpdate i:param s:returnUrl s:returnPath**

Requests that when the parameter 'param' changes, a message is sent back to the returnUrl and returnPath with the parameters

i:param f:value

where the returned parameters have the same meaning as for the /get command (above).

### **/unregisterUpdate i:param s:returnUrl s:returnPath**

Cancels a request made via /registerUpdate (above).

### **/getAllRegistered**

Behaves exactly as if every parameter registered for updates with /registerUpdate had changed. A message will be sent for every such parameter.

### **/exec s:func ...**

Executes the Lua function 'func', which is assumed to be defined by the MIDI & OSC scripting system (see the separate documentation, available from the Expert Sleepers website). OSC parameters following 'func' are passed through to the Lua function, as can best be managed given the varying limitations of the two. Specifically, the following table describes the mapping from OSC types to Lua types:

<b>OSC</b>	<b>Lua</b>
bool	bool
float	number
double	number
int32	number
int64	number
string	string
nil	nil

### **/call s:func s:returnUrl s:returnPath ...**

As /exec, but also responds to the returnUrl and returnPath with the results of the Lua function call. The following table describes the mapping from Lua return values to OSC types:

<b>Lua</b>	<b>OSC</b>
number	float
string	string

Lua types not in the above table are not handled.

### **/startNote i:note f:velocity**

Starts playback of a note - equivalent of a MIDI note on message.

'note' is the MIDI note number; 'velocity' should be between 0.0 and 1.0. E.g.

```
/startNote 60 0.7
```

### **/stopNote i:note f:velocity**

Stops playback of a note - equivalent of a MIDI note off message.

'note' is the MIDI note number; 'velocity' should be between 0.0 and 1.0. E.g.

```
/stopNote 60 0.0
```

# Version History

**1.0.0 19/5/2009**

- Initial version.



# Contact

The Expert Sleepers website is here:

<http://www.expert-sleepers.co.uk/>

Or you can email

[info@expertsleepers.co.uk](mailto:info@expertsleepers.co.uk)

Or you can use the forum, which is here:

<http://www.kvraudio.com/forum/viewforum.php?f=85>

# Acknowledgements

The software described in this manual makes use of the following open source projects. The author is greatly indebted to them for their efforts and generosity.

Below are reproduced the various copyright notices and disclaimers that accompany these software projects, in accordance with their terms of use.

## Lua



Copyright (C) 1994-2008 Lua.org, PUC-Rio.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## oscpack

oscpack -- Open Sound Control packet manipulation library  
<http://www.audiomulch.com/~rossb/code/oscpack>

Copyright (c) 2004 Ross Bencina <[rossb@audiomulch.com](mailto:rossb@audiomulch.com)>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Any person wishing to distribute modifications to the Software is requested to send the modifications to the original developer so that they can be incorporated into the canonical version.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.