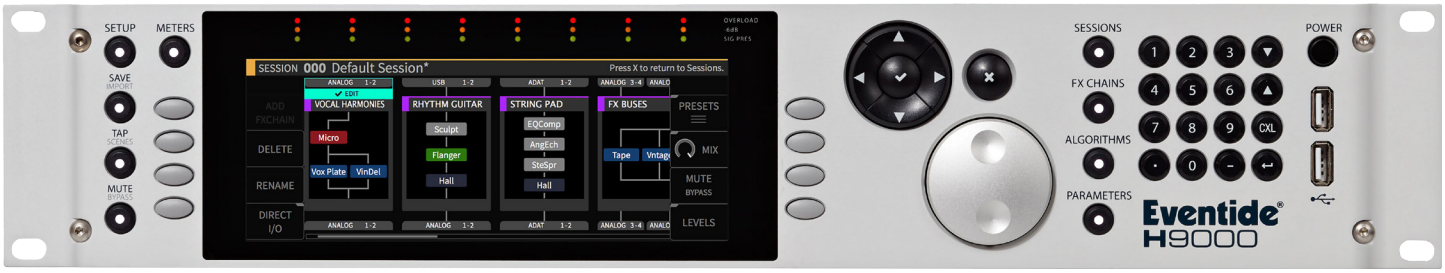
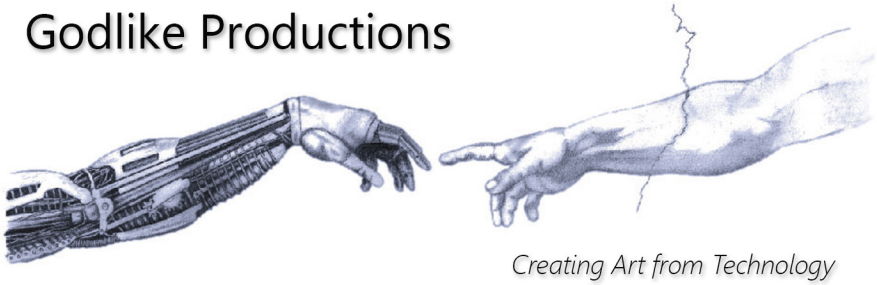


VSIG Module

Unofficial Reference

Eventide®

Godlike Productions



Revision 1.11

Godlike Productions Contact

All Customers:

Godlike Productions
PO Box 1520
Midland DC, WA, 6936, AUSTRALIA

Email: info@godlike.com.au

Eventide Website: https://www.eventideaudio.com/

https://godlike.com.au

support@godlike.com.au

https://www.facebook.com/GodlikeAustralia

www.twitter.com/GodlikeAust

https://www.youtube.com/GodlikeAu

Contents

Chapter 1 - Bridge

Chapter 2 - Control Math

Chapter 3 - Control Process

Chapter 4 - Delay

Chapter 5 - Detector

Chapter 6 - Dynamic

Chapter 7 - External

Chapter 8 - Filter

Chapter 9 - Interface

Chapter 10 - Math

Chapter 11 - Miscellaneous

Chapter 12 - Mixer

Chapter 13 - Oscillator

Chapter 14 - Pitch Shift

Chapter 15 - Reverb

Chapter 16 - System

Chapter 17 - Change Log

Chapter 1 - Bridge

a_fltr_c(afc) - Audio to Control (Filtered)

Description

This module converts an audio signal into a control signal. Before doing the conversion, the audio signal is lowpass filtered. This module is useful in allowing an audio input signal or internal oscillator module to control signal processing parameters that are only accessible via control signals.

Godlike Productions Comments

Specifiers

Header	Description
time_constant	The time constant of the filter that is used on the audio signal before it is converted to a control signal. It is specified in seconds.

Audio Inputs

Header	Description
in	Audio Input

Audio Outputs

Header	Description

Control Inputs

Header	Description

Control Outputs

Header	Description
out	The resultant control output signal

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
time_constant	0	100	0	0.001
	ramp_time			
in				
MONO				
AUDIOINPUT				
out				
CONTROLOUTPUT	out			

a_to_c(a_c) - Audio to Control (Non-Filtered)

Description

This module converts an audio signal into a control signal. No filtering is done on the audio signal. This module is useful in allowing an audio input signal or internal oscaillator module to control signal processing parameters that are only accessible via control signals.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	Audio Input

Audio Outputs

Header	Description

Control Inputs

Header	Description

Control Outputs

Header	Description
out	The resultant control output signal

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
CONTROLOUTPUT	out			

c_fltr_a(cfa) - Control to Audio (Filtered)

Description

This module converts a control signal into an audio signal. After doing the conversion, the audio signal is lowpass filtered. This module is useful in taking a user input, like a button press or pedal input, and controlling the modulation of a particular module. The lowpass filter built into the module will tend to smooth out the "roughness" associated with control signals.

Godlike Productions Comments

Specifiers

Header	Description
time_constant	The time constant of the filter that is used on the control signal after it is converted to an audio signal. It is specified in seconds.

Audio Inputs

Header	Description

Audio Outputs

Header	Description
out	The resultant audio output

Control Inputs

Header	Description
out	The control signal to be converted

Control Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
time_constant	0	100	0	0.01
	ramp_time			
state_clear	0	1	0	1
INT	state_clear			
in	-1	1		
CONTROLINPUT	%n			
out				
MONO				
AUDIOOUTPUT				

c_to_a(c_a) - Control to Audio (Non-Filtered)

Description

This module converts a control signal into an audio signal. No filtering is done on the control signal. This module is useful in taking a user input, like a button press or pedal input, and controlling the modulation of a particular module.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description

Audio Outputs

Header	Description
out	The resultant audio output

Control Inputs

Header	Description
out	The control signal to be converted

Control Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
MONO				
in	-1	1		
CONTROLINPUT	%n			

c_tweak_a(c_a) - Control to Audio (Non-Filtered)

Description

This module converts a control signal into an audio signal. No filtering is done on the control signal. Before conversion, the control signal is divided by 1024. The advantage is that you can generate an audio signal with finer precision.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description

Audio Outputs

Header	Description
out	The resultant audio output

Control Inputs

Header	Description
out	The control signal to be converted

Control Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
MONO				
in	-1024	1024		0.0001
CONTROLINPUT	%n			

Chapter 2 - Control Math

c_abs(cab) - Control rate absolute value.

Description
This module implements a control rate absolute value.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
input	input.

Control Outputs

Header	Description
output	abs of input.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
input				

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	in			
output				
CONTROLOUTPUT	out			

c_adder(add) - Add Control Signals

Description

This module adds together a specified number of control signals. This is needed in creating patches where more than one source can affect a single parameter.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	Specifies how many inputs are to be added together. At least two inputs must be specified. The maximum allowed is 32.

Control Inputs

Header	Description
in1,in2,...inN	The input control signals that are to be added together.

Control Outputs

Header	Description
out	The sum of all of the input control signals

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	2	32	2	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	number_inputs			
in				
		@ninputs		
CONTROLINPUT	in~n			
out				
CONTROLOUTPUT	out			

c_and(and) - Logical And Control Signals

Description

This module executes a logical AND of two control signals. A value of 1.0 is defined to be true and a value of 0.0 is defined to be false. If both inputs have a value of 1.0 or greater, the output is set to 1.0, otherwise it is set to 0.0.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1,in2	The input control signals that are to be logically anded together.

Control Outputs

Header	Description
out	The logical and of the input control signals

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2				
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			

c_bound(bnd) - Control signal bounder

Description

This module limits an input to an upper and lower bound If min > max output value is the minimum value

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
input	The input control
maximum	The maximum value the output will be
minimum	The minimum value

Control Outputs

Header	Description
out	The bounded output signal

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
input				

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	input			
minimum				
CONTROLINPUT	minimum			
maximum				
CONTROLINPUT	maximum			
out				
CONTROLOUTPUT	out			

c_cmp2(cmp) - Compare Two Control Signals

Description

This module compares the value of two input control signals. There is an output for each possible equality or inequality. If (in1 condition in2) is true, a value of 1.0 is output. If (in1 condition in2) is false, a value of 0.0 is output.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1,in2	The input control signals to be compared.

Control Outputs

Header	Description
eq	1 if (in1 = in2), else 0
ne	1 if (in1=/= in2), else 0
ge	1 if (in1 >= in2), else 0
gt	1 if (in1 > in2), else 0
le	1 if (in1 <= in2), else 0
lt	1 if (in1 < in2), else 0

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2				
CONTROLINPUT	in2			
eq				
CONTROLOUTPUT	eq			
ne				
CONTROLOUTPUT	ne			
gt				
CONTROLOUTPUT	gt			
lt				
CONTROLOUTPUT	lt			
ge				
CONTROLOUTPUT	ge			
le				
CONTROLOUTPUT	le			

c_comparator(cmp) - Compare Two Control Signals

Description

This module compares the value of two input control signals. If the first is greater than that of the second, a value of 1.0 (TRUE) is output. Otherwise, a value of 0.0 (FALSE) is output.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1,in2	The input control signals to be compared. in1 is the “positive” input.

Control Outputs

Header	Description
out	The logical and of the input control signals

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2				
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	

c_constant(con) - Constant Control Signal Output

Description

This module generates a constant value control signal output. This is often necessary to generate a bias value in various control schemes.

Godlike Productions Comments

Specifiers

Header	Description
value	The value the output is to be set to.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	The constant output.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
value	-32768	32767	0	0.001

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	value			
out				
CONTROLOUTPUT	out			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_db2lin(dbl) - Convert dB to linear.

Description

This module converts dB valued input control signals to linear valued output control signals. Both input and output signals are considered 32bit fixed point values with the binary point bethween bit 15 and 16. Values between 32767 (90dB) and 2^-15 (-90dB) can be handled.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
dbvalue	dB value of the linear input.

Control Outputs

Header	Description
linvalue	Linear value to be converted to dB.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
dbvalue				

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	in~n			
linvalue				
CONTROLOUTPUT	linvalue			

c_divide(div) - Control Signal Divider

Description

This module divides one control signal by another. If the divisor (in2) is zero, the output is set to maximum value (+/-32767.0) Like the other control signal math operators, this is useful for creating various user interactions with the DSP parameters.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1	The dividend.
in2	The divisor.

Control Outputs

Header	Description
out	The value of in1/in2.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2				
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			

c_ftop(ftp) -

Description
This module converts its input signal from frequency to pitch (cents)

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	Input control signal to be converted

Control Outputs

Header	Description
out	Frequency -> Pitch

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	16	32767		

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
CONTROLOUTPUT	out			

c_graph (gph) - Graphical control input array editor

Description

This module allows you to edit an arbitrary number of points (up to 32) on a display graph control. It also provides an offset control input to add/subtract a value from all numbers before results are output. The screen width of the control is also variable. Displayed label (x-values) can be specified on control inputs created for each point. See also C_DISPLAY.

Godlike Productions Comments

Specifiers

Header	Description
8_char_name	Control name; appears in first line of graph display
arrow_text	Text to be displayed between point x and y. The â€˜^â€™ character will cause a down-arrow to be displayed in its place.
format_labels	These are formatting strings for the x and y values displayed. The standard "%.0f" style formatting applies.
format_points	These are formatting strings for the x and y values displayed. The standard "%.0f" style formatting applies.
number_points	(1->32) Number of control inputs (and outputs) = points on the graph.
point_1..N	(point min->point max) The default y-values for the points.
point_max	(-32768->32767) Max values for editing & outputting points.
point_min	(-32768->32767) Min values for editing & outputting points.
point_res	(0->1) Point resolution. (in .001 increments)
screen_width	(2->4) Screen width of control in quarters. (2=half screen, 4=full)

Control Inputs

Header	Description
label1..N	(-32768->32768) Label (x-value) to be displayed by the format labels string when that point is being edited.
offset	Value added to points before being displayed & outputted. The internal value of the point does not change.

Control Outputs

Header	Description
out1..N	(internal y-value N + offset) bound by point min & max

User Objects

Header	Description
	Actual displayed graph control

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
width	2	4		
	screen_width			
statics0				
STRING	8_char_name			
statics1				
STRING	arrow_text			
sformat0				
STRING	format_labels			
sformat1				
STRING	format_points			
npoints	1	32		
INT	number_points			
gmin	-32768	32767	0	0.001
FLOAT	point_min			
gmax	-32768	32767	0	0.001
FLOAT	point_max			
gres	0	1	1	
FLOAT	point_res			
point	@gmin	@gmax	0	@gres
		@npoints		
FLOAT	point~n			
offset				
CONTROLINPUT	offset			
label				
		@npoints		
CONTROLINPUT	label~n			
out				
		@npoints		

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

c_graph

c_lin2db(ldb) - Convert linear to dB.

Description

This module converts linear valued input control signals to dB valued output control signals. Both input and output signals are considered 32bit fixed point values with the binary point between bit 15 and 16. Values between 32767 (90dB) and 2^-15 (-90dB) can be handled.

Godlike Productions Comments

If this block feeds an interface module (for example level), ensure that this never receives 0, or it will crash emote. Place a c_bound before this block with a min value of 0.0000001 (or similar).

Specifiers

Header	Description

Control Inputs

Header	Description
linvalue	Linear value to be converted to dB.

Control Outputs

Header	Description
dbvalue	dB value of the linear input.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
linvalue				
	in~n			
dbvalue				
CONTROLOUTPUT	dbvalue			

c_log(log) - Find the base(in2) logarithm of in1

Description

This module finds the base(in2) logarithm of in1. In1 and in2 are restricted to positive numbers. If the resulting value is greater than +/- 32767, the output value is limited.

Godlike Productions Comments

If this block feeds an interface module (for example level), ensure that this never receives 0, or it will crash emote. Place a c_bound before this block with a min value of 0.0000001 (or similar).

Specifiers

Header	Description

Control Inputs

Header	Description
in1	The input control signal to the logarithm.
in2	The control signal representing the base of the logarithm.

Control Outputs

Header	Description
out	The result of log base (in2) of in1.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2	1	32767		
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			

c_master(mst) - Create N number scaled control signals

Description

This module creates n number outputs of scaled numbers

Godlike Productions Comments

Specifiers

Header	Description
instart	first (ie lowest if lower) input value
instop	last (ie highest if higher) input value
noutputs	Specifies how many outputs are to be created. At least one output must be specified. The maximum allowed is 32.

Control Inputs

Header	Description
input	input value to have % taken of
offsetN	Offset values to be added after % processing but before bounds checking
outstartN	first (ie lowest if lower) output value
outstopN	last (ie highest if higher) output value

Control Outputs

Header	Description
outputN	(% of scaled input applied to output range + offset) within scale tolerances

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
noutputs	1	32	1	1

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	number_outs			
instart	-32768	32767	0	0.0001
FLOAT	in_start			
instop	-32768	32767	0	0.0001
FLOAT	in_stop			
in				
CONTROLINPUT	master_in			
outstart	-32768	32767	0	0.0001
		@noutputs		
CONTROLINPUT	out_start~n			
outstop	-32768	32767	0	0.0001
		@noutputs		
CONTROLINPUT	out_stop~n			
offset	-32768	32767	0	1
		@noutputs		
CONTROLINPUT	offset~n			
output				
		@noutputs		
CONTROLOUTPUT	output~n			

c_master_taper(mst) - Create N number taper-scaled control signals

Description

This module creates n number outputs of taper-scaled numbers

Godlike Productions Comments

Specifiers

Header	Description
alphaN	0 to 32767: tapering parameters for each output. alphaN < 1: smaller values give increasingly more resolution to the top of the range and less resolution to the bottom. alphaN = 1: linear relationship b/t input / output (same as c_master module) alphaN > 1: larger values give increasingly more resolution to the bottome of the range and less resolution to the top.
instart	first (ie lowest if lower) input value
instop	last (ie highest if higher) input value
noutputs	Specifies how many outputs are to be created. At least one output must be specified. The maximum allowed is 32.

Control Inputs

Header	Description
input	input value to have % taken of
offsetN	Offset values to be added after % processing but before bounds checking
outstartN	first (ie lowest if lower) output value
outstopN	last (ie highest if higher) output value

Control Outputs

Header	Description
outputN	(% of scaled input applied to output range + offset) within scale tolerances

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
noutputs	1	32	1	1
	number_outs			
instart	-32768	32767	0	0.0001
FLOAT	in_start			
instop	-32768	32767	0	0.0001
FLOAT	in_stop			
alpha	0	32767	1	0.0001
		@noutputs		
FLOAT	alpha~n			
in				
CONTROLINPUT	master_in			
outstart	-32768	32767	0	0.0001
		@noutputs		
CONTROLINPUT	out_start~n			
outstop	-32768	32767	0	0.0001
		@noutputs		
CONTROLINPUT	out_stop~n			
offset	-32768	32767	0	1
		@noutputs		
CONTROLINPUT	offset~n			
output				
		@noutputs		
CONTROLOUTPUT	output~n			

c_minmax(max) - gives the maximum and minimum signal.

Description

Godlike Productions Comments

Specifiers

Header	Description
ninputs	number of inputs

Control Inputs

Header	Description
in1...inN	The input control signals.

Control Outputs

Header	Description
max	The greatest of all the inputs.
mn	The smallest of all the inputs.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	2	32	2	1

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	number_inputs			
in				
		@ninputs		
CONTROLINPUT	in~n			
minimum				
CONTROLOUTPUT	minimum			
maximum				
CONTROLOUTPUT	maximum			

c_mod(cmod) - Control Signal Modulus

Description

This module performs modular arithmetic upon the control input IN. Undefined results (i.e. 5 MOD 0) are output as zero.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input control signal to perform modular arithmetic on.
modulo	The input control signal representing modulo.

Control Outputs

Header	Description
out	The remainder of in/modulo

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	input_one			
modulo				
CONTROLINPUT	modulo			
out				
CONTROLOUTPUT	output			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_multiply(mul) - Multiply Two Control Signals

Description

This module multiplies two control signals. If the resulting value is greater than +/- 32767, the output value is limited.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1,in2	Input signals that are going to be multiplied together. Range: -32768.0 to 32767.0.

Control Outputs

Header	Description
out	The result of multiplying the two signals. If the resultant value exceeds -32768 or +32767, it will be limited at those values.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				
	in1			
in2				
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			

c_not(not) - Logical NOT of Control Signal

Description

This module executes a logical NOT of the input control signal. An input value 1.0 or greater yields an output value of zero, otherwise the output is 1.0.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input control signal to be logically inverted.

Control Outputs

Header	Description
out	The logical not of the input control signal

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
CONTROLOUTPUT	out			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_or(or) - Logical OR Control Signals

Description

This module executes a logical OR of two control signals. A value of 1.0 is defined to be true and a value of 0.0 is defined to be false. If either input has a value of 1.0 or greater, the output is set to 1.0, otherwise it is set to 0.0.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1,in2	The input control signals that are to be logically ORed together.

Control Outputs

Header	Description
out	The logical OR of the input control signals

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2				
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			

c_power(pow) - Raise In1 to the Power of In2

Description

This module raises control signal in1 to the power of control signal in2. If the resulting value is greater than +/- 32767, the output value is limited.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1	The base control signal.
in2	The exponent control signal.

Control Outputs

Header	Description
out	The result of in1 to the power of in2.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2				
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_ptof(ptf) -

Description
This module converts its input signal from pitch (cents) to frequency

Godlike Productions Comments
This block accepts inputs of cents, but also accepts negative numbers, which will lower the frequency. -12800 cents is approximately 0.01Hz

Specifiers

Header	Description

Control Inputs

Header	Description
in	Input control signal to be converted

Control Outputs

Header	Description
out	Pitch -> Frequency

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	-32768	13162	0	

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
CONTROLOUTPUT				

c_quantize(qnt) - Control quantizer

Description

This module will quantize a control signal into discrete steps. The user specifies the stepsize and origin.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
input	The input control
origin	An absoute step
stepsize	distance between steps

Control Outputs

Header	Description
out	The quantized output signal

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
input				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	input			
stepsize	0.0001			
CONTROLINPUT	stepsize			
origin				
CONTROLINPUT	origin			
out				
CONTROLOUTPUT	out			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	

c_random(rnd) - Create Random Number Control Signals

Description

This module creates n number outputs of random numbers

Godlike Productions Comments

Specifiers

Header	Description
max_number	Highest random number generated
min_number	Lowest random number generated
noutputs	Specifies how many outputs are to be created. At least one output must be specified. The maximum allowed is 32.

Control Inputs

Header	Description
delay	# of cycles to skip between rands
reset	(re)Seed random number generator

Control Outputs

Header	Description
randN	Our generated random numbers

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
noutputs	1	32	1	1

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	number_randoms			
min_number	-32768	32767	-32768	1
INT	min_number			
max_number	-32768	32767	32767	1
INT	max_number			
reset				
CONTROLINPUT	reset			
delay	0	32767	0	1
CONTROLINPUT	reset			
rand				
		@noutputs		
CONTROLOUTPUT	rand~n			

c_sample_rate - Sample Rate Signal Output

Description

This module outputs the current sample rate of the host environment in kHz.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description

Control Outputs

Header	Description
out	The sample rate output in kHz.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	out			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_sincos(csc) - Sine,cosine

Description
This module will produce a the sine and cosine of its input at its outputs

Godlike Productions Comments
This block expects ange in turns. Ie an input of 2 is the same as 360 degrees. 0.25 is the same as 90 degrees. Alternative way is it multiplies the input by 2 x Pi and that number is in radians. The block will take the sine and cosine of the input x 2 x pi radians.

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input

Control Outputs

Header	Description
out	The output

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	-1	1	0	0.1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
sine				
CONTROLOUTPUT				
cosine				
CONTROLOUTPUT				

c_sqrt(csq) - Square root

Description

This module will produce a the square root of its input at its output

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input

Control Outputs

Header	Description
out	The output

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	0	32767	1	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
CONTROLOUTPUT	out			

c_subtract(sub) - Subtract Two Control Signals

Description

This module computes the difference of two control signals. The output is equal to in1-in2.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1,in2	The input control signals that are to be subtracted.

Control Outputs

Header	Description
out	The output equal to in1 - in2.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	in1			
in2				
CONTROLINPUT	in2			
out				
CONTROLOUTPUT	out			

c_xor(xor) - Logical Exclusive OR Control Signals

Description

This module execute a logical Exclusive OR of two control signals. A value of 1.0 is defined to be true and a value of 0.0 is defined to be false. If one of the inputs is TRUE and the other is FALSE, the output is set to TRUE, otherwise it is set to FALSE.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in1,in2	The input control signals that are to be logically xor'ed together.

Control Outputs

Header	Description
out	The logical xor of the input control signals

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	in1			
in2				
CONTROLINPUT	in2			
out	0	1		1
CONTROLOUTPUT	out			

surroundfeedbackcontroller(sfc) - works with the surround system positioner and hyperplex module to set up a feedback scheme for a surround effect

Description

The Surround Feedback Controller allows you to set up a feedback scheme for a surround effect. When you have a multi channel effect the feedback/crossfeed setup must be a unitary matrix to avoid oscillation. This can be done using the Hyperplex module, but the parameter setup is not obvious. The Surround Feedback Controller module allows you to specify an N channel system, takes the location of each speaker in the playback system, and generates the mixing angles for the hyperplex module to create a natural sounding feedback.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
angle(n)	The left/right angle of the nth speaker in degrees. 0 degrees is straight ahead, negative values move the speaker toward the left, positive vaues move the speaker toward the right.
elevation(n)	The up/down angle of the nth speaker in degrees. 0 degrees is straight ahead, negative values move the speaker down, positive values move the speaker up.

Control Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
numchannels	2	32	8	1
	num_channels			
numthetas	@numchan-nels/1/#sub/@ numchan-nels/2/#div/#mul	@numchan-nels/1/#sub/@ numchan-nels/2/#div/#mul	@numchan-nels/1/#sub/@ numchan-nels/2/#div/#mul	@numchan-nels/1/#sub/@ numchan-nels/2/#div/#mul
INT	num_thetas			
angle	-180	180	0	1
	degrees	@numchannels		
CONTROLINPUT	angle~n			
elevation	-90	90	0	1
	degrees	@numchannels		
CONTROLINPUT	elevation~n			
mixingangle				
		@numthetas		
CONTROLOUTPUT	mixing_angle~n			
obj				
USEROBJECTPARENT	obj			

Chapter 3 - Control Process

c_1shot(1sh) - One-shot

Description

This module will produce a 1 each time the input goes from <1 to >=1. It will then drop back to 0 and remain there until the input goes below 1 and back up >=1 again.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input control which we are converting to a one-shot control signal.

Control Outputs

Header	Description
out	The one-shot result.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
	in			
out				
CONTROLOUTPUT	out			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_adsr(cnv) - Control Signal ADSR

Description

This module implements an ADSR-type envelope generator for control signals. It has 3 states: - Attack: Rising until it reaches max level (1). - Decay/sustain: falling until it reaches the sustain level (and waiting for the gate_off). - Release: falling back to the min level (0) until another gate_on.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
Attack	This input controls the attack rate.
Decay	This input controls the decay rate.
Gate	Value <1 is considered to be "0". Value >= 1 is considered to be "1". A rising edge triggers attack mode. A falling edge triggers release mode.
Release	This input controls the attack rate.

Control Outputs

Header	Description
out	The current value of the counter.
state	The state the ADSR is in: 0 = release 1 = attack 2 = decay/sustain

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
attack	0	1		0.01
	attack			
decay	0	1		0.01
CONTROLINPUT	decay			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
sustain	0	1		0.01
CONTROLINPUT	sustain			
release	0	1		0.01
CONTROLINPUT	release			
gate	0	1		1
			off,on	
CONTROLINPUT	gate			
out				
CONTROLOUTPUT	out			
state				
CONTROLOUTPUT	state			

c_change(chg) - Detect Change in Input

Description

This module will produce a 1 each time the input changes from the previous value. If the input value has not changed from the previous value, the output returns to 0.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input value on which to detect a change.

Control Outputs

Header	Description
out	The result (0 or 1).

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	-32768	32767		1
	%n			
out				
CONTROLOUTPUT	out			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_count2(cnt) - Control Signal Counter

Description

This module implements a control signal counter. It will count up to a specified value and then stop. The counting mechanism is controlled by a "clock" control signal input. Whenever the clock transitions from 0 to 1, the counter will increment by a preset amount. The counter has an output corresponding to its current counting value, and an output that indicates if the maximum count has been reached. This module may find use in complex control schemes that may require delayed reactions to user inputs. For example, a patch might be created that causes one sweep to be triggerd a second after the user presses a button on the front panel.

Godlike Productions Comments

I don't know if there are any difference from c_counter

Specifiers

Header	Description

Control Inputs

Header	Description
clock	This input controls the counting of the clock mechanism. Each time this input transitions from below 0.5 to above 0.5, the increment value is added to the current count.
incr	This controls how much is added to the count value for each transition of the clock input.
maxcount	This determines the maximum allowed value of the counter. The counter will stop once it has reached this value. If counting down (incr<0), maxcount is effectively 0.
reset	When this input is greater than 0.0, And a clock occurs, the counter is reset.

Control Outputs

Header	Description
out	The current value of the counter.
timeout	Set to 1.0 if at the maxcount, otherwise, set to zero.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clock				
	clock			
incr	-32000	32000	1	1
CONTROLINPUT	increment			
maxcount		32000	1	1
CONTROLINPUT	maxcount			
reset				
CONTROLINPUT	reset			
out				
CONTROLOUTPUT	out			
timeout				
CONTROLOUTPUT	timeout			

c_counter(cnt) - Control Signal Counter

Description

This module implements a control signal counter. It will count up to a specified value and then stop. The counting mechanism is controlled by a "clock" control signal input. Whenever the clock transitions from 0 to 1, the counter will increment by a preset amount. The counter has an output corresponding to its current counting value, and an output that indicates if the maximum count has been reached. This module may find use in complex control schemes that may require delayed reactions to user inputs.For example, a patch might be created that causes one sweep to be triggerd a second after the user presses a button on the front panel.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
clock	This input controls the counting of the clock mechanism.Each time this input transitions from below 0.5 to above 0.5, the increment value is added to the current count.
incr	This controls how much is added to the count value for each transition of the clock input.
maxcount	This determines the maximum allowed value of the counter.The counter will stop once it has reached this value.
reset	When this input is greater than 0.0, and a clock occurs, the counter is reset.

Control Outputs

Header	Description
out	The current value of the counter.
timeout	Set to 1.0 if the maxcount has been reached, otherwise, set to zero.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clock				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	clock			
incr	-32000	32000	1	1
CONTROLINPUT	increment			
maxcount		32000	1	1
CONTROLINPUT	maxcount			
reset				
CONTROLINPUT	reset			
out				
CONTROLOUTPUT	out			
timeout				
CONTROLOUTPUT	timeout			

c_curve control signal map

Description

An arbitrary relationship between an input value and an output value. This relationship is called a map. Your map is formed by ajusting points along the map. You specify how many points. The points are placed equally along the input. For each input point, you adjust the output value for that point. For an input value that is between two points, the output is found on a straight line between the points. (Linear interpolation) This map works on control signals that are between -1 and 1. Use this module for special tapers on knobs or external controls.

Godlike Productions Comments

Specifiers

Header	Description
npoint	Specifies how many data points there are. The minimum is 1 and the max is 32.

Control Inputs

Header	Description
in	value to be mapped. This should be between -1 and 1

Control Outputs

Header	Description
out	The output. Also between -1 and 1.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	-1	1	0	0.01
	in			
npoints	1	32	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
INT	number_points			
point	-1	1	0	0.01
		@npoints		
FLOAT	point_~n			
out	-1	1		
CONTROLOUTPUT	out			

c_dtimer(cnt):

Description
A control rate running timer that reports running time and previous stop time in seconds.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
clock	timer trigger input, starts or continues the timer depending on mode. Triggers on the rising edge. Timer runs when clock is >= 1.
mode	0-1, selcts the way the timer works: 0 - "restart" : restarts the timer at zero when triggered to start again 1 - "continue" : continues adding time to the previously stored end_time

Control Outputs

Header	Description
stoptime	the previous value where the timer was stopped
time	difference between start and stop time

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clock				
	clock			
mode				
			restart,continue	
CONTROLINPUT	mode			
time_out				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	time_out			
stoptime_out				
CONTROLOUTPUT	stoptime_out			

c_flop(flp) - RS (T) Flip-flop

Description

This module will remain in a constant state while the control inputs are low. It may switch states, depending on which of the control inputs goes high. The three inputs are reset, set, and toggle. When SET>=1, the output will go to 1. When RESET>=1 (provided SET<1), the output will go to 0. When TOGGLE>=1 (provided SET and RESET are both <1), the output will toggle from 0 to 1, or from 1 to 0.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
reset	The input control that sets the output to 0.
set	The input control that sets the output to 1.
toggle	The input control that changes the output.

Control Outputs

Header	Description
out	The current state of the flip-flop.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
reset				
	reset			
set				
CONTROLINPUT	set			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
toggle				
CONTROLINPUT	toggle			
out				
CONTROLOUTPUT	out			

c_hold(hld) - One-shot and hold

Description

This module will produce a 1 each time the input goes from <1 to >=1 and leave it high for a specified number of control loops. It will then drop back to 0 and remain there until the input goes below 1 and back up >=1 again.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input control that we are converting to a one-shot control signal.
iterations	The number of control loops the output will remain high.

Control Outputs

Header	Description
out	The one-shot result.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	0	1		1
	in			
iterations	0	32767		1
CONTROLINPUT	iterations			
out	0	1		1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	out			

c_hold_msec(hld) - One-shot and hold (msec based)

Description

This module will produce a 1 each time the input goes from <1 to >=1 and leave it high for a specified number of milliseconds. It will then drop back to 0 and remain there until the input goes below 1 and back up >=1 again.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
holdtime	The number of milliseconds the output will remain high.
in	The input control which we are converting to a one-shot control signal.

Control Outputs

Header	Description
out	The one-shot result.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	0	1	0	1
	in			
holdtime	1	50000	500	1
	mS			
CONTROLINPUT	holdtime			
out	0	1	0	1

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
CONTROLOUTPUT	out			

c_impulse(cim) - An impulse train

Description

This module will produce a sequence of impulses... its output will usually remain at 0, but every so often it'll go up to 1 for a single cycle and then drop back to 0. This would be useful for triggering a sample-and-hold, for example. The "nyquist frequency" of this codegen would be half the frequency of calls to the foreground loop. Since we can't predict that frequency, we cannot exactly limit the frequency of impulses, but we CAN guarantee that each 1 is followed by a 0, effectively limiting the frequency.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
freq	The number of impulses per second.

Control Outputs

Header	Description
out	The impulse train.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
freq	0	20000	0	0.01
	freq			
out	0	1		

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	out			

c_ioselect(cioselect) - Input/output configuration control

Description

This module implements input/output channel configuration selection.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
input_inputmode	mono(0) or stereo(1)
input_outputmode	mono(0) or stereo(1)

Control Outputs

Header	Description
output_in1_sens	inactive(0) or active(1)
output_in2_sens	inactive(0) or active(1)
output_out1_sens	inactive(0) or active(1)
output_out2_sens	inactive(0) or active(1)

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
input_inputmode	0	1	0	1
	inputmode			
input_outputmode	0	1	0	1
CONTROLINPUT	outputmode			
output_in1_sens				
CONTROLOUTPUT	in1_sens			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
output_in2_sens				
CONTROLOUTPUT	in2_sens			
output_out1_sens				
CONTROLOUTPUT	out1_sens			
output_out2_sens				
CONTROLOUTPUT	out2_sens			

c_keysel(keysel) - Detect Change in Input

Description
This module implements key and scale selection for pitch-based algorithms.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
detect	The input value on which to detect a change.

Control Outputs

Header	Description
keyout	The key (0-11)
scaleout	The scale (0-17)

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
detect	0	1		1
	%n			
pitchin	-32768	32768		1
CONTROLINPUT	%n			
levelin	0	1		0.0001
CONTROLINPUT	%n			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
threshold	-99	0		1
CONTROLINPUT	%n_dB			
scalein	0	17	0	1
			Ionian_(Maj),Dorian,Phrygian,Lydian,-Mixolydian,Aeolian_(Min),Locrian,-Major,Minor,Harmonic_Min,Melodic_Min,Chromatic,Wholetone,Penta_Maj,Penta_Min,Enigmatic,Neapolitan,Hungarian	
CONTROLINPUT	scale			
keyin	0	11	0	1
			C,C#,D,D#,E,F,F#,G,G#,A,A#,B	
CONTROLINPUT	key			
scaleout				
CONTROLOUTPUT	scaleout			
keyout				
CONTROLOUTPUT	keyout			
obj				
USEROBJECTPARENT	obj			

c_many

Description

This module takes one control input and produced a number of outputs, each being a scaled representation of the input. The relationship between the input and each output is: output_n = input_n * mult_n + offset_n This module may be used in place of c_master. Either of these modules is useful when a single knob is used to control a number of differing parameters.

Godlike Productions Comments

Specifiers

Header	Description
inmax	(-32767->+32767) the maximum value for the input
inmin	(-32767->+32767) the minimum value for the input
noutputs	(1->32) the number of control outputs

Control Inputs

Header	Description
in	(-32767->+32767) master control input
mult_1..n	(-32767->+32767) multiplier for output 1..n
offset_1..n	(-32767->+32767) offset for output 1..n

Control Outputs

Header	Description
output_1..n	(-32767->+32767) slave control outputs

User Objects

Header	Description
	A userobject to display the in and out values, suitable for placing on a menupage.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
noutputs	1	32	1	1
	number inputs			
inmax	-32768	32767	0	1
INT	input max			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
inmin	-32768	32767	0	1
INT	input min			
in				
CONTROLINPUT	in			
mult	-32768	32767	1	1
		@noutputs		
CONTROLINPUT	mult~n			
offset	-32768	32767	1	1
		@noutputs		
CONTROLINPUT	offset~n			
output				
		@noutputs		
CONTROLOUTPUT	out~n			

c_merge(mrg) - Merge Control Signals

Description

This module merges together a specified number of control signals. A merge is when the output is the value of the last control that changed. In the case of a tie, the last on the list.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	Specifies how many inputs are to be merged together. At least two inputs must be specified. The maximum allowed is 32.

Control Inputs

Header	Description
in1,in2,...inN	The input control signals that are to be added together.

Control Outputs

Header	Description
out	The last input to have changed

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	2	32	2	1
	number_inputs			
in				
		@ninputs		
CONTROLINPUT	in~n			
out				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	out			

c_oswitch(cos) - Select One of N Control Outputs

Description

This module switches an input between N control outputs, depending on the value of the "select" control signal. A select value of 0,1,2 passes the value of in to out0,1,2, etc. Each output has a corresponding defaultout control input which determines the value to be broadcast by each output when it is unselected. Select values other than integers will cause the input to be split between two of the outputs. More specifically, Each output will be interpolated between 'in' and 'defaultoutN' like so:
outN = (1.0 - fraction) * in + fraction * (defaultoutN)
outN+1 = fraction * in + (1.0 - fraction) * (defaultoutN+1)
For example, select = 1.8 would have N = 1, N+1 = 2, and fraction = 0.8.
When defaultout = 0 for all outputs, the math simplifies to:
outN = (1.0 - fraction) * in
outN+1 = fraction * in
In this case, a select value of 0.5 would send half the input to out0 and half to out1.

Godlike Productions Comments

Specifiers

Header	Description
noutputs	Specifies how many outputs there are. The minimum is 1 and the maximum is 32.

Control Inputs

Header	Description
defaultout1,2,...N	The control signals to be broadcast by each control output when unselected. Also used to interpolate for non-integer values of select.
in	The input control signal.
select	Selects which output(s) will receive the input. May be fractional to split the input between two outputs.

Control Outputs

Header	Description
out1,out2,...outN	The output control signals that are to be switched.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
noutputs	1	100	1	1
	number_outputs			
select	0	@noutputs/1/#s-ub	0	1
CONTROLINPUT	select			
in				
CONTROLINPUT	input			
defaultout				
		@noutputs		
CONTROLINPUT	default_output			
out				
		@noutputs		
CONTROLOUTPUT	out~n			

c_reset - generate a reset pulse

Description

This module generates a reset pulse on loading, lasting for the number of ms in <time>.

Godlike Productions Comments

Specifiers

Header	Description
time	Should be specified in ms

Control Inputs

Header	Description

Control Outputs

Header	Description
nout	Output a 1 when time has elapsed.
out	Output a 1 until the duration specified by time has elapsed. Then output a 0.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out	0	1		1
	out			
nout	0	1		1
CONTROLOUTPUT	nout			
time	1	32000	100	1
INT	time			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_samp (smp)- Sample-and-hold for control signals.

Description

This module will sample the IN input signal, as long as the NEWSAMP input remains high (>=1). When NEWSAMP is low (<1), the output will remain unchanged.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The input control to be sampled.
newsamp	Tells when to take a new sample of IN.

Control Outputs

Header	Description
out	The currently-held sample.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
	in			
newsamp				
			thru,hold	
CONTROLINPUT	sample			
out				
CONTROLOUTPUT	out			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

c_smooth(smu) - Control-signal smoother

Description

This module will make a control signal smoother, by only letting it change slowly. It will interpolate between the old and the new value. The output is calculated by: newout = in*speed + oldout*(1-speed) We need to recalculate this every time the input is different from the old output, regardless as to whether the input or the speed has changed.

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
in	The new value the signal is trying to go to.
speed	How fast the signal is allowed to change.

Control Outputs

Header	Description
out	The (new) output control signal.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
	in			
speed	0	1		
CONTROLINPUT	speed			
out				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	out			

c_switch(swt) - Select One of N Control Inputs

Description

This module switches between one of N control inputs, depending on the value of the "select" control signal. A select value of 0,1,2 passes the value of in1,2,3, etc, to the output. Select values other than integers will cause the output to interpolate between two of the inputs. For example, if the select value is set to 1.5, the output will be a 50 % mix of in2 and in3.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	Specifies how many inputs there are. The minimum is 1 and the max is 32.

Control Inputs

Header	Description
in1,in2,...inN	The input control signals that are to be switched.
select	Selects which input is to be passed on to the output.

Control Outputs

Header	Description
out	The output.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	1	100	1	1
	number_inputs			
select	0	@ninputs/1/#sub	0	1
CONTROLINPUT	select			
in				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
		@ninputs		
CONTROLINPUT	input~n			
out				
CONTROLOUTPUT	out			

c_table(tbl) - Control Signal Table Look-up

Description

This module implements a table look-up for control signals. A variable number of table entries are stored with this module. The output will assume the value of one of the table entries, depending on the state of the select signal. If the select signal is 0,1,2,etc. the output will have the value of table entry 0,1,2 etc. If the select signal has a fractional value, the output will interpolate between two table entries.

Godlike Productions Comments

Specifiers

Header	Description
nentries	Specifies the number of table entries. The minimum is 1 and the max is 32.

Control Inputs

Header	Description
select	Selects which table entry is to be passed on to the output.

Control Outputs

Header	Description
out	The output.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nentries	1	1024	1	1
	entries			
select	0	@nentries/1/#sub		1
CONTROLINPUT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	-32768	32767	0	0.01
		@nentries		
FLOAT	value~n			
out				
CONTROLOUTPUT	out			

c_tablen(tbn) - Control Signal Table Look-up

Description

This module implements an n-output table look-up for control signals. A variable number of table entries are stored with this module. Each output will assume the value of one of the table entries, depending on the state of the select signal. If the select signal is 0,1,2,etc. the ouput will have the value of table entry 0,1,2 etc. If the Select signal has a fractional value, the output will interpolate between two table entries.

Godlike Productions Comments

Limited to 999 table entries as of 24 August 2022. This is a bug that Eventide are aware of.

Specifiers

Header	Description
nentries	Specifies the number of table entries. The minimum is 1 and the max is 1024.
noutputs	Specifies the number of module outputs. Each output has its own select signal. The minimum is 1 and the max is 32.

Control Inputs

Header	Description
select(n)	Selects which table entry is to be passed on to the output.

Control Outputs

Header	Description
out*n)	The output.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nentries	1	1024	1	1
	entries			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
noutputs	1	32	1	1
INT	outputs			
select	0	@nentries/1/#sub		1
		@noutputs		
CONTROLINPUT	%n			
in	-32768	32767	0	0.01
		@nentries		
FLOAT	value~n			
out				
		@noutputs		
CONTROLOUTPUT	out~n			

c_timer(tim) - Real time clock

Description

This module will produce an output in seconds showing how long the RUN input was 1.0. If RESET goes from below 1.0 to 1.0 the output will be set to zero

Godlike Productions Comments

Specifiers

Header	Description

Control Inputs

Header	Description
reset	set count to zero on +ve edge
run	count when equal to or above 1.0

Control Outputs

Header	Description
out	The count in seconds

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
run	0	1	0	1
	run			
reset	0	1	0	1
CONTROLINPUT	reset			
out				
CONTROLOUTPUT	out			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

Chapter 4 - Delay

allpass(apf) - Allpass Filter

Description

This module implements an allpass filter of the type described in Manfred Schroeder's seminal paper on digital reverb simulation. As such, this module is intended to be used as a building block in reverb and room simulations. In effect, this module is less like a filter and more like a repeating delay line. It is called an allpass filter because it has the unique characteristic of having a FLAT frequency response. This enables a user to cascade several allpass filters in series without generating excess coloration of the sound. This technique is typically used in reverberators to generate diffusion, a dense grouping of echoes.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Determines how much delay is used by this module (in ms). The delay can then be adjusted up to this maximum value. The most delay that can be specified is 660 milliseconds.

Audio Inputs

Header	Description
in	Allpass Input

Audio Outputs

Header	Description
out	Allpass Output

Control Inputs

Header	Description
delayamt	This controls the amount of delay in the feedback loop. Adjustment is in milliseconds.
g	Controls the feedback gain. 0 is no feedback and 1 is 100 per cent feedback. Negative numbers invert the phase of the feedback.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
maxdelay	1	660	1	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	1	0.01
	ms			
CONTROLINPUT	delay			
g	-1	1		0.01
CONTROLINPUT	g			

bucket_brigade_delay - Bucket Brigade Delay Emulation

Description

This module emulates the characteristics of a Bucket Brigade chip. It can be used for building echo effects, choruses, flangers, reverbs, etc. In addition to all the good things that come from using BBDs to make delays like squishy modulation and dark repeats this module also comes with all the baggage. It will alias, it will come out distorted, and it will be noisy if you make it so. If you don't want this to sound bad it is essential that you put a lowpass filter before and after the BBD module so that it doesn't alias and so that the clock glitches are filtered out. Typical setups used over and over again in analog delay effects look like -

Low Pass ---> BBD ---> Low Pass

Compressor ---> Low Pass ---> BBD ---> Low Pass ---> Expander

The cutoff frequency of the two9 low pass filters should be between 1/3 and 1/2 of the lowest bucket brigade clock frequency to get a clean sound. It should also have a steep roll off - i.e using the cheby1 module with an order of 3 or higher.

The CPU usage of this module will increase as the BBD clock frequency increases (so as delay times go down). On resource constrained systems you can get around this by using the processOff input to switch beteen BBDs with low and high stage counts.

This module will clip internally at high signal levels close to 0dB. With no additional filtering the modules frequency response is limited to 18kHz at 48k and 16.5kHz at 44.1. The frequency response is full bandwidth at 88.1kHz/96kHz.

Godlike Productions Comments

The transfer efficiency controls how much leakage occurs between stages. In hardware this is equivalent to real charge loss as charge is transferred between transistors on the BBD chip.

Specifiers

Header	Description
upSampling	The amount of upsampling to use inside the module. This will help reduce unwanted harmonics generated as a result of converting from the BBD sampling rate to the system sampling rate. The inteded harmonics (those generated by the BBD clocking) are unaffected. Supported values are 2, 4, and 8. Odd numbers are mapped to their lower even neighbor (i.e 3 -> 2). Any value between 4 and 8 is equal to an upsampling factor of 4.
nStages	The number of stages in the bucket brigade. This will affect some aspects of the performance and the delay times possible. With a constant clock the delay times are given by: N/(2*Fcl). Where N is the number of stages and Fcl is the clock frequency.
nTaps	Number of taps to read out of the bucket brigade. Each tap is read after a specified number of stages (as given in tapStages).
minClockFreq	The minimum clock frequency to expect. This will affect at what clock frequency the BBD module has the highest noise level.

Audio Inputs

Header	Description
in	The audio input

Header	Description
clockMod	Modulation signal for the clock frequency. This should be in the range -1 to 1. The modulated clock frequency is determined by: clockFrequency + clockMod*clockModDepth.

Audio Outputs

Header	Description
out	The delayed output of each tap.

Control Inputs

Header	Description
processOff	Turns processing for this module On or Off. When off it will ouput zeros. 1 Turns it off, 0 is on. This is usefule on resource constrained systems for switching between two bbd modules (with high and low number of stages) to get shorter/longer delay times.
tapStages	The number of stages after which each tap is read out of the BBD.
clockFrequency	The frequency of the clock which controls the timing of the BBD. Delay times are given by: N/(2*Fcl) where N is the number of stages and Fcl is the clock frequency. GIVEN IN KHZ!
clockModDepth	The depth of the clock modulation. Modulated clock frequencies are determined by: clockFrequency + clockMod*clockModDepth. GIVEN IN KHZ!
transferEfficiency	How much energy is lost at each transfer between stages in the bucket brigade device given as a percent between 0 and 100. In a read device this affects the overall level and high frequency roll off. Since an overall level control isn't this just affects the high frequency roll off close to the BBD nyquest frequency (half the clock frequency). Higher values equal less high frequency roll off.
maxNoiseLevel	Controls the amount of noise present in the BBD. This is given in dB and represents the worst case noise level which happens at the given value of minClockFreq. As the clock frequency increases from this value, the level of the noise decreases.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
upSampling	2	8	2	1

bucket_brigade_delay

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	upSampling			
nStages	64	16384	512	1
INT	Number of stages in the bucket brigade.			
nTaps	1	6	1	1
INT	Number of outputs from the bucket brigade.			
minClockFreq	0.25	800	10	0.01
	Minimum allowable clock frequency (kHz).			
in	-1.0	1.0		
MONO				
AUDIOINPUT				
clockMod	-1.0	1.0		
MONO				
AUDIOINPUT				
outn				
MONO				
AUDIOOUTPUT				
processOff	0	1	0	1
CONTROLINPUT	Turn off			
clockFrequency	0.25	800	12	0.01
	kHz			
CONTROLINPUT	clockFrequency			
clockModDepth	-800	800	0	0.01
	kHz			
CONTROLINPUT	clockModDepth			
transferEfficiency	0	100.0	100.0	0.1
	%			
CONTROLINPUT	transferEfficiency			
maxNoiseLevel	-120.0	-30.0	-60.0	0.1
	dB			
CONTROLINPUT				
tapStagesn	1	@nStages	@nStages	1
CONTROLINPUT	tapStagesn			

bucket_brigade_delay

comb(com) - Comb Filter

Description

Like the allpass filter, the comb filter module is a building block to be used in creating reverb simulations. Also like the allpass filter, the comb filter is a delay line with feedback. The difference is, the comb filter does NOT have a flat frequency response. In fact, the freuqency response is periodic, resembling the teeth of a comb. The comb module is useful as a simple repeating delay. To build a reverb, several comb filters are typically connected in parallel. This parallel combination then typically is connected to several allpass filters in series.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Determines how much delay is used by this module (in ms). The delay can then be adjusted up to this maximum value. The most delay that can be specified is 660 milliseconds.

Audio Inputs

Header	Description
in	Comb Input

Audio Outputs

Header	Description
out	Comb Output

Control Inputs

Header	Description
delayamt	This controls the amount of delay in the feedback loop. Adjustment is in milliseconds.
feedback	Controls the feedback gain. 0 is no feedback and 1 is 100 per cent feedback. Negative numbers invert the phase of the feedback.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	660	10	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	10	0.01
	ms			
CONTROLINPUT	delay			
feedback	-1	1	0.5	0.01
CONTROLINPUT	feedback			

delay(dly) - Audio Delay

Description

This module implements a simple audio delay line. Any audio signal applied to the input appears at the output a specified amount of time later. The amount of delay is controllable via the delayamt control signal input. Note: changing the delay value while audio is present may cause clicks in the audio. If it is desired to smoothly change the delay time, use the moddelay module.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	1 to 660 milliseconds. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The audio input to be delayed.

Audio Outputs

Header	Description
out	A delayed version of the input signal.

Control Inputs

Header	Description
delayamt	0 to maxdelay milliseconds. Controls how much the audio will be delayed.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
maxdelay	1	22000/32500/#speed/#if	100	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	100	0.01
	ms			
CONTROLINPUT	%n			
obj				
USEROBJECTPARENT	obj			

delay_xf(dxf) - Multi-Tap Crossfading Delay Line

Description

This is an n tap delay line with with modulation, gain control, and crossfading outputs. Crossfading is activated for all delay and gain changes including modulation. Xfade times of less than about 20 mS will result in a slight pitching sound whereas xfade times greater than that will sound like skipping audio without the clicks and pops.

Godlike Productions Comments

Specifiers

Header	Description
noutputs	the number of individual delay taps.

Audio Inputs

Header	Description
in	the audio input to the delay line.

Audio Outputs

Header	Description
out1..n	the output for each delay tap.

Control Inputs

Header	Description
gain1..n	-144 dB to 0 dB. Controls the relative volume of each delay tap.
modamt1..n	0 to maxdelay milliseconds. Controls how much the delay will be affected by the modulation.
tapdls1..n	0 to maxdelay milliseconds. The delay time for each individual delay tap.
xfadetime	0.0 to 2000.0 milliseconds. Controls how long it takes to update the delay tap to a new delay time.

Control OutputsMod Inputs

Header	Description
mod1..n	the modulation signal for each delay tap.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	22000/32500/#speed/#if	100	1
	max_delay			
noutputs	1	16	1	1
INT	number_outputs			
in				
MONO				
AUDIOINPUT				
xfadetime	0.1	2000	20	0.01
	ms			
CONTROLINPUT	xfadetime			
mod				
MOD		@noutputs		
AUDIOINPUT				
out				
MONO		@noutputs		
AUDIOOUTPUT				
modamt	0	@maxdelay	0	1
	ms	@noutputs		
CONTROLINPUT	modamt			
tapdls	0	@maxdelay	0	0.1
	ms	@noutputs		
CONTROLINPUT	delay~n			
gain	-144	0	0	0.1
	db	@noutputs		
CONTROLINPUT	gain~n			

delaysampler2(dlysmp2) - Audio Sampler in Delay memory

Description

This module implements an audio sampler using delay memory, with the express purpose of looping a recorded section of audio. Salient features are loop recording with dubbing, variable speed record / playback / dubbing, and real-time adjustment of the head and tail of the loop.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	5000 to <max_delay_max> milliseconds. Specifies the maximum length audio to be recorded.
maxsplice	0 to 100 milliseconds. Specifies how long the loop splice (xfade) between the head and tail can be.

Audio Inputs

Header	Description
in	The audio input to be delayed.
looptrigger	Triggers a loop around from the beginning, but will preserve state if in Dub, Useful for automatic MIDI synced loop arounds.
playtrigger	The trigger signal for play, >0.5 will trigger a play (or play re-trigger)
rectrigger	The trigger signal for recording, >0.5 will trigger a record (or dub) Currently, this will only trigger the inital recording (rectrigger from a stopped state) when playspeed > 0, i.e. the forward direction.
stoptrigger	The trigger signal for stopping record or play, >0.5 will trigger a stop

Audio Outputs

Header	Description
out	Recorded input signal whenever Play is hit.
posout	audio rate output of the current position. Will be an integer value, so it's actually a small fraction of the max 1.0 of audio data.
speedout	audio rate ouput of the current speed. b/t -1 and 1

Control Inputs

Header	Description
beginpt	0 to maxdelay milliseconds. The beginning point (starting position) of the loop. Rectrigger/playtrigger will trigger/re-trigger at this postion. The audio will loop back to this position the next time around if the control is moved during the loop playback. Of course, this acts as the end of the loop for reverse playspeeds.
dubfade	0 to 100 milliseconds. The fade in/out time for the dubbed signal. Also, serves as the xfade time b/t the dubbed signal and existing loop signal if there is any decaying of the loop.
endpt	-maxdelay to 0 milliseconds. The amount of time trimmmed off the back of the loop that determines the end point of the loop. For reverse playspeeds this will act as the beginning of the loop.
maxlength	0 to maxdelay milliseconds. The maximum length of the audio that can be recorded.

Control OutputsMod Inputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	2000	32500	5000	1
	max_delay			
maxsplice	0	100	1	0.1
	ms			
INT	max_splicelength			
in				
MONO				
AUDIOINPUT				
rectrigger				
MOD				
AUDIOINPUT				
stoptrigger				
MOD				
AUDIOINPUT				
playtrigger				
MOD				
AUDIOINPUT				
looptrigger				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
posout				
MOD				
AUDIOOUTPUT				
speedout				
MOD				
AUDIOOUTPUT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxlength	0	@maxdelay/@ maxsplice/#sub	@maxdelay/@ maxsplice/#sub	0.1
	ms			
CONTROLINPUT	max_loop_length			
beginpt	0	@maxdelay/@ maxsplice/#sub	0	0.1
	ms			
CONTROLINPUT	start_at			
endpt	@maxdelay/@ maxsplice/#sub /#neg	0	0	0.1
	ms			
CONTROLINPUT	end_at__			
dubfade	0	100	1	0.1
	ms			
CONTROLINPUT	dub_fadetime			
splicelength	0	@maxsplice	0	0.1
	ms			
CONTROLINPUT	splicelength			
playspeed	-100	100	100	0.1
	%%			
CONTROLINPUT	speed			
decay	0	100	0	0.1
	%%			
CONTROLINPUT	decay			
loopmode	0	1	0	1
CONTROLINPUT	Loop_mode			
playmode	0	1	0	1
			stutter,pause	
CONTROLINPUT	playmode			
timeoutmode	0	2	0	1
			autostop,autoplay,autodub	
CONTROLINPUT	Time_Out_Mode			
dubmode	0	1	0	1
			latching,momentary	
CONTROLINPUT	dubmode			
statusout	0	100	0	1
CONTROLOUTPUT	status___			
position	0	@maxdelay	0	0.1
CONTROLOUTPUT	Position			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
reclength	0	@maxdelay	0	0.1
CONTROLOUTPUT	Length_			
obj				
USEROBJECTPARENT	obj			

delaysampler2n(dlysmp2n) - Mulitiple Track Audio Sampler in Delay memory

Description

This module implements an audio sampler using delay memory, with the express purpose of recording and looping multiple independent audio channels using the same transport controls. Salient features are loop recording with dubbing, variable speed record / playback / dubbing, real-time adjustment of the head and tail of the loop, and undo/redo of the last recording/dub action.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	0.1 to <max_delay_max> seconds. Specifies the maximum length audio to be recorded, divided equally amongst all tracks. Total record time will be cut in half b/c of undo/redo. Higher sample rates will also
maxsplice	0 to 100 milliseconds. Specifies how long the loop splice (xfade) between the head and tail can be.

Audio Inputs

Header	Description
looptrigger	Triggers a loop around from the beginning, but will preserve state if in Dub, Useful for automatic MIDI synced loop arounds, >0.5 will trigger a loop around.
playtrigger	The trigger signal for play, >0.5 will trigger a play (or play re-trigger)
rectrigger	The trigger signal for recording, >0.5 will trigger a record (or dub) Currently, this will only trigger the inital recording (rectrigger from a stopped state) when playspeed > 0, i.e. the forward direction.
stoptrigger	The trigger signal for stopping record or play, >0.5 will trigger a stop

Audio Outputs

Header	Description
posout	audio rate output of the current position. Will be an integer value, so it's actually a small fraction of the max 1.0 of audio data.
speedout	audio rate ouput of the current speed. b/t -1 and 1

Control Inputs

Header	Description
beginpt	0 to ((maxdelay*0.5)/numtracks-maxsplice) seconds. The beginning point (starting position) of the loop. Rectrigger/playtrigger will trigger/re-trigger at this postion. The audio will loop back to this position the next time around if the control is moved during the loop playback. Of course, this acts as the end of the loop for reverse playspeeds.
dubfade	0 to 100 milliseconds. The fade in/out time for the dubbed signal. Also, serves as the xfade time b/t the dubbed signal and existing loop signal if there is any decaying of the loop.

Header	Description
endpt	-((maxdelay*0.5)/numtracks-maxsplice) to 0 seconds. The amount of time trimmmed off the back of the loop that determines the end point of the loop. For reverse playspeeds this will act as the beginning of the loop.
maxlength	0 to ((maxdelay*0.5)/numtracks-maxsplice) seconds. The maximum length of the audio that can be recorded.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
numtracks	1	32	2	1
	numtracks			
maxdelay	0.1	720	12	0.001
	sec			
INT	max_delay			
maxsplice	0	100	1	0.1
	msec			
INT	max_splicelength			
in				
MONO		@numtracks		
AUDIOINPUT				
rectrigger				
MOD				
AUDIOINPUT				
stoptrigger				
MOD				
AUDIOINPUT				
playtrigger				
MOD				
AUDIOINPUT				
looptrigger				
MOD				
AUDIOINPUT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
undo_redo				
MOD				
AUDIOINPUT				
begin_mod				
MOD				
AUDIOINPUT				
end_mod				
AUDIOINPUT				
out				
		@numtracks		
AUDIOOUTPUT				
posout				
AUDIOOUTPUT				
speedout				
AUDIOOUTPUT				
maxlength	0	@maxdelay/2/#div/@numtracks/#div/@max-splice/0.001/#mul/#sub	@maxdelay/2/#div/@numtracks/#div/@max-splice/0.001/#mul/#sub	0.0001
	sec			
CONTROLINPUT	max_loop_length			
beginpt	0	@maxdelay/2/#div/@numtracks/#div/@max-splice/0.001/#mul/#sub	0	0.0001
	sec			
CONTROLINPUT	start_at			
endpt	@maxdelay/2/#div/@numtracks/#div/@max-splice/0.001/#mul/#sub/#neg	0	0	0.0001
	sec			
CONTROLINPUT	end_at__			
dubfade	0	100	1	0.1
	msec			
CONTROLINPUT	dub_fadetime			
splicelength	0	@maxsplice	0	0.1
	msec			
CONTROLINPUT	splicelength			
playspeed	-100	100	100	0.1
	%%			
CONTROLINPUT	speed			
decay	0	100	0	0.1
	%%			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	decay			
loopmode	0	1	0	1
CONTROLINPUT	Loop_mode			
playmode	0	1	0	1
CONTROLINPUT	playmode			
timeoutmode	0	2	0	1
CONTROLINPUT	Time_Out_Mode			
dubmode	0	1	0	1
CONTROLINPUT	dubmode			
statusout	0	100	0	1
CONTROLOUTPUT	status___			
position	0	@maxde-lay/2/#div/@ numtracks/#div	0	0.1
CONTROLOUTPUT	Position			
reclength	0	@maxde-lay/2/#div/@ numtracks/#div	0	0.1
CONTROLOUTPUT	Length__			
obj				
USEROBJECTPARENT	obj			

easytaps(etp) - Easy Multitap Delay Line

Description

The easytaps module is a multitap delay line with a simplified user interface. It allows the user to contol the delay, amplitude, and pans by specifiying "shapes" rather than specifiying parameters for each individual tap.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Specifies the amount of delay memory (in ms) allocated to the module.

Audio Inputs

Header	Description
in	The audio input to be processed.

Audio Outputs

Header	Description

Control Inputs

Header	Description

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
taps	1	256	1	1
	number_taps			
maxdelay	0	10/524288/480/#div/#trun-c/#speed/1/#add/#div/#mul	2000	1
	ms			
INT	max_delay			
tapnorm	0	1	0	1
INT	tap_normalization			
in				
MONO				
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				
AUDIOOUTPUT				
parallel_delay				
MONO				
AUDIOOUTPUT				
numbertaps	0	@taps	8	1
CONTROLINPUT	taps			
length	0	@maxdelay	600	1
CONTROLINPUT	length			
randomizing	0	1	0	
CONTROLINPUT	random			
delayalpha	0	1	0.5	
CONTROLINPUT	delay_alpha			
width	-1	1	1	
CONTROLINPUT	width			
ampalpha	0	1	0.5	
CONTROLINPUT	amp_alpha			

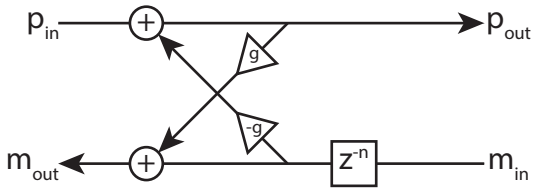
Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
ampshape	0	6	0	
CONTROLINPUT	amp_shape			
delayshape	0	6	0	
CONTROLINPUT	delay_shape			
panshape	0	7	5	
CONTROLINPUT	pan			
declick	0	8	5	1
CONTROLINPUT	declick			
quantize	0	@maxdelay	0	0.01
	ms			
CONTROLINPUT	quantize			
obj				
USEROBJECTPARENT	obj			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

lattice - Lattice Filter

Description

The lattice module implements one stage of a lattice filter with delay line memory. Connecting pout to m_in and taking the output from mout will give an allpass response, the same as the allpass module. Chaining lattice filters together allows nested allpass structures useful for quickly building up echo density in a larger reverb structure. Signal flow graph:



Further detail: Audio input flows into p_in and through the top rail. Lattice instances are connected in series, pout feeding p_in of the next instance, until the end which links the top rail (pout) to its own bottom rail (m_in), wherein the audio flows in the opposite direction. The bottom rail output (mout) at the initial stage, should have an allpass transfer function. The instance has a small delay line receiving input from the bottom rail. It appears instances along the top rail can

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Specifies the maximum delay in this lattice stage in milliseconds.

Audio Inputs

Header	Description

Audio Outputs

Header	Description

Control Inputs

Header	Description
g	The reflection coefficient, or gain, of this stage.

Control OutputsMod Inputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	660	1	1
	max_delay			
p_in				
MONO				
AUDIOINPUT				
m_in				
MONO				
AUDIOINPUT				
pout				
MONO				
AUDIOOUTPUT				
mout				
MONO				
AUDIOOUTPUT				
dly_amt	0	@maxdelay	1	0.01
	msec			
CONTROLINPUT	delay amount			
g	0	1	0.5	0.001
CONTROLINPUT	reflection			

lms - LMS Adaptive Filter

Description

The lms module implements an LMS adaptive filter. The audio input is copied into a delay buffer, the delay buffer is FIR-filtered into an estimate, the estimate is subtracted from the desired signal resulting in the error estimate, and finally the error is weighted and multiplied by the delay contents to adjust the respective FIR taps. The taps eventually converge to an estimate of the correlation between the corresponding delays and the desired signal.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio source to remove noise from.

Audio Outputs

Header	Description
out	The audio source with modeled noise removed.

Control Inputs

Header	Description

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
num_taps	1	660	1	1
	numtaps			
in				
MONO				
AUDIOINPUT				
noise_in				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
noise_out				
MOD				
AUDIOOUTPUT				
adaptation_gain	0	1	0	0.001
CONTROLINPUT	adapt			

longdelay - Audio Delay

Description

This module implements a simple audio delay line. It is very similar to the delay module, the principal difference being the maximum delay allowed. The original DSP-based version of this module interfaced with an external bulk delay device. The C version does not have this luxury, so the maximum delay allowed, LONGDELAY_MAXSIZE, should be set judiciously so as not to chew up too much system RAM.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Specifies the maximum length of the audio delay, from 1 to 120 seconds.

Audio Inputs

Header	Description
in	The audio input to be delayed.

Audio Outputs

Header	Description
out	A delayed version of the input signal.

Control Inputs

Header	Description
delayamt	The amount the audio will be delayed, from 0 to maxdelay seconds.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
maxdelay	1	120	1	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	1	0.01
	sec			
CONTROLINPUT	%n			

microdelay(udl) - Modulatable Micro-Delay

Description

The microdelay module provides a precisely adjustable delay amount. The delay is adjustable in increments of 1/256 of an audio sample. This is done with a high-order interpolation filter. This module also provides high-qualtity delay modulation, maintaining full-bandwidth and minimizing aliasing artifacts. If this high quality is essential, this module should be used instead of the moddelay module.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	0 to 10 seconds at 48kHz. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The delay line input.
mod	The delay modulation input.

Audio Outputs

Header	Description
out	The delay line output.

Control Inputs

Header	Description
delayamt	0 to maxdelay milliseconds. Provides adjustment of delay time. The accuracy is better than 100
modamt	0 to maxdelay milliseconds. Controls how much the mod input will affect the delay amount.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	0	10/524288/480/#div/#trun-c/#speed/1/#add/#div/#mul	20	1
	max_delay			
modsw	0	1	0	1
INT	mod_select			
in				
MONO				
AUDIOINPUT				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	20	0.001
	ms			
CONTROLINPUT	delay			
modamt	0	@maxdelay	0	0.001
	ms			
CONTROLINPUT	moddelay			

moddelay(mdl) - Modulatable Delay

Description

The moddelay module is an audio delay line that can have its delay amount modulated by a mod rate signal. This is useful for creating flanging and chorus effects. This module implements its delay modulation with linear interpolation. This can result in some attenuation of high frequencies as well as potential for aliasing. In most applications, this is not a problem. However, if utmost audio quality is important, use the "microdelay" module.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	0 to 2000 milliseconds. The maximum amount of delay for this module.

Audio Inputs

Header	Description
in	The delay input.
mod	Modulation input for delay modulation

Audio Outputs

Header	Description
out	The delay output.

Control Inputs

Header	Description
delayamt	0 to maxdelay milliseconds. Controls the amount of delay.
modamt	0 to maxdelay milliseconds. Controls how much the delay time will be modulated by the mod input.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	0	10/524288/480/#div/#trun-c/#speed/1/#add/#div/#mul	20	1
	max_delay			
declick	0	2	1	
INT	declicking_on_off			
in				
MONO				
AUDIOINPUT				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	20	0.1
	ms			
CONTROLINPUT	delay			
modamt	@maxde-lay/#neg	@maxdelay	0	0.1
	ms			
CONTROLINPUT	modulation			
glidespeed	0	1000	1000	0.1
	ms/s			
CONTROLINPUT	glide			

multidelay(mdy) - Multi-Mode Multi-Tap Delay Line

Description

The multidelay module is a multiple mode multitap delay with modulation and buffer clearing. Each tap can be set to either regular modulation with smoothed delay changes, crossfading modulation with crossfaded delay changes, or reverse delay with modulated length. There are a total of 16 possible taps with individual outputs for each tap. Each tap can also be independently cleared.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	0 to 32500 ms. This is the maximum delay time for each tap.
noutputs	1 to 16. This specifies how many taps the delay line has.

Audio Inputs

Header	Description
in	The audio input to the delay line.
mod(n)	This audio input will modulate the amount of delay time for a particular tap. This is useful to create chorus, flanging, vibrato and other more dramatic effects. There is

Audio Outputs

Header	Description
out(n)	This is the output of the delay line. There is one output per delay tap.

Control Inputs

Header	Description
modamt(n)	#NAME?
processoff	0 or 1, turns off processing
xfadetime	0.1 to 2000.0 ms. This is the length of the crossfade in both xfade mode and reverse mode.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	22000/32500/#speed/#if	100	1
	max_delay			
noutputs	1	16	1	1
INT	number_outputs			
in				
MONO				
AUDIOINPUT				
mod				
MOD		@noutputs		
AUDIOINPUT				
revsync				
MOD				
AUDIOINPUT				
out				
MONO		@noutputs		
AUDIOOUTPUT				
mode	0			
		@noutputs		
CONTROLINPUT	mode			
tapdls	0	@maxdelay	0	0.1
	ms	@noutputs		
CONTROLINPUT	delay~n			
modamt	@maxde-lay/#neg	@maxdelay	0	1
	ms	@noutputs		
CONTROLINPUT	modamt			
clearmem	0	1	0	1
		@noutputs		
CONTROLINPUT	clear			
xfadetime	0.1	2000	20	0.01
	ms			
CONTROLINPUT	xfadetime			
glidespeed	0	2000	1000	0.1
	ms/s			
CONTROLINPUT	glide			
glidesmooth	0	10	1	0.1
	s			

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
CONTROLINPUT	glide_smooth			
processoff	0	1	0	1
CONTROLINPUT	turn_off			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

multitap(mtp) - Multi-Tap Delay Line

Description

This module implements a multi-tap delay line, with a selectable number of delay taps. Each tap has adjustable level, pan and delay. This can be used to create early reflections for room simulation, strange reverse reverb effect and much more. This module has its own built-in user object that creates a graphical display on the LCD screen. It allows adjustment of level, pan, and delay for each tap.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio input to the multitap delay.

Audio Outputs

Header	Description

Control Inputs

Header	Description

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description
	Creates a graphical interface to the multitap delay. Simply assign this

Module Entries

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
taps	2	100	2	
	number_taps			
in				
MONO				
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				
AUDIOOUTPUT				
params				
		@taps/3/#mul		
FLOAT				
obj				
USEROBJECTPARENT	obj			

picodelay(pdl) - Audio Delay

Description

This module implements a simple audio delay line like in the delay module. This modules though allows for sample accurate setting of the delay value via the delayamt control input.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	0 to 2048 samples. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The audio input to be delayed.

Audio Outputs

Header	Description
out	A delayed version of the input signal.

Control Inputs

Header	Description
delayamt	0 to maxdelay samples. Controls how much the audio will be delayed.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	0	16777216		

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	0	1
	samples			
CONTROLINPUT	delay			

precisiondelay(pdl) - Modulatable Micro-Delay

Description

The microdelay module provides a precisely adjustable delay amount. The delay is adjustable in increments of 1/256 of an audio sample. This is done with a high-order interpolation filter. This module also provides high-quality delay modulation, maintaining full-bandwidth and minimizing aliasing artifacts. If this high quality is essential, this module should be used instead of the moddelay module.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	0 to 10 seconds at 48kHz. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The delay line input.

Audio Outputs

Header	Description
out	The delay line output.

Control Inputs

Header	Description
delayamt	0 to maxdelay milliseconds. Provides adjustment of delay time. The accuracy is better than 100

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
maxdelay	0	10/524288/480/#div/#trunc/#speed/1/#add/#div/#mul	20	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayamt	0	@maxdelay	20	0.001
	ms			
CONTROLINPUT	delay			
xfadelen	1	200	10	0.1
	ms			
CONTROLINPUT	xfade			
actual				
CONTROLOUTPUT	actual			
obj				
USEROBJECTPARENT	obj			

rampdelay - Audio Delay

Description

This module implements a long delay that can gradually vary. Between-sample interpolation is done via convolution with an oversampled windowed sinc function.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Specifies the maximum length of the audio delay, from 1 to 10920 milliseconds.

Audio Inputs

Header	Description
in	The audio input to be delayed.

Audio Outputs

Header	Description
out	A delayed version of the input signal.

Control Inputs

Header	Description

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	10920	1000	1
	max_delay			
in				
MONO				
AUDIOINPUT				
delrate				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delout				
MOD				
AUDIOOUTPUT				
Delay	0	@maxdelay	60	0.05
	ms			
CONTROLINPUT	%n			
SetDelay	0	@maxdelay	0	0.1
	ms			
CONTROLINPUT	%n			
curdelay				
CONTROLOUTPUT	curdelay			

revdly - Reverse Audio Delay

Description

This module implements an audio delay line with two reading tails moving in the opposite direction than that of the writing head. The tails move uniformly in the direction of increasing delay. When the most remote tail reaches the maximum delay length, the least remote tail assumes the role of most remote and the end tail is moved to the beginning of the delay line. Prior to this transition, reads from the two tails should cross-fade so that the most remote tail will completely fade out at the time to swap it to the front.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Specifies the maximum length of the audio delay, from 1 to 10920 milliseconds.

Audio Inputs

Header	Description
in	The audio input to be reversed.

Audio Outputs

Header	Description
out	A reversed version of the input signal.

Control Inputs

Header	Description
reset	One-shot switch to return the delay tail readers to an initial state.
xfadetime	The duration of cross fade that occurs between the reversed sections of delay readers, from 1 to 100 milliseconds.

Control OutputsMod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
maxdelay	1	10920	100	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
reverseamt	1	@maxdelay	100	0.01
	ms			
CONTROLINPUT	%n			
xfadetime	1	100	10	0.1
	ms			
CONTROLINPUT	xfade time			
reset	0	1	0	1
CONTROLINPUT	reset sw			

Chapter 5 - Detector

envdetect - Envelope Detection

Description
This module detects the envelope of the audio signal and outputs the envelope value as an audio signal The output stays within 0-1 range

Godlike Productions Comments
I cannot get this block to work. Either nothing comes out or it crashes the H9000. Use with caution.

Specifiers

Header	Description

Audio Inputs

Header	Description
in	Audio Input

Audio Outputs

Header	Description
out	Envelope Output

Control Inputs

Header	Description
attackcntl	Attack time constant in milliseconds

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
MONO				
AUDIOOUTPUT				
thresmincntl	-60	60	-60	0.1
CONTROLINPUT	threshold min			
thresmaxcntl	-60	60	60	0.1
CONTROLINPUT	threshold max			
attackcntl	10	1000	10	0.1
	ms			
CONTROLINPUT	attack			
releasecntl	10	1000	100	0.1
	ms			
CONTROLINPUT	release			
obj				
USEROBJECTPARENT	obj			

multimeter - Multimeter

Description
Multiple audio meters: peak, hold, and rms in a single convenient module with meter decay control for synchronous and smoothe decay properties.

Godlike Productions Comments
The audio outputs don't appear to work as expected for modulating other audio sources, such as using these to trigger envelope followers in a vocoder. More investigation required.

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to the meter.

Audio Outputs

Header	Description
peak_out	The peak meter.
hold_out	The peak hold meter.
rms_out	The rms meter

Control Inputs

Header	Description
attack	0 to 100 seconds. Peak attack. This controls the speed at which the output of the peak meter responds to increasing signal level. The time specified is the time it takes for the output to reach 67% of the value at the input. This is usually set to a very small value.
decay	0 to 100 seconds. Peak decay. This adjusts the speed at which the peak detector responds to decreasses in signal level. This is usually set to a larger value, depending on the application.
hold	0 to 100 seconds. Peak Hold. The length of time the maximum peak is held on in the hold output.
rms_time	5 to 5000 milliseconds. The length of the MS averaging period.
meter_decay	0 to 1000 dB/sec. Overall linear slew decay of all 3 meters.

Control Outputs

Header	Description
peak_dBout	Control rate dB peak signal output
hold_dBout	Control rate dB peak hold output
rms_dBout	Control rate dB rms output

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in	-1.0	1.0		
MONO				
AUDIOINPUT				
peak_out				
MONO				
AUDIOOUTPUT				
hold_out				
MONO				
AUDIOOUTPUT				
rms_out				
MONO				
AUDIOOUTPUT				
attack	0	100	0.01	0.001
	sec			
CONTROLINPUT	peak attack			
decay	0	100	0.1	0.001
	sec			
CONTROLINPUT	peak decay			
hold	0	100	1	0.001
	sec			
CONTROLINPUT	peak hold			
rms_time	5.0	5000.0	100.0	0.001
	msec			
CONTROLINPUT	rms averaging time			
meter_decay	0	1000	0.1	0.001
	dB/sec			
CONTROLINPUT	meter decay			
peak_dBOut				
CONTROLOUTPUT	peak_dBOut			
hold_dBOut				
CONTROLOUTPUT	hold_dBOut			
rms_dBOut				
CONTROLOUTPUT	rms_dBOut			

peak(pkd) - Peak Detector

Description

The peak detect module is an adjustable rectifier of audio data. It is typically used to get an indication of the level of an audio signal. This can then be used as a modulation source to create effects that vary with input level.

Godlike Productions Comments

The linear state machine doesn't appear to work particularly well, especially with short attack times. Recommend sticking with the differential equation type, especially for envelope follower type aplications.

Specifiers

Header	Description
Type	Selects between old style linear state machine (0) and newer differential equation type peak detector (1)

Audio Inputs

Header	Description
in	The input to the peak detector.

Audio Outputs

Header	Description
out	The output of the peak detector.

Control Inputs

Header	Description
attackcntl	0 to 100 seconds. This controls the speed at which the output of the peak
decaycntl	0 to 100 seconds. This adjust the speed at which the peak detector responds to

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
type	0	1	1	1

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	type			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
attackcntl	0	100	0.1	0.001
	sec			
CONTROLINPUT	attack			
decaycntl	0	100	0.1	0.001
CONTROLINPUT				
obj				
USEROBJECTPARENT	obj			

peakdetect(pkd) - Peak Detector

Description

The peak detect module is an adjustable rectifier of audio data. It is typically used to get an indication of the level of an audio signal. This can then be used as a modulation source to create effects that vary with input level.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to the peak detector.

Audio Outputs

Header	Description
out	The output of the peak detector.

Control Inputs

Header	Description
attackcntl	0 to 100 seconds. This controls the speed at which the output of the peak
decaycntl	0 to 100 seconds. This adjust the speed at which the peak detector responds to

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
MOD				
AUDIOOUTPUT				
attackcntl	0	100	0.1	0.01
	sec			
CONTROLINPUT	attack			
decaycntl	0	100	0.1	0.01
	sec			
CONTROLINPUT	decay			
obj				
USEROBJECTPARENT	obj			

pitchdetect(pdt) - Pitch Detector

Description

This module measure the pitch and other qualities of an audio signal. The resultant control and mod signals can be used to create effects that vary with changes in the musical pitch of the input. Also, the pitch detector can be used to control oscillators so as to create a pitch triggered musical synthesizer.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio input to be pitch detected.

Audio Outputs

Header	Description

Control Inputs

Header	Description
gatelevel	-100 to 0 dB. The gatelevel control sets the level at which the pitch detector will output new pitch values. If the input signal level falls below the level set here, the pitch detector outputs will latch on to the old values.
minpitch	0 to 47. The minpitch control is used to optimize the pitch detection algorithm. It sets the minimum pitch that pitch detector is likely to detect. The values are as follows: 0 - C0 1 - C0# 2 - D0 ... ... 46 - A3# 47 - B3

Control Outputs

Header	Description
amp	amp' is the r.m.s. amplitude relative to full scale ('amp' equal to 1 would be a square wave 'hitting the rail').
freq	The output of the pitch detector given as a frequency in Hertz.
period	The output of the pitch detector given as a period. The value is in milliseconds.
pitch	The output of the pitch detector given in cents relative to middle C.
timbre	is a measurement of the brightness of the tone independent of its pitch. A sine wave has a 'timbre' equal to 1, other wave shapes result in a higher 'timbre'.

Header	Description
tonality	A value representing how periodic the input signal is. A value of 1.0 is given for signals which are purely periodic. Lower values represent signals that are less periodic. The smallest value would be given for very noise-like signals.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
freqout				
MOD				
AUDIOOUTPUT				
periodout				
MOD				
AUDIOOUTPUT				
minpitch	0	47	23	1
CONTROLINPUT	minpitch			
maxpitch	0	47	12	1
			c4,c#4,d4,d#4,e4,f4,f#4,g4,g#4,a4,a#4,b4,c5,c#5,d5,d#5,e5,f5,f#5,g5,g#5,a5,a#5,b5,c6,c#6,d6,d#6,e6,f6,f#6,g6,g#6,a6,a#6,b6,c7,c#7,d7,d#7,e7,f7,f#7,g7,g#7,a7,a#7,b7	
CONTROLINPUT	maxpitch			
gatelevel	-100	0	-60	1
	db			
CONTROLINPUT	gate			
pitch				
CONTROLOUTPUT	pitch			
period				
CONTROLOUTPUT	period			
freq				
CONTROLOUTPUT	freq			
amp				
CONTROLOUTPUT	amp			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
tonality				
CONTROLOUTPUT	tonality			
timbre				
CONTROLOUTPUT	timbre			
obj				
USEROBJECTPARENT	obj			

Chapter 6 - Dynamic

compr_expndr - Compressor / Expander

Description
This a dynamic range combination compressor/expander with separate inputs for the signal whose gain is to be processed and for the the detection (sidechain) input.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be compressed.
sidechain	The audio input whose level is measured and is used to alter the dynamics of the "in" audio input.

Audio Outputs

Header	Description
out	This is the output of the compressor.

Control Inputs

Header	Description
attack	0 to 10.000 seconds. This control determines how fast the compressor/expander will respond to increasing level at the sidechain input.
compr_knee	0 to 24 dB. This control adjusts the width in dB of the compressor soft knee. The compressor soft knee is a region, above the compressor threshold, over which the ratio transitions from 1:1 to the selected ratio.
compr_ratio	1 to 100. This controls the amount of gain reduction that occurs once the sidechain input has gone above the compressor threshold. For ratio > 1, we get a compressor and the value entered selects how many dB of gain reduction occur for every dB the sidechain input is above the threshold.
compr_threshold	-100 to 0 dB. This control adjusts the threshold above which gain reduction begins taking place. For sidechain signals above this threshold with compressor ratio > 1 (a compressor), the gain of the input is reduced.
declickXfade	0 to 200 msecs. declicking xfade time to make changes to gain, threshold, knee, and ratio smoother
expndr_knee	0 to 24 dB. This control adjusts the width in dB of the expander soft knee. The expander soft knee is a region, below the expndr_threshold, over which the ratio transitions from 1:1 to the selected ratio.
expndr_ratio	1 to 100. This controls the amount of gain reduction that occurs once the sidechain input has gone below expndr_threshold. For ratio > 1, we get an expander and the value entered selects how many dB of gain reduction occur for every dB the sidechain input is below.

Header	Description
expndr_threshold	-100 to 0 dB. This control adjusts the threshold below which gain reduction begins taking place. It is internally upper bounded by the current setting of compr_threshold. For sidechain signals below expndr_threshold, with expndr_ratio > 1 (an expander), the gain of the input is reduced.
gain	-24 to 48 dB. This control allows gain to be added to the output signal to make up for the gain lost by gain reduction, or for convenience, allows for some gain reduction.
processoff	0 or 1, turns off processing
release	0 to 10.000 seconds. This control determines how fast the compressor/expander will respond to decreasing level at the sidechain input.

Control Outputs

Header	Description

Mod Outputs

Header	Description
lingain	the gain applied to the signal. Includes additional gain set by control.
loggain	the gain in logarithmic terms. Does not include the additional gain.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
sidechain				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
inlevel				
MOD				
AUDIOOUTPUT				
compr_threshold	-96	0	-40	1
	db			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	compressor threshold			
compr_knee	0	24	0	6
	db			
CONTROLINPUT	compressor knee			
compr_ratio	1	100	10	1
	db			
CONTROLINPUT	compressor ratio			
expndr_threshold	-96	0	-40	1
	db			
CONTROLINPUT	expander threshold			
expndr_knee	0	24	0	6
	db			
CONTROLINPUT	expander knee			
expndr_ratio	1	100	10	1
	db			
CONTROLINPUT	expander ratio			
gain	-24	48	0	1
	db			
CONTROLINPUT	makeup gain			
attack	0	10	0.01	0.001
	sec			
CONTROLINPUT	attack			
release	0	10	0.01	0.001
	sec			
CONTROLINPUT	release			
declickXfade	0	200	0	1
	msec			
CONTROLINPUT	declicking xfade			
processoff	0	1	0	1
CONTROLINPUT	turn_off			
obj				
USEROBJECTPARENT	obj			

compressor(cpr) - Compressor

Description

This a dynamic range compressor with separate inputs for the signal whose gain is to be processed and for the the detection (sidechain) input.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be compressed.
sidechain	The audio input whose level is measured and is used to alter the dynamics of the "in" audio input.

Audio Outputs

Header	Description
out	This is the output of the compressor.

Control Inputs

Header	Description
attackcntl	0 to 10.000 seconds. This control determines how fast the compressor will respond to increasing level at the sidechain input.
decaycntl	0 to 10.000 seconds. This control determines how fast the compressor will respond to decreasing level at the sidechain input.
declickXfade	declicking xfade time to make changes to gain, threshold, knee, and ratio smoother
gaincntl	0 to 24 dB. This control allows gain to be added to the output signal to make up for the gain lost by gain reduction.
kneecntl	0 to 24 dB. This control adjusts the width in dB of the soft knee. The soft knee is a region, above the threshold, over which the ratio transitions from 1:1 to the selected ratio.
processoff	0 or 1, turns off processing
ratiocntl	1 to 100. This controls the amount of gain reduction that occurs once the sidechain input has gone above the threshold. The value entered selects how many dB of gain reduction occur for every dB the sidechain input is above the threshold.
threshcntl	-100 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected.

Control Outputs

Header	Description

Mod Outputs

Header	Description
lingain	the gain applied to the signal. Includes additional gain set by control, but scaled down by the maximum output gain, -48 dB.
loggain	the gain in logarithmic terms. Does not include the additional gain.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
sidechain				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
inlevel				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-40	1
	db			
CONTROLINPUT	threshold			
kneecntl	0	24	0	6
	db			
CONTROLINPUT	knee			
ratiocntl	1	100	10	1
	db			
CONTROLINPUT	ratio			
gaincntl	-24	48	0	1
	db			
CONTROLINPUT	gain			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
attackcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	attack			
decaycntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	decay			
declickXfade	0	200	0	1
	msec			
CONTROLINPUT	declicking xfade			
processoff	0	1	0	1
CONTROLINPUT	turn_off			
obj				
USEROBJECTPARENT	obj			

dbxcompander_c(dbc) - DBX Style Compressor Section

Description

This module models the compression section of an RMS DBX compander circuit found in tape machines. It must be used with the associated dbxcompander_e expander model. This set of modules must be setup correctly with matching thresholds and ratios and the controls should not be changed while audio is running through it. We may want to consider a different scheme for this.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be compressed.

Audio Outputs

Header	Description
inlevel	This is the measured input level.
lingain	This is the gain in Linear domain.
loggain	This is the gain in Log domain.
out	This is the output of the compressor.

Control Inputs

Header	Description
ratiocntl	Because DBX is feedback this should be infinity to one to get a real compression value of 2:1. However, I left this in to tune it a bit. time constant - 0 to 10.000 seconds. This control determines how fast the compressor will respond to increasing level at the sidechain input.
threshcntl	-100 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected.

Control Outputs

Header	Description

Mod Outputs

Header	Description

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
inlevel				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-40	1
	db			
CONTROLINPUT	threshold			
ratiocntl	1	100	100	1
	db			
CONTROLINPUT	ratio			
timeconstantcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	timeconstant			

dbxcompander_c2(dc2) - DBX Style Compressor Section

Description

This module models the compression section of the DBX compander module used in the H910 and H949. It must be used with the associated dbxcompander_e2 expander model. This set of modules must be setup correctly with matching thresholds and ratios and the controls should not be changed while audio is running through it. We may want to consider a different scheme for this.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be compressed.

Audio Outputs

Header	Description
inlevel	This is the measured input level.
lingain	This is the gain in Linear domain.
loggain	This is the gain in Log domain.
out	This is the output of the compressor.

Control Inputs

Header	Description
ratiocntl	Because DBX is feedback this should be infinity to one to get a real compression value of 2:1. However, I left this in to tune it a bit. time constant - 0 to 10.000 seconds. This control determines how fast the compressor will respond to increasing level at the sidechain input.
threshcntl	-100 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected.

Control Outputs

Header	Description

Mod Outputs

Header	Description

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
inlevel				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-40	1
	db			
CONTROLINPUT	threshold			
ratiocntl	1	100	100	1
	db			
CONTROLINPUT	ratio			
timeconstantcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	timeconstant			

dbxcompander_e(dbe) - DBX Style Expander Section

Description

This module models the expansion section of an RMS DBX compander circuit found in tape machines. It must be used with the associated dbxcompander_c compressor module. This set of modules must be setup correctly with matching thresholds and ratios and the controls should not be changed while audio is running through it. We may want to consider a different scheme for this.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be expanded.

Audio Outputs

Header	Description
inlevel	This is the measured input level.
lingain	This is the gain in Linear domain.
loggain	This is the gain in Log domain.
out	This is the output of the expander.

Control Inputs

Header	Description
threshcntl	-100 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected. time constant - 0 to 10.000 seconds. This control determines how fast the compressor will respond to increasing level at the sidechain input.

Control Outputs

Header	Description

Mod Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
inlevel				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-40	1
	db			
CONTROLINPUT	threshold			
timeconstantcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	timeconstant			

dbxcompander_c2(dc2) - DBX Style Compressor Section

Description

This module models the compression section of the DBX compander module used in the H910 and H949. It must be used with the associated dbxcompander_e2 expander model. This set of modules must be setup correctly with matching thresholds and ratios and the controls should not be changed while audio is running through it. We may want to consider a different scheme for this.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be compressed.

Audio Outputs

Header	Description
inlevel	This is the measured input level.
lingain	This is the gain in Linear domain.
loggain	This is the gain in Log domain.
out	This is the output of the compressor.

Control Inputs

Header	Description
ratiocntl	Because DBX is feedback this should be infinity to one to get a real compression value of 2:1. However, I left this in to tune it a bit. time constant - 0 to 10.000 seconds. This control determines how fast the compressor will respond to increasing level at the sidechain input.
threshcntl	-100 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected.

Control Outputs

Header	Description

Mod Outputs

Header	Description

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
inlevel				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-40	1
	db			
CONTROLINPUT	threshold			
timeconstantcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	timeconstant			

ducker(dck) - Ducker

Description

The ducker module is the basic building block for most dynamics control patches. It is essentially a dynamic range compressor with separate inputs for the signal whose gain is to be processed and for the detection (sidechain) input. By connecting the input and the sidechain to the same source, a basic compressor is built. By connecting a dry signal to the sidechain and a processed signal to the input, the processed signal can be ducked (have its gain reduced) during louder passages of audio.

Godlike Productions Comments

There is a possible bug with this block, causing digital crackling. Suggest using the compressor block instead. See https://www.eventideaudio.com/forums/topic/h9000-radio-compress/#post-165785 for more information.

Audio Inputs

Header	Description
in	The audio input to be ducked.
sidechain	The audio input whose level is measured and is used to alter the dynamics of the "in" audio input.

Audio Outputs

Header	Description
out	This is the output of the ducker.

Control Inputs

Header	Description
attackcntl	0 to 10.000 seconds. This control determines how fast the ducker will respond to increasing level at the sidechain input.
decaycntl	0.1 to 10.000 seconds. This control determines how fast the ducker will respond to decreasing level at the sidechain input.
gaincntl	0 to 24 dB. This control allows gain to be added to the output signal to make up for the gain lost by gain reduction.
ratiocntl	1 to 100. This controls the amount of gain reduction that osccurs once the sidechain input has gone above the threshold. The value entered selects how many dB of gain reduction occur for every dB the sidechain input is above the threshold.
threshcntl	-100 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected.

Control Outputs

Header	Description

Mod Outputs

Header	Description
lingain	the gain applied to the signal. Includes additional gain set by control.
loggain	the gain in logarithmic terms. Does not include the additional gain.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
sidechain				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-40	1
	db			
CONTROLINPUT	threshold			
ratiocntl	1	100	10	1
	db			
CONTROLINPUT	ratio			
gaincntl	-24	24	0	1
	db			
CONTROLINPUT	gain			
attackcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	attack			
decaycntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	decay			

expander(epr) - Expander

Description

This a dynamic range expander with separate inputs for the signal whose gain is to be processed and for the the detection (sidechain) input.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be expanded.
sidechain	The audio input whose level is measured and is used to alter the dynamics of the "in" audio input.

Audio Outputs

Header	Description
out	This is the output of the expander.

Control Inputs

Header	Description
attackcntl	0 to 10.000 seconds. This control determines how fast the expander will respond to increasing level at the sidechain input.
decaycntl	0 to 10.000 seconds. This control determines how fast the expander will respond to decreasing level at the sidechain input.
kneecntl	0 to 24 dB. This control adjusts the width in dB of the soft knee. The soft knee is a region, below the threshold, over which the ratio transitions from 1:1 to the selected ratio.
ratiocntl	1 to 100. This controls the amount of gain reduction that occurs once the sidechain input has gone below the threshold. The value entered selects how many dB of gain reduction occur for every dB the sidechain input is below the threshold.
threshcntl	-100 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected.

Control Outputs

Header	Description

Mod Outputs

Header	Description
lingain	The gain applied to the signal.
loggain	The gain applied in logarithmic terms.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
sidechain				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-24	1
	db			
CONTROLINPUT	threshold			
kneecntl	0	24	0	6
	db			
CONTROLINPUT	knee			
ratiocntl	1	100	3	1
CONTROLINPUT	ratio			
rangecntl	-96	0	-40	1
	db			
CONTROLINPUT	range			
attackcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	attack			
decaycntl	0	10	0.02	0.001
	sec			
CONTROLINPUT	decay			

expander2

Description

This a dynamic range expander with separate inputs for the signal whose gain is to be processed and for the the detection (sidechain) input. It has more analog-like peak detection of the sidechain input compared to the expander module.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be expanded.
sidechain	The audio input whose level is measured and is used to alter the dynamics of the "in" audio input.

Audio Outputs

Header	Description
out	This is the output of the expander.

Control Inputs

Header	Description
attackcntl	0 to 10.000 seconds. This control determines how fast the expander will respond to increasing level at the sidechain input.
decaycntl	0 to 10.000 seconds. This control determines how fast the expander will respond to decreasing level at the sidechain input.
gaincntl	-24 to 24 dB. Output gain.
kneecntl	0 to 24 dB. This control adjusts the width in dB of the soft knee. The soft knee is a region, below the threshold, over which the ratio transitions from 1:1 to the selected ratio.
ratiocntl	1 to 100. This controls the amount of gain reduction that occurs once the sidechain input has gone below the threshold. The value entered selects how many dB of gain reduction occur for every dB the sidechain input is below the threshold.
threshcntl	-96 to 0 dB. This control adjusts the threshold at which gain reduction begins taking place. For sidechain signals below this threshold, the gain of the input is not affected.

Control Outputs

Header	Description

Mod Outputs

Header	Description
lingain	The gain applied to the signal.
loggain	The gain applied in logarithmic terms.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
sidechain				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
loggain				
MOD				
AUDIOOUTPUT				
lingain				
MOD				
AUDIOOUTPUT				
inlevel				
MOD				
AUDIOOUTPUT				
threshcntl	-96	0	-40	1
	db			
CONTROLINPUT	threshold			
kneecntl	0	24	0	6
	db			
CONTROLINPUT	knee			
ratiocntl	1	100	10	1
	db			
CONTROLINPUT	ratio			
gaincntl	-24	24	0	1
	db			
CONTROLINPUT	gain			
attackcntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	attack			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
decaycntl	0	10	0.01	0.001
	sec			
CONTROLINPUT	decay			
obj				
USEROBJECTPARENT	obj			

gate - Noise Gate

Description

The gate module applies dynamic gain to audio. The gain is determined by whether an input level estimate is above or below a threshold. This effect serves as noise reduction when the threshold is set at the level of background noise and the gain to apply below the threshold is zero. This implementation adds a hysteresis zone to prevent the system from rapidly oscillating between attack and decay states. This splits the threshold into two, for attack and decay separately. Level estimates between the two retain the system state previously calculated. The level estimator and the gain level smoother are two independent one-pole averaging filters, both employing attack and decay coefficients.

Godlike Productions Comments

Audio Inputs

Header	Description
in	Channel input

Audio Outputs

Header	Description
out	Channel output

Control Inputs

Header	Description
attack	Controls the 1/e duration of transition from attenuation to pass-through. The value range is 0.0 to 10.0 sec.
decay	Controls the 1/e duration of transition from pass-through to attenuation. The value range is 0.0 to 10.0 sec.
decaycntl	pass-through to attenuation. The value range
decaylevel	is 0.0 to 10.0 sec.
hysteresis	Controls the level relative to the threshold below which the decay process is activated. The value range is 0 to 20 dB.
speed	Controls the 1/e averaging time of the channel level filter. The value range is 0.001 to 10.0 sec.
thresh	Controls the threshold at which the gating takes place. Levels above the threshold pass through, and levels below are attenuated. The value range is -100 to 0 dB.

Control Outputs

Header	Description

Mod Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
gain				
MOD				
AUDIOOUTPUT				
thresh	-100	0	-40	1
	db			
CONTROLINPUT	threshold			
decay	0	10	0.1	0.001
	sec			
CONTROLINPUT	decay time			
attack	0	10	0.1	0.001
	sec			
CONTROLINPUT	attack time			
hysteresis	0	20	3	1
	db			
CONTROLINPUT	hysteresis			
speed	0.001	10	0.1	0.001
	sec			
CONTROLINPUT	speed time			
obj				
USEROBJECTPARENT	obj			

gate2(gat) - Audio Noise Gate

Description

This module implements a noise gate function. If the input is below a spcified threshold it will silence (or gate) the output. This noise gate has adjustable attack and release times that control how fast the gate will turn on or off. With proper settings, this makes the gating function much less audible.

Godlike Productions Comments

Audio Inputs

Header	Description
in	The audio input to be gated.
sidechain	The audio input whose level is measured and is used to alter the dynamics of the "in" audio input.

Audio Outputs

Header	Description
out	The noise gated output.

Control Inputs

Header	Description
attack	0.0 to 10.000 seconds. Controls how fast the gate transitions from the "off" state to the "on" state.
decay	0.0 to 10.000 seconds. Controls how fast the gate transitions from the "on" state to the "off" state.
hysteresis	0 to 20 dB. Controls how much the input must drop below the trigger level before the gate can be turned on. This is used to prevent spurious triggering of the gate function.
speed	0.001 to 10.000 seconds. Controls the trigger sensitivity. "Speed" sets the decay rate of a peak detector used in the gate. Setting this to large values will be similiar to a gate "hold" function.
thresh	-100 to 0 dB. Controls the threshold at which the gating takes place. When the input is above the threshold, the gate is turned on, allowing audio to pass. When the input is below the threshold, the gate is turned off.

Control Outputs

Header	Description

Mod Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
sidechain				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
gain				
MOD				
AUDIOOUTPUT				
attack	0	10	0.1	0.001
	s			
CONTROLINPUT	attack			
decay	0	10	0.1	0.001
	s			
CONTROLINPUT	decay			
thresh	-100	0	-40	1
	db			
CONTROLINPUT	threshold			
hysteresis	0	20	3	1
	db			
CONTROLINPUT	hysteresis			
speed	0.001	10	0.1	0.001
	s			
CONTROLINPUT	speed			
obj				
USEROBJECTPARENT	obj			

Chapter 7 - External

midiclock(mck) - MIDI realtime services

Description
This module allows access to the following MIDI realtime functions: MIDI clock, MIDI start and MIDI stop. These allow a process to be controlled by, and synchronized to, an external MIDI sequencer or other controller.

Godlike Productions Comments
BPMOUT works only while Emote is active as a plugin and Tempo Mode is on. Beatout drifts and STARTOUT and STOPOUT don't respond to the DAW's transport. In Emote Standalone with MIDI DIN or USB BPM doesn't follow DAW's tempo, but BEATOUT is better and STARTOUT and STOPOUT respond to DAW's transport. BEATOUT loses sync after some time and doesn't always recover. In depth discussion at https://www.eventideaudio.com/forums/topic/h9000vsig3-bugs/#post-163459 and below

Specifiers

Header	Description

Audio Outputs

Header	Description
clockout	Clock output
startout	Start output
stopout	Stop output

Control Inputs

Header	Description
time_in	A reference time value, to be modified acording to MIDI clock
freq_in	A reference frequency value, to be modified acording to MIDI clock
start_in	A trigger to be merged with MIDI start
stop_in	A trigger to be merged with MIDI stop
remote_mode	Allows the user to enable/disable either MIDI or local controls. 0: both, 1: local only, 2: MIDI only
clock_in	Allows the user to vary MIDIclock out (where supported). 120 BPM is 1.0

Control Outputs

Header	Description
time_out	The reference time value modified acording to MIDI clock. 120 BPM is 1:1
freq_out	The reference frequency value modified acording to MIDI clock. 120 BPM is 1:1
start+out	A combination of start_in and MIDI start
stop_out	A combination of stop_in and MIDI stop
bpm_out	The BPM value received from MIDI clock

User Objects

Header	Description
remote_obj	allow the user to vary remote_mode

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clockout				
startout				
AUDIOOUTPUT				
stopout				
AUDIOOUTPUT				
syncout				
AUDIOOUTPUT				
time_out				
CONTROLOUTPUT	time_out			
freq_out				
CONTROLOUTPUT	freq_out			
start_out				
CONTROLOUTPUT	start_out			
stop_out				
CONTROLOUTPUT	stop_out			
bpm_out				
CONTROLOUTPUT	bpm_out			
beat_out				
CONTROLOUTPUT	beat_out			
tap_count				
CONTROLOUTPUT	tap_count			
mclock_ok				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	mclock_ok			
time_in	-32767	32767	1	0.1
CONTROLINPUT	time_in			
freq_in	-32767	32767	1	0.1
CONTROLINPUT	freq_in			
start_in	0	1		1
CONTROLINPUT	start_in			
stop_in	0	1		1
CONTROLINPUT	stop_in			
remote_mode	0	4	0	1
CONTROLINPUT	remote_mode			
clock_in	0	10	1	0.1
CONTROLINPUT	clock_in			

midicout(out) - MIDI control output

Description
Takes a control input and puts it out the MIDI Port.

Godlike Productions Comments

Specifiers

Header	Description

Audio Outputs

Header	Description

Control Inputs

Header	Description
input	0 to 1 The signal to output over MIDI
mode	0 - single message 1 - double message (only for number 0 thru 31)
channel	0 - global setting, 1-16 - channel number
number	0 thru 121

Control Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
channel				
	channel			
mode				

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
			single,double	
CONTROLINPUT	mode			
number	0	121	16	1
CONTROLINPUT	number			
input	0	1		0.001
CONTROLINPUT	input			

midinote(mnt) - midi note interface

Description

Turns MIDI notes messages into control signals. This gives you the ability make a patch that you can play from a midi keyboard.

Godlike Productions Comments

Specifiers

Header	Description
nvoices	how many voice outputs do we have.

Audio Outputs

Header	Description
gate_mod_out	mod rate gate output (0-1)
pitch_mod_out	mod rate pitch output (0-1)
vel_mod_out	mod rate velocity output (0-1)
pres_mod_out	mod rate pressure output (0-1)

Control Inputs

Header	Description
channel	0: global setting, 1-16: channel number, 17: omni
notemode	0: global setting, 1: mono (many channels, 1 note per channel), 2: poly (1 channel, many notes)
polymode	When in poly mode, how to assign voices. 0: normal - voices are assigned in round robin fashion, 1: ordered - First note played is always Voice1. Then Voice2 and so on., 2: spread1 - All notes play if any keys are pressed. Voice1 is the highest, Voice2 is the next highest and so on. The note assignment happens every time you press a key. If you just press one note, all voices play the same note., 3: spread2 - Just like spread1 except that note assignement also happens when you lift up a key. When you lift up from a chord, you're left with all voices playing one note.
pressure	0: global setting, 1: channel pressure, 2: key pressure
pitchbend	0: global setting, 1-25: 0 thru 24 notes

Control Outputs

Header	Description
gate[nvoices]	1 if voice is on (key down). 0 is off.
pitch[nvoices]	the pitch in cents relative to lowest MIDI note. 0 to 12800 cents.
vel[nvoices]	the velocity of the note (both on and off). 0 to 1.
pres[nvoices]	the pressure on the note. 0 to 1.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nvoices	1	16	1	1
	number_voices			
gate_mod_out				
MONO		@nvoices		
AUDIOOUTPUT				
pitch_mod_out				
MONO		@nvoices		
AUDIOOUTPUT				
vel_mod_out				
MONO		@nvoices		
AUDIOOUTPUT				
pres_mod_out				
MONO		@nvoices		
AUDIOOUTPUT				
channel	0	17	0	1
			global,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,omni	
CONTROLINPUT	channel			
notemode	0	2	0	1
CONTROLINPUT	notemode			
polymode	0	3	0	1
CONTROLINPUT	polymode			
pressure	0	2	0	1
CONTROLINPUT	pressure			
pitchbend	0	25	0	1
			global,0__notes,1__notes,2__notes,3__notes,4__notes,5__notes,6__notes,7__notes,8__notes,9__notes,10_notes,11_notes,12_notes,13_notes,14_notes,15_notes,16_notes,17_notes,18_notes,19_notes,20_notes,21_notes,22_notes,23_notes,24_notes	
CONTROLINPUT	pitchbend			
gate				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT	gate			
pitch				
		@nvoices		
CONTROLOUTPUT	pitch			
vel				
		@nvoices		
CONTROLOUTPUT	vel			
pres				
		@nvoices		
CONTROLOUTPUT	pres			
obj				
USEROBJECTPARENT	obj			

midinout(out) - MIDI note output

Description
Takes note/presure input and puts it out the MIDI Port.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	1 to 32 number of notes to send

Audio Outputs

Header	Description

Control Inputs

Header	Description
channel	0: global setting, 1-16: channel number
gate	0 to 1. Not on/off with .2 ~ .8 hysteresis.
pitch	0 thru 12700 (MIDI note * 100)
velocity	0 thru 127
pressure	0 thru 127. Key pressure (sent on own change)

Control Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	1	32	1	1
	number_notes			
channel				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
			base,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16	
CONTROLINPUT	channel__			
gate	0	1	0	0.1
		@ninputs		
CONTROLINPUT	gate~n___			
pitch	0	12700	0	1
		@ninputs		
CONTROLINPUT	pitch~n__			
velocity	0	127	64	1
		@ninputs		
CONTROLINPUT	velocity~n			
pressure	0	127	0	1
		@ninputs		
CONTROLINPUT	pressure~n			

Chapter 8 - Filter

cheby1(cheby1) - Chebyshev Type I Filter

Description

Analog style Chebyshev Type I lowpass filter design and implementation. Chebyshev filters are designed using three parameters, the order (number of poles), the amount of ripple allowed in the passband, and the cutoff frequency. Chebyshev Type I filters are very common in analog audio circuits as they are relatively easy to implement, and provide a very steep rolloff after the cutoff frequency. This steep rolloff comes at the expense of having ripple (oscillations in the frequency response) in the passband.

Godlike Productions Comments

Specifiers

Header	Description
order	The order of the filter, also number of poles. More poles gives a steeper roll off, at the expense of more peaks and troughs in the passband ripple.
prewarp	Optional prewarping of the filter cutoff frequency. Becuase the filter is designed analog and then transformed via the bilinear transform the high freqs will be warped (because infinity in analog is mapped to nyquist in digital). If 1 this will warp the specified analog cutoff frequency so that the final digital cutoff frequency matches the given one exactly. Note that although this will cause that exact frequency to match, it might alter the response of the filter at other points (i.e. peaks in the passband ripple that were previously correct could become off). This defaults to off and should only be used with high cutoff frequencies (roughly > 10kHz) when it is critical that that frequency is exactly right.

Audio Inputs

Header	Description
in	The modules audio input.

Audio Outputs

Header	Description
out	The modules audio output.

Control Inputs

Header	Description
cutoff	The cutoff frequency in Hz. This works a little differently for chebyshev filters since it does not specify the -3dB point, but the rather the point at which the frequency response descends below the ripple value for the final time.
ripple	The amount of ripple to allow in the passband, specified in dB. The ripple in the passband will oscillate between a magnitude of 0dB and -ripple.

Header	Description

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
order	1	8	1	1
prewarp	0	1	0	1
INT				
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
cutoff	20	20000	1000	0.01
CONTROLINPUT	fc			
ripple	0.01	6	1	0.01
CONTROLINPUT	rip			

dcblocker(hct) - DC Blocking Filter

Description

Provides a simple, first order,adjustable high-pass filter. The gain at DC is always 0. This filter is very useful for blocking dc offset and can be especially useful in feedback loops.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to the dcblocker module.

Audio Outputs

Header	Description
out	The output. The input with the DC offset removed

Control Inputs

Header	Description
freq	0 to 20000 Hertz. Controls the cutoff frequency of the dcblocker. The cutoff is defined as the point at which the frequency response drops 3 dB.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

dcblocker

Header

Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
out				
AUDIOOUTPUT				
freq	0	20000	5	0.1
	hz			
CONTROLINPUT	%n			

de_emphasis(dee) - De-Emphasis

Description

Provides, to within 0.5 dB, the standard 50 and 15 microsecond de-emphasis. De-emphasis is also performed by the DAC 8x oversampling filter. Pre-emphasis is also provided by the analog input section. This module serves to provide de-emphasis for internally generated digital signals. There are times when you want to process signals without the emphasis we put in. You would use this module to de-emphasis. Then process. and then use the pre-emphasis module to restore the signal to normal.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to the de-emphasis module.

Audio Outputs

Header	Description
out	The de-emphasized output.

Control Inputs

Header	Description

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

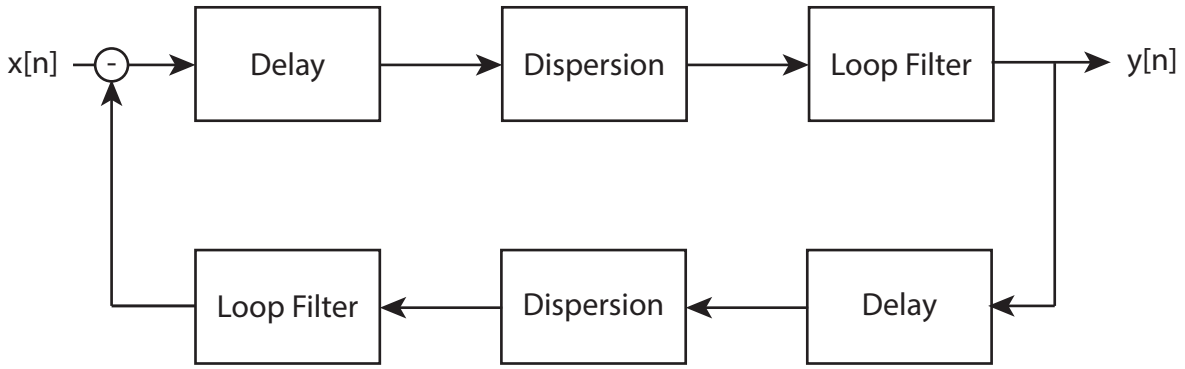
Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				

dispfilt(dfr) - Dispersion Filter

Description

Creates audio dispersion i.e. frequency dependent group delay. This was created for use in a spring reverb effect. This dispersion curve was modeled from an actual Accutronics Type 8 spring reverb and is set by the coefficients of a series of allpass filters. Spring reverbs have two dominant modes of wave propagation that exist in two spectral bands. The lower band delays the higher freqs. more than the lower ones, and the higher band does the opposite. This dispersion curve currently only implements the lower band of a Type 8 spring with dispersion up to about 4kHz. A decent spring sound (lower band) can be simulated by placing two dispersion modules, two delays, and two loop filters (usually lowpass or bandpass) in a waveguide-like structure, shown below:



Future expansion should include curves for the Type 8 higher band as well as other types of Spring reverbs e.g. an Accutronics Type 4, etc.

Godlike Productions Comments

Specifiers

Header	Description
mode	0 or 1, Spring dispersion curve mode. 0: Less life-like heuristally pruned "cycle lite" version 1: more like-like and accuracte spring dispersion curve, but also more expensive (about twice the 0 mode option).
nsects	number of second order sections

Audio Inputs

Header	Description
in	audio input

Audio Outputs

Header	Description
out	audio output

Control Inputs

Header	Description

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mode	0	1	0	1
	mode			
nsects	1	44/24/@ mode/#if	24	1
INT	sects			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

distort(dst) - Guitar Distortion

Description

This module implements a guitar distortion effect. The audio input is mapped to an audio output through a continuous curve. To approximate tube loading, and to allow for new effects, the module can morph between two different curves.

Godlike Productions Comments

Specifiers

Header	Description
morphon	Enables/disables curve morphing.
oversamp	introduces varying levels of oversampling with a gradual lowpass filter. 0 - no oversampling or filtering 1 - just applies the FIR lowpass with no oversampling 2-3 - oversamples by 2 4-7 - oversamples by 4 8 - oversamples by 8
preventWaveFolding	controls whether or not the signal is clipped to the optimal values based on the curve, or just clipped to -1 and +1. If the signal is clipped to +/- 1, some curves may start wavefolding, especially with higher pregain values.

Audio Inputs

Header	Description
in	The audio input to be distorted (typically guitar).

Audio Outputs

Header	Description
out	The distorted output.

Control Inputs

Header	Description
bias	Can be used to add DC to the input to change the harmonic characteristics of the output (note that it will also add DC to the output).
curve1	Chooses the primary mapping curve.
curve2	If enabled, chooses the secondary mapping curve.
pregain	Can be used to increase or decrease gain of the input before it is mapped through the curve(s).
expansion	Presumably expansion to compensate for compression through the distortion. Not documented.

Mod Inputs

Header	Description

Mod Outputs

Header	Description
mod	If curve morphing is enabled, the value of mod will determine at what point between the curves the output will be. (for example, if mod = 0.1, the audio output will be 0.9*out1 + 0.1*out2).

User Objects

Header	Description

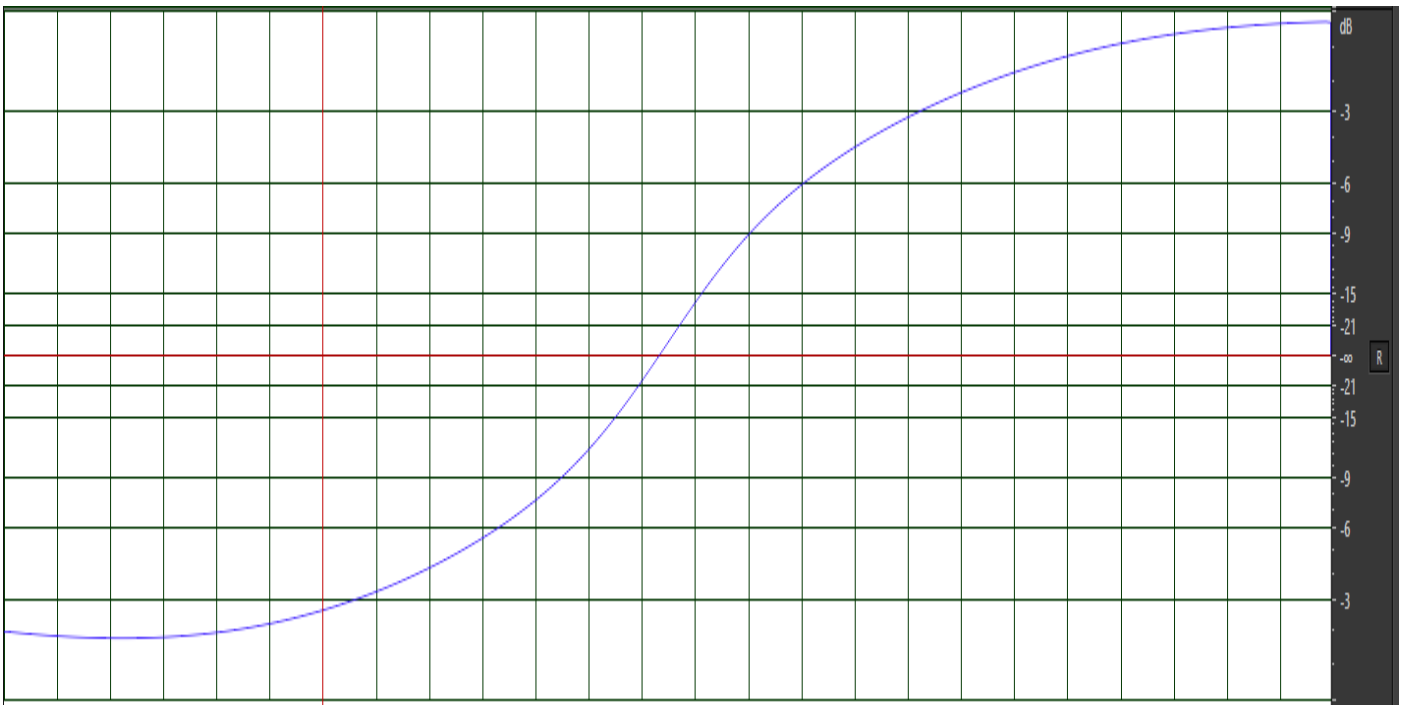
Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
morphon	0	1	0	1
	morphenable			
oversamp	0	8	0	1
INT	oversampling			
preventWaveFolding	0	1	0	1
INT	prevent wave folding			
in				
MONO				
AUDIOINPUT				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
curve1	0	9		
CONTROLINPUT				
curve2	0	@morphon/9/#mul		
CONTROLINPUT				
pregain	-96	48	0	1
	db			

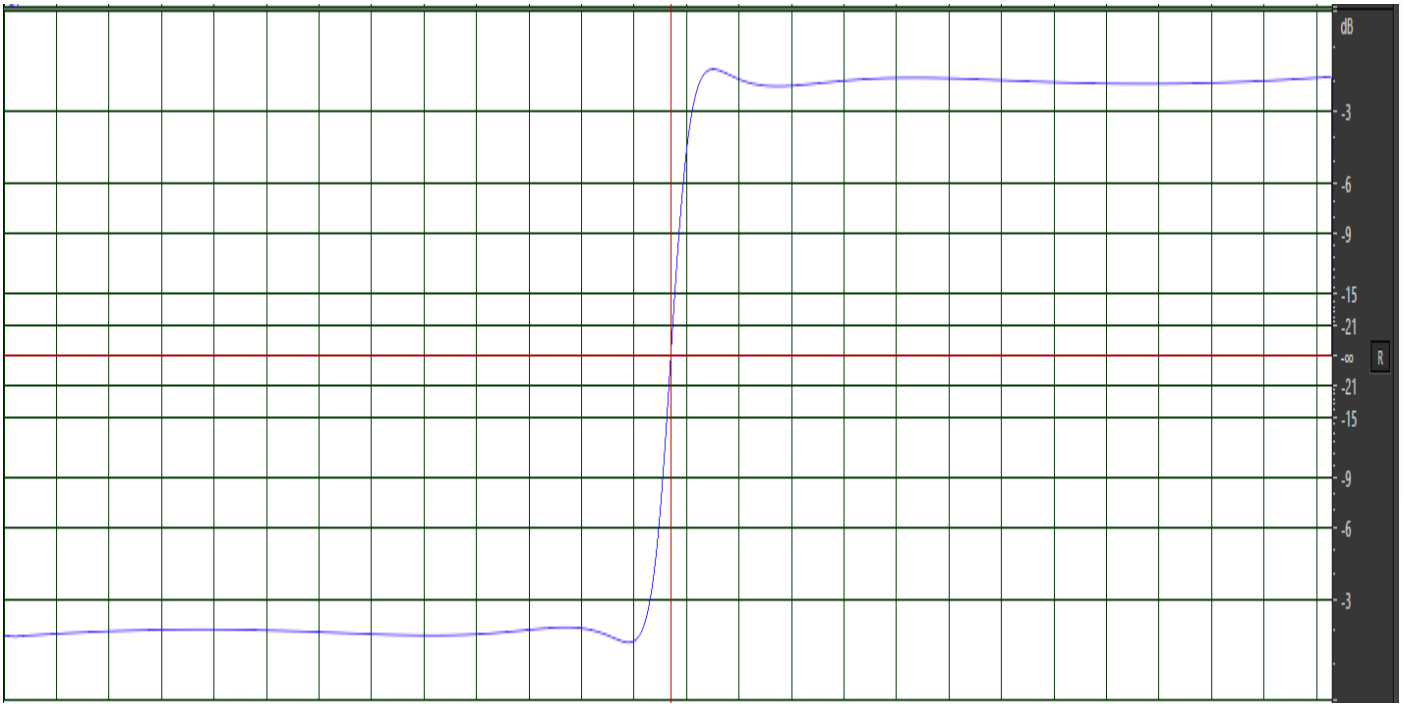
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	pregain			
bias	-0.1	0.1	0	0.00001
CONTROLINPUT	bias			
expansion	-32768	32767	0	0.00001
CONTROLINPUT	expansion			

Filter Curves

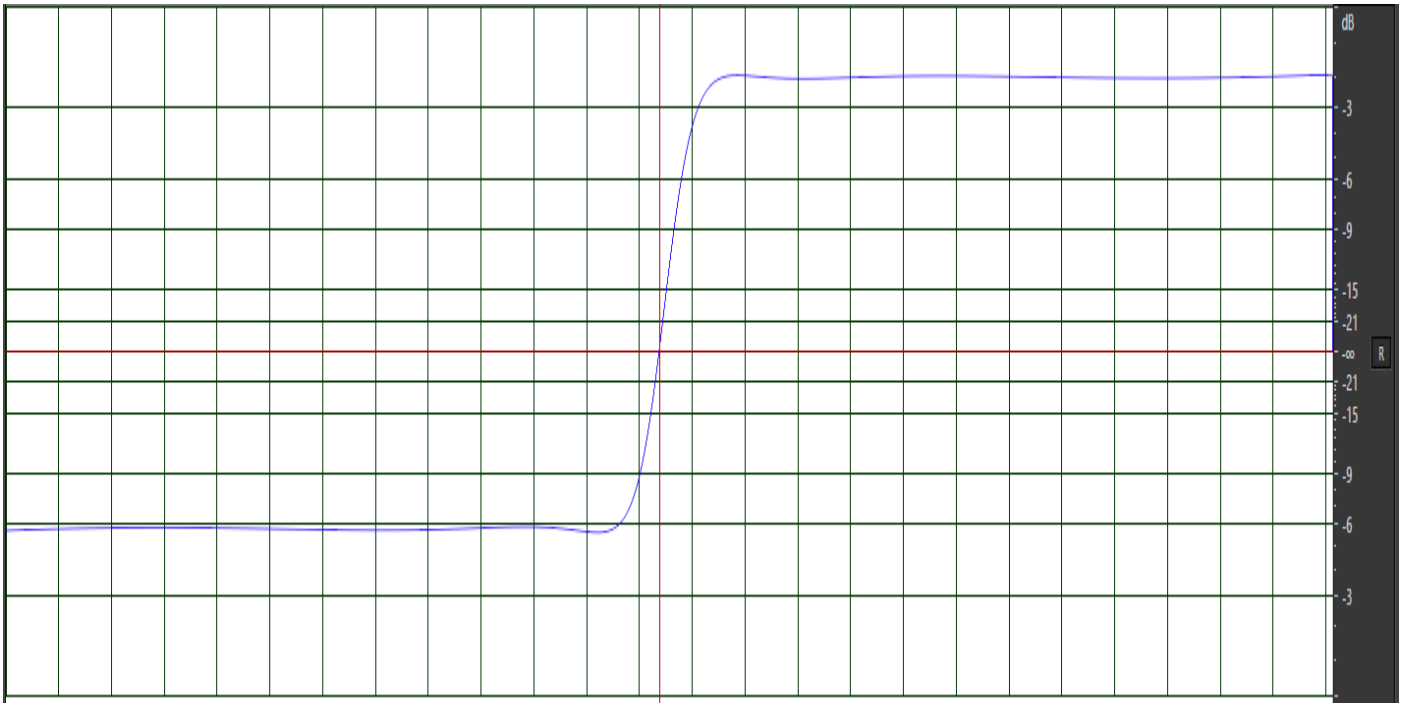
The diagrams below show the output (vertical) as the input increases from -full scale to positive full scale (horizontal). The DC offsets are as recorded from the H9000 outputs.



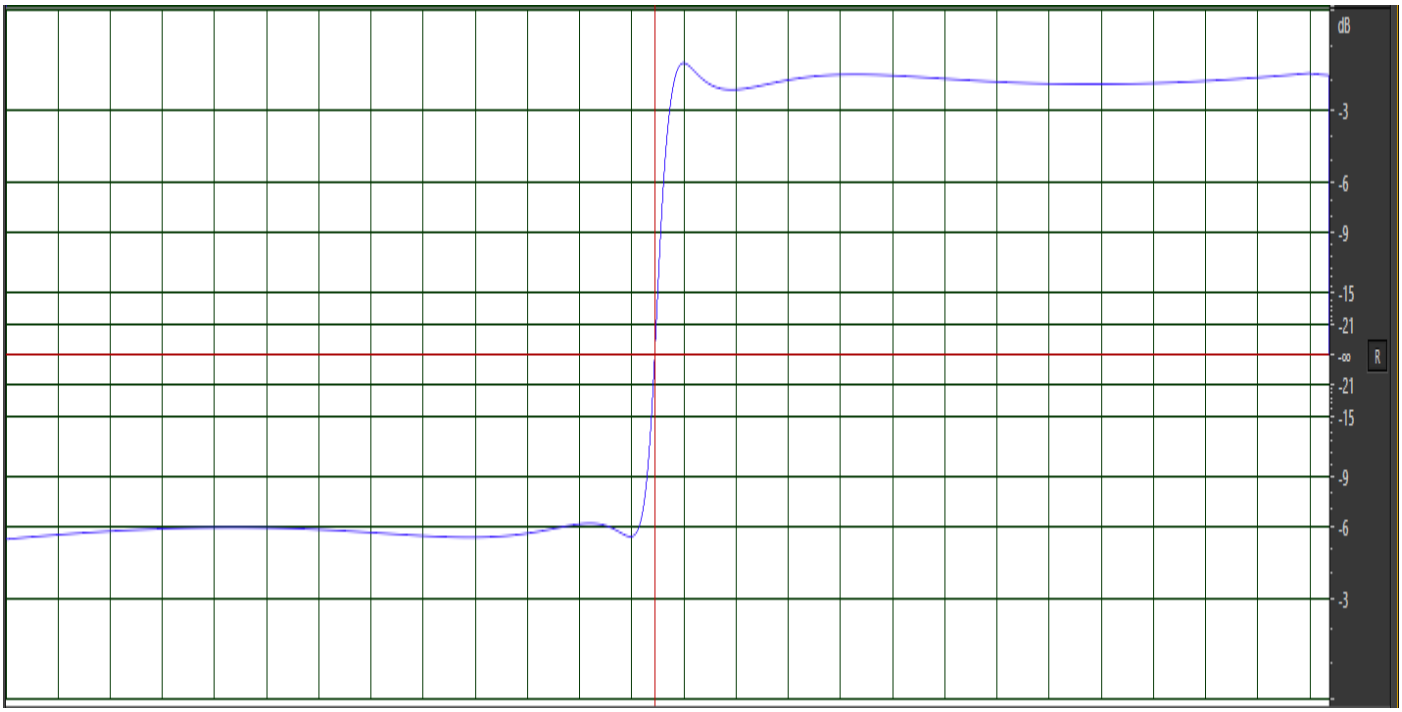
Curve 0: 0dB Gain



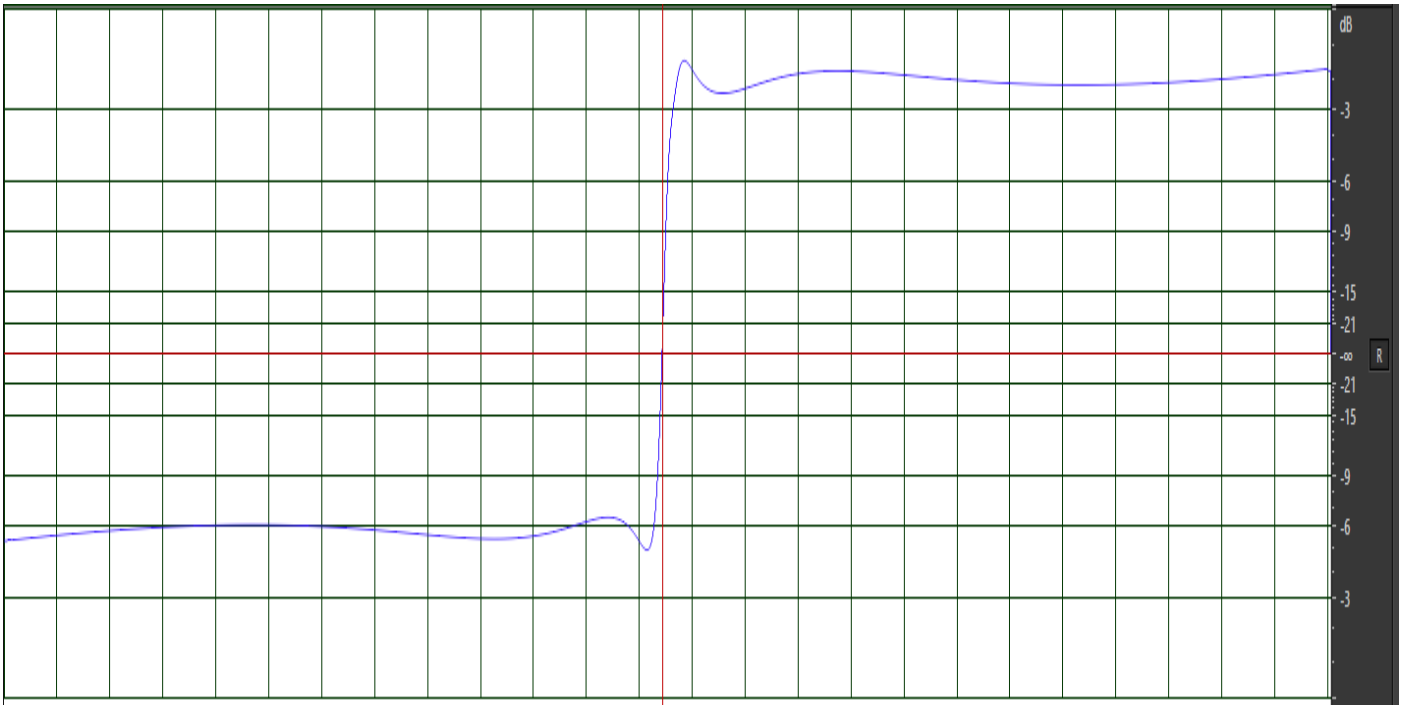
Curve 1: 0dB Gain



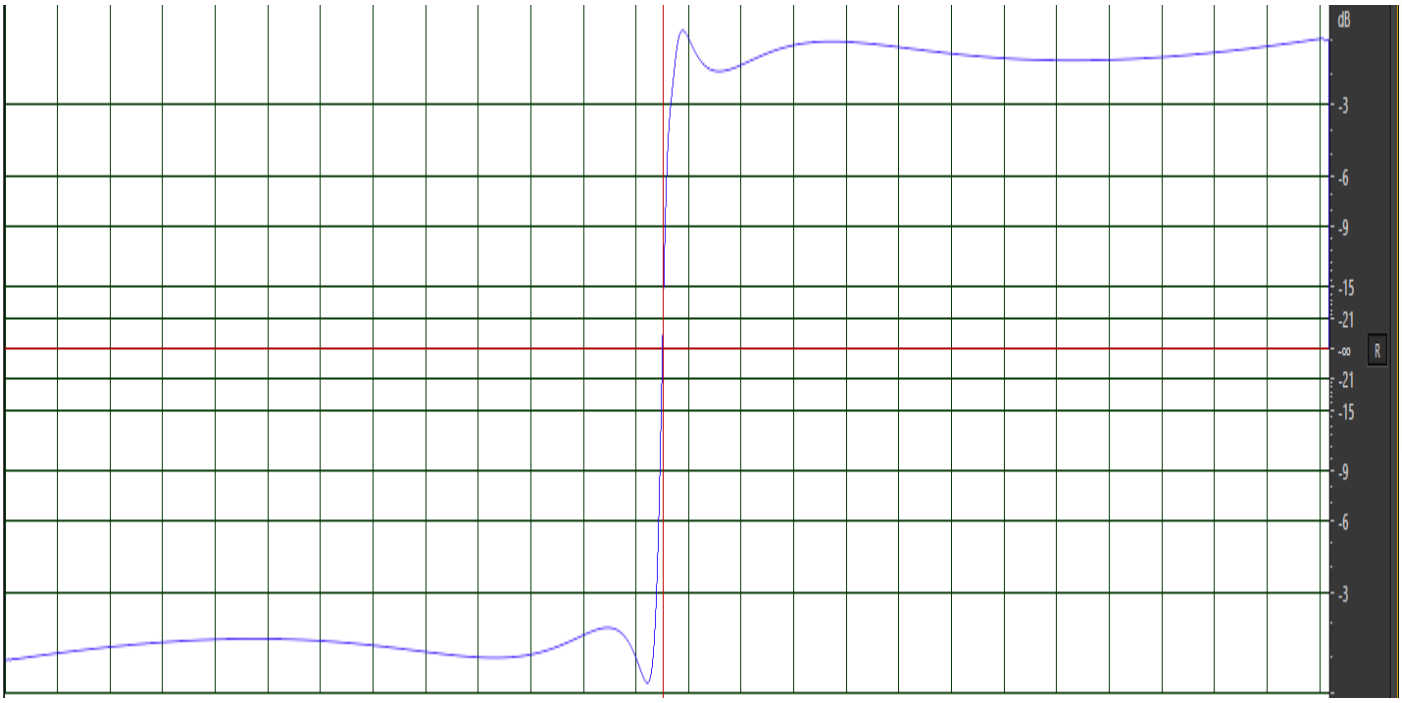
Curve 2: 0dB Gain



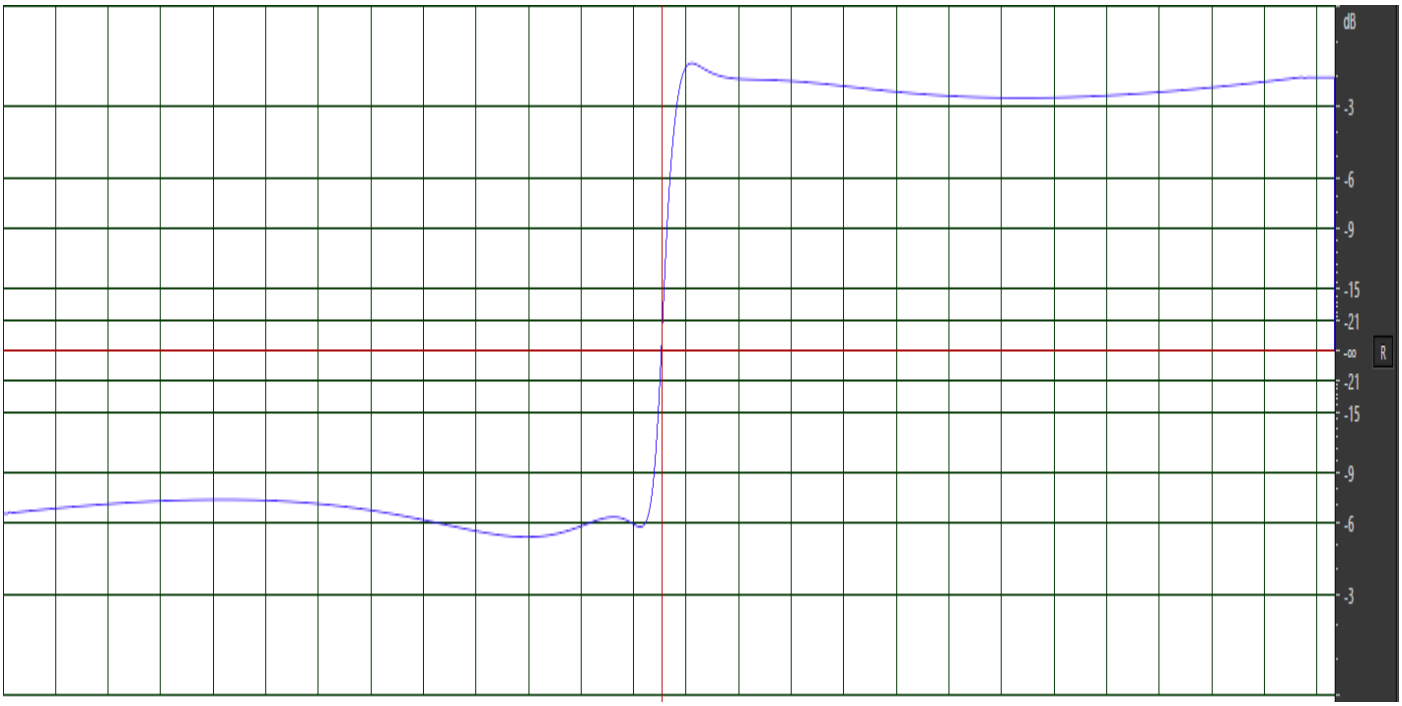
Curve 3: 0dB Gain



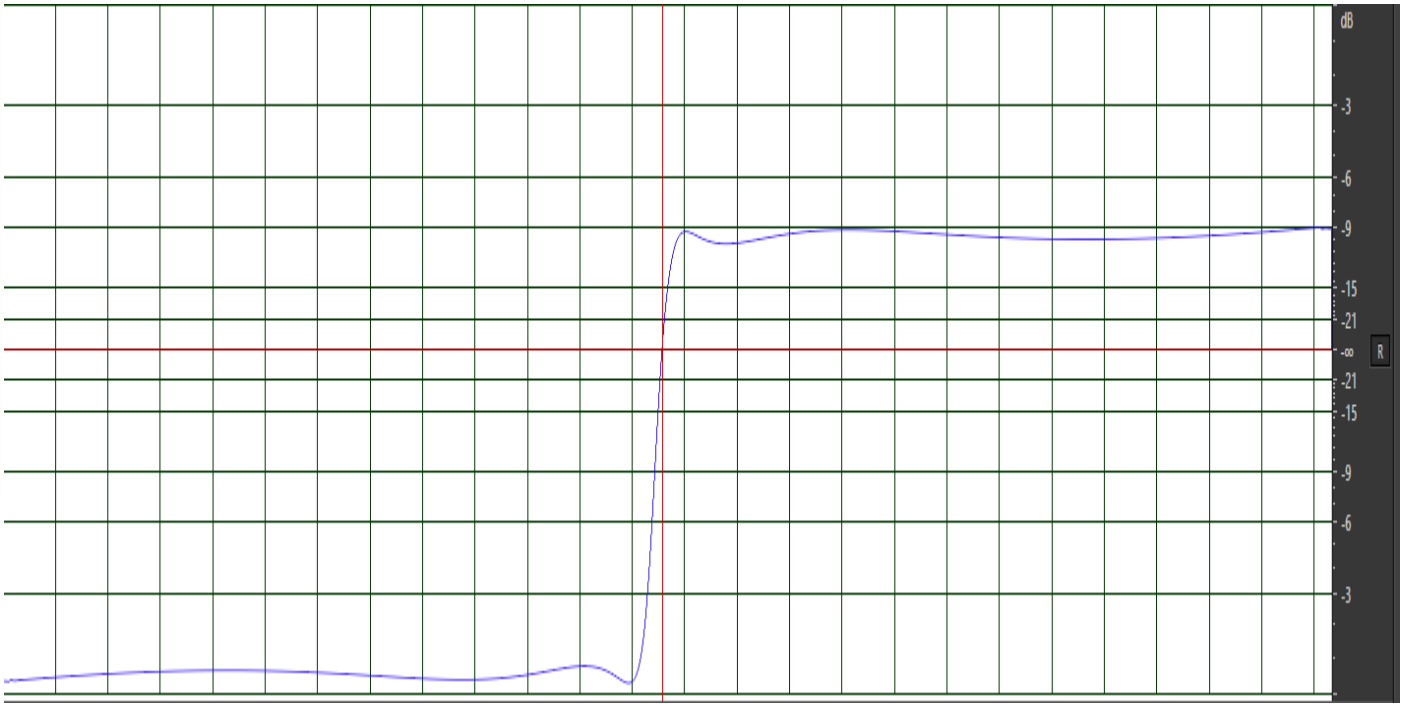
Curve 4: 0dB Gain



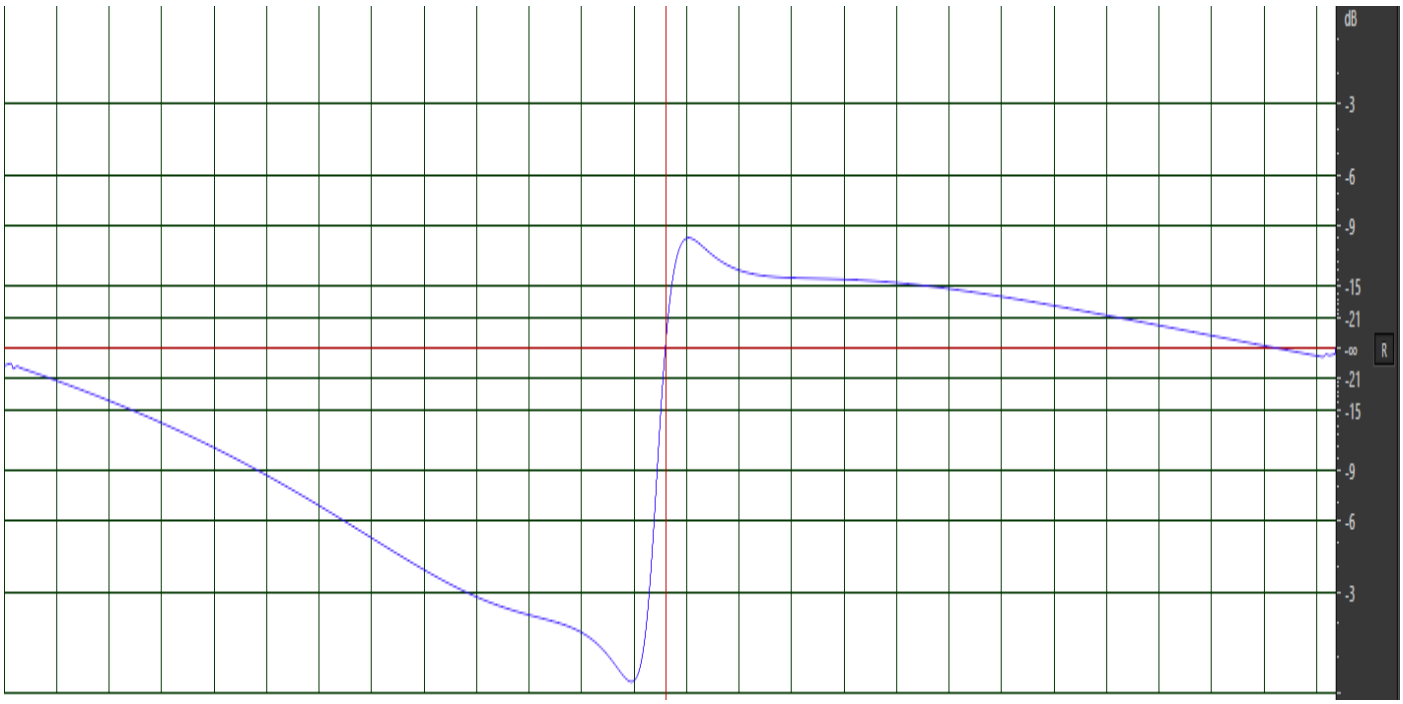
Curve 5: 0dB Gain



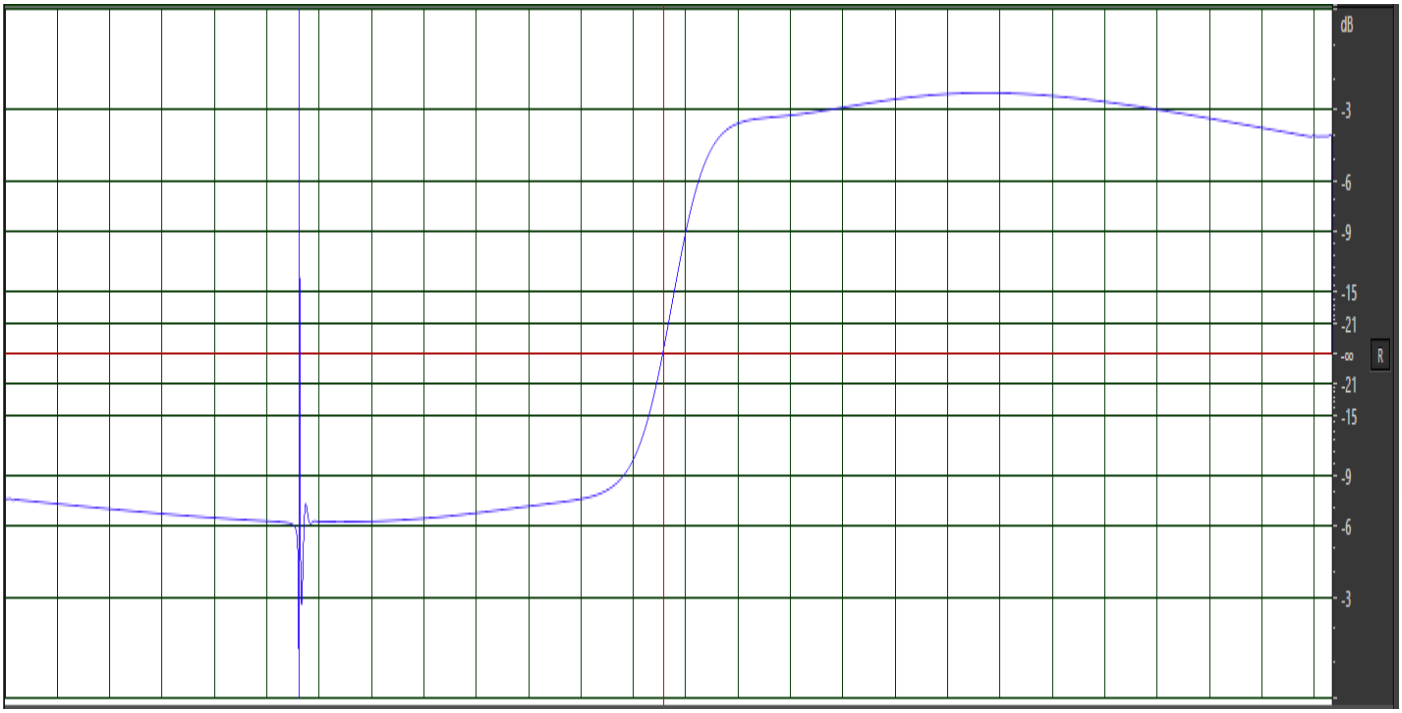
Curve 6: 0dB Gain



Curve 7: 0dB Gain



Curve 8: 0dB Gain



Curve 9: 0dB Gain

eq(eq) - Equalizer

Description

This module implements a standard boost/cut type audio equalizer. The equalizer type can be selected to be low shelving (boost/cut below specified freq), peaking (boost/cut a particular freq band), or high shelving (boost/cut above specified freq).

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the EQ clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in	The eq input.

Audio Outputs

Header	Description
out	The eq output.

Control Inputs

Header	Description
freq	20 to 20000 Hertz. Adjust the frequency range over which the equalizer acts. The specific effect of this parameter depends upon the type control setting.
qfactor	0.5 to 1000. Controls the shape of the equalizer frequency response. For a peaking equalizer this directly controls the width of the band to be boost/cut. The bandwidth is equal to the freq setting divided by q. This means that higher q settings result in narrower band equalization. For low and high shelving eq, the q setting affects the steepness of the eq at the spcified frequency. Settings above 1.0 will also begin to have a resonant peak at the selected frequency. For high q settings, these eq types will sound very "bandpassish".
boost	-18.0 to 18.0 dB. Controls how much the specified band will be boost or cut. Positive values means the band will be boosted (increased in level) and negative values mean the band will be cut.
type	This controls the type of eq this module will perform. The values are as follows: 0 - Low Shelving EQ, 1 - Peaking EQ (Presence), 2 - High Shelving EQ.

Header Description

Mod Inputs

Header Description

Mod Outputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clipvalue	0	10000	10000	0.1
	clip_value			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	20	32000/20000/#ifhi	1000	0.1
	hz			
CONTROLINPUT	freq			
qfactor	0.01	1000	1	0.0001
CONTROLINPUT	q			
boost	-18	18	0	0.1
	db			
CONTROLINPUT	boost			
type	0	2		
CONTROLINPUT	type			
obj				
USEROBJECTPARENT	obj			

eqn(eqn) - Series Equalizer

Description

This module contains multiple eq type modules connected serially. Up to eight standard boost/cut type audio equalizers can be implemented in series per module. The equalizer types can be selected to be low shelving (boost/cut below specified freq), peaking (boost/cut a particular freq band), or high shelving (boost/cut above specified freq). The eq modules are implemented to minimize clicks caused by parameter changes.

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the EQ clips. Range is 0 to 10000.
neqs	The number of serial eqs desired.

Audio Inputs

Header	Description
in	The eq bank's input.

Audio Outputs

Header	Description
out	The eq bank's output.

Control Inputs

Header	Description
freq[n]	20 to 20000 Hertz. Adjust the frequency range over which the equalizer acts. The specific effect of this parameter depends upon the type control setting.
qfactor[n]	0.5 to 1000. Controls the shape of the equalizer frequency response. For a peaking equalizer this directly controls the width of the band to be boost/cut. The bandwidth is equal to the freq setting divided by q. This means that higher q settings result in narrower band equalization. For low and high shelving eq, the q setting affects the steepness of the eq at the spcified frequency. Settings above 1.0 will also begin to have a resonant peak at the selected frequency. For high q settings, these eq types will sound very "bandpassish".
boost[n]	-18.0 to 18.0 dB. Controls how much the specified band will be boost or cut. Positive values means the band will be boosted (increased in level) and negative values mean the band will be cut.
type[n]	This controls the type of eq this module will perform. The values are as follows: 0 - Low Shelving EQ, 1 - Peaking EQ (Presence), 2 - High Shelving EQ.

Header Description

Mod Inputs

Header Description

Mod Outputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
neqs	1	8	1	1
	num_eqs			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	20	32000/20000/#ifhi	1000	0.1
	hz	@neqs		
CONTROLINPUT	freq~n			
qfactor	0.5	1000	1	0.1
		@neqs		
CONTROLINPUT	q~n			
boost	-18	18	0	0.1
	db	@neqs		
CONTROLINPUT	boost~n			
type	0	3		
		@neqs		

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	type~n			

eqs(eqs) - Double Precision Series Equalizer

Description

This module contains multiple double precision eq and filter sections connected serially. Up to eight standard boost/cut type audio equalizers or filters can be implemented in series per module. The equalizer types can be selected to be 6 dB/Oct low cut, 12 dB/Oct low cut, low shelf, classic peaking, modern peaking, bandpass, notch, high shelf, 12 dB/Oct high cut, 6 dB/Oct high cut, or bypass. The eq modules are implemented to minimize clicks caused by parameter changes. There is 12 dB of internal headroom to avoid internal clipping.

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the EQ clips. Range is 0 to 10000.
neqs	The number of serial eqs desired.

Audio Inputs

Header	Description
in	The eq bank's input.

Audio Outputs

Header	Description
out	The eq bank's output.

Control Inputs

Header	Description
freq[n]	5 to 20000 Hertz. Adjust the frequency range over which the equalizer acts. The specific effect of this parameter depends upon the type control setting.
qfactor[n]	0.5 to 1000. Controls the shape of the equalizer frequency response. For a peaking equalizer this directly controls the width of the band to be boost/cut. The bandwidth is equal to the freq setting divided by q. This means that higher q settings result in narrower band equalization. For low and high shelving eq, the q setting affects the steepness of the eq at the spcified frequency. Settings above 1.0 will also begin to have a resonant peak at the selected frequency. For high q settings, these eq types will sound very "bandpassish". This control will not affect the 6dB/Oct high or low cut filters.
boost[n]	-24.0 to 24.0 dB. Controls how much the specified band will be boost or cut. Positive values means the band will be boosted (increased in level) and negative values mean the band will be cut. This control does not affect the high or low cut filters or the bandpass filter. For the notch filter a lower boost will create a shallower notch and a higher boost will create a deeper notch all the way up to +24 dB = ultimate notch.

Header	Description
type[n]	This controls the type of eq this module will perform. The values are as follows: 0 - 6dB/Octave Low Cut Filter, 1 - 12 dB/Octave Low Cut Filter, 2 - Low Shelving EQ, 3 - Classic Peaking EQ (Presence), 4 - Modern Peaking EQ (Presence), 5 - Bandpass Filter, 6 - Notch Filter, 7 - High Shelving EQ, 8 - 12 dB/Octave High Cut Filter, 9 - 6 dB/Octave High Cut Filter, 10 - Bypass.
processoff	0 or 1, turns off processing

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
neqs	1	8	1	1
	num_eqs			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	5	32000/20000/#ifhi	1000	0.01
	hz	@neqs		
CONTROLINPUT	freq~n			
qfactor	0.5	1000	1	0.1
		@neqs		
CONTROLINPUT	q~n			
boost	-24	24	0	0.1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	db	@neqs		
CONTROLINPUT	boost~n			
type	0	10		
		@neqs		
CONTROLINPUT	type~n			
processoff	0	1	0	1
CONTROLINPUT	turn_off			

filter(flt) - Audio Frequency Filter

Description

Provides a configurable and adjustable second order filter. This module can be configured as a low-pass, high-pass, band-pass, or band-reject (notch) filter.

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the EQ clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in	The filter input.

Audio Outputs

Header	Description
out	The filter output.

Control Inputs

Header	Description
freq	1 to 20000 Hertz. Adjust the frequency range over which the filter acts. The specific effect of this parameter depends upon the type control setting.
qfactor	0.5 to 1000. Controls the shape of the filter frequency response. For a band-pass filter this directly width of the band to be passed. The bandwidth is equal to the freq setting divided by q. This means that higher q settings result in a narrower band filter. For lowpass and highpass filters, the q setting affects the steepness of the eq at the spcified frequency. Settings above 1.0 will also begin to have a resonant peak at the selected frequency. For high q settings, these eq types will sound very “bandpassish”.
type	This controls the type of filtering this module will perform. The values are as follows: 0 - Lowpass (Passes frequencies below freq setting), 1 - Bandpass (Passes frequencies in a band centered at freq setting), 2 - Highpass (Passes frequencies above freq setting), 3 - Notch (Rejects frequencies at and around freq setting)

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clipvalue	0	10000	10000	0.1
	clip_value			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	1	20000	1000	0.1
	hz			
CONTROLINPUT	frequency			
qfactor	0.1	1000	1	0.1
CONTROLINPUT	q			
type	0	3		
			lowpass,bandpass,highpass,notch	
CONTROLINPUT	type			

fir(fir) - Finite Impulse Response filter

Description

This module provides an FIR filter which can act as a Lowpass, Highpass, Bandpass or Stopband filter. Its impulse response is determined by the windowing method and the user has the choice among various windows.

Godlike Productions Comments

Specifiers

Header	Description
maxtaps	[3 .. 127], Maximum number of filter taps. maxtaps is constrained to be an odd # only, to guarantee Type 1 FIR filters (Type II or even numbered FIR filters don't have complementary output ability). The actual number will be set by a control input (NumTaps). Maxtaps will however determine the group delay of the filtered and dry outputs equal to (numtaps-1)/2 samples.
declick	0 or 1, set to 1 turn on declick xfading, with xfade time set by the xfade control.

Audio Inputs

Header	Description
in	The audio stream to be processed through the FIR filter.

Audio Outputs

Header	Description
outmn	The main audio stream after filtering.
outcp	The complementary audio stream after filtering.
outdry	A phase aligned dry output to the filtered output.

Control Inputs

Header	Description
lowcut	Hz, [0 ..fs / 2]. Low frequency cutoff for Highpass filter part.
highcut	Hz, [0 ..fs / 2]. High frequency cutoff for Lowpass filter part.
numtaps	[3 ..maxtaps]. Number of taps actually used in the filter. Please note : Even #s of taps result in Type 2 FIR filters, that don't have a transformation to a true complementary signal that is not the opposite filtering operation like the Type 1 odd tap # FIRs. Because of this, the FIR module will always round up the nearest odd number.
window	[0 .. 5]. Type of window to use to get filter's impulse response: 0 Rectangular; 1 Triangular; 2 von Hann; 3 Hamming; 4 Blackmann; 5 Kaiser;
beta	[0.01 .. 20.0]. Additional parameter for the Kaiser Window.
xfade	[0 .. 100] ms. Sets the xfade time used to declick changes to lowcut and highcut (if declick is On). Also sets the xfade for the mute fade for turning the process off regardless of declick is on or not.

Header	Description
processoff	0 or 1, turns off processing

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxtaps	3	127	63	2
	max_taps			
declick	0	1	0	1
INT	declick_on			
in				
MONO				
AUDIOINPUT				
outmn				
MONO				
AUDIOOUTPUT				
outcp				
MONO				
AUDIOOUTPUT				
outdry				
MONO				
AUDIOOUTPUT				
lowcut	0	20000	1000	0.1
	Hz			
CONTROLINPUT	lo_cut			
highcut	0	20000	10000	0.1
	Hz			
CONTROLINPUT	hi_cut			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
numtaps	3	@maxtaps	31	2
CONTROLINPUT	num_taps			
windowtype	0	5	3	
			Rectangular,Triangular,Von_Han-n,Hamming,Blackman,Kasier	
CONTROLINPUT	Window			
beta	0.01	20	5	0.01
CONTROLINPUT	beta			
xfade	0	@de-click/100/#mul	0	1
CONTROLINPUT	declick_xfade			
processoff	0	1	0	1
CONTROLINPUT	turn_off			

h3000svf(hsv) - State Variable Implementation from the H3000

Description

This module implements a classic Chamberlin style state-variable audio filter with the same control scheme commonly found in the H3000. It provides simultaneous lowpass, bandpass, highpass, and notch outputs. It has a variable "Q" control which ranges from 0.5 to 1000 and is set to the same taper as that in the H3000 Band Delays algorithm (and others). It also has variable Frequency and mod rate frequency and q factor modulation inputs. At high "Q" values this filter will also self oscillate. This is the module to use to create any type of swept filter effects.

Godlike Productions Comments

There is something up with this filter. I get high resonance amounts at very low levels of Q. The documentation in Vsig 3.3.3 is wrong for this filter.

Specifiers

Header	Description
stabilityprotect	No documentation in the json file. Most likely a scalable stability protection mechanism reducing feedback gain. Turning this off may allow filter instability (ie runaway feedback), 1 may allow self oscillation at the margin of stability and 2 most likely to guarantee stability and not allow self oscillation.

Audio Inputs

Header	Description
in	The filter input.
pmod	Modulation input for cutoff frequency. (-1 to 1)
qmod	Modulation input for q factor (-1 to 1)

Audio Outputs

Header	Description
low	The lowpass filter output
band	The bandpass filter output
high	The highpass filter output
notch	The notch filter output.

Control Inputs

Header	Description
pitch	0 to 10700 Cents (MIDI note * 100 I suspect). Controls the center frequency (or cutoff) of all filter outputs.
pitchmodamount	-12800 to 12800 Cents. Adjusts how much the pmod input will modulate the filter centre frequency.
q	0.5 to 1000. Adjusts the sharpness of the filter. The bandwidth of the filter is Freq/Q
qmodamt	-1000 to 1000. Controls how much the qmod input modulate the filter q.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
stabilityprotect	0	2	1	
	stabilityprotect_on_off			
in				
MONO				
AUDIOINPUT				
pmod				
MOD				
AUDIOINPUT				
qmod				
MOD				
AUDIOINPUT				
low				
MONO				
AUDIOOUTPUT				
band				
MONO				
AUDIOOUTPUT				
high				
MONO				
AUDIOOUTPUT				
notch				
MONO				
AUDIOOUTPUT				
pitch	0	10700	5700	1
	cents			
CONTROLINPUT	pitch			
pitchmodamt	-12800	12800	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	cents			
CONTROLINPUT	pitch_mod			
q	0.5	1000	1	0.1
CONTROLINPUT	q			
qmodamt	-1000	1000	0	0.1
CONTROLINPUT	q_mod			

harmonix - Discrete Harmonics Generator

Description

The harmonix module generates a number of harmonics of an input tone. It does so by iteratively applying the recurrence relation for Chebshev's T-polynomial: T[n](x) = 2*x*T[n-1](x) - T[n-2](x), given T[0](x) = 1 and T[1](x) = x. The property of the T-polynomial exploited is this: T[n](cos(w)) = cos(n*w). This works best if the input is a pure sinusoid with exactly unit amplitude. This rarely happens, which leads to a DC offset for even orders of T. The remedy here is to apply a DC-block filter to the output of every other harmonic.

Godlike Productions Comments

Specifiers

Header	Description
nharmonics	Specifies the number of harmonics to generate.
avgtime	Specifies the reciprocal of the DC-block co-efficient.

Audio Inputs

Header	Description
in	Input tone

Audio Outputs

Header	Description
out[n]	The harmonics of the input tone

Control Inputs

Header	Description
dummy	null

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nharmonics	4	64	10	2
	number_harmonics			
avgtime	1	5000	100	1
FLOAT	average_time			
in				
MONO				
AUDIOINPUT				
out				
MONO		@nharmonics		
AUDIOOUTPUT				
dummy	0	1	0	1
	nil			
CONTROLINPUT	null			

highcut(hct) - Highcut Filter

Description

Provides a simple, first order,adjustable low-pass filter. The gain at DC is always 0 dB. This filter is very useful for gently rolling off the high end to produce a warm, analog sound.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to the highcut module.

Audio Outputs

Header	Description
out	The filtered output.

Control Inputs

Header	Description
freq	0 to 20000 Hertz. Controls the cutoff frequency of the highcut. The cutoff is defined as the point at which the frequency response drops 3 dB.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
out				
AUDIOOUTPUT				
freq	0	20000	0	0.1
	hz			
CONTROLINPUT	%n			
obj				
USEROBJECTPARENT	obj			

iir(iir) - Audio Frequency Filter

Description

Provides a configurable and adjustable high order filter. This module can be configured as a low-pass, high-pass, band-pass, or band-reject (notch) filter. Uses first order noise shaping to reduce perceptable noise and eliminate DC offset problems.

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the EQ clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in	The filter input.

Audio Outputs

Header	Description
out	The filter output.

Control Inputs

Header	Description
freq	20 to 20000 Hertz. Adjust the frequency range over which the filter acts. The specific effect of this parameter depends upon the type control setting.
qfactor	0.5 to 1000. Controls the shape of the filter frequency response. For a band-pass filter this directly width of the band to be passed. The bandwidth is equal to the freq setting divided by q. This means that higher q settings result in a narrower band filter. For lowpass and highpass filters, the q setting affects the steepness of the eq at the spcified frequency. Settings above 1.0 will also begin to have a resonant peak at the selected frequency. For high q settings, these eq types will sound very “bandpassish”.
type	This controls the type of filtering this module will perform. The values are as follows: 0 - Lowpass (Passes frequencies below freq setting), 1 - Bandpass (Passes frequencies in a band centered at freq setting), 2 - Highpass (Passes frequencies above freq setting), 3 - Notch (Rejects frequencies at and around freq setting)

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
n_sections	1	20	1	1
INT	number_sections			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
freq	20	20000	1000	0.1
	hz			
CONTROLINPUT	frequency			
qfactor	0.5	1000	1	0.001
CONTROLINPUT	q			
type	0	3		
CONTROLINPUT	type			

mod_slew (msl) - slew limit mod signal

Description

This module will limit the slew rate of a mod signal. The slew rate is how fast the signal changes.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description

Audio Outputs

Header	Description

Control Inputs

Header	Description
pslew	Positive slew limit. This is used when input is higher than output. How fast in milliseconds does it take to go from 0 to full scale. 0 disables function.
nslew	Negative slew limit. This is used when input is lower than output. How fast in milliseconds does it take to go from 0 to full scale.

Mod Inputs

Header	Description
in	Mod signal to be processed.

Mod Outputs

Header	Description
out	Slew limited Mod Output

User Objects

Header **Description**

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MOD				
out				
MOD				
AUDIOOUTPUT				
pslew	0	60	1	0.001
	s			
CONTROLINPUT	positive			
nslew	0	60	1	0.001
	s			
CONTROLINPUT	negative			

modfilter(mfr) - Modulatable Filter

Description

This module implements a classic state-variable audio filter. It provides simultaneous lowpass, band-pass, highpass, and notch outputs. It has variable Q (1/bandwidth) and frequency and has mod rate frequency and q factor modulation inputs. This is the module to use to create any type of swept filter effects.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The filter input.
fmod	Modulation input for cutoff frequency. (-1 to 1)
qmod	Modulation input for q factor (-1 to 1)

Audio Outputs

Header	Description
low	The lowpass filter output
band	The bandpass filter output
high	The highpass filter output
notch	The notch filter output.

Control Inputs

Header	Description
freq	0 to 20000 Hertz. Controls the center frequency (or cutoff) of all filter outputs.
processoff	0 or 1, turns off processing
freqmodamt	-20000 to 20000 Hertz. Adjusts how much the freqmod input will modulate the filter center frequency.
q	0.5 to 1000. Adjusts the sharpness of the filter. The bandwidth of the filter is Freq/Q
qmodamt	-1000 to 1000. Controls how much the qmod input modulate the filter q.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
fmod				
MOD				
AUDIOINPUT				
qmod				
MOD				
AUDIOINPUT				
low				
MONO				
AUDIOOUTPUT				
band				
MONO				
AUDIOOUTPUT				
high				
MONO				
AUDIOOUTPUT				
notch				
MONO				
AUDIOOUTPUT				
freq	0	20000	1000	0.1
	hz			
CONTROLINPUT	freq			
freqmodamt	-20000	20000	0	0.1
	hz			
CONTROLINPUT	freq_mod			
q	0.5	1000	1	0.1
CONTROLINPUT	q			
qmodamt	-1000	1000	0	0.1
CONTROLINPUT	q_mod			
processoff	0	1	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	turn_off			

modfiltp(mfp) - Modulatable Filter with pitch input

Description

This module implements a classic state-variable audio filter. It provides simultaneous lowpass, band-pass, highpass, and notch outputs. It has variable Q (1/bandwidth) and frequency and has mod rate frequency and q factor modulation inputs. This is the module to use to create any type of swept filter effects.

Godlike Productions Comments

Specifiers

Header	Description
stabilityprotect	No documentation in the json file. Most likely a scalable stability protection mechanism reducing feedback gain. Turning this off may allow filter instability (ie runaway feedback), 1 may allow self oscillation at the margin of stability and 2 most likely to guarantee stability and not allow self oscillation.

Audio Inputs

Header	Description
in	The filter input.
pmod	Modulation input for cutoff frequency. (-1 to 1)
qmod	Modulation input for q factor (-1 to 1)

Audio Outputs

Header	Description
low	The lowpass filter output
band	The bandpass filter output
high	The highpass filter output
notch	The notch filter output.

Control Inputs

Header	Description
pitch	0 to 10700 Cents (MIDI note * 100 I suspect). Controls the center frequency (or cutoff) of all filter outputs.
pitchmodamt	-12800 to 12800 Cents. Cadjusts how much the pmod input will modulate the filter centre frequency.
q	0.5 to 1000. Adjusts the sharpness of the filter. The bandwidth of the filter is Freq/Q
qmodamt	-1000 to 1000. Controls how much the qmod input modulate the filter q.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
stabilityprotect	0	2	1	
	stabilityprotect_on_off			
in				
MONO				
AUDIOINPUT				
pmod				
MOD				
AUDIOINPUT				
qmod				
MOD				
AUDIOINPUT				
low				
MONO				
AUDIOOUTPUT				
band				
MONO				
AUDIOOUTPUT				
high				
MONO				
AUDIOOUTPUT				
notch				
MONO				
AUDIOOUTPUT				
pitch	0	10700	5700	1
	cents			
CONTROLINPUT	pitch			
pitchmodamt	-12800	12800	0	1
	cents			
CONTROLINPUT	pitch_mod			
q	0.5	1000	1	0.1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	q			
qmodamt	-1000	1000	0	0.1
CONTROLINPUT	q_mod			

modhighcut(mhc) - Highcut Filter

Description

Provides a simple, first order,adjustable low-pass filter, with a mod input for smooth changes of the cut-off freqquency. The gain at DC is always 0 dB. This filter is very useful for gently rolling off the high end to produce a warm, analog sound.

Godlike Productions Comments

Specifiers

Header	Description
modsw	Selects the modulation type. 0 is the old chunk rate modulation of the cutoff frequency and 1 is for true audio rate modulation. Defaults to chunk rate.

Audio Inputs

Header	Description
in	The input to the highcut module.
mod	Modulation input for cutoff frequency.

Audio Outputs

Header	Description
out	The filtered output.

Control Inputs

Header	Description
freq	0 to 20000 Hertz. Controls the cutoff frequency of the highcut. The cutoff is defined as the point at which the frequency response drops 3 dB.
freqmodamt	0 to 20000 Hertz. Adjusts how much the mod input will modulate the filter center frequency.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
modsw	0	1	0	1
	mod_select			
in				
AUDIOINPUT				
mod				
MOD				
AUDIOINPUT				
out				
AUDIOOUTPUT				
frequency	0	20000		
CONTROLINPUT				
frequencymodamt	0	20000		
CONTROLINPUT				

modiirq(mfq) - Audio Frequency Filter

Description

This module has been marked as deprecated as of VSIG 3.4.2. It will remain available until another modulatable version is available. If audio modulation is not required, it is recommended to use the iir module instead.

Provides a configurable and adjustable high order filter with a modulatable cutoff frequecy. This module can be configured as a low-pass, high-pass filter or band-pass filter. Uses first order noise shaping to reduce perceptable noise and eliminate DC offset problems.

Godlike Productions Comments

While Biquads can be used to emulate more complex filters they suffer from instability at low cut-off frequencies as co-efficients are highly sensitive to quantization errors. It gets worse with more sections. Normally this is handled by limiting the lower limit of the cutoff frequency.

Specifiers

Header	Description
n_sections	1 to 13 (Vsig docs say 20, but 13 is the max). Maximum number of 2nd order biquad sections that the filter can use. The more sections that are specified in this section, the more DSP horsepower that is used up, so set it to the maximum number of sections that you believe you might use.

Audio Inputs

Header	Description
in	The filter input.

Audio Outputs

Header	Description
out	The filter output.

Control Inputs

Header	Description
qfactor	0.5 to 1000. Controls the shape of the filter frequency response. The band-width is equal to the freq setting divided by q. This means that higher q settings result in a narrower band filter. For lowpass and highpass filters, the q setting affects the steepness of the eq at the spcified frequency. Settings above 1.0 will also begin to have a resonant peak at the selected frequency. For high q settings, these eq types will sound very “bandpassish”.
type	This controls the type of filtering this module will perform. The values are as follows: 0 - Lowpass (Passes frequencies below freq setting), 1 - Bandpass (Passes frequencies in a band centered at freq setting), 2 - Highpass (Passes frequencies above freq setting).

Header	Description
sections	1 to max_#_sections. The number of 2nd-order biquad sections that are in use by the filter. The top limit is set by max_#_sections.
processoff	0 or 1, turns off processing

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
modfreq				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
n_sections	1	13	2	1
INT	max_sections			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
qfactor	0.5	1000	0.7	0.01
CONTROLINPUT	q			
type	0	2		
CONTROLINPUT	type			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
sections	1	@n_sections	2	1
CONTROLINPUT	sections			
processoff	0	1	0	1
CONTROLINPUT	turn_off			

onepole(opl) - One Pole Filter

Description

Provides a simple, first order, adjustable one pole filter, with a mod input for smooth changes of the cut-off frequency and selectable filter type.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to the onepole module.
mod	Modulation input for cutoff frequency.

Audio Outputs

Header	Description
out	The filtered output.

Control Inputs

Header	Description
freq	0 to 20000 Hertz. Controls the cutoff frequency of the highcut. The cutoff is defined as the point at which the frequency response drops 3 dB.
freqmodamt	0 to 20000 Hertz. Adjusts how much the mod input will modulate the filter center frequency.
type	Lowpass, highpass, or allpass

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
mod				
MOD				
AUDIOINPUT				
out				
AUDIOOUTPUT				
type	0	2		
CONTROLINPUT				
frequency	0	20000		
CONTROLINPUT				
frequencymodamt	0	20000		
CONTROLINPUT				

phaseshift(pha) - Phase Shifter

Description

This module is used to build the classic phase-shift effect. This effect is created by connecting several allpass filter stages in series, and sweeping the frequency parameter. In order to hear the phase shift effect (moving notches), the output of the phaseshift module must be summed with the input.

Godlike Productions Comments

Specifiers

Header	Description
poles	2 to 50 poles. Controls how many poles the phase shifter will have. More poles will cause greater phase shift effect.

Audio Inputs

Header	Description
in	The input to the phase shifter.
mod	The modulation input that controls

Audio Outputs

Header	Description
out	The output of the phase shifter.

Control Inputs

Header	Description
depth	Controls how much the mod input will affect the pole frequency.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
poles	2	50	2	1
	number_poles			
in				
MONO				
AUDIOINPUT				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
depth	0	1	1	0.01
	%%			
CONTROLINPUT	%n			

phaseshift2(pha) - Phase Shifter

Description

This module is an update to the original phaseshift module. Like the original, it is a series connection of allpass filters, the number of which is given by the specifier poles. However, unlike the original, which set the g coefficient directly via the control input, this module lets you specify the coefficient using the pole frequency (the point in an allpass with the highest group delay). The mod input affects this parameter, which is then used to calculate the coefficient. The old module also had some bad terminology which confused some operational aspects of allpass filters.

Godlike Productions Comments

Specifiers

Header	Description
poles	1 to 50 poles. Controls how many poles the phase shifter will have. More poles will cause greater phase shift effect.

Audio Inputs

Header	Description
in	The input to the phase shifter.
mod	The modulation input that moves the

Audio Outputs

Header	Description
out	The output of the phase shifter.

Control Inputs

Header	Description
polefreq	The initial frequency of the pole of the allpass. Specified in Hz.
depth	Controls the frequency range over which the pole will move. The modulated pole frequency is the sum of the initial pole frequency, and the depth multiplied by the mod input. Specified in Hz.

Mod Inputs

Header	Description

Mod Outputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
poles	1	50	1	1
	number_poles			
in				
MONO				
AUDIOINPUT				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
polefreq	1	100000	1000	0.01
CONTROLINPUT	polefreq			
depth	-100000	100000	1000	0.01
CONTROLINPUT	depth			

pre_emphasis(pre) - Pre-Emphasis Filter

Description

Provides, to within 0.5 dB, the standard 50 and 15 microsecond pre-emphasis. On older Eventide products De-emphasis was performed by the DAC 8x oversampling filter. Pre-emphasis was also provided by the analog input section. This module serves to provide pre-emphasis for internally digitally generated signals, or to do pre-emphasis / de-emphasis for bit reduction effects.

Godlike Productions Comments

Specifiers

Header	Description
filter_on	0 or 1, Turns pre-emphasis on if desired. Defaults to off so legacy algorithms (sigfiles) run without any emphasis in modern systems.

Audio Inputs

Header	Description
in	The input to the pre-emphasis module.

Audio Outputs

Header	Description
out	The pre-emphasized output.

Control Inputs

Header	Description

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
filter_on	0	1	0	1
	filter_on			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

quadrature2(qad) - Quadrature Transformer

Description

This module phase shifts the input by -90 degrees via a hilbert transform type filter. Since this type of filter behaves as a band-pass the output signal at low frequencies may be attenuated significantly as it will be with very high frequencies too.

Godlike Productions Comments

Specifiers

Header	Description
nquality	1 to 4. This specifies the quality of the filter used to achieve the phase shift. A value of 1 gives the best quality but is very resource intensive.

Audio Inputs

Header	Description
in	The input to be phase shifted.

Audio Outputs

Header	Description
norm	A delayed version of the input.
quad	A 90 degree phase-shifted versiono f the input. This phase shift is relative to the norm output.

Control Inputs

Header	Description
subsample	The subsample control allows the input to the module to be downsampled without filtering. This can be used when (better) results are desired in a narrow low-frequency band. 0 - No down sampling performed. 1 - Down sample 2:1. 2 - Down sample 4:1. 3 - Down sample 8:1. The setting of the nquality specifier effects these options.

Mod Inputs

Header	Description

Mod Outputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nquality	1	4	1	1
	nquality			
in				
AUDIOINPUT				
norm				
AUDIOOUTPUT				
quad				
AUDIOOUTPUT				
subsample	0	3	2	1
CONTROLINPUT	subsample			

simpletone(smt) - Simple Tone

Description

This is a simple two pole tone module which means that while it has independent low and high frequencies the region in-between these frequencies is a straight line between the two levels. This module should be clickless and, if used for cutting only, should produce a gain curve which is strictly less than 0. This last feature makes it useful for placing in reverb structures.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio input.

Audio Outputs

Header	Description
out	The filter output.

Control Inputs

Header	Description
low_freq	20 to 1000 Hertz. Controls the frequency at which the low frequency attenuation works.
high_freq	20 to 1000 Hertz. Controls the frequency at which the high frequency attenuation works.
low_level	-20 to 0db. Controls how much frequencies below low atten will be attenuated. This is used to quiet an overly rumbly reverb.
high_level	-20 to 0db. Controls how much frequencies above high atten will be attenuated. This is used to diminish the high sizzle of the reverb and to produce a warmer, more natural sound.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
low_freq	20	1000	100	0.1
	Hz			
CONTROLINPUT	low_freq_			
high_freq	1000	20000	2000	0.1
	Hz			
CONTROLINPUT	high_freq			
low_atten	-20	0	-1	1
	dB			
CONTROLINPUT	low_decay_			
high_atten	-20	0	-2	1
	dB			
CONTROLINPUT	high_decay			
obj				
USEROBJECTPARENT	obj			

slew (slw) - slew limit

Description

This module will limit the slew rate of a signal. The slew rate is how fast the signal changes.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	signal to be processed

Audio Outputs

Header	Description
out	Slew limited output.

Control Inputs

Header	Description
pslew	Positive slew limit. This is used when input is higher than output. How fast in milliseconds does it take to go from 0 to full scale. 0 disables function.
nslew	Negative slew limit. This is used when input is lower than output. How fast in milliseconds does it take to go from 0 to full scale.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

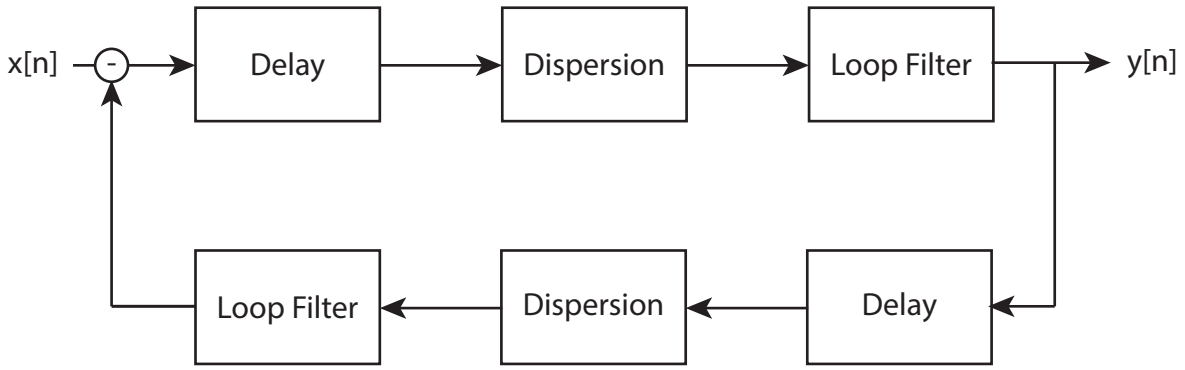
Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
pslew	0	30000	1	0.01
	ms			
CONTROLINPUT	positive			
nslew	0	30000	1	0.01
	ms			
CONTROLINPUT	negative			
state_clear	0	1	0	1
INT	state_clear			

spring_dispersion - Spring Reverb Dispersion Filter

Description

Creates audio dispersion i.e. frequency dependent group delay. This was created for use in a spring reverb effect. This dispersion curves was modeled from an actual Accutronics Type 8 and Type 4 spring reverb tank springs. The particular group delay curve is set by the coefficients of a series of allpass filters. Notes on use case: Spring reverbs have two dominant modes of wave propagation that exist in two spectral bands. The lower band delays the higher freqs more than the lower ones, and the higher band does the opposite. This dispersion curve currently only implements the lower band of the Type 8 spring with dispersion up to about 4kHz OR the Type 4 spring with dispersion up to about 6 kHz. A decent spring sound (lower band) can be simulated by placing two dispersion modules, two delays, and two loop filters (usually lowpass or bandpass) in a waveguide-like structure, shown below:



Future expansion should include curves for the Type 8 higher band as well as other types of Spring reverbs e.g. an Accutronics Type 4, etc.

Godlike Productions Comments

Specifiers

Header	Description
mode	0 or 1, Spring dispersion curve mode. 0: Less life-like heuristally pruned "cycle lite" version 1: more like-like and accuracte spring dispersion curve, but also more expensive (about twice the 0 mode option).

Audio Inputs

Header	Description
in	audio input

Audio Outputs

Header	Description
out	audio output

Control Inputs

Header	Description
type	0 or 1, Spring Type. 0: Type 8 spring, dispersion up to 4 kHz, 1: Type 4 spring, dispersion up to 6 kHz.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mode	0	1	0	1
	mode			
type	0	1	1	1
			Type8,Type4	
CONTROLINPUT	type			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

tone(ton) - Audio Tone Control

Description

The tone control module provides a simple tone control equalizer. It has a gentle low and high shelving control with adjutable frequencies.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The tone control input.

Audio Outputs

Header	Description
out	The tone control output.

Control Inputs

Header	Description
low_freq	0 to 20000 Hertz. Adjusts the frequency at which the low shelving filter begins affecting the audio.
low_level	-20 dB to +20 dB. Control how much the low frequencies are boosted or cut. This is like a “bass” control.
high_freq	0 to 20000 Hertz. Adjusts the frequency at which the high shelving filter begins affecting the audio.
high_level	-20 dB to +20 dB. Control how much the high frequencies are bossted or cut. This is like a “treble” control.

Mod Inputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
declick	0	1	0	1
	declick			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
low_freq	0	20000	100	0.1
	hz			
CONTROLINPUT	low_freq			
low_level	-18	18	0	0.1
	dB			
CONTROLINPUT	low_level			
high_freq	20	20000	2000	0.1
	hz			
CONTROLINPUT	hi_freq			
high_level	-18	18	0	0.1
	dB			
CONTROLINPUT	hi_level			

Chapter 9 - Interface

extcontrol

Description

Brings an external control source into the system. This codegen has two outputs. One is a control signal that varies from 0 to 1. The other is a userobject. You include the userobject in the head or a collection. This will bring you a set of userobjects that allow you to program what external source produces the output control signal. When using the editor, you will be asked for 4 numbers called spec1, spec2, spec3, spec4. You should put 0 for all of them since you can go to the parameter menu to set what the source is. When writing a sigfile, you will need to supply 4 ints. The following table shows what you can set the first number to. the remaining 3 should be 0. 0 off 256 high 512 mid 768 low 1024 mod1 1280 mod2 1536 mod3 1792 mod4 2048 pedal 2304 tip 2560 ring 2816 user 1 3072 user 2

Godlike Productions Comments

Specifiers

Header	Description
spec1	The integer to select the external control source.
spec2	Set to 0
spec3	Set to 0
spec4	Set to 0

Control Inputs

Header	Description

Control Outputs

Header	Description
out	Controller output

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
description				
	description			
spec1	-32768	32767	0	1
INT	spec1			
spec2	-32768	32767	0	1
INT	spec2			
spec3	-32768	32767	0	1
INT	spec3			
spec4	-32768	32767	0	1
INT	spec4			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

footswitch (ftsw) - Footswitch Module

Description
Represents a hardware footswitch.

Godlike Productions Comments

Specifiers

Header	Description
name	The name of the footswitch, which will appear on the footswitch mapping selection page.
type	The action type of the footswitch. 0: momentary switch 1: latching switch
latch_off_text	For latching footswitches, the text to display when the parameter is off.
latch_on_text	For latching footswitches, the text to display when the parameter is on

Control Inputs

Header	Description

Control Outputs

Header	Description
out	The state of the footswitch, 0 or 1. 1 means the parameter is engaged.

User Objects

Header	Description
obj	The userobject for the footswitch. Must be connceted to a menupage or directly to the head.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
name				
STRING				
type	0	1	0	1
INT	footswitch type			
latch_off_text			off	

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
STRING	latch off text			
latch_on_text			on	
STRING	latch on text			
obj				
USEROBJECT				

gang (gng) - Gang of User Interface Objects

Description

Brings a group of objects together into a gang. This works much like a MENUPAGE module where you insert knobs into a menu. You take knobs and insert them into the gang, and then you insert the gang into a menu as one object. It will appear as though your menu has those knobs. But as you move around the knobs, there will be a point where all knobs in the gang are selected at once. If you turn the wheel, all the knobs will change. You then can move to an individual knob and adjust it seperately. In addition to knobs, you can use the different faders in a gang and you can mix different types. You can also insert monitors and textlines but they can't be adjusted. Some limitations: All of the objects in a gang must fit on one page. What is left over can't be accessed. You can't put menupages or other gangs into a gang. They will be ignored You can't put your gang on the head collection.

Godlike Productions Comments

Specifiers

Header	Description
description	19 characters. This is the text that will show in the upper right of the PARAMETER screen when this menu page is selected.
name	8 characters or less of text which is used for the soft key in first level menupages. If this is a menupage that is inserted into an existing menupage this text will not be used. Creating this text is recommended because this name might be used in later gerneration Ultra-Harmonizers

Control Inputs

Header	Description

Control Outputs

Header	Description

User Objects

Header	Description
obj	user object

Module Entries

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
description				
name				
STRING				
nsubuos				
INT				
obj				
		@nsubuos		
USEROBJECTCHILD				
obj				
USEROBJECTPARENT				

hfader(fdr) - Horizontal Fader Knob fdr

Description

This module creates a horizontal fader in a PARAMETER menu. Selecting the fader in the appropriate PARAMETER menu allows the control output of the fader knob to be varied. The display for an hfader takes one half of a PARAMETER menu line (the same size as a knob module).

Godlike Productions Comments

The documentation available in VSIG 3.4.2 has been updated, but still wrong. I've updated as best I can. The documentation for vfader appears to be correct, so refer to that.

Specifiers

Header	Description
statement	description for future use. The documentation shows a menutext specifier, however this is not available on the block.
name	Text desplayed when hfader appears on H9000's LCD display in a PARAMETER menu page. Size: 6 characters.
min	Value for the bottom of the fader. Range -32768 to 32768.
max	Value for the top of the fader. Range -32768 to 32768
resolution	Step rate. Range 0 to 32767
default	Value that the hfader will be set to when it is first used. Range: from min to max.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	hfader output

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	menu statement			
name				
STRING	name			
min	-32768	32767	-32768	
FLOAT	min value			
max	-32768	32767	32767	
FLOAT	max value			
res	0	32767	1	
FLOAT	resolution			
default	@min	@max	0	
FLOAT	default			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

hmenupage (hmn) - Versatile Menupage

Description

On the H9000, this module behaves identically to a menupage, and its two unique control inputs are not recognized. It is included with the H9000 library for legacy support. On older units, this module would allow for dynamic menupage names and visibility.

Godlike Productions Comments

Specifiers

Header	Description
description	The menupage name.
tag	Tag seems to be unused. 8 char name.
nsubuos	The number of interface objects that can be connected.

Control Inputs

Header	Description
textnum	Not supported in H9000
explevel	Not supported in H9000

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
obj				
nstrings				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
INT				
textnum	0	@nstrings/1/#sub	1	1
CONTROLINPUT	key_name			
explevel	-20	20	0	1
CONTROLINPUT	explevel			
description				
STRING	description			
tag				
STRING	8 char name			
nsubuos				
INT	# entries			
obj				
		@nsubuos		
USEROBJECTCHILD				
text				
		@nstrings		
STRING	text_~n			

hmonitor(vmn) - horizontal monitor

Description
A control signal monitor that is shown much like a bargraph.

Godlike Productions Comments
Ensure that audio signals are not wired directly into hmonitor or vmonitor (via atoc or atoc(filt) or emote will crash. Use the default peak detect block before the monitor to keep it safe for emote. If this block is fed by a module using log (for example level), ensure that this never receives 0, or it will crash emote. Place a c_bound before the log block with a min value of 0.0000001 (or similar).

While some fixes were made in 3.4.2, there are still bugs with hmon, in that %f will not display a value. A workaround is to use a monitor object wired in parallel to show a value. In 3.4.2, there is no way to get a hmenu to display a value or a scale.

Specifiers

Header	Description
minimum	Minimum expected value
maximum	Maximum expected value
name	The name of this userobject (currently not used)
tag	A short name for this object. A %f will include a number (the value at in)

Control Inputs

Header	Description
in	The signal to be monitored

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	%n			
minimum			0	0.001
FLOAT	minimum			
maximum			1	0.001
FLOAT	maximum			
name			hmon:%f	
STRING	name			
tag			hmon	
STRING	tag			
obj				
USEROBJECTPARENT	obj			

knob(knb) - Manual Adjust Of A Control Signal

Description

This module is associated with a line of text that appears on a PARAMETER menu. Selecting that line of text in the appropriate PARAMETER menu allows the control output of the knob module to be varied. The knob has one output and no inputs (aside from the menus in which the knob appears).

Godlike Productions Comments

Specifiers

Header	Description
statement	Text description for PARAMETER menu. Use %f to place the value into the description.
name	A short name for the knob which will be used if the knob is connected directly to the HEAD.
min	minimum value. Range: -32768 to max.
max	maximum value. Range: min to 32767.0.
resolution	step rate. Range: 0 to 32767.0.
default	value which the knob will be set to when first used. Range: min to max.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	knob output

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				
	menu statement			

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
name				
STRING	name			
min	-32768	32767	-32768	
FLOAT	min value			
max	-32768	32767	32767	
FLOAT	max value			
res	0	32767	1	
FLOAT	resolution			
default	@min	@max	0	
FLOAT	default			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

menupage (mnu) - Collection of User Interface Objects

Description

This module is used to give the user control over the appearance of the user interface of a program's parameters. A Collection is a way of grouping user interface objects (knobs, faders, etc.) into a meaningful interface. Knobs that are "plugged-in" to a collection will show up on the same page of parameters, (if they don't fit, they will appear as multiply pages under a single softkey. In order for the objects in a collection to show up on the display, the collection itself must be plugged into another collection. The arrangement of all of the collections is hierarchical. The top of the tree is a collection contained in the "head" operator. When the interface is being built up, all collections "under" the head collection are assigned to softkeys. Any collections under these collections become "pages" under the softkeys.

Godlike Productions Comments

Specifiers

Header	Description
name	The menupage name.
tag	Tag seems to be unused. 8 char name.
nsubuos	The number of interface objects that can be connected.

Control Inputs

Header	Description

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
obj				
name				
STRING				
tag				
STRING				
nsubuos				
INT				
obj				
		@nsubuos		
USEROBJECTCHILD				

METER(mtr) - a analog style meter

Description
A control signal monitor that is shown much like a analog style meter.

Godlike Productions Comments

Specifiers

Header	Description
name	The name of this userobject (currently not used)
tag	A short name for this object. A %f will include a number (the value at in)
minimum	Minimum expected value
maximum	Maximum expected value

Control Inputs

Header	Description
in	The signal to be monitored

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
	%n			
minimum			0	0.001
FLOAT	minimum			
maximum			1	0.001

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
FLOAT	maximum			
name			meter:%f	
STRING	name			
tag			meter	
STRING	tag			
obj				
USEROBJECTPARENT	obj			

mknob(mkb) - Multiple numeric knob

Description

MKNOB allows the designer to create a number of numeric knobs with the same minimum, maximum values and name, along with a containing menupage, all this being contained in a single module. Each of the knobs is similar to an SKNOB3. When a number of similar knobs are required, the use of an MKNOB saves time and reduces preset size and loading time. If a text knob is required, look at MTEXTKNOB. Simple scaling arithmetic can be performed on the output of each knob, allowing these knobs to act as PERCENTKNOBs, or to give an "off by one" value, etc. A lockout input is provided to prevent input changes being processed, for example during preset startup. Each knob also has an associated control input, which can override the user object value. The minimum and maximum values are also control inputs, so the algorithm can modify the constraints according to the results of other calculations. The module exposes individual user objects for the knobs plus a menu user object that aggregates all of them in one page.

Godlike Productions Comments

Appearance settings do not work as of 24 Aug 2022. As per https://www.eventideaudio.com/forums/topic/h9000vsig3-bugs/

Specifiers

Header	Description
description	This is the text that will appear on the screen. If "~n" is contained within the text it will be replaced by the knob number, e.g. knob~n gives knob1, knob2 etc. %f type formatting statements may be used to display the raw values of the knobs before offset and multiplier are applied.
noutputs	The number of knobs required. Minimum is 1, maximum is 100.
appearance	determines the style of the knobs: 0 - numeric, 1 - horizontal fader, 2 - vertical fader, 3 - round. Note that the use of types 1 - 3 requires that minimum and maximum be set to meaningful values, as these determine the start and end of the display range.
resolution	The amount by which the value changes for each "click" of the knob.
defalt1,defalt2,etc	These are the values that the knobs will display when the preset is loaded.

Control Inputs

Header	Description
maximum	The maximum value for the knob. Range minimum to 32767. This applies to the on-screen value, and the output value before applying scaling.
minimum	The minimum value for the knob. Range -32767 to maximum. This applies to the on-screen value, and the output value before applying scaling.
offset,multiplier	These inputs allow simple arithmetic to be performed on the knob value before it is sent to its output. The final output value is given by (<screen value> times multiplier) plus offset. Range -32768 to 32767. The default values are multiplier=1, offset=0, meaning that these values have no effect. Set multiplier to 0.01 to act like a PERCENTKNOB.
lockout	If this value is greater than 0.1, changes on the knob inputs will be ignored.
in1,in2,etc	Changes on these inputs will update the displayed value and the outputs.

Control Outputs

Header	Description
out1,out2,etc	These are the outputs of the knobs. Their outputs will range from minimum to maximum, before allowing for offset and multiplier.

User Objects

Header	Description
obj	The userobject for each knob. Connect it to a menupage or directly to the HEAD.
menu	This is a menupage containing the knobs. Menu can be connected to a menupage, and will put all knobs into the menu.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
description				
	description			
noutputs	1	64	1	1
INT	num outputs			
appearance	0	3	0	
INT	appearance			
min	-32767	@max	-32767	1
CONTROLINPUT	min value			
max	@min	32767	32767	1
CONTROLINPUT	max value			
offset			0	1
CONTROLINPUT	offset amount			
multiplier			1	1
CONTROLINPUT	multiplier			
lockout	0	1	0	1
CONTROLINPUT	lockout			
resolution			1	
FLOAT	resolution			

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
in	@min	@max	0	1
		@noutputs		
CONTROLINPUT	in_~n___			
defalt				
FLOAT	default_~n			
out				
		@noutputs		
CONTROLOUTPUT	@description			
obj				
		@noutputs		
USEROBJECTPARENT	obj~n			
menu				
USEROBJECTPARENT	menu			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

MOMENTARY (MOM) - momentary button

Description
this control will output 1 for as long as it remains pressed

Godlike Productions Comments

Specifiers

Header	Description
name	The name of this userobject (currently not used)
tag	A short name for this object.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	trigger output

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
name			momentary	
	name			
statement			momentary	
STRING	statement			
tag			momentary	

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
STRING	tag			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

monitor(mon) - Monitor A Control Signal

Description

This module creates a one line display on a selected parameter menu to allow a numerical output which will describe the value of its control input. When the module is installed in a program using the patch editor, the editor will prompt for the parameter menupage to be used and for the text used in the menu.

Godlike Productions Comments

Specifiers

Header	Description
statement	Statement, including %f format, which describes how the monitor signal will be displayed. Size: 19 characters.
tag	8 characters or less of text which describes the monitored signal. This text is not displayed by the H9000 at any time but may be used in future products.

Control Inputs

Header	Description
in	Signal to be displayed. Range: -32768.0 to 32767.0.

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	%n			
statement			mon:%f	
STRING	statement			
tag			mon	
STRING	tag			
obj				
USEROBJECTPARENT	obj			

mtextknob - Multiple Text Knob

Description

mtextknob provides a list of related textknobs with common metadata. They are presented as user objects -obj1..-objN, and also as a menu user object -menu, that can be added to the root user object.

Godlike Productions Comments

Specifiers

Header	Description
description	This is the text that will appear on the screen. Textknobs use the format "NAME: %s", where the text before the colon will appear below the textknob, and the text after the colon will appear inside the textknob. "~n" format is supported, and will be replaced by the number of each knob. %s type formatting statements will display the raw values of the knobs before the offset is applied.
noutputs	The number of knobs required. Minimum is 1, maximum is 100.
defalt1,defalt2,etc	These are the values that the knobs will display when the preset is loaded.

Control Inputs

Header	Description
offset	These inputs allow simple arithmetic to be performed on the knob value before it is sent to its output. The final output value is given by <screen value> plus offset. Range -32768 to 32767. The default value is offset=0, meaning that these values have no effect.
lockout	If this value is greater than 0.1, changes on the knob inputs will be ignored.
in1,in2,etc	Changes on these inputs will update the displayed value and the outputs.

Control Outputs

Header	Description
out1,out2,etc	These are the outputs of the knobs. Their outputs will range from minimum to maximum, before allowing for offset and multiplier.

User Objects

Header	Description
menu	These are the userobject outputs for the knobs. Since the menu output also contains the knobs, these individual outputs are not normally used.
obj1,obj2,etc	

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
description				
	description			
nstrings	1	100	1	1
INT	num strings			
noutputs	1	100	1	1
INT	num outputs			
defalt	0	@nstrings	0	1
		@noutputs		
FLOAT	default_~n			
offset			0	1
CONTROLINPUT	offset amount			
in	0	@nstrings	0	1
		@noutputs		
CONTROLINPUT	in_~n___			
lockout	0	1	0	1
CONTROLINPUT	lockout			
out				
		@noutputs		
CONTROLOUTPUT	out_~n___			
text				
	text	@nstrings		
STRING	line_~n_text			
obj				
		@noutputs		
USEROBJECTPARENT	obj~n			
menu				
USEROBJECTPARENT	menu			

multiknob(mkb) - multi-value knob

Description

This module is a knob storing a variable number of preset values, the active one being chosen by a control input. It may be useful as: (1) a user changeable "lookup table" (2) to allow a preset to offer a number of built-in "tweaks", typically driven by a textknob, giving the name of the tweak. The appearance of the knob is determined by a specifier.

Godlike Productions Comments

There is a major bug with multiknob. If you include it on a vsig graph, it will crash a processor on the H9000 and stop Emote connecting with the H9000, requiring the H9000 to be power cycled.A program called ERROR will be created in a random empty user algorithm ID, which cannot be deleted from either Emote or the H9000. To resolve this, you must load an algorithm from VSIG over the ID number (this will create 2 programs at the same ID). You can then delete one of these new programs, which will delete both of these identical programs. TLDR: Don't use this block until Eventide fix this bug.

Specifiers

Header	Description
type	How knob is displayed. 0: normal, 1: vfader, 2: hfader, 3: round.
num_tweaks	number of stored values

Control Inputs

Header	Description
tweaknum	active stored value number

Control Outputs

Header	Description
out	the knob value

User Objects

Header	Description
obj	allows the user to vary remote_mode

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				
	menu statement			
tag				
STRING	tag			
minimum	-32768	32767	-32768	
FLOAT	min value			
maximum	-32768	32767	32767	
FLOAT	max value			
resolution	0	32767	1	
FLOAT	resolution			
appearance	0	3	0	
INT	appearance			
num_tweaks	0	50	0	
INT	number of tweaks			
tweaknum				
CONTROLINPUT	tweaknum			
val	@minimum	@maximum	0	@resolution
		@num_tweaks		
FLOAT	val_~n			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

percentknob(pkb) - Percent Knob

Description

This is a modification of the standard knob. In this knob the output is divided by 100. So, if you insert this knob and specify a min of 0 and a max of 100, then the display will go from 0 to 100 but the output will go from 0 to 1. If you specify a resolution of .1 then the output resolution will be .001 although the display will show steps of .1.

Godlike Productions Comments

Specifiers

Header	Description
statement	Text description for PARAMETER menu, use %?.?f format. Note that in order to make your text actually print a % on the screen you will have to use %% in the text. So, to make a display that would show Volume 45.6% you will need Volume %3.1f%%.
name	8 character description, for future use
min	minimum value. Range: -32768 to 0.0
max	maximum value. Range: min to 32767.0
resolution	step rate. Range: 0 to 32767.0
default	Value which the knob will be set to when first used. Range: min to max.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	knob output = display/100

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	menu statement			
name				
STRING	name			
min	-32768	32767	-32768	
FLOAT	min value			
max	-32768	32767	32767	
FLOAT	max value			
res	0	32767	1	
FLOAT	resolution			
default	@min	@max	0	
FLOAT	default			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

presetchange(prc) - Preset Change

Description
For the H9. Detects change of preset.

Godlike Productions Comments
There is nothing in the .json file for this block. I suspect this is for the H9, and a signal is sent when the preset changes

Specifiers

Header	Description

Control Inputs

Header	Description

Control Outputs

Header	Description
out	There is no entry in the documentation. Suspect this sends out a 1 when a preset changes (or perhaps sends out the preset number)

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
	out			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

rfader(fdr) - Round Knob

Description

Rotating the knob causes the line to rotate. Up to four rfaders may be pictured on a single screen. Two rfaders may share a display page with four knobs.

Godlike Productions Comments

Specifiers

Header	Description
statement	Description for future use.
name	6 character description for PARAMETER menu.
min	Minimum value. Range: -32768.0 to 32768.0.
max	Maximum value. Range: min to 32767.0.
resolution	Step rate. Range: 0 to 32767.0.
default	value which the knob will be set to when first used. Range: min to max.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	knob output

User Objects

Header	Description
obj	The userobject for the fader.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				
	menu statement			
name				
STRING	name			
min	-32768	32767	-32768	

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
FLOAT	min value			
max	-32768	32767	32767	
FLOAT	max value			
res	0	32767	1	
FLOAT	resolution			
default	@min	@max	0	
FLOAT	default			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

skmonitor(tmn) - Combination knob and monitor

Description

A control signal monitor that shows words for values. Each integer value from zero can be linked to a word that is displayed.

Godlike Productions Comments

Specifiers

Header	Description
nvalues	how many values are represented
name	the name of this userobject (use %s to include text)
tag	A short name for this object.
text[nvalues]	The names of each value starting with 0

Control Inputs

Header	Description
in	The signal to be monitored

Control Outputs

Header	Description
out	knob output

User Objects

Header	Description
obj	The userobject for the monitor.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
tag			skmonitor	
	tag			
in				
CONTROLINPUT				
nvalues	0	1000	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
INT	num_values			
textnum	0	@nvalues		
CONTROLINPUT				
defalt	@minimum	@maximum		
INT				
minimum				
CONTROLINPUT				
maximum				
CONTROLINPUT				
resolution				
CONTROLINPUT				
mode	0	7	0	1
INT	%n			
merge_in	@minimum	@maximum	0	1
CONTROLINPUT	%n			
text			text	
STRING	text~n			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

sknob3(sk3)

Description

sknob3 is, to a certain extent, a pairing of a knob and a monitor. a common use case of sknob3 is to connect the output of the knob to ctrlin another common use is to use the output of the knob as an input to a math expression and the output of that expression as ctrlin. sknob and sknob2 are legacy objects that have been subsumed under sknob3 min, max, merge can be specifiers or control ins

Godlike Productions Comments

Merge input has an initialization bug that Eventide are looking into per https://www.eventideaudio.com/forums/topic/h9000vsig3-bugs/#post-157408

Specifiers

Header	Description
statement	The text that will display below the knob. Like many other interface modules, you may use %f notation to display the number coming in from ctrlin (eg. %1.2f)
name	The name of the knob's H9000 parameter page if the userobject output is connected directly to the head.
knob_type	Determines the appearance of the knob. 0: numeric, 1: hfader, 2: vfader, 3: round.
resolution	the step size of the knob
default	The initial value for the knob

Control Inputs

Header	Description
ctrlin	the input to the monitor. This value is displayed in the ui.
min	The minimum value to be output
max	The maximum value to be output
merge	can be used to control the value of both the knob and the monitor

Control Outputs

Header	Description
out	the output of the knob

User Objects

Header	Description
obj	A display object for the knob

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	menu statement			
name				
STRING	name			
type	0	3	0	1
FLOAT	fader type			
res	0	32767	1	
FLOAT	resolution			
default	-32768	32767	0	
FLOAT	default			
ctrlin				
CONTROLINPUT	control in			
min				
CONTROLINPUT	minimum in			
max				
CONTROLINPUT	maximum in			
merge				
CONTROLINPUT	merge in			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

smonitor(smn) - Combined text and numeric monitor

Description

Formats a control input according to a list of format strings. The format string is selected with the textnum input. This module operates like MONITOR and may be used to display a numeric value. In addition, the statement string that determines the format of the result may be selected by means of a control input, allowing the same value to be displayed in a number of forms. It may also be used as a TMONITOR by ignoring the numeric input and selecting the desired text string. Or, as both..

Godlike Productions Comments

Specifiers

Header	Description
shortname	8 character description, for future use.
nvalues	number of selectable values. Range: 0 to 100.
text1,text2,...textN	Text description for PARAMETER menu, for value #N. Use %f format to display in value, or plain text.

Control Inputs

Header	Description
in	input to display as numeric value
textnum	input to select text string. Range: 0 to nvalues-1. If the control signal is .5 or less, it shows the first text value. If it is between .5 and 1.5 it will show the next text value, etc.

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this monitor.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
name			smon	

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	name			
tag			smonitor	
STRING	tag			
in				
CONTROLINPUT				
nvalues	0	240	0	1
INT	num_values			
textnum	0	@nvalues/1/#sub		
CONTROLINPUT				
text			text	
		@nvalues		
STRING	text~n			
obj				
USEROBJECTPARENT	obj			

stextknob(stx) - Versatile Textknob

Description

An stextknob is a combination of a textknob and a tmonitor, in that it produces a user-controllable output and displays the value of an input. If the input is connected to the output, it will act much like a normal textknob. However, its reason to exist is the fact that the output can be bounded or otherwise processed before being fed to the input. In these cases, the relationship between the input, output and displayed values can be complex. stextknob is a text knob (i.e. allows the user to select from an enumeration of choices) that also has a control input that can override the selected value. The displayed value uses the statement as a printf-style format string with the selected choice string.

Godlike Productions Comments

Specifiers

Header	Description
description	Description for PARAMETER menu, use %s format (26 chars)
tag	Description, for future use (8 chars)
nvalues	Number of selectable values (1->100)
default	Value knob will be set to when first loaded (0->nvalues-1)
value1..n	Text to be displayed in place of %s, for value 1..N (40 chars)

Control Inputs

Header	Description
Textnum	This input determines the displayed text (0->nvalues-1)

Control Outputs

Header	Description
textnum_out	Knob output

User Objects

Header	Description
obj	The userobject for this stextknob

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
description			stknob:%s	

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	description			
tag			stknob	
STRING	tag			
nvalues	1	100	1	1
INT	# of strings			
textnum_out			0	
CONTROLOUTPUT	textnum out			
textnum	0	@nvalues/1/#sub	0	1
CONTROLINPUT	text num			
default			0	
INT	default choice			
text				
		@nvalues		
STRING	choices			
obj				
USEROBJECTPARENT	obj			

taperknob(knb) - Tapered Knob

Description

taperknob gives you a knob with a number of different tapers. You supply a minimimum and maximum and the number of steps in between. The steps are not uniform and the non-uniformity is controlled by taper. A 0 taper is linear. Taperknob allows you to select the resolution of the knob by setting the number of steps. The actual value of the knob at each step is automatically generated by the tapknob module. You only need to select the taper, min->max and number of steps.

Godlike Productions Comments

This block replaces the deprecated tapknob

Specifiers

Header	Description
statement	How the value of the knob will be displayed. Use %.?f format for the number where ? is the number of displaed digits.
name	The name of the knob.
min	The value you will get if the knob is rotated fully counter-clockwise. This value can actually be higher than the maximum value.
max	Value that the knob generates if the knob is rotated fully clockwise. See min.
steps	This is the number of different values that the knob can have along its range before the taper is applied
res	The resolution of the real/post-taper value of this knob. The output will be quantized to this resolution.
taper	What kind of taper to use on the knob. 0 - linear taper, 1 - square taper, 2 - cube taper, 3 - inverse square taper, 4 - inverse cube taper, 5 - S taper, 6 - concave convex 7 - quarter sine 8 - quarter cosine
default	value which the knob will be set to when first used. Range: from min to max.
alpha	Only applies to the concave convex taper. Values form 0 to 1 yield a decreasingly convex taper, and values greater than 1 yield an increasingly concave taper.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	The tapered output value.

User Objects

Header	Description
obj	The userobject for this knob.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				
	menu statement			
name				
STRING	name			
min	-32768	32767	-32768	1
INT	min value			
max	-32768	32767	32767	1
INT	max value			
steps	2	32767	2	1
INT	steps			
res	0	32767	1	1
INT	resolution			
default	@min	@max	0	1
INT	default			
taper	0	8	0	1
INT	taper			
alpha	0	32767	1	0.0001
FLOAT	alpha			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

tapknob(knb) - Tapered Knob

Description

Tapknob has been deprecated as of VSIG 3.4.2. Use Taperknob instead.

tapknob gives you a knob with a number of different tapers. You supply a minimimum and maximum and the number of steps in between. The steps are not uniform and the non-uniformity is controlled by taper. A 0 taper is linear. tapknob allows you to select the resolution of the knob by setting the number of steps. The actual value of the knob at each step is automatically generated by the tapknob module. You only need to select the taper, min->max and number of steps.

Godlike Productions Comments

Specifiers

Header	Description
statement	26 character description for PARAMETER menu. Use %.?s format for the number where ? is the number of displayed digits. This should fit into 19 characters
name	8 character description, for future use.
min	Minimum value. This is actually a misnomer for this particular knob. This value is the value you will get if the knob is rotated fully counter-clockwise. This value can actually be higher than the maximum value. Range: -32768.0 to 0.
max	Value that the knob generates if the knob is rotated fully clockwise. See min. Range: min to 32767.0.
steps	This is the number of different values that the knob can generate. This controls the rate of knob change vs knob rotation speed. Range: 1 to 32767 number of steps.
taper	What kind of taper to use on the knob. 0 - linear taper, 1 - square taper, 2 - cube taper, 3 - inverse square taper, 4 - inverse cube taper, 5 - S taper.
default	value which the knob will be set to when first used. Range: from min to max.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	knob output

User Objects

Header	Description
obj	The userobject for this knob.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				
	menu statement			
name				
STRING	name			
min	-32768	32767	-32768	1
INT	min value			
max	-32768	32767	32767	1
INT	max value			
steps	0	32767	1	1
INT	steps			
default	@min	@max	0	1
INT	default			
taper	0	6	0	1
INT	taper			
alpha	0	32767	1	0.0001
FLOAT	alpha			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

textblock(txt) - Some text to display on the screen

Description

Use this module to place a block of text on a display page. You control how high and wide. There is a number you specify that tells the system how many lines of text. If there are more than 4 lines, the system will allow you to scroll thru the text. The system will read all the lines and use width. In your patch, you place this module's userobject on to some menupage. (you cannot put it directly on the head) The maximum characters per line is 40. However, If you have more than 40 lines, You will only see 39. The system places an arrow on the left to indicate you can scroll. Each line uses 42 bytes of storage. Even if there is 1 character.

Godlike Productions Comments

Specifiers

Header	Description
nlines	how many lines of text, 1 to 100
text(n)	lines of text from 0 to 40 characters

Control Inputs

Header	Description

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this knob. Connect it to a menupage or directly to the HEAD.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nlines	1	100	1	1

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	number_lines			
text			text	
		@nlines		
STRING	line_~n_text			
obj				
USEROBJECTPARENT				

textknob(tkb) - Text Knob

Description

You can insert a knob into your program that allows a user to select from multiple text items. The knob outputs a number which indicates which text string was selected. You'll have to use multiply or c_table to make the number into the value you'll probably need. One interesting application for this module is to allow a user to select a note, as in "MODULES1C, 4 middle C" or "A, 4 A-440". On the H9000, a textknob with nchoices = 2 will appear as a binary switch.

Godlike Productions Comments

Specifiers

Header	Description
statement	This is the text that will appear on the screen. Textknobs use the format "NAME: %s", where the text before the colon will appear below the textknob, and the text after the colon will appear inside the textknob. %s type formatting statements will display the raw values of the knobs before the offset is applied.
name	8 character description, for future use.
nchoices	number of selectable values. Range: 0 to 100.
default	value knob will be set to when first used. Range: 0 to nvalues-1.
value1,value2...valueN	Text to be displayed in place of %s, for value #N.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	knob output

User Objects

Header	Description
obj	The userobject for this textknob.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	menu statement			
name				
STRING	name			
nchoices	0	100	0	1
INT	# of values			
default	0	@nchoices/1/#sub	0	1
INT	default choice			
choices				
		@nchoices		
STRING	choices			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

textline(txt) - 20 character message to display on the screen

Description

Use this module to display a message as part of a page. It is just one line of text 20 characters wide. You put this on a menupage to display.

Godlike Productions Comments

Specifiers

Header	Description
text	line of text, 20 characters.

Control Inputs

Header	Description

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for textline. Connect to a menupage to show the text on the H9000 screen or Emote.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
text			text	
	textline			
obj				
USEROBJECTPARENT	obj			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

texttrigger(TTG) - trigger button with variable name

Description
press the button will create a control signal trigger

Godlike Productions Comments
The documentation available in VSIG 3.3.3 is incorrect. I've updated as best I can.

Specifiers

Header	Description
nstrings	The number of options for the trigger. Opens up textn
textn	The text to show when selected from the input.

Control Inputs

Header	Description
textnum	An integer to select which of the textn to show on the button.

Control Outputs

Header	Description
out	trigger output

User Objects

Header	Description
obj	The userobject for texttrigger. Connect to a menupage to show the control on the H9000 screen or Emote.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nstrings	0	10	0	1
	num_values			
textnum	0	@nstrings		
CONTROLINPUT				
text			text	

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
		@nstrings		
STRING	text~n			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

tjknobs (tjk) - H9 knob matrix.

Description
Knob object for H9 Pedal. Will be deprecated. Recommend not using.

Godlike Productions Comments
This knob is for the H9 pedal and will be deprecated as per https://www.eventideaudio.com/forums/topic/h9000vsig3-bugs/#post-164090. See the module entries for information. All entries are specifiers.

Specifiers

Header	Description
Note	See module entries for list of specifiers

Control Inputs

Header	Description

Control Outputs

Header	Description
realn	Raw knob output where n = 1 to 10
pretapern	Tapered knob output where n = 1 to 10

User Objects

Header	Description
obj	The userobject for tjknobs

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
real1				
pretaper1				
CONTROLOUTPUT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
display1				
CONTROLINPUT				
textnum1	0	@nvalues1/1/#s-ub		
CONTROLINPUT				
nvalues1				
INT				
text1_				
		@nvalues1		
STRING	text~n			
label1_				
		@nvalues1		
STRING	label~n			
min1				
FLOAT				
max1				
FLOAT				
res1				
FLOAT				
default1				
FLOAT				
pretapermin1				
FLOAT				
pretapermax1				
FLOAT				
pretaperdefault1				
FLOAT				
tag1				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
STRING				
real_in1				
CONTROLINPUT				
real2				
CONTROLOUTPUT				
pretaper2				
CONTROLOUTPUT				
display2				
CONTROLINPUT				
textnum2	0	@nvalues2/1/#s-ub		
CONTROLINPUT				
nvalues2				
INT				
text2_				
STRING	text~n			
label2_				
STRING	label~n			
min2				
FLOAT				
max2				
FLOAT				
res2				
FLOAT				
default2				
FLOAT				
pretapermin2				
FLOAT				
pretapermax2				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
FLOAT				
pretaperdefault2				
FLOAT				
tag2				
STRING				
real_in2				
CONTROLINPUT				
real3				
CONTROLOUTPUT				
pretaper3				
CONTROLOUTPUT				
display3				
CONTROLINPUT				
textnum3	0	@nvalues3/1/#s-ub		
CONTROLINPUT				
nvalues3				
INT				
text3_				
		@nvalues3		
STRING				
label3_				
		@nvalues3		
STRING	label~n			
min3				
FLOAT				
max3				
FLOAT				
res3				
FLOAT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
default3				
FLOAT				
pretapermin3				
FLOAT				
pretapermax3				
FLOAT				
pretaperdefault3				
FLOAT				
tag3				
STRING				
real_in3				
CONTROLINPUT				

tmonitor(txt) - 20 character message to display on the screen

Description

Use this module to display a message as part of a page. It is just one line of text 20 characters wide. You put this on a menupage to display.

Godlike Productions Comments

Specifiers

Header	Description
name	The name of this object. %s displays the message fron textn.
tag	A short name for this object.
nvalues	The number of options of messages to show.
textn	The text to show when selected from the input.

Control Inputs

Header	Description
in	Integer input to select which of textn to show in a message. Similar to text-trigger except is not a button.

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for tmonitor. Connect to a menupage to show the control on the H9000 screen or Emote.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
name			tmonitor:%s	

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
STRING	name			
tag			tmonitor	
STRING	tag			
nvalues	0	100	0	1
INT	num_values			
text			text	
		@nvalues		
STRING	text~n			
obj				
USEROBJECTPARENT				
	Min	Max	Default	Resolution
	Units	Size	Choices	

trigger(TRG) - trigger button

Description
pressing the button will create a control signal trigger

Godlike Productions Comments

Specifiers

Header	Description
name	The name of this userobject
tag	A shortname for this object.
statement	The statement for the trigger.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	trigger output

User Objects

Header	Description
obj	The userobject of trigger. Connect to menupage to show the trigger button.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
name			trigger	
	name			
statement			trigger	
STRING	statement			
tag			trigger	

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
STRING	tag			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

uo_switch(uos) - Userobject switch

Description

This module allows multiple knobs, triggers or menupages to be connected to a single Userobject input. A control input determines which of them is displayed and hence active. It can be used to enable or disable entire menupages based on a control setting, or to allow the makeup of a screen page to be determined by the preset. A Userobject is a control which may be placed on the PARAMETER screen by connecting the Userobject output to the userobject inputs on HEAD module (right hand side of screen in graphics view), usually via a menupage. Restrictions in use: 1) uo_switch inputs may not mix menupages and non-menupages when output is connected to HEAD. 2) uo_switch inputs may not mix triggers and non-triggers when output is connected to HEAD. 3) uo_switch inputs may only be blank when all non-blank inputs are menupages and output is connected to HEAD.

Godlike Productions Comments

Specifiers

Header	Description
nsubuos	The number of userobjects to show. These are the number of menu pages you can select between.

Control Inputs

Header	Description
in	select the active userobject. Range 0 to nsubuos-1.

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject output. This passes whichever subuos referred to by the input integer.
subuosn	The menupages to select between.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
obj				
nsubuos				
INT				
subuos				
		@nsubuos		
USEROBJECTCHILD				
in				
CONTROLINPUT				

vfader(fdr) - Vertical Fader Knob

Description

A Knob with graphical representation. Rotating the knob causes the 'low 1' to slide up or down. Up to six vfaders may be pictured on a single screen. Three vfaders may share a display page with four knobs.

Godlike Productions Comments

If this block is fed by a module using log (for example level), ensure that this never receives 0, or it will crash emote. Place a c_bound before the log block with a min value of 0.0000001 (or similar).

Specifiers

Header	Description
statement	description for future use.
name	6 character description for PARAMETER menu
min	Value for the bottom of the fader. Range -32768 to 32768.
max	Value for the top of the fader. Range -32768 to 32768
resolution	Step rate. Range 0 to 32767
default	Value that the vfader will be set to when it is first used. Range: from min to max.

Control Inputs

Header	Description

Control Outputs

Header	Description
out	knob output.

User Objects

Header	Description
obj	The userobject for this fader. Connect to a menupage to show the control on the H9000 screen or Emote.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
statement				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	menu statement			
name				
STRING	name			
min	-32768	32767	-32768	
FLOAT	min value			
max	-32768	32767	32767	
FLOAT	max value			
res	0	32767	1	
FLOAT	resolution			
default	@min	@max	0	
FLOAT	default			
out				
CONTROLOUTPUT	out			
obj				
USEROBJECTPARENT	obj			

vmonitor(vmn) - Vertical monitor

Description
A control signal monitor that is shown much like a vertical fader.

Godlike Productions Comments
Ensure that audio signals are not wired directly into hmonitor or vmonitor (via atoc or atoc(filt) or emote will crash. Use the default peak detect block before the monitor to keep it safe for emote. This block still has bugs in VSIG 3.4.2, specifically it will not display values. Use a monitor along side of it.

Specifiers

Header	Description
name	the name of this userobject (currently not used)
tag	A short name for this object. A %f will include a number.
min	minimum expected value
max	maximum expected value

Control Inputs

Header	Description
in	The signal to be monitored

Control Outputs

Header	Description

User Objects

Header	Description
obj	The userobject for this fader. Connect to a menupage to show the control on the H9000 screen or Emote.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
	%n			

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
min			0	0.001
FLOAT	minimum			
max			1	0.001
FLOAT	maximum			
name			vmonitor:%f	
STRING	name			
tag			vmonitor	
STRING	tag			
obj				
USEROBJECTPARENT	obj			

Chapter 10 - Math

Introduction

Most of the math functions in this chapter are designed to do calculations on audio rate streams. Many of the blocks allow you to type constants into the block directly (such as In1 or In2), and while this may seem an efficient way to work, this will lead to unpredictable results.
In all cases and constant block should be used to introduce a constant or DC signal into a math block. The constant block correctly formats and generates a proper audio stream that can be utilized by the math functions in this chapter.

abs(abs) - Absolute Value Code of Audio Signal

Description
This module takes the arithmetic absolute value of an audio input signal. This is equivalent to full-wave rectification. This can be used as an crude frequency doubler or in level detection applications.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	Audio Input

Audio Outputs

Header	Description
out	Absolute value of input

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				

add(add) - Add Two Audio Signals

Description

This module adds the two audio signals “in1” and “in2”. It is the simplest way of mixing two signals together. It is often used for creating feedback loops around delay lines.

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in1	Audio Input
in2	Audio Input

Audio Outputs

Header	Description
out	Sum of inputs

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clipvalue	0	10000	10000	0.1
	clip_value			
in1				
LEFT				
AUDIOINPUT				
in2				
RIGHT				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

adder(add) - Add Multiple Audio Signals

Description

This module adds two or more audio signals together. The number of signals to be added is specified by the "ninputs" specifier.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	The number of inputs
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
inn	Audio Inputs

Audio Outputs

Header	Description
out	Sum of inputs

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
ninputs	2	50	2	1
	number_inputs			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in				
MONO		@ninputs		
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

ampmod(mod) - Amplitude Modulator

Description

The ampmod module will uses one audio input (mod) to control the amplitude of another (in). This is equivalent to one signal being multiplied by the other. The ampmod module is useful for creating tremolo effects, autopanning, envelope control, and many other applications.

Godlike Productions Comments

Specifiers

Header	Description
offset_time_constant	Controls the time constant of the tracking filter to smoothly change the offset parameter. Given in milliseconds.

Audio Inputs

Header	Description
in	Audio Input
mod	Modulation Input

Audio Outputs

Header	Description
out	Modulated Output

Control Inputs

Header	Description
modamt	This control signal scales the mod input before it is multiplied with the input signal. In combination with the offset control, this can be used to control the depth of amplitude modulation.
modlimit	The modlimit control sets a +/- ceiling on how far the amplitude modulation can go, up to 10 times the input signal. The new equation will be ampltmod = (offset + mod*modamt) if (ampltmod > modlimit) ampltmod = modlimit if (ampltmod < -modlimit) ampltmod = -modlimit out = in*ampltmod
offset	The offset control signal determines the amplitude of the output signal in the absence of any modulation signal. Its value is added the mod input scaled by the modamt. Mathematically, this is: out = in * (offset + mod * modamt)

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
offset_time_constant	0	5000	0	0.01
	ramp_time			
in				
MONO				
AUDIOINPUT				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
modamt	-10	10		0.01
CONTROLINPUT	modulation			
offset	-10	10		0.01
CONTROLINPUT	offset			
modlimit	0	10	1	
CONTROLINPUT				

and(and) - Logical AND Control Signals

Description

This module execute a logical AND of two audio signals. A value of 1.0 is defined to be true and a value of 0.0 is defined to be false. If either input has a value greater than the threshold, the output is set to 1.0, otherwise it is set to 0.0.

Godlike Productions Comments

The json file mistakenly lists a control output. This block does not have a control output.

Specifiers

Header	Description

Audio Inputs

Header	Description
in1,in2	The input control signals that are to be logically ANDed together.

Audio Outputs

Header	Description
output	(in1 > thresh) && (in2 > thresh)

Control Inputs

Header	Description
threshold	The decibel value above which represents true

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				
MONO				
in2				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
thresh	-96	0	0	0.01
	db			
CONTROLINPUT	threshold			

audiotaper(aut)

Description

Tapers an audio rate signal the same way that c_master_taper tapers control rate signals. An alpha value is used to smoothly bend the incoming signal in different directions. This can be very useful for shaping lfo signals. NOTE: The signal is assumed to be between 0 - 1, and outputs a signal in the range 0 - 1.

Godlike Productions Comments

The json file lists the output as an input

Specifiers

Header	Description
alpha	Controls the tapering of the signal.

Audio Inputs

Header	Description
in	The input signal to be tapered.

Audio Outputs

Header	Description
out	The tapered signal.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
alpha	0	32767	1	0.0001
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

bound(bnd) - Signal bounder

Description
limit the amplitude of the signal to a min and a max.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The signal to be bounded.

Audio Outputs

Header	Description
out	The bounded output.

Control Inputs

Header	Description
maximum	-1.0 to 1.0. The maximum value
minimum	-1.0 to 1.0. The minimum value

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
LEFT				
out				
MONO				
AUDIOOUTPUT				
minimum	-1	1	-1	0.0001
CONTROLINPUT	minimum			
maximum	-1	1	1	0.0001
CONTROLINPUT	maximum			

comparator(cmp) - Audio Signal Comparator

Description

This module compares the value of one audio signal to that of another. If the value of the first signal is greater, the output is set to fullscale. If the value is smaller, the output is set to minus fullscale. The comparator has a hysteresis control that makes the “turn-on” value greater than the “turn-off” value. This prevents undo oscillation of the output.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in1	The “positive” input. If this is greater than in2, the output goes positive
in2	The “negative” input.

Audio Outputs

Header	Description
out	Comparator Output

Control Inputs

Header	Description
hysteresis	This controls the amount of hysteresis. A setting of zero will make the turn-on and turn-off thresholds identical. A setting of 1.0 will cause the thresholds to be +/1 fullscale.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				
MONO				
in2				
MONO				
AUDIOINPUT				
hysteresis	0	1		0.01
CONTROLINPUT	hysteresis			
out				
MONO				
AUDIOOUTPUT				

comparator2(cmp) - Audio Signal Comparator

Description

This module compares the value of one audio signal to that of another. If the value of the first signal is the same as the second, the output is set to fullscale, otherwise the output is set to zero. The comparator has a spread control that allows a narrow band of values of one input to be viewed as equal to the other input.

Godlike Productions Comments

The json does not outline all of the inputs or outputs.

Specifiers

Header	Description

Audio Inputs

Header	Description
in1	The "first" input. If this is the same as in2, the output goes positive, otherwise it is 0.
in2	The "second" input.

Audio Outputs

Header	Description

Control Inputs

Header	Description
inspread	The controls the range of values considered equivalent. A setting of zero will make the turn-on and turn-off thresholds identical. A setting of 1.0 will cause the thresholds to be +/1 fullscale.
constval	This is not documented in the json file. Probably a default value to output if there is no input.
outspread	This is not documented in the json file. Perhaps a spread of the output - similar to hysteresis.

Control Outputs

Header	Description

Mod Outputs

Header	Description
out	Comparator Output

Header	Description
constout	Constant output. Not documented in json file. Perhaps mirrors the constval input.

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				
MONO				
in2				
MONO				
AUDIOINPUT				
inspread	0	1		0.01
CONTROLINPUT				
constval	-1	1		
CONTROLINPUT				
outspread	0	1		
CONTROLINPUT				
out				
MONO				
AUDIOOUTPUT				
constout				
MONO				
AUDIOOUTPUT				

constant(con) - Constant Audio Signal

Description

This module's purpose in life is to create and audio signal output that remain's at a fixed (DC) value. This is often needed in creating different modulation schemes where an offset may need to be added to generate the proper range of values.

Godlike Productions Comments

Specifiers

Header	Description
value	The value to be assigned to the audio output. This is a number in the range of +/- 1.

Audio Inputs

Header	Description

Audio Outputs

Header	Description
out	Constant Output

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
value	-1	1	0	0.0001
	value			
out				
MOD				
AUDIOOUTPUT				

cosine(cos) - Cosine function

Description

This module implements a sinus function. More specifically it outputs the following formula if(func == 1) output[n] := cos(pi/2*input[n]) else output[n] := cos(pi*input[n]) A specifier sets the actual number of terms used in the polynomial series to compute the cosine. In this way, the user has control over the accuray of the computation which has a direct impact on the CPU load

Godlike Productions Comments

Specifiers

Header	Description
func	{0,1}. 0 -> cos(pi*t); 1 -> cos(pi/2*t).
nios	1 to 32 sine functions can be handled.
nterms	3 to 7 polynomial terms. Specifies the number of harmonics generated.

Audio Inputs

Header	Description
inn	The audio inputs

Audio Outputs

Header	Description
outn	The audio outputs = cos(pi/2*in).

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nios	1	32	1	2
	n_ios			
nterms	2	4	2	1
INT	n_terms			
func	0	1	1	0
INT	sin_func			
in				
MONO		@nios		
AUDIOINPUT				
out				
MONO		@nios		
AUDIOOUTPUT				

curve - Mapping or waveshaping function.

Description

An arbitrary relationship between an input value and an output value. This relationship is called a map. Your map is formed by ajusting points along the map. You specify how many points. The points are placed equally along the input. For each input point, you adjust the output value for that point. For an input value that is between two points, the output is found on a straight line between the points. (Linear interpolation) This module is full bandwidth. You can use it to make interesting distortion effects.

Godlike Productions Comments

Specifiers

Header	Description
npoint	the number of points in the mapping function. Maximum of 32 points.

Audio Inputs

Header	Description
in	The input to be mapped or shaped.

Audio Outputs

Header	Description
out	The mapped output.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description
obj	the map of the function

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
npoints	1	32	1	1
INT	number_points			
point	-1	1	0	0.001
		@npoints		
INT	point_~n			
obj				
USEROBJECTPARENT	obj			

differentiator(du) - the differential of a signal

Description

This module performs the calculus function of differentiation. That is: the output at any one point is the slope of the input at that point. Another description is that the output is a measure of how the input changes. If you put a ramp into this module, you will get a dc value out since a ramp is a constant rate of change. The compliment of a differentiator is an integrator. You can use HIGHCUT with a very low frequency to perform this function.

Godlike Productions Comments

Specifiers

Header	Description
	The time constant, or ramp time. This is the maximum speed that the output can change. Small values are true differentiators. Larger values slow the response and will better track the input (ie less differentiation)

Audio Inputs

Header	Description
in	The audio input

Audio Outputs

Header	Description
out	The derivative of the input

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
time_constant	0	100	0	0.01
	ramp_time			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

dither - Dither and PCM Word Truncation

Description

The dither module prepares the audio streaming through it for transfer to an integer format. We generate a noise signal of amplitude on the order of the DAC LSB and specified probability density function, then add the signal to the audio, and finally truncate the sum to the PCM width of the output format. This module should be employed only at the end of the processing chain. The goal of dither is to decorrelate the artificial signal generated by cutting the audio word length from the signal itself. Further processing of a dithered and truncated signal should be dithered again. The dither signal is generated by bitwise exclusive-or with a delayed dither word. The seed to this sequence is calculated at initialization by the host. This module is stereo so that the same dither signal cannot be applied to both channels. This is to say, the dither applied to the left and right channels is statistically independent.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
leftin	Left channel input
rightin	Right channel input

Audio Outputs

Header	Description
leftout	Left channel output
rightout	Right channel output

Control Inputs

Header	Description
dither	Controls the probability density function of the dither signal. This is an indexed value mapped as follows: 0 -> rectangular PDF [0..1), 1 -> triangular PDF [0..1)
wrdlen	Controls the length of the output PCM word in bits. This is an indexed value mapped as follows: 0 -> 16 bits, 1 -> 18 bits, 2 -> 20 bits, 3 -> 24 bits

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
leftin				
LEFT				
rightin				
RIGHT				
AUDIOINPUT				
leftout				
LEFT				
AUDIOOUTPUT				
rightout				
RIGHT				
AUDIOOUTPUT				
wrdlen	0	3		
			16,18,20,24	
CONTROLINPUT	word length			
dither	0	1		
			rectangular,triangular	
CONTROLINPUT	dither type			

exp(exp) - Exponential Function

Description

The exp module passes an audio signal through an exponentially shaped function. This function is the complement of the log function, meaning that a signal passed through the log function and then through the exp function will be restored to its original state. The actual function that is used is: out = 2 ^ [(in-1)*16] for in >= 0 out = -2 ^ [(-in-1)*16] for in < 0

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The exp input.

Audio Outputs

Header	Description
out	The exp output.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				

exp_mod(exp) - Exponential Function for Mod Signals

Description

The exp module passes an mod-type audio signal through an exponentially shaped function. This function is the complement of the log function, meaning that a signal passed through the log function and then through the exp function will be restored to its original state. The only difference between the exp_mod module and the exp module is that the exp_mod is only calculated at the "mod" sample rate which is 1/4 the audio sample rate. The actual function that is used is: out = 2 ^ [(in-1)*16] for in >= 0 out = -2 ^ [(-in-1)*16] for in < 0

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The exp input.

Audio Outputs

Header	Description
out	The exp output.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MOD				
out				
MOD				
AUDIOOUTPUT				

gain(gan) - Audio Gain Adjust

Description

This module applies a fixed amount of gain to an audio signal. The amount of gain is user selectable and may be any positive or negative amount within the allowable range. The gain is in decibels (dB).

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in	The audio input.

Audio Outputs

Header	Description
out	The processed output.

Control Inputs

Header	Description
gain	-144 to +96 dB. Controls the amount the signal is amplified or attenuated. Positive values amplify while negative ones attenuate.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
clipvalue	0	10000	10000	0.1
	clip_value			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
gain	-144	96	0	1
	db			
CONTROLINPUT	%n			
obj				
USEROBJECTPARENT	obj			

integrator - Audio Delay

Description

This module implements a simple digital integrator. That is, the output is a straight running sum of the input samples. An input level-triggered reset signal can load an initial constant to the running sum.

Godlike Productions Comments

This block is not used in any factory presets. It is not windowed and acts like a low level RC Integrator.

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio input to be integrated.

Audio Outputs

Header	Description
out	The integration of the input signal.

Control Inputs

Header	Description
initial_condition	The value to initialize the state to from -1 to 1, either when block is loaded, or when a reset signal is received.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
reset				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
initial_condition	-1	1	0	0.001
CONTROLINPUT	%n			

log(log) - Logarithm Function

Description

The log module passes an audio signal through an logarithmically shaped function. This function is the complement of the exp function, meaning that a signal passed through the log function and then through the exp function will be restored to its original state. The actual function that is used is:
out = [logbase2(in)/16] + 1 for in >= 0 out = -[logbase2(-in)/16] - 1 for in <= 0

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The log input.

Audio Outputs

Header	Description
out	The log output.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
out				
AUDIOOUTPUT				

log_mod(log) - Logarithm Function for Mod Signals

Description

The log module passes an audio signal through an logarithmically shaped function. This function is the complement of the exp function, meaning that a signal passed through the log function and then through the exp function will be restored to its original state. The only difference between the log_mod module and the log module is that the log_mod is only calculated at the "mod" sample rate which is 1/4 the audio sample rate. The actual function that is used is: out = [logbase2(in)/16] + 1 for in >= 0 out = -[logbase2(-in)/16] - 1 for in >= 0

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The log input.

Audio Outputs

Header	Description
out	The log output.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MOD				
out				
MOD				
AUDIOOUTPUT				

m_curve - Mapping for mod signals

Description

An arbitrary relationship between an input value and an output value. This relationship is called a map. Your map is formed by ajusting points along the map. You specify how many points. The points are placed equally along the input. For each input point, you adjust the output value for that point. For an input value that is between two points, the output is found on a straight line between the points. (Linear interpolation) This module is for mod type signals that go from -1 to 1. This module is good to warp the outputs of envelopes and LFOs

Godlike Productions Comments

This is pretty cool actually as the map can be adjusted in real time, but it is at mod frequency (ie every 4th sample). Documentation is wrong. This block does not have an obj output.

Specifiers

Header	Description
npoint	the number of points in the mapping function. (Max of 32 that are equally spaced)

Audio Inputs

Header	Description
in	The input to be mapped. Can be from -1 to 1

Audio Outputs

Header	Description
out	The mapped output.

Control Inputs

Header	Description
pointn	The output value of each point.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
npoints	1	32	1	1
INT	number_points			
point	-1	1	0	0.01
		@npoints		
CONTROLINPUT	point_~n			

m_ucurve - Unipolar Mapping for mod signals

Description

An arbitrary relationship between an input value and an output value. This relationship is called a map. Your map is formed by ajusting points along the map. You specify how many points. The points are placed equally along the input. For each input point, you adjust the output value for that point. For an input value that is between two points, the output is found on a straight line between the points. (Linear interpolation) This module is for mod type signals that go from 0 to 1. If input is less than one, the output is the value of point zero. This module is good to warp the outputs of envelopes and LFOs

Godlike Productions Comments

Same as m_curve except unipolar.

Specifiers

Header	Description
npoint	the number of points in the mapping function. (Max of 32 that are equally spaced)

Audio Inputs

Header	Description
in	The input to the mapping module. Should be from 0 to 1.

Audio Outputs

Header	Description
out	The mapped output.

Control Inputs

Header	Description
pointn	The output value of each point.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description
obj	the map of the function

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
npoints	1	32	1	1
INT	number_points			
point	-1	1	0	0.01
		@npoints		
CONTROLINPUT	point_~n			
obj				
USEROBJECTPARENT	obj			

multiply(mul) - Multiply two input audio signals.

Description
The multiplication operation.

Godlike Productions Comments
There seems to be limitation on upward multiplication of audio signals, by, say a constant. Gain may be more appropriate for upward multiplication.

Specifiers

Header	Description

Audio Inputs

Header	Description
in1	First operand to be multiplied.
in2	Second operand to be multiplied.

Audio Outputs

Header	Description
out	in1*in2.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				
MONO				
in2				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

noiseshape - Audio Delay

Description

This module implements a simple first-order noise shaper. The quantization operation is the floor function, and the noise shape filter feeding back to the input is a first-order FIR highpass, i.e. a filter with one zero at DC.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio input to noise shape.

Audio Outputs

Header	Description
out	The quantized and noise-shaped

Control Inputs

Header	Description
bits	The number of bits from unity full-scale to quantize to, from 1 to 24.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
bits	1	24	16	1
CONTROLINPUT	%n			

not(not) - Logical NOT Control Signals

Description

This module execute a logical NOT of an audio signal. A value of 1.0 is defined to be true and a value of 0.0 is defined to be false. If the input has a value greater than the threshold, the output is set to 0.0, otherwise it is set to 1.0.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input control signal that is to be not-ed

Audio Outputs

Header	Description
output	(in < thresh)

Control Inputs

Header	Description
threshold	The decibel value above which represents true

Control Outputs

Header	Description
out	The logical NOT of the input control signal

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
thresh	-96	0	0	0.01
	db			
CONTROLINPUT	threshold			

or(or) - Logical OR Control Signals

Description

This module execute a logical OR of two audio signals. A value of 1.0 is defined to be true and a value of 0.0 is defined to be false. If either input has a value greater than the threshold, the output is set to 1.0, otherwise it is set to 0.0.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in1,in2	The input control signals that are to be logically ORed together.

Audio Outputs

Header	Description
output	(in1 > thresh) || (in2 > thresh)

Control Inputs

Header	Description
threshold	The decibel value above which represents true

Control Outputs

Header	Description
out	The logical OR of the input control signals

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				
MONO				
in2				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
thresh	-96	0	0	0.01
	db			
CONTROLINPUT	threshold			

Overload - tests signals for values over a threshold

Description

This module takes the arithmetic absolute value of an audio input signal. This is equivalent to full-wave rectification. This can be used as an crude frequency doubler or in level detection applications. The highest sample peak in a chunk is reported to the control code. If it is greater than the user-determined threshold, the control output ol is written as high, o/w it is written as 0.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	Audio input

Audio Outputs

Header	Description

Control Inputs

Header	Description
thresh	The decibel value above which the count starts. If the signal exceeds this level it is set to 1 for the length of the chunk, otherwise the output is 0.

Control Outputs

Header	Description
ol	1 when overload is detected. 0 otherwise.

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
thresh	-96	0	0	0.01
	db			
CONTROLINPUT	threshold			
ol				
CONTROLOUTPUT	ol			

Overload2 - tests signals for a number of consecutive values over a threshold

Description

This module takes the arithmetic absolute value of an audio input signal. This is equivalent to full-wave rectification. This can be used as an crude frequency doubler or in level detection applications. The highest sample peak in a chunk is reported to the control code. If it is greater than the user-determined threshold, the control output ol is written as high, o/w it is written as 0.

Godlike Productions Comments

There is a bug in this module in v 3.3.3 of VSIG and 2.0.5 of H9000. The output never falls back to FALSE after it has been triggered. Eventide are aware of the bug, and it will be fixed in a future update.

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio input to be tested for overload.

Audio Outputs

Header	Description

Control Inputs

Header	Description
thresh	The decibel value above which the count starts. If the signal remains above this value for more than the number of samples in count, then TRUE is set on the output.
count	The number of consecutive samples above the threshold before the output is set true.

Control Outputs

Header	Description
ol	1 when overload is detected. 0 otherwise.

Mod Outputs

Header	Description

User Objects

Header	Description

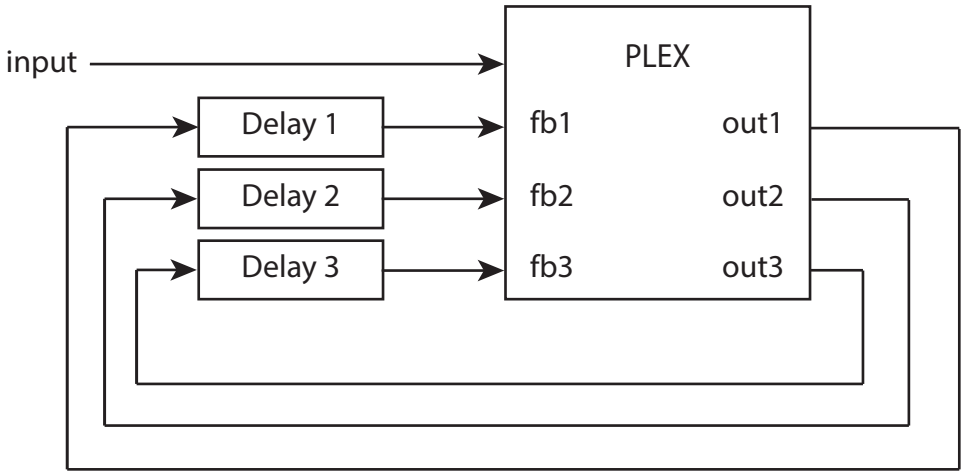
Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
thresh	-96	0	0	0.01
	db			
CONTROLINPUT	threshold			
count	1	10	1	1
	samples			
CONTROLINPUT	sample count			
ol				
CONTROLOUTPUT	ol			

plex(plx) - Reverberation Tool

Description

The plex module provides a simple way of creating high quality reverberators.To create a reverberator, the outputs of a bunch of delay lines are fed into the plex module and the outputs of the plex module are fed back to the inputs of the delay line. The plex module combines the delay outputs to produce an exponentially increasing density of echoes, hence a dense reverb. The signal that is to be used as the main input to the reverberator is connected to the "in" of the plex module. Here is a typical patch.



You can make things even more interesting by experimenting with add other modules besides delay lines in the feedback structure.

Godlike Productions Comments

Specifiers

Header	Description
size	2 through 24. Specifies how many feedback loops to create.

Audio Inputs

Header	Description
in	The main audio input to the reverb structure.
fbn	The feedback inputs

Audio Outputs

Header	Description
out(n)	The outs to be connected to the delay line inputs.

Control Inputs

Header	Description
feedback	-1.0 to 1.0.Controls the overall feedback amount.

Header Description

Control Outputs

Header Description

Mod Outputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
size	2	24	2	1
	plex_size			
in				
MONO				
AUDIOINPUT				
fb				
MONO		@size		
AUDIOINPUT				
out				
MONO		@size		
AUDIOOUTPUT				
feedback	-1	1	0.9	0.0001
	%			
CONTROLINPUT				

quadrature(qad) - Hilbert Quadrature Transformer

Description

Provides a -90 degree phase shift of all frequency components up to 1/4 nyquist.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to be transformed.

Audio Outputs

Header	Description
norm	A delayed version of the input
quad	A 90 degree phase-shifted version of the input. This phase shift is relative to the norm output.

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
norm				
MONO				
AUDIOOUTPUT				
quad				
MONO				
AUDIOOUTPUT				

quantize(qnt) - Audio Bit Quantizer

Description

The quantize module truncates a digital audio signal to the specified number of bits. If truncation distortion is not desired, use the noiseshape module. This module is useful in simulating low-resolution digitization or for creating stepped waveforms from an LFO.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to be quantized.

Audio Outputs

Header	Description
out	The quantized output.

Control Inputs

Header	Description
bits	-24 to 24. Controls how many bits the signal will be quantized to. The remaining low order bits will be truncated (set to zero). Negative values correct for the inherent bias such that very small signals will always be 0, positive values do not, which results in a fair amount of noise with low bit values.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
bits	-24	24	16	1
CONTROLINPUT	bits			

rms - Root Mean Square

Description

This module calculates the rms value of an incoming audio signal. The signal's MS value is determined using modpp::acc_precision precision math, and then a recursive equation is used to approximate the root. In order to eliminate oscillation on the output, the output of the equation rms(n) = ms(n)/(rms(n-1)+noise) is averaged with rms(n-1). To promote readability and simplicity, these two equations are separately executed, rather than combining them into a more complex equation. In the interests of speed, only one rms value is calculated per iteration. The remaining three intermediary values are linearly interpolated between the previous and current rms value. Satisfactory results are given above a minimum averaging period of approximately 10 ms.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to take the rms value.

Audio Outputs

Header	Description
out	The rms value.

Control Inputs

Header	Description
avrgtime	5 to 5000 ms. This determines the length of the MS averaging period.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
avrgtime	5	5000	100	0.01
	ms			
CONTROLINPUT	avrg_time			

scale(scl) - Audio Signal Scaler (Attenuator)

Description
This module is used to adjust the amplitude of an audio signal.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input to be attenuated.

Audio Outputs

Header	Description
out	The attenuated output.

Control Inputs

Header	Description
amp	-1.0 to 1.0. Controls the amount of attenuation for the input. The gain adjustment is a LINEAR value (not dB), with 1.0 being no attenuation. Negative numbers will invert the phase of the signal.

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
amp	-1	1	1	0.0001
CONTROLINPUT	%n			

sinus(sin) - Sinus function

Description

This module implements a sinus function. More specifically it outputs the following formula if(func == 1) output[n] := sin(pi/2*input[n]) else output[n] := sin(pi*input[n]) A specifier sets the actual number of terms used in the polynomial series to compute the sine. In this way, the user has control over the accuracy of the computation which has a direct impact on the CPU load.

Godlike Productions Comments

Specifiers

Header	Description
func	{0,1}. 0 -> sin(pi*t); 1 -> sin(pi/2*t).
nios	1 to 32 sine functions can be handled.
nterms	2 to 4, 3rd to 7th order polynomial terms. Specifies the number of harmonics generated.

Audio Inputs

Header	Description
in	The audio input.

Audio Outputs

Header	Description
out	The audio output = sin(pi/2*in).

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nterms	2	4	2	1
	n_terms			
nios	1	32	1	1
INT	n_inouts			
func	0	1	0	1
INT	sin_func			
in				
MONO		@nios		
AUDIOINPUT				
out				
MONO		@nios		
AUDIOOUTPUT				

softclip(clp) - Softclipper

Description

This module implements a series of softclippers. The softlcipper in use is chosen by the specifier, Each option has a gain of 1 at drive = 0 and an output swing of -1 to 1 at drive = 100.

Godlike Productions Comments

There is a hyperbolic tan function available in here if needed.

Specifiers

Header	Description
select	0 selects a symmetric polynomial softclipper from 0 to 9th order. 0th order is a standard hardclipper, 1st order is a cubic, 2nd order is a 5th, ... , 9th order is to the 19th power. The coefficients are chosen for the smoothest saturation possible for a polynomial. 1 selects a hyperbolic tangent soft clipper. These are popular because they're infinitely differentiable, but they only asymptotically approach -1 and 1 and are expensive to compute.

Audio Inputs

Header	Description
in	The audio input to be clipped.

Audio Outputs

Header	Description
out	A clipped version of the input signal.

Control Inputs

Header	Description
drive	0 to 100. Unity gain at 0, full swing output at 100.
order	0 to 9. polynomial order
processoff	0 or 1, turns off processing

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
select	0	1	0	1
INT	clip_select			
drive	0	100	0	0.1
CONTROLINPUT	%n			
order	0	9	0	1
CONTROLINPUT	%n			
processoff	0	1	0	1
CONTROLINPUT	turn_off			

sqrt(sqt) - Positive Square Root of Audio Signal

Description

This module takes the positive square root value of an audio input signal.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	Audio input

Audio Outputs

Header	Description
out	Square root value of the input

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				

subtract(sub)- Subtract Two Audio Signals

Description
This module simply subtracts one audio signal from another.

Godlike Productions Comments

Specifiers

Header	Description
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in1	The first input.
in2	The second input.

Audio Outputs

Header	Description
out	The output (in1 - in2)

Control Inputs

Header	Description

Control Outputs

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
clipvalue	0	10000	10000	0.1
	clip_value			
in1				
AUDIOINPUT				
in2				
AUDIOINPUT				
out				
AUDIOOUTPUT				

xformer(xfr) - This is a simple model of a saturating transformer

Description

This module simulates the behavior of a saturating transformer. Specifically, it integrates the signal to simulate the flux through the transformer core, saturates the flux, and differentiates the signal back into an electrical potential.

Godlike Productions Comments

This can probably be used to do division. There is no audio rate division function. It will probably be computationally expensive though, but maybe not, as this seems to use differentiation and integration, which should be pretty low cost on a digital system.

This block can probably be used for analog style saturation and may possibly act as a wide band-pass filter (need to test frequency sweep)

For more information on transformer physics - https://www.electronics-tutorials.ws/transformer/transformer-basics.html and https://www.jensen-transformers.com/wp-content/uploads/2014/08/Audio-Transformers-Chapter.pdf

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The voltage across the transormers primary winding.

Audio Outputs

Header	Description
out	The output voltage across the transformers secondary

Control Inputs

Header	Description
n1	100 to 20000 turns. The number of turns on the primary winding. This has the effect of setting the corner frequency for the integration.
ratio	0.1 to 10. Sets the transformer ratio, which is the number of turns on the secondary winding divided by the number of turns on the primary winding. This effects both the gain through the transformer and the corner frequency of the differntiation.
area	This control sets the cross sectional area of the transformer core in square cm. This affects how easily the transformer saturates. (maybe)
satpoint	The controls the saturation point of the transformer

Control Outputs

Header	Description

Header	Description

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
out				
AUDIOOUTPUT				
n1	100	20000	1000	1
	number			
CONTROLINPUT	primturns			
ratio	0	10	1	0.1
	ratio			
CONTROLINPUT	ratio			
area	0	10	1	0.01
	sqarecm			
CONTROLINPUT	csarea			
satpoint	-1	1	1	0.01
	something			
CONTROLINPUT	satpoint			

xor(xor) - Logical XOR Control Signals

Description

This module execute a logical XOR of two audio signals. A value of 1.0 is defined to be true and a value of 0.0 is defined to be false. If either input has a value greater than the threshold, the output is set to 1.0, otherwise it is set to 0.0.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in1,in2	The input control signals that are to be logically XORed together.

Audio Outputs

Header	Description
output	(in1 > thresh) != (in2 > thresh)

Control Inputs

Header	Description
threshold	The decibel value above which represents true

Control Outputs

Header	Description
out	The logical XOR of the input control signals

Mod Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in1				
MONO				
in2				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
thresh	-96	0	0	0.01
	db			
CONTROLINPUT	threshold			

Chapter 11 - Miscellaneous

auxxin (axi)

Description
Aux assigner for H9 Pedal

Godlike Productions Comments
There is nothing in the .json file for this. I suspect this is used for the H9 for control of how the Aux Switch acts. The module has no output, only an internal input and WHICH. I suspect this directs the Aux in to which parameter it is assigned to.

Specifiers

Header	Description
which	Parameter number that the aux controls.

Audio Inputs

Header	Description
in	Audio input

Audio Outputs

Header	Description

Control Inputs

Header	Description

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
which				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
INT				

flipflop(ffp) - Audio octave Divider

Description

The flipflop module generates a relatively crude method of producing a tone an octave below that of its input. If the input is a reasonably simple periodic tone, the output of this module will be a square-wave one octave lower. This module operates by changes the state of its output every time the input transitions through zero. For this reason, this module works best by providing it with a simple input signal such as a sine or square wave. To get reasonable performance from other signals, it is usually a good idea to put a lowpass filter before this module.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The flipflop input.

Audio Outputs

Header	Description
out	The flipflop output.

Control Inputs

Header	Description
hysteresis	0 to 1.0. Controls the amount of hysteresis, or the immunity to spurious triggering. A value of zero result in no hysteresis, while a value of 1.0 will never change state.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
hysteresis	0	1		
CONTROLINPUT				
out				
MONO				
AUDIOOUTPUT				

oneshot (sht) - Triggered timer

Description

When a low to high transition occurs, output goes high for a select period of time. This is retriggrable in that the output will stay high for the time period from the last low to high transisition

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	Trigger input

Audio Outputs

Header	Description
out	Output triggered for a set time

Control Inputs

Header	Description
time	How long to stay high
threshold	trigger threshold

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
time	0	30000	1	0.01

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	ms			
CONTROLINPUT	time			
threshold	-1	1	0.5	0.001
CONTROLINPUT	threshold			

samphold (smp)- Sample-and-hold for audio signals.

Description

This module will sample the IN input signal, as long as the NEWSAMP output remains high (>=1 bit in a 24 bit number). When NEWSAMP is low (<1 bit in a 24 bit number), the output will remain unchanged.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The input control to be sampled.
newsamp	Tells when to take a new sample of IN

Audio Outputs

Header	Description
out	The currently-held sample.

Control Inputs

Header	Description

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
newsamp				
MOD				
AUDIOINPUT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
MONO				
AUDIOOUTPUT				

seqcount - Mod Signal Sequencer Counter

Description

This module implements a leading-edge-triggered stepped counter. The counter will progress from zero upwards to one less the reciprocal of the number of steps, and then it will automatically wrap around in a modulo fashion. The state can be reset to zero via a level-triggered reset line.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	Audio rate input that controls when the sequencer state will increment. The increment occurs when the signal value crosses the positive threshold control, but the signal must cross the negative threshold before the positive threshold triggers the next increment.
resetin	Audio rate input that is a level-triggered reset signal. Reset is triggered when the absolute value is greater than or equal to 0.5.

Audio Outputs

Header	Description
clkout	Audio rate output that shows the incremental state of the sequencer.
errout	An elaborate diagnostic comprising the threshold polarity, the incremental state, and the input signal into one sample.

Control Inputs

Header	Description
steps	The modulo number of incremental states in the sequencer, from 2 to 6000.
threshold	The input threshold past which the state
polarity	A switch for the input and output range. When set to zero, both input and output are ranged from [0..1] instead of [-1..1]. Polarity must match the condition of the input signal for proper operation.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
resetin				
MONO				
AUDIOINPUT				
clkout				
MONO				
AUDIOOUTPUT				
errout				
MONO				
AUDIOOUTPUT				
steps	2	6000	16	1
	steps			
CONTROLINPUT	number_steps			
threshold	0	1	1	0.0001
CONTROLINPUT	thresh			
polarity	0	1	1	1
CONTROLINPUT	polar			

sequencer - Mod Signal Sequencer

Description

This module implements a direct map from input to output. The map contains a finite number of discrete steps, and input values from zero to one linearly translate to the table index. The sequencer table value at the specific index is retrieved and sent to an averaging lowpass filter for glide effect. This can be used to translate a sequence of incremental cues into a recurring pattern that can translate into, say, a melody.

Godlike Productions Comments

The sequencer steps can be modulated in realtime, so this can be much more flexible than it appears.

Specifiers

Header	Description
n_steps	Specifies the number of entries in the sequencer table, from 2 to 50.

Audio Inputs

Header	Description
in	The absolute value of the input maps in steps from 0 to n_steps-1 for full scale.

Audio Outputs

Header	Description
out	The table-mapped output, post-glide.

Control Inputs

Header	Description
fullscale	The value of full-scale for the value[n] elements. The value[n] elements will be normalized by the reciprocal for the corresponding output map target.
glide	Time constant for calculating an averaging filter coefficient to be applied to the output, in milliseconds.
value[n]	Table of n_steps sequencer map values.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
n_steps	2	50	2	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	number_steps			
in				
MOD				
AUDIOINPUT				
out				
MOD				
AUDIOOUTPUT				
fullscale	-32000	32000	1	0.001
CONTROLINPUT	%n			
glide	0	100	0	1
	msec			
CONTROLINPUT	%n			
value	-32768	32767	0	0.001
		@n_steps		
CONTROLINPUT	value~n			

transient_split

Description
This module takes an input audio stream and splits it into two output streams, one containing only Tonal audio, and the other containing only Transient audio. The split is completely complementary, that is, if you add together the two outputs, you'll get the original input (albeit with some processing latency).

Godlike Productions Comments
This has disappeared from VSIG 3.4.2 :(

Specifiers

Header	Description

Audio Inputs

Header	Description
in	the audio input to be split into tonal and transient outputs

Audio Outputs

Header	Description
out_tonal	The tonal audio output
out_transient	The transient audio output

Control Inputs

Header	Description
resolution_prefer	The time or frequency resolution preference, two choices: 0: time - prefer time resolution with sharper transients with possible low frequency tonal leakage into transient. 1: frequency - prefer frequency resolution with potentially slightly smeared transients, but better low frequency tonal separation.
source_type	Coarse Tunings for the Tonal / Transient separation. These tunings change how the focus control acts in guiding the separation decision across time and frequency of the audio. Choose from the following: 0: general, 1: kick, 2: snare, 3: tom, 4: cymbal, 5: full drum kit, 6: electronic beat, 7: hand drum, 8: percussion set, 9: bass, 10: piano, 11: guitar, 12: vocal.
focus	0 to 100 (tonal to transient). "Focuses" the sonic energy towards either the Tonal or Transient audio channel, with extreme settings pushing all the energy into either Transient (100 value) or Tonal (0 value). However the real separation magic occurs in the middle settings, where Focus sets the main transition region or decision point where audio splits (in time and frequency) into the separate Tonal and Transient streams. By way of example, pushing more energy into the Transient channel will create musical auto-swelling in the Tonal channel and give longer Transient tails. Alternatively, pushing more energy into the Tonal Channel will trim the Transient audio into staccato transients devoid of tonal resonance.

Header	Description
trans_decay	0 to 1 sec. The Transient Decay control limits how quickly audio is allowed to transitionv(in time and frequency) from Transient to Tonal, thus increasing the decay on the Transients. Larger values of Trans Decay limit the transition rate substantially. This control could equally be named Tonal Swell, as larger values will also increase the auto-swell period in the Tonal Channel.
trans_attack	0 to 1 sec. The Transient Attack control limits how quickly audio is allowed to transition (in time and frequency) from Tonal to Transient, thus increasing the attack time on the Transients. Larger values of Trans_attack limit the transition rate substantially, such most of the transients stay in the Tonal output channel. This control is most useful at small values to smooth out spurious artifacts.
split_off	0 or 1. Turns off split processing, at which point Focus simply becomes a mix control into the Tonal and Transient outputs. So, focus at 50 simply equally divides the signal into the two outputs.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out_tonal				
MONO				
AUDIOOUTPUT				
out_transient				
MONO				
AUDIOOUTPUT				
resolution_prefer	0	1	0	1
CONTROLINPUT	resolution preference			
source_type	0	12	0	1
			general,kick,snare,tom,cymbal,full kit,electronic beat,hand drum,percussion set,bass,piano,guitar,vocal	
CONTROLINPUT	source_type			
focus	0	100	50	1
CONTROLINPUT	focus			
trans_decay	0	1	0.1	0.001

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	sec			
CONTROLINPUT	transient decay			
trans_attack	0	1	0.005	0.001
	sec			
CONTROLINPUT	transient attack			
split_off	0	1	0	1
CONTROLINPUT	split process off			

unzipper - Parameter change smoother unz

Description

This module acts as an interface between two identical modules or sets of modules and their inputs and outputs. At any given time, one set of modules is active and the other is inactive. Should one of the control inputs change, the new value will be fed to the inactive set of modules which will become active and the audio outputs will be smoothly crossfaded from the old one to the new one. This reduces or eliminates the effect of "zipper" noise resulting from a parameter change, but of course doubles the number of modules required. This module is intended for advanced programmers who will understand its function.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
sina[n]	numsigs the audio inputs from one set of modules.
sinb[n]	numsigs the audio inputs from the other set of modules.

Audio Outputs

Header	Description
sout[n]	numsigs the audio inputs from the active set of modules.

Control Inputs

Header	Description
cin[n]	numsigs the control inputs to be monitored.

Control Outputs

Header	Description
couta[n]	numsigs the control inputs to be sent to one set of modules.
coutb[n]	numsigs the control inputs to be sent to the other set of modules.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
numsigs	1	8	1	1
	num_sigs			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
numctrl	1	32	1	1
INT	num_ctrl			
rate	1	1000	80	1
FLOAT	rate			
sina				
MONO		@numsigs		
AUDIOINPUT				
sinb				
MONO		@numsigs		
AUDIOINPUT				
sout				
MONO		@numsigs		
AUDIOOUTPUT				
cin	-32768	32767	0	
		@numctrl		
CONTROLINPUT				
couta				
		@numctrl		
CONTROLOUTPUT				
coutb				
		@numctrl		
CONTROLOUTPUT				

Chapter 12 - Mixer

audiomux(mux) - Audio Signal Crossbar Module

Description

The audiomux module allows you to create complex routings of audio signals. You can individually select which of n-inputs gets routed to each of m-outputs.

Godlike Productions Comments

Specifiers

Header	Description
moutputs	1 to 64 outputs. Specifies how many audio outputs are available to route to.
ninputs	1 to 64 inputs. Specifies how many input audio signals exist to be routed.

Audio Inputs

Header	Description
in(n)	The inputs to be routed.

Audio Outputs

Header	Description
out(m)	the routed outputs.

Control Inputs

Header	Description
select(m)	1 to ninputs. Control for each output selects the input channel number that gets routed to that output. One input can be routed to multiple outputs.

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	1	64	1	1
	number_inputs			
moutputs	1	64	1	1
INT	number_outputs			
in				
MONO		@ninputs		
AUDIOINPUT				
out				
MONO		@moutputs		
AUDIOOUTPUT				
select	1	@ninputs		
		@moutputs		
CONTROLINPUT				

crossin(mix) - Crossfades between two inputs

Description

A mod input selects one of two inputs. As the mod input goes from 0 to 1, the output is crossfaded betweeen input1 and input2.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in1	The first input two be mixed.
in2	The second input two be mixed.

Audio Outputs

Header	Description
out	The mix output.

Control Inputs

Header	Description

Mod Inputs

Header	Description
select	Controls which input is fed to the output. 0 selects input 1, 1 select input 2, -1 outputs -in2

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
in1				
LEFT				
in2				
RIGHT				
AUDIOINPUT				
select				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

crossout(crs) - Crossfades between two outputs

Description
A mod input selects which of two outputs to send an input to.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in1	The input to send.

Audio Outputs

Header	Description
out1	output when mod is 0
out2	output when mod is 1

Control Inputs

Header	Description

Mod Inputs

Header	Description
select	Controls which output is feed from the input. 0 selects output1, 1 selects output2, -1 selects output2 = -input

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
in				
select				
MOD				
AUDIOINPUT				
out1				
AUDIOOUTPUT				
out2				
AUDIOOUTPUT				

crossoq(crs) - Crossfades between four outputs

Description
A mod input selects which of two outputs to send an input to

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in1	The input to send.

Audio Outputs

Header	Description
out1	output when mod is 0
out2	output when mod is 1

Control Inputs

Header	Description

Mod Inputs

Header	Description
select	Controls which output is feed from the input. 0 selects output1, 1 selects output2, 1 selects output2 = -input

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
in				
frontrear				
MOD				
AUDIOINPUT				
leftright				
MOD				
AUDIOINPUT				
frontleft				
AUDIOOUTPUT				
frontright				
AUDIOOUTPUT				
rearleft				
AUDIOOUTPUT				
rearright				
AUDIOOUTPUT				

iswitch(isw) - Input switch

Description

This module allows up to 50 input signals to be switched to a single output signal without clicks.

Godlike Productions Comments

Specifiers

Header	Description
in[n]	0 to 1024 input signals to select from.

Audio Inputs

Header	Description
in[n]	The audio input to select from

Audio Outputs

Header	Description
out	Carries the selected input signal.

Control Inputs

Header	Description
select	0 to numouts. Tells to which output to switch the input to.

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
numins	2	1024	2	1
	num_outs			
select	0	@numins/1/#sub	0	1
CONTROLINPUT	select			
in				
MONO		@numins		
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				

mix(mix) - Two-Input Audio Mixer

Description

The mix module provides a way to add (mix) two audio signals, with adjustable attenuation on each of the two inputs. If you need to mix more than two inputs, use "mixer".

Godlike Productions Comments

Specifiers

Header	Description
time_constant	The time constant of the mixer.
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in1	The first input two be mixed.
in2	The second input two be mixed.

Audio Outputs

Header	Description
out	The mix output.

Control Inputs

Header	Description
amp1	-1.0 to 1.0. Controls the amount of attenuation for input 1. The gain adjustment is a LINEAR value (not dB), with 1.0 being no attenuation. Negative numbers will invert the phase of the signal.
amp2	-1.0 to 1.0. Attenuation adjust for in2.

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
time_constant	0	100	20	0.01
	ramp_time			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in1				
LEFT				
AUDIOINPUT				
in2				
RIGHT				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
amp1	-1	1	0.5	0.01
CONTROLINPUT	atten1			
amp2	-1	1	0.5	0.01
CONTROLINPUT	atten2			

mixer(mix) - Multi-Input Audio Mixer

Description

The mixer module provides a way to add (mix) two or more audio signals, with adjustable attenuation on each of input. The mixing is done with a dB control.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	1 to 50 inputs. Specifies how many audio signals are to be mixed.
time_constant	The time constant of the mixer.
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in(n)	The inputs to be mixed.

Audio Outputs

Header	Description
out	The mix output.

Control Inputs

Header	Description
gain(n)	-100.0 to 0.0. Controls the amount of attenuation for each input. The gain adjustment is a dB value. 0 dB = no attenuation, full level.

Mod Inputs

Header	Description

User Objects

Header	Description
obj	Userobject for connection to menupage.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	1	50	1	1
	number_inputs			
time_constant	0	100	20	0.01
	mSec			
FLOAT	ramp_time			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in				
MONO		@ninputs		
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
gain	-100	0	-3	0.1
	db	@ninputs		
CONTROLINPUT	gain~n			
obj				
USEROBJECTPARENT	obj			

mixn(mix) - Multi-Input Audio Mixer with linear level control

Description

The mixn module provides a way to add (mix) two or more audio signals, with adjustable attenuation on each of input. The mixing is done with a linear control.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	1 to 50 inputs. Specifies how many audio signals are to be mixed.
time_constant	The time constant of the mixer.
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in(n)	The inputs to be mixed.

Audio Outputs

Header	Description
out	The mix output.

Control Inputs

Header	Description
amp(n)	-1.0 to 1.0. Controls the amount of attenuation for each input. The gain adjustment is a linear value. 1/-1 = no attenuation, full level, with -1 being phase inverted.

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	1	50	1	1
	number_inputs			
time_constant	0	100	20	0.01
	mSec			
FLOAT	ramp_time			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in				
MONO		@ninputs		
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
amp	-1	1	0.5	0.0001
		@ninputs		
CONTROLINPUT	amp~n			
obj				
USEROBJECTPARENT	obj			

oswitch(osw) - Output switch

Description

This module allows a single input signal to be switched to either one of N output signals without clicks.

Godlike Productions Comments

Specifiers

Header	Description
numouts	0 to 1024 output signals to switch to.

Audio Inputs

Header	Description
in	The audio input to be switched.

Audio Outputs

Header	Description
out(n)	The n outputs.

Control Inputs

Header	Description
select	0 to numouts. Tells to which output to switch the input to.

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
numouts	2	1024	2	1
	num_outs			
in				
MONO				
AUDIOINPUT				
out				
MONO		@numouts		
AUDIOOUTPUT				
select	0	@numouts/1/#sub	0	1
CONTROLINPUT	Select			

quadmixr - Multi-Input Quad-output Mixer

Description

This module mixes a specified number of input channels into four output channels. Each input channel has its own gain, left/right pan, and front/rear pan.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	Specifies the number of audio inputs, from 1 to 50.
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in[inputs]	The channels of audio input.

Audio Outputs

Header	Description
frontleft	The front left output channel.
frontright	The front right output channel.
rearleft	The rear left output channel.
rearright	The rear right output channel.

Control Inputs

Header	Description
gain[ninputs]	The gain to apply to the corresponding input channel, from -100 to 0 dB.
lrpan[ninputs]	The left/right pan position to apply to the corresponding input channel, from -1 (full left) to 1 (full right).
frpan[ninputs]	The front/rear pan position to apply to the corresponding input channel, from -1 (full rear) to 1 (full front).

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	1	50	1	1
	number_inputs			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in				
MONO		@ninputs		
AUDIOINPUT				
frontleft				
LEFT				
AUDIOOUTPUT				
frontright				
RIGHT				
AUDIOOUTPUT				
rearleft				
LEFT				
AUDIOOUTPUT				
rearright				
RIGHT				
AUDIOOUTPUT				
gain	-100	0	-3	1
	db	@ninputs		
CONTROLINPUT	gain~n			
lrpan	-1	1	0	0.01
		@ninputs		
CONTROLINPUT	lrpan~n			
frpan	-1	1	0	0.01
		@ninputs		
CONTROLINPUT	frpan~n			

stereomixer(smx)- Multi-Input Stereo Audio Mixer

Description

The mix module provides a way to add (mix) two or more audio signals, with adjustable attenuation and pan on each input. The mixing is done with a dB control.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	2 to 50 inputs. Specifies how many audio signals are to be mixed.
time_constant	Smoothing time constant in mS
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
in(n)	The inputs to be mixed.

Audio Outputs

Header	Description
left	The left mix output.
right	The right mix output.

Control Inputs

Header	Description
gain(n)	-100.0 to 0.0. Controls the amount of attenuation for each input. The gain adjustment is a dB value. 0 dB = no attenuation, full level.
pan(n)	-1.0 to 1.0. Controls the left/right pan position for each of the inputs. Pan values of -1.0, 0.0, and 1.0 correspond to pan positions of left, center and right, respectively.

Mod Inputs

Header	Description

User Objects

Header	Description
obj	Userobject for connection to menupage.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	2	50	2	1
	number_inputs			
time_constant	0	100	20	0.01
	mSec			
FLOAT	ramp_time			
clipvalue	0	10000	10000	0.1
FLOAT	clip_value			
in				
MONO		@ninputs		
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				
AUDIOOUTPUT				
gain	-100	0	-3	1
	db	@ninputs		
CONTROLINPUT	gain~n			
pan	-1	1	0	0.1
		@ninputs		
CONTROLINPUT	pan~n			
processoff	0	1	0	1
CONTROLINPUT	turn_off			
obj				
USEROBJECTPARENT	obj			

stereopanner

Description
Applies selected pan law to a stereo input signal.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	2 to 50 inputs. Specifies how many audio signals are to be mixed.
time_constant	Smoothing time constant in mS
clipvalue	No documentation in the json file. Perhaps a level at which the block clips. Range is 0 to 10000.

Audio Inputs

Header	Description
left_in	The left input.
right_in	The right input.

Audio Outputs

Header	Description
left_out	The left output.
right_out	The right output.

Control Inputs

Header	Description
panmode	0 (left/right) or 1 (Mid/Side). Controls the signal mode over which the pan effect occurs. Mid/Side will only have an effect if the left and right inputs are different.
pan	-1 to 1. -1 is left, 0 Center, 1 is right for panmode 0 (L/R) -1 is mid, 0 Center, 1 is side for panmode 1 (M/S)
pan_width	-1 to 1. Controls how wide the pan mod input will swing. final pan = pan-width*panmod.
panlaw	The pan law the panning effect follows. Choices are: 0 - equal gain 1 - equal power 6dB 2 - equal power 4.5dB 3 - equal power 3dB
panlaw_comp	0 or 1, Off or On. Pan law compensation for the loss of gain when center panned.
pan_limit	-144 to 0dB, lowest gain value the selected pan law is allowed to apply.

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
left_in	-1.0	1.0		
LEFT				
AUDIOINPUT				
right_in	-1.0	1.0		
RIGHT				
AUDIOINPUT				
left_out				
LEFT				
AUDIOOUTPUT				
right_out				
RIGHT				
AUDIOOUTPUT				
panmode	0	1	0	1
CONTROLINPUT	panmode			
pan	-1.0	1.0	0	0.1
CONTROLINPUT	pan			
pan_width	-1.0	1.0	1.0	0.1
CONTROLINPUT	pan width			
panlaw	0	3	0	1
INTEGER				
CONTROLINPUT	panlaw			
panlaw_comp	0	1	1	0.001
CONTROLINPUT	compensation			
pan_limit	-144	0	-144	0.001
	dB			
CONTROLINPUT	panning limit			

stereotaps - Weighted Multi-Tap Filter

Description

This module filters its mono input channel through multiple taps of a delay line, each tap with its own delay length, amplitude, and stereo pan.

Godlike Productions Comments

Specifiers

Header	Description
taps	Specifies the number of taps from the delay line, from 2 to 50.

Audio Inputs

Header	Description
in	The mono stream of audio input.

Audio Outputs

Header	Description
left	The left mix output.
right	The right mix output.

Control Inputs

Header	Description
delayamt[taps]	The delay to apply to the corresponding tap, from 0 to 660 milliseconds.
amp[taps]	The linear weight to apply to the corresponding delayed tap, from -1 to 1.
pan[taps]	The left/right pan position to apply to the corresponding delayed tap, from -1 (full left) to 1 (full right).

Mod Inputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
taps	2	50	2	1
	number_taps			
in				
MONO				
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				
AUDIOOUTPUT				
delayamt	0	660	100	0.1
	ms	@taps		
CONTROLINPUT	delay~n			
amp	-1	1	1	0.01
		@taps		
CONTROLINPUT	amp~n			
pan	-1	1	1	0.1
		@taps		
CONTROLINPUT	pan~n			

surroundmixer(smx) - Multi-Input Multi-Output Surround Mixer

Description

The surround mixer mixes N inputs to M outputs based on the distance from each input to each output. This can be useful for creating reverbs an other things.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	Specifies the number of inputs.
moutputs	Specifies the number of outputs

Audio Inputs

Header	Description
in(n)	The inputs to be mixed.
inX(n)	The X dimension of the nth input.
inY(n)	The Y dimension of the nth input.
inZ(n)	The Z dimension of the nth input.
outX(m)	The X dimension of the mth output.
outY(m)	The Y dimension of the mth output.
outZ(m)	The Z dimension of the mth output.

Audio Outputs

Header	Description
out(m)	The mth mix output.

Control Inputs

Header	Description
size	The total size of the space the signals are mixed in, this controls the separation of the channels
pan	When pan is 0 this works as a mixer, when it's 100 it works as a panner

Mod Inputs

Header	Description

User Objects

Header	Description
obj	Userobject for connection to menupage.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	1	50	5	1
	number_inputs			
moutputs	1	50	5	1
INT	number_outputs			
in				
MONO		@ninputs		
AUDIOINPUT				
inX				
MONO		@ninputs		
AUDIOINPUT				
inY				
MONO		@ninputs		
AUDIOINPUT				
inZ				
MONO		@ninputs		
AUDIOINPUT				
outX				
MONO		@moutputs		
AUDIOINPUT				
outY				
MONO		@moutputs		
AUDIOINPUT				
outZ				
MONO		@moutputs		
AUDIOINPUT				
out				
MONO		@moutputs		
AUDIOOUTPUT				
size	0	100	50	0.1
	%			
CONTROLINPUT	size			
pan	0	100	0	0.1
	%			
CONTROLINPUT	pan			
obj				
USEROBJECTPARENT	obj			

surroundsystempositioner(ssp) - creates an interface to set up a surround speaker layout for the surroundmixer module

Description

The Surround System Positioner allows you to set up and save a speaker arrangement for your surround system and use that as a target in the surroundmixer module.

Godlike Productions Comments

Specifiers

Header	Description
moutputs	1 to 50 outputs. Specifies how many independent outputs are created.

Audio Inputs

Header	Description

Audio Outputs

Header	Description
outX(m)	The X dimension of the mth output.
outY(m)	The Y dimension of the mth output.
outZ(m)	The Z dimension of the mth output.

Control Inputs

Header	Description
angle(m)	The left/right angle of the mth speaker in degrees. 0 degrees is straight ahead, negative values move the speaker toward the left, positive values move the speaker toward the right.
elevation(m)	The up/down angle of the mth speaker in degrees. 0 degrees is straight ahead, negative values move the speaker down, positive values move the speaker up.

Mod Inputs

Header	Description

User Objects

Header	Description
obj	Userobject for connection to menupage.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
moutputs	1	50	5	1
	number_outputs			
outX				
MONO		@moutputs		
AUDIOOUTPUT				
outY				
MONO		@moutputs		
AUDIOOUTPUT				
outZ				
MONO		@moutputs		
AUDIOOUTPUT				
angle	-180	180	0	1
	degrees	@moutputs		
CONTROLINPUT	angle~n			
elevation	-90	90	0	1
	degrees	@moutputs		
CONTROLINPUT	elevation			
obj				
USEROBJECTPARENT	obj			

switch(swi) - Audio Signal Switch

Description

This module selects one of N audio inputs to be passed to the output.

Godlike Productions Comments

Specifiers

Header	Description
ninputs	2 to 1024. Specifies how many inputs the switch will select from.

Audio Inputs

Header	Description
in(n)	The audio inputs that will be switched.

Audio Outputs

Header	Description
out	The output of the switch.

Control Inputs

Header	Description
select	0 to ninputs-1. This controls which of the audio inputs will be passed along to the switch output. A value of zero

Mod Inputs

Header	Description

User Objects

Header	Description
obj	Userobject for connection to menupage.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
ninputs	2	100	2	1
	number_inputs			
select	0	@ninputs/1/#sub	0	1
CONTROLINPUT	%n			
in				
MONO		@ninputs		
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
obj				
USEROBJECTPARENT	obj			

Chapter 13 - Oscillator

adsr(adr) - Audio Signal ADSR

Description

This module is an ADSR envelope generator for audio signals. As an ADSR it has four states: attack, decay, sustain, release. The trigger input controls the ADSR. When the signal at the trigger input is above 0.5 the ADSR will go through the attack, decay, sustain sequence. Once the signal drops below 0.5 it will enter release. The envelope level is applied to the 'in' signal and sent to 'out'.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The audio signal to apply the envelope to.
trigger	The input which controls the ADSR.

Audio Outputs

Header	Description
out	The output signal.

Control Inputs

Header	Description
attacktime	0 to 10000 msec. This input controls how long the envelope level takes to go from zero to 1.0 during the attack phase.
decaylevel	0 to 1.0. This input controls the final envelope level when the ADSR completes the decay phase.
decaytime	0 to 10000 msec. This input controls how long the envelope level takes to go from 1.0 to the decay level during the decay phase.
retriggertime	0 to 10000 msec. This input controls how much time must pass from the start of one attack for another attack to be recognized.
sustainlevel	0 to 1.0. This input controls the final envelope level when the ADSR is in the sustain phase.
sustaintime	0 to 10000 msec. This input controls how long the envelope level takes to go from the decay level to the sustain level.
releasetime	0 to 10000 msec. This input controls how long the envelope level takes to go to zero after the trigger signal drops below 0.5.

Control Outputs

Header	Description

User Objects

Header	Description
obj	Userobject output

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
trigger				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
attacktime	0	10000	2	0.1
	msec			
CONTROLINPUT	attack			
decaytime	0	10000	5	0.1
	msec			
CONTROLINPUT	decay			
decaylevel	0	1	0.75	0.01
	%%			
CONTROLINPUT	dlevel *100			
sustaintime	0	10000	20	0.1
	msec			
CONTROLINPUT	sustain			
sustainlevel	0	1	0.5	0.01
	%%			
CONTROLINPUT	slevel *100			
releasetime	0	10000	25	0.1
	msec			

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
CONTROLINPUT	release			
retriggertime	0	10000	0	0.1
	msec			
CONTROLINPUT	retrigger			
obj				
USEROBJECTPARENT	obj			

analoglfos - Analog style LFO module

Description

This module gives the user access to multiple styles of analog lfos. These can be lfos found in specific products (i.e the Instant Phaser and Flanger) or any other analog style oscillator. This works by providing a mode switch, and a phase increment that is available to all of the modes. What each oscillator mode does with that phase increment is entirely up to it. For accuracy, the phase increment is calculated using kahan summation, which keeps track of the inherent errors in floating point math.

Godlike Productions Comments

Specifiers

Header	Description
increment_mode	Selects whether or not the modules phase counter is externally or internally driven. If externally driven, the audio input MUST be a phase counter that goes from 0 to 1 at the desired rate. 0. Internally driven 1. Externally driven
lfo_mode	selects the oscillator to use... 0. The Instant Flanger LFO 1. The Instant Phaser LFO

Audio Inputs

Header	Description
in	The phase counter input if using the module in the externally driven mode. NOTE: This input is expected to be a ramp form 0 to 1 at the desired LFO rate. Furthermore, certain lfo modes need to know the rate of the LFO so whatever is setting the rate of the external phase counter must set the rate in this module as well.
retrigger	A signal to retrigger the oscillator and any associated states of filters, etc. A value of > 0.5 will cause the LFO to reset to its initial state.

Audio Outputs

Header	Description
out	The output of the LFO

Control Inputs

Header	Description
dutycycle	The duty cycle of the LFO. NOTE: The phaser and flanger LFOs were designed to work specifically at one duty cycle value (that of the actual hardware units). Unfortunately, becuase of the precise modeling that went in to these they may not work at other duty cycle values so when using these two LFOs they should be set accordingly: Phaser - 0.7 Flanger - 0.44
rate	The rate of the LFO in Hz.

Header Description

Control Outputs

Header Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
increment_mode	0	1	0	1
lfo_mode	0	1	0	1
INT				
in				
MONO				
AUDIOINPUT				
retrigger				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
rate	0.01	20000	1000	0.01
CONTROLINPUT	rate			
dutycycle	0.01	1	0.5	0.01
CONTROLINPUT	dutycycle			

envelope(env) - Envelope Generator

Description

The envelope generator module will produce a single cycle of a specified waveshape after being triggered by an audio input signal. This is useful in creating various triggered effects from delay sweeps to sound synthesizers. Typically the output of the envelope generator will be connected to the modulation input of a filter, delay, amplitude modulator, etc.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
mod	Modulates the frequency (length) of the envelope
trigger	Triggers the production of the envelope output.

Audio Outputs

Header	Description
out	The envelope generator output.

Control Inputs

Header	Description
dutycycle	0 to 1.0 Controls the duty cycle of the generated envelope. This control does not affect the sine wave. A setting of 0.5 will produce a 50 percent duty cycle, i.e. the waveform will be symetrical.
freq	0 to 1000 Hertz. Controls the rate of the envelope generator. Since the envelope generator output is essentially a single cycle of the selected waveform, this directly controls the length of the envelope. A setting of 1000 Hertz will produce an envelope that is 1 milliseconds long. A setting of 1 Hertz will produce an envelope that is 1 second long.
hysteresis	0 to 20 dB. Controls how much the input must drop below the trigger level before a new trigger will be allowed. This is used to prevent spurious triggering of the envelope.
modamt	-1000 to 1000 Hertz. This controls how much the mod input affects the rate of the envelope generator.
speed	0.000 to 10.000 seconds. Controls the trigger sensitivity. The setting is an averaging time for the peak detector that feeds the trigger mechanism.
wave	0 to 9. Selects the waveshape to be used. The values are as follows: 0 - Sine Shape, 1 - Triangle, 2 - Square, 3 - peak, 4 - half sin, 5 - warp sin, 6 - full sin, 7 - full triangle, 8 - full square, 9 - full peak

Header	Description
togglemode	Controls the mode of operation of the envelope generator. The settings are as follows: 0 - One-Shot Mode. Resets waveform to beginning after each trigger. 1 - Toggle Mode, On alternate triggers will scan through wave forward then backward.
thresh	-100 to 0dB. Set the threshold at which the envelope will be triggered.

Control Outputs

Header	Description

User Objects

Header	Description
obj	Userobject output

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
trigger				
MOD				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	0	1000	0.1	0.001
	hz			
CONTROLINPUT	freq			
modamt	-1000	1000	0.1	0.001
	hz			
CONTROLINPUT	mod			
wave	0	10	0	1
			sine,triangle,square,peak,half_sin,warp_sin,full_sin,full_tri,full_sqr,-full_peak	
CONTROLINPUT	wave			
dutycycle	0	1	0.5	0.01

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	%%			
CONTROLINPUT	dutycycle			
togglemode	0	1		
CONTROLINPUT	mode			
thresh	-100	0	-40	1
	db			
CONTROLINPUT	threshold			
hysteresis	0	20	3	1
	db			
CONTROLINPUT	hysteresis			
speed	0.0001	10	3	0.0001
	s			
CONTROLINPUT	trig			
obj				
USEROBJECTPARENT	obj			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

H3000OSC(hos) - H3000 Audio Oscillator

Description

The H3000 Oscillator mimics the oscillators found in the H3000. The module produces a waveform of variable shape and frequency. The waveform is produced at the audio sample rate. If you are using this to slowly modulate a parameter, the more efficient LFO may work just as well. This oscillator works at audio rate in order to duplicate all H3000 functionality, including modfactory. This enables it to be used as either an oscillator or an LFO.

Godlike Productions Comments

Waveforms 6-19 output one or half cycle when triggered. They are not free running, like waveforms 1-5. Triggering from an inpulse does not work and allow triggering of the entire waveform. A squarewave must be used. Oscillator Block set to waveform 2 works fine if the frequencies are the same, though if retriggering is required, you will require another H3000 Oscillator set to waveform 2. Duty cycle does not slew, it controls width of top and bottom half of waveforms. Can be used for LFO's if audiorate is important.

Specifiers

Header	Description

Audio Inputs

Header	Description
mod	Modulates the frequency of the oscillator.
trigger	Either triggers or retriggers the waveform.

Audio Outputs

Header	Description
out	The oscillator output. level is at +20dBm

Control Inputs

Header	Description
dutycycle	0 to 1.0. Controls the duty cycle of the oscillator. This control does not affect the sine wave. A setting of 0.5 will produce a 50 percent duty cycle, i.e. the waveform will be symmetrical.
freq	0 to 20000 Hertz. Controls the rate of the oscillator.
modamt	-20000 to 20000 Hertz. This controls how much the mod input affects the rate of the oscillator.
wave	0 to 22. Selects the waveshape to be used. The values are as follows: 0 -Sine, 1 - Triangle, 2 - Square, 3 - Peak, 4 - Ramp, 5 - Exponential Ramp, 6 - Lenear Ramp, 7 - Decaying Ramp, 8 - Whole Sine, 9 - Whole Triangle, 10 - Whole Square, 11 - Whole Peak, 12 - Whole Ramp, 13 - Whole Exponential Ramp, 14 - Half Sine, 15 - Half Triangle, 16 - Half Square, 17 - Half Peak, 18 - Toggle Ramp, 19 - Exponential Toggle Ramp, 20 - Random, 21 - Sample and Hold, 22 - Triggered Sample and Hold
polarity	0 for unipolar and 1 for bipolar

Header	Description

Control Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
trigger				
MOD				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	0	32000/20000/#ifhi	1000	0.1
	hz			
CONTROLINPUT	freq			
modamt	-32000/-20000/#ifhi	32000/20000/#ifhi	1000	0.1
	hz			
CONTROLINPUT	mod			
wave	0			
			sine,triangle,square,peak,ramp,-exp_ramp,linear_ramp,decaying_ramp,whole_sine,whole_triangle,whole_square,whole_peak,whole_ramp,whole_exp_ramp,half_sine,half_triangle,half_square,half_peak,toggle_ramp,exp_toggle_ramp,random,samphold,triggered_samphold	

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	wave			
dutycycle	0	1	0.5	0.01
	%%			
CONTROLINPUT	dutycycle			
phaseoffset	-180	180	0	0.1
	degrees			
CONTROLINPUT	phaseoffset			
polarity				
CONTROLINPUT	polarity			

H3000OSCLITE(hos) - H3000 Audio Oscillator

Description

The H3000 Oscillator mimics the osciallators found in the H3000. This module is a super cheap version that creates sine and random waveforms only.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description

Audio Outputs

Header	Description
out	The oscillator output. level is at +20dBm

Control Inputs

Header	Description
freq	0 to 20000 Hertz. Controls the rate of the oscillator.
wave	0 to 1. Selects the waveshape to be used. The values are as follows: 0 - Sine, 1 - Random.

Control Outputs

Header	Description

User Objects

Header **Description**

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
out				
MONO				
freq	0	32000/20000/#ifhi	1000	0.1
	hz			
CONTROLINPUT	freq			
wave	0			
CONTROLINPUT	wave			

impulse(imp) - Impulse Generator

Description

The impulse generator module creates a pulse train a variable frequency.The width of the pulses is a single audio sample. This module is useful in testing reverb patches.By patching the output of the impulse generator into the reverb input, the character of the reverb can be more easily assessed.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
mod	A modulation input that affects the frequency of the impulse generator.

Audio Outputs

Header	Description
out	The impulse generator output.

Control Inputs

Header	Description
freq	0 to 20000 Hertz. Controls the frequency of the pulse train.For example, a setting of 10 Hertz will produce 10 pulses per second.
freqmod	0 to 20000 Hertz. Controls the rate of the impulse generator.

Control Outputs

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mod				
MOD				
out				
MONO				
AUDIOOUTPUT				
freq	0	20000	0.1	0.001
	Hz			
CONTROLINPUT	freq			
freqmod	0	20000	0.1	0.001
	Hz			
CONTROLINPUT	freq_mod			

lfo(lfo) - Low Frequency Oscillator

Description

The lfo module will produce a waveform of variable shape and frequency. The waveform is produced at the "mod" rate, 1/4 the audio sample rate. See also oscillator. Typically the output of the lfo will be connected to the modulation input of a filter, delay, amplitude modulator, etc. It is useful in creating flangers, chorus effects, autopanning, etc.

Godlike Productions Comments

LFO's operate at 1/4 audio frequency. Note that the signal is not held. Intermediate samples are 0, and cannot be filtered with the comparatators.

Specifiers

Header	Description

Audio Inputs

Header	Description
mod	Modulates the frequency of the LFO.

Audio Outputs

Header	Description
out	The LFO generator output.

Control Inputs

Header	Description
dutycycle	0 to 1.0 Controls the duty cycle of the LFO. This control does not affect the sine wave. A setting of 0.5 will produce a 50 percent duty cycle, i.e.the waveform will be symmetrical.
freq	0 to 1000 Hertz. Controls the rate of the LFO.
modamt	-1000 to 1000 Hertz. This controls how much the mod input affects the rate of the LFO.
wave	0 to 7. Selects the waveshape to be used. The values are as follows: 0 - Sine Shape, 1 - Triangle, 2 - Square, 3 - Peak, 4 - Warp Sin, 5 - Warp Tri, 6 - Half Sin, 7 - Half Peak
polarity	Controls whether the output is unipolar (only produces positive values) or bipolar (produces both positive and negative values). 0 = Unipolar, 1 = Bipolar

Control Outputs

Header	Description

User Objects

Header	Description
obj	Userobject output

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mod				
MOD				
out				
MOD				
AUDIOOUTPUT				
freq	0	1000	0.1	0.001
	hz			
CONTROLINPUT	freq			
modamt	-1000	1000	0.1	0.001
	hz			
CONTROLINPUT	mod			
wave				
			sine,triangle,square,peak,warp_sin,warp_tri,half_sin,half_peak	
CONTROLINPUT	wave			
dutycycle	0	1	0.5	0.01
	%%			
CONTROLINPUT	dutycycle *100			
polarity				
CONTROLINPUT	polarity			
obj				
USEROBJECTPARENT	obj			

lfo2 - Retriggerable Low-Frequency Oscillator

Description

The LFO-2 module produces a waveform for use as a modulation signal. The waveform is updated once every four samples. The principal difference between this lfo2 and the other lfo is that it is capable of being reset by two audio signals representing the reset signal and the phase location to reset to.

Godlike Productions Comments

LFO's operate at 1/4 audio frequency. Note that the signal is not held. Intermediate samples are 0, and cannot be filtered with the comparatators.

Specifiers

Header	Description

Audio Inputs

Header	Description
mod	Control signal for modulating the frequency of the LFO signal.
trig	Control signal for resetting the LFO phase, level triggered on transition from zero to one.
trigval	Control signal indicating the phase point to reset to, from -1.0 to 1.0 representing -180 degrees to +180 degrees.

Audio Outputs

Header	Description
out	The LFO output.

Control Inputs

Header	Description
dutycycle	The duty cycle of the LFO, from 0 to 1.
freq	The LFO frequency, from 0 to 1000 Hertz.
modamt	The extent that the mod signal offsets the LFO frequency, from -1000 to 1000 Hertz.
wave	The waveform of the LFO output, enumerated as {sine, triangle, square, peak, warped sine, warped triangle, half sine, half peak }
polarity	The range of the LFO output, zero for the range [0..1] and one for the range [-1..1].

Control Outputs

Header	Description

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mod				
MONO				
trig				
MONO				
AUDIOINPUT				
trigval				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	0	1000	0.1	0.001
	Hz			
CONTROLINPUT	frequency			
modamt	-1000	1000	0.1	0.001
	Hz			
CONTROLINPUT	mod amount			
wave	0	7	0	1
CONTROLINPUT	waveform			
dutycycle	0	1	0.5	0.01
CONTROLINPUT	duty cycle			
polarity	0	1	1	0.01
CONTROLINPUT	polarity			

noise(noi) - Noise Generator

Description

This code generator produces an audio output signal that is pseudo random white noise. Noise Generating Algorithm: x[n+1] = { x[n] + 2*y[n] } mod 2^24 (always odd); y[n+1] = { y[n] + x[n] } mod 2^24 or; y[n+1] = { 2*y[n] + y[n-1] } mod 2^24 (alternates odd, even)

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description

Audio Outputs

Header	Description
out	The output of the noise generator.

Control Inputs

Header	Description

Control Outputs

Header	Description

User Objects

Header Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
out				
MONO				

oscillator(osc) - Audio Oscillator

Description

The oscillator module produces a waveform of variable shape and frequency. The waveform is produced at the audio sample. If you are using this to slowly modulate a parameter, the more efficient LFO may work just as well. Typically the oscillator will be used to generate an audio range waveform. It is useful in creating synthesis effects and for audio range modulations.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
mod	Modulates the frequency of the oscillator.

Audio Outputs

Header	Description
out	The oscillator output. level is at +20dBm

Control Inputs

Header	Description
dutycycle	0 to 1.0 Controls the duty cycle of the oscillator. This control does not affect the sine wave. A setting of 0.5 will produce a 50 percent duty cycle, i.e.the waveform will be symmetrical.
freq	0 to 20000 Hertz. Controls the rate of the oscillator.
modamt	-20000 to 20000 Hertz. This controls how much the mod input affects the rate of the oscillator.
wave	0 to 2. Selects the waveshape to be used. The values are as follows: 0 - Sine Shape, 1 - Triangle, 2 - Square.

Control Outputs

Header	Description

Header	Description

User Objects

Header	Description
obj	Userobject output

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mod				
MOD				
out				
MONO				
AUDIOOUTPUT				
freq	0	32000/20000/#ifhi	1000	0.1
	hz			
CONTROLINPUT	freq			
modamt	-32000/-20000/#ifhi	32000/20000/#ifhi	1000	0.1
	hz			
CONTROLINPUT	mod			
wave				
			sine,triangle,square	
CONTROLINPUT	wave			
dutycycle	0	1	0.5	0.01
	%%			
CONTROLINPUT	dutycycle *100			
obj				
USEROBJECTPARENT	obj			

phasor - Audio-rate Phasor Oscillator with Triangle Waveshapping

Description

The phasor module will produce an audio-rate positive phasor. This phasor is then shaped based on the value of the duty cycle, allowing you to change the slope from positive to negative and any triangle shape in between. A duty-cycle of 0 produces a negative ramp. A duty-cycle of 1 produces a positive ramp. A duty-cycle of 0.5 produces a symmetric triangle wave. You can also specify the number of outputs and the phase offset, duty cycle, and polarity for each output. Allowing you generate many different phasors that are synced together.

Godlike Productions Comments

Fixed phase offset documentation. The range is wrong, this can rotate a complete circle. Polarity and phase-offset were not imported from .json file. This block can generate sawtooth waveform with slew modulation. Phase can be modulated in real time, so potential use as audio rate oscillator bank (up to 1000 Hz)

Specifiers

Header	Description
noutputs	Specifies the number of outputs
min_freq	Sets the minimum frequency.
max_freq	Sets the maximum frequency.

Audio Inputs

Header	Description
in	Controls the frequency of the LFO. Set to some between 0 and 1 to get full range of min to max frequency.

Audio Outputs

Header	Description
out	The generated oscillator output. Can be more than one.

Control Inputs

Header	Description
dutycycle	Value 0 to 1 that controls the triangle waveshaping.
polarity	Controls whether the output is unipolar (wholly positive) or bipolar (full range -1 -> 1). 0: Unipolar 1: Bipolar
phase_offset	Sets the phase offset of an output phase. Has a range of 0? to 360?

Control Outputs

Header	Description

Header	Description

User Objects

Header	Description
obj	Userobject output

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
noutputs	1	16	1	1
	number_outputs			
in				
MOD				
AUDIOINPUT				
out				
MONO		@noutputs		
AUDIOOUTPUT				
polarity	0	1	0	1
		@noutputs		
CONTROLINPUT	polarity~n			
dutycycle	0	1	0.5	0.001
		@noutputs		
CONTROLINPUT	dutycycle~n			
min_freq	0	1000	0	1
FLOAT	minimum_frequency			
max_freq	0	1000	1000	1
FLOAT	maximum_frequency			
phase_offset	0	360	0	0.1
	degrees	@noutputs		
CONTROLINPUT	phase_offset~n			
obj				
USEROBJECTPARENT	obj			

rlfo - Retriggerable Low-Frequency Oscillator

Description

The retriggerable LFO module produces a waveform for use as a modulation signal. The waveform is updated once every four samples. The principal difference between this module and the other LFOs is that it is capable of being reset to an arbitrary point, defined by two audio signals representing the reset signal and the phase location to reset to.

Godlike Productions Comments

LFO's operate at 1/4 audio frequency. Note that the signal is not held. Intermediate samples are 0, and cannot be filtered with the comparatators.

Specifiers

Header	Description

Audio Inputs

Header	Description
mod	Control signal for modulating the frequency of the LFO signal.
trig	Control signal for resetting the LFO phase, level triggered on transition from zero to one.
trigval	Control signal indicating the phase point to reset to, from -1.0 to 1.0 representing -180 degrees to +180 degrees.

Audio Outputs

Header	Description
out	The LFO output.

Control Inputs

Header	Description
dutycycle	The duty cycle of the LFO, from 0 to 1.
freq	The LFO frequency, from 0 to 1000 Hertz.
modamt	The extent that the mod signal offsets the LFO frequency, from -1000 to 1000 Hertz.
wave	The waveform of the LFO output, enumerated as {sine, triangle, square, peak, warped sine, warped triangle, half sine, half peak }
polarity	The range of the LFO output, zero for the range [0..1] and one for the range [-1..1].
trigslew	The retrigger transition time as a fraction of the LFO period, from 0 to 1000 percent.

Control Outputs

Header	Description

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mod				
MOD				
trig				
MOD				
AUDIOINPUT				
trigval				
MOD				
AUDIOINPUT				
out				
MOD				
AUDIOOUTPUT				
freq	0	1000	0.1	0.001
	Hz			
CONTROLINPUT	frequency			
modamt	-1000	1000	0.1	0.001
	Hz			
CONTROLINPUT	mod amount			
wave	0	7	0	1
CONTROLINPUT	waveform			
dutycycle	0	1	0.5	0.01
CONTROLINPUT	duty cycle			
polarity	0	1	1	1
CONTROLINPUT	polarity			
trigslew	0	1000	10	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	pct			
CONTROLINPUT	trig slew			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

slfo - Synchronized Low-Frequency Oscillator

Description

The synchronized LFO module produces a waveform for use as a modulation signal. The waveform is updated once every four samples. The principal difference between this and the other LFOs is that multiple instances can be configured so that one instance can transmit its phase state to other instances in a master/slave relationship, so that the group of instances are synchronized.

Godlike Productions Comments

LFO's operate at 1/4 audio frequency. Note that the signal is not held. Intermediate samples are 0, and cannot be filtered with the comparatators.

Specifiers

Header	Description

Audio Inputs

Header	Description
smod	Control signal for setting the LFO phase, used in slave mode. In master mode, this is a frequency modulation signal.
trig	Control signal for resetting the LFO phase, level triggered on transition from zero to one.

Audio Outputs

Header	Description
out	The LFO output.
sync	The control signal output for connection to smod of slave instances of slfo.

Control Inputs

Header	Description
dutycycle	The duty cycle of the LFO, from 0 to 1.
freq	The LFO frequency, from 0 to 1000 Hertz.
modamt	The extent that the mod signal offsets the LFO frequency, from -1000 to 1000 Hertz.
wave	The waveform of the LFO output, enumerated as {sine, triangle, square, peak, warped sine, warped triangle, half sine, half peak }
polarity	The range of the LFO output, zero for the range [0..1] and one for the range [-1..1].
trigslew	The retrigger transition time as a fraction of the LFO period, from 0 to 1000 percent.
mode	The master/slave mode switch, 0/1 respectively.
phase	The output phase offset, from -180 to 180 degrees.

Control Outputs

Header	Description

Header	Description

User Objects

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
smod				
MOD				
trig				
MOD				
AUDIOINPUT				
out				
MOD				
AUDIOOUTPUT				
sync				
MOD				
AUDIOOUTPUT				
freq	0	1000	0.1	0.001
	Hz			
CONTROLINPUT	frequency			
modamt	-1000	1000	0.1	0.001
	Hz			
CONTROLINPUT	mod amount			
wave	0	7	0	1
CONTROLINPUT	waveform			
dutycycle	0	1	0.5	0.01
CONTROLINPUT	duty cycle			
polarity	0	1	1	1
CONTROLINPUT	polarity			
trigslew	0	1000	10	1

Name Type (A) Type	Min Units Description	Max Size	Default Choices	Resolution
	pct			
CONTROLINPUT	trig slew			
mode	0	1	0	1
CONTROLINPUT	master/slave			
phase	-180	180	0	0.1
	deg			
CONTROLINPUT	phase			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

waveform - Programmable Waveform Oscillator

Description

This module implements a phase counter that circulates through a 32-point wavetable that is defined at launch time. Linear interpolation is applied to the wavetable rendering. A controller drives the waveform frequency. The input signal drives frequency modulation. The wavetable is defined through an array of floating- point values with the external name point[32].

Godlike Productions Comments

Specifiers

Header	Description
point[n]	The value for the waveform point. Range [-1 .. 1].

Audio Inputs

Header	Description
mod	The frequency modulation signal.

Audio Outputs

Header	Description
out	The oscillator output.

Control Inputs

Header	Description
freq	The frequency to circulate through the wavetable, from 0 to 20000 Hz.
modamt	The extent to which a full scale (1.0) value of mod will modulate the effective frequency, from -20000 to 20000 Hz.

Control Outputs

Header	Description

User Objects

Header	Description
obj	Userobject output

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
mod				
MOD				
out				
MONO				
AUDIOOUTPUT				
freq	0	20000	1000	0.1
	Hz			
CONTROLINPUT	%n			
modamt	-20000	20000	1000	0.1
	Hz			
CONTROLINPUT	%n			
point	-1	1	0	0.001
		32		
FLOAT	point~n			
obj				
USEROBJECTPARENT	obj			

Chapter 14 - Pitch Shift

detune(dly) - Audio Signal Detuner

Description

The detune module is used to add small amounts of pitch shift to an audio signal. This module is is intended as an efficient (i.e. it doesn't use a lot of processing time) method of detuning a signal. In order to accomplish this, this module has traded off deglitch quality (glitches are a common artifact of pitch shifters) for processing efficiency. If totally glitch-free audio is needed, the pitch-shift module should be used instead of the detune module.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	1 to 660 milliseconds. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The audio input to be detuned.

Audio Outputs

Header	Description
out	A detuned version of the input signal.

Control Inputs

Header	Description
delay_ctl	0 to maxdelay milliseconds. Controls how much the audio will be delayed.
length_ctl	1 to maxdelay milliseconds. Controls the splice length for the detuning algorithm. Longer settings of this parameter will provide less "glitches" but will add more delay to the signal. Smaller settings may cause more glithes but will give a tighter sound (i.e. smaller delay). Extreme small settings may introduce modulation effects into the audio. Note also that the delay introduced as a result of this control will be variable, that is the audio delay will be continually changing from 0 thru the amount set by the length control.
pitch_ctl	-100 to 100 cents. Controls the amount of pitch detuning to be applied to the audio input. The adjustment is in "cents". A cent is one one-hundredth of a semitone. Positive values will shift the pitch upward and negative values will shift it downward.

Header	Description

Control Outputs

Header	Description

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	660	20	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delay_ctl	0	@maxdelay	20	1
	ms			
CONTROLINPUT	delay			
length_ctl	1	@maxdelay	20	1
	ms			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	length			
pitch_ctl	-100	100	15	1
	cents			
CONTROLINPUT	pitch			
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

diatonic(dsh) - Diatonic Pitch Shifter

Description

The diatonic pitch shift module shifts the musical pitch of an audio signal while maintaing the proper harmonic relationship to a diatonic scale. To accomplish this, the user specifies the desired key and the desired musical interval. The pitch shifter takes care of finding out what note is being played and automatically adjusts the amount of pitch shift so that the resultant note is in key. This pitch shifter can have from 1 to 12 independant pitch shift outputs. This can be useful for creating anything from two to five-part harmonies. The pitch shifter also has a built-in pitch detector whose results are made available through various control outputs.

Godlike Productions Comments

Specifiers

Header	Description
lowlatencymode	On Factor pedals, H9, and Eclipse the pitchshift module used a smaller splice buffer than on the H8k. This had two main effects - 1) A lower minimum delay value (@44.1k ~3ms H9 vs ~6ms H8K) and 2) A higher low note value for the pitch tracking (minpitch of ~9 for the H9 version and ~7 for the H8K). Another possible side effect would be slightly more artifacts in the lower latency version since a smaller splice buffer means the module xfades/splices more often. How noticeable these would be would be very situationally dependent. 0 - Higher latency, better low note tracking H8K version (default) 1 - Lower latency H9, Factor, and Eclipse version.
nvoices	1 to 12. This specifies how many independant output or "voices" this module will have.
maxdelay	500 to 32500 milliseconds. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The audio input to be pitch shifted.
mod(n)	This audio input will modulate the amount of pitch shift for a particular voice. This is useful to create vibrato effects. There is one mod input for each pitch shift voice.

Audio Outputs

Header	Description
out(n)	This is the output of the pitch shifter. There is one output per pitch shift voice.

Control Inputs

Header	Description
delayamt(n)	0 to 2000 milliseconds. These control the amount of delay for each pitch shift voice.

Header	Description
gatelevel	-100 to 0 dB. This control affects only the pitch detection output of this module. The gatelevel control determines at what level the pitch detector will output pitch readings. If the input signal level falls below the level set here, the pitch detect outputs will latch on to the old values.
glide	Controls the "glide" rate of the pitch adjustments. The adjustment is in seconds and controls the time constant that is used to smooth out changes in the amount of pitch shift that may come from changes in the interval.
key	This specifies the key the user will be playing in. The values are as follows: 0 - C, 1 - C#, 2 - D, 3 - D#, 4 - E, 5 - F, 6 - F#, 7 - G, 8 - G#, 9 - A, 10 - A#, 11 - B
minpitch	0 to 47. The minpitch control is used to optimize the pitch shifting algorithm. It sets the minimum pitch that pitch shifter is likely to detect. The values are as follows: 0 - C0, 1 - C0#, 2 - D0, ..., ..., 46 - A3#, 47 - B3
modamt(n)	-2400 to 2400 cents. These control the amount of modulation to be applied to each pitch shift voice. adjustment is in cents and it represents the amount of pitch shift that would be added to each voice if the mod input was fully on.
quantization	-2 to 2. Controls whether the output pitch is quantized to remain exactly within key or whether it simply track the input pitch. A value of zero corresponds to no quantizing. A value of 1 corresponds to full quantizing.
scale	This control input selects the type of scale of "mode" the user will be playing in. The scales are as follows: 0 - Ionian (Root Major), 1 - Dorian, 2 - Phrygian, 3 - Lydian, 4 - Mixolydian, 5 - Aeolian (Relative Minor), 6 - Locrian, 7 - Major, 8 - Minor, 9 - Harmonic Minor, 10 - Melodic Minor, 11 - Chromatic, 12 - Whole-tone, 13 - Pentatonic Major, 14 - Pentatonic Minor, 15 - Enigmatic, 16 - Neapolitan, 17 - Hungarian
shift(n)	-21 to 21. These control signals adjust the pitch shift interval for each voice. The values are as follows: -21 -3 octaves, -20 -21st, ..., ..., -7 octave down, -6 seventh down, -5 sixth down, -4 fifth down, -3 fourth down, -2 third down, -1 second down, 0 unison, 1 second up, 2 third, 3 fourth, 4 fifth, 5 sixth, 6 seventh, 7 octave up, 8 ninth, ..., ..., 21 3 octaves up
tune	-1200 to 1200 cents. This control allows the diatonic shift to be tuned to a pitch reference other than A440 Hertz. The tuning is adjusted in cents referenced to A440.
xfadetime	0 to 100 milliseconds. This control signal is used to optimize the sound of the pitch shifters. Larger settings may result in smoother overall sound but may add a "flanged" sound to the audio. Smaller settings will result in a crisper sound but may allow more audible pitch shifting artifacts.
userscalen_note	This control selects the notes of a custom scale. The notes must be in ascending order and can only cover an octive in range from the lowest note to the highest note with no gaps. The scale must have at least five notes. If the scale does not follow these rules it will be ignored and a major scale used instead. If the scale has seven or less notes then the intervals will be derived from the scale. If it has more than seven notes a default major scale interval pattern will be used.

Header	Description

Control Outputs

Header	Description
amp	amp' is the r.m.s. amplitude relative to full scale ('amp' equal to 1 would be a square wave 'hitting the rail').
freq	The output of the pitch detector given as a frequency in Hertz.
period	The output of the pitch detector given as a period. The value is in milliseconds.
pitch	The output of the pitch detector given in cents relative to middle C.
timbre	is a measurement of the brightness of the tone independent of its pitch. A sine wave has a 'timbre' equal to 1, other wave shapes result in a higher 'timbre'.
tonality	A value representing how periodic the input signal is. A value a 1.0 is given for signals which are purely periodic. Lower values represent signals that are less periodic. The smallest value would be given for very noise-like signals.

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
lowlatencymode	0	1	0	1
	lower latency			
maxdelay	500	22000/32500/#speed/#if	2000	1
	ms			
INT	max_delay			
nvoices	1	12	1	1
INT	number_voices			
in				
MONO				
AUDIOINPUT				
scale	0	17	0	1
			Ionian_(Maj),Dorian,Phrygian,Lydian,-MixoLydian,Aeolian_(Min),Locrian,-Major,Minor,Harmonic_Min,Melodic_Min,Chromatic,Whole-tone,Penta_Maj,Penta_Min,Enigmatic,Neapolitan,Hungarian	

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	scale			
key	0	11	0	1
			C,C#,D,D#,E,F,F#,G,G#,A,A#,B	
CONTROLINPUT	key			
quantization	-2	2	0	1
CONTROLINPUT	quantize			
glide	0	100	0	0.1
	sec			
CONTROLINPUT	glide			
tune	-1200	1200	0	1
	cents			
CONTROLINPUT	tune			
minpitch	0	47	21	1
			c0,c#0,d0,d#0,e0,f0,f#0,g0,g#0,a0,a#0,b0,c1,c#1,d1,d#1,e1,f1,f#1,g1,g#1,a1,a#1,b1,c2,c#2,d2,d#2,e2,f2,f#2,g2,g#2,a2,a#2,b2,c3,c#3,d3,d#3,e3,f3,f#3,g3,g#3,a3,a#3,b3	
CONTROLINPUT	minpitch			
gatelevel	-100	0	-60	1
	db			
CONTROLINPUT	gate			
xfadetime	0	100	10	1
	ms			
CONTROLINPUT	xfadetime			
userscale1_c	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_c			
userscale1_db	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_db			
userscale1_d	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_d			
userscale1_eb	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_eb			
userscale1_e	-2400	2400	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	cents			
CONTROLINPUT	userscale1_e			
userscale1_f	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_f			
userscale1_gb	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_gb			
userscale1_g	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_g			
userscale1_ab	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_ab			
userscale1_a	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_a			
userscale1_bb	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_bb			
userscale1_b	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale1_b			
userscale2_c	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_c			
userscale2_db	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_db			
userscale2_d	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_d			
userscale2_eb	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_eb			
userscale2_e	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_e			
userscale2_f	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_f			
userscale2_gb	-2400	2400	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	cents			
CONTROLINPUT	userscale2_gb			
userscale2_g	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_g			
userscale2_ab	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_ab			
userscale2_a	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_a			
userscale2_bb	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_bb			
userscale2_b	-2400	2400	0	1
	cents			
CONTROLINPUT	userscale2_b			
mod				
MOD		@nvoices		
AUDIOINPUT				
out				
MONO		@nvoices		
AUDIOOUTPUT				
shift	-21	23	21	1
		@nvoices	-3_oct,-21st,-20th,-19th,-18th,-17th,-16th,-2_oct,-14th,-13th,-12th,-11th,-10th,-9th,-1_oct,-7th,-6th,-5th,-4th,-3rd,-2nd,unison,+2nd,+3rd,+4th,+5th,+6th,+7th,+1_oct,+9th,+10th,+11th,+12th,+13th,+14th,+2_oct,+16th,+17th,+18th,+19th,+20th,+21st,+3_oct,USER_SCALE1,USER_SCALE2	
CONTROLINPUT	shift~n 21			
modamt	-2400	2400	1	1
	cents	@nvoices		
CONTROLINPUT	mod~n			
delayamt	0	@maxdelay	20	1
	ms	@nvoices		
CONTROLINPUT	delay~n			
pitch				
CONTROLOUTPUT	pitch out			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
period				
CONTROLOUTPUT	period out			
freq				
CONTROLOUTPUT	frequency out			
amp				
CONTROLOUTPUT	amplitude out			
tonality				
CONTROLOUTPUT	tonality out			
timbre				
CONTROLOUTPUT	timbre out			
obj				
USEROBJECTPARENT	obj			

dtonic(dtn) - Diatonic Pitch Shifter II (diatonic pitch shifter + scales)

Description

The diatonic pitch shift module shifts the musical pitch of an audio signal while maintaing the proper harmonic relationship to a diatonic scale. To accomplish this, the user specifies the desired key and the desired musical interval. The pitch shifter takes care of finding out what note is being played and automatically adjusts the amount of pitch shift so that the resultant note is in key. This pitch shifter can have from 1 to 12 independant pitch shift outputs. This can be useful for creating anything from two to five-part harmonies. The pitch shifter also has a built-in pitch detector whose results are made available through various control outputs.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	maximum delay memory that will be used
nvoices	1 to 12. This specifies how many independant output or "voices" this module will have.
steps	The number of steps in the scale.

Audio Inputs

Header	Description
in	The audio input to be pitch shifted.
mod(n)	This audio input will modulate the amount of pitch shift for a particular voice. This is useful to create vibrato effects. There is one mod input for each pitch shift voice.

Audio Outputs

Header	Description
out(n)	This is the output of the pitch shifter. There is one output per pitch shift voice.

Control Inputs

Header	Description
delayamt(n)	0 to 100 milliseconds. These control the amount of delay for each pitch shift voice.
gatelevel	-100 to 0 dB. This control affects only the pitch detection output of this module. The gatelevel control determines at what level the pitch detector will output pitch readings. If the input signal level falls below the level set here, the pitch detect outputs will latch on to the old values.
hysteresis	0 to 50 cents. This controls how much the pitch may move outside the range of the current note before the current note is changed to reflect that pitch. Hystersis can prevent the oscillating between adjacent notes that can occur when the pitch is in between two notes. Notes are typically 100 to 200 cents apart so 50 cents of hysteresis would allow the pitch to wander a lot before a higher or lower note was recognized.

Header	Description
key	This specifies the key the user will be playing in. The values are as follows (enharmonic keys are actually separate because of the different tuning options): 0 - C, 1 - C#, 2 - Db, 3 - D, 4 - D#, 5 - Eb, 6 - E, 7 - F, 8 - F#, 9 - Gb, 10 - G, 11 - G#, 12 - Ab, 13 - A, 14 - A#, 15 - Bb, 16 - B
minpitch	0 to 47. The minpitch control is used to optimize the pitch shifting algorithm. It sets the minimum pitch that pitch shifter is likely to detect. The values are as follows: 0 - C0 1 - C0# 2 - D0 ... ... 46 - A3# 47 - B3
modamt(n)	-2400 to 2400 cents. These control the amount of modulation to be applied to each pitch shift voice. adjustment is in cents and it represents the amount of pitch shift that would be added to each voice if the mod input was fully on.
quantization	-2 to 2. Controls whether the output pitch is quantized to remain exactly within key or whether it simply track the input pitch. A value of zero corresponds to no quantizing. A value of 1 corresponds to full quantizing.
scale	0 to 18. This control selects the scale, or mode, the user will be in. The scales are as follows: 0 - User, 1 - Ionian (Root Major), 2 - Dorian, 3 - Phrygian, 4 - Lydian, 5 - Mixolydian, 6 - Aeolian (Relative Minor), 7 - Locrian, 8 - Major, 9 - Minor, 10 - Harmonic Minor, 11 - Melodic Minor, 12 - Chromatic, 13 - Whole-tone, 14 - Pentatonic Major, 15 - Pentatonic Minor, 16 - Enigmatic, 17 - Neapolitan, 18 - Hungarian
shift(n)	-21 to 21. These control signals adjust the pitch shift interval for each voice. The values are as follows: -21 -3 octaves, -20 -21st, ..., ..., -7 octave down, -6 seventh down, -5 sixth down, -4 fifth down, -3 fourth down, -2 third down, -1 second down, 0 unison, 1 second up, 2 third, 3 fourth, 4 fifth, 5 sixth, 6 seventh, 7 octave up, 8 ninth, ..., ..., 21 3 octaves up
tune	392 to 494 cents. This control allows the diatonic shift to be tuned to a pitch reference other than A440 Hertz. The tuning is adjusted in cents referenced to A440.
xfadetime	0 to 100 milliseconds. This control signal is used to optimize the sound of the pitch shifters. Larger settings may result in smoother overall sound but may add a "flanged" sound to the audio. Smaller settings will result in a crisper sound but may allow more audible pitch shifting artifacts.
pitchtracking	on or off. This control allows you to turn off pitch tracking, for use with non pitched, or polyphonic sources. When pitch tracking is off the algorithm acts as though the input pitch is the root note of the selected scale.
userscale	This control selects the notes of a custom scale. The notes must be in ascending order and can only cover an octive in range from the lowest note to the highest note with no gaps. The scale must have at least five notes. If the scale does not follow these rules it will be ignored and a major scale used instead. If the scale has seven or less notes then the intervals will be derived from the scale. If it has more than seven notes a default major scale interval pattern will be used.
correctrate	0 to 2000 msec. This controls how rapidly the pitch is corrected to bring it into tune. The correctrate is how long it takes for the pitch to change 100 cents. Actual pitch correction values will be much less than 100 cents and thus will be performed more rapidly.

Header	Description
mincorrect	0 to 50 cents. This controls the smallest error in pitch that will be corrected for. If the pitch deviates from a note by less than this value no correction will be made. This value may be used to let vibrato through or to enable correction only on notes that are badly out of tune.
intervalglide	1 to 2000 msec. This controls how rapidly the pitch is changed to bring it into key. The intervalglide is how long it take for the pitch to change 100 cents.
tuning	0 to 4. This selects the tuning system used. The values are as follows: 0 - Equal Temperament, 1 - Just Major, 2 - Just Minor, 3 - Pythagorean, 4 - Meantone
pcent(n)	-2400 to 2400 cents. Control the amount of shifting in cents to be applied to each voice.

Control Outputs

Header	Description
amp	amp' is the r.m.s. amplitude relative to full scale ('amp' equal to 1 would be a square wave 'hitting the rail').
freq	The output of the pitch detector given as a frequency in Hertz.
period	The output of the pitch detector given as a period. The value is in milliseconds.
pitch	The output of the pitch detector given in cents relative to middle C.
timbre	is a measurement of the brightness of the tone independent of its pitch. A sine wave has a 'timbre' equal to 1, other wave shapes result in a higher 'timbre'.
tonality	A value representing how periodic the input signal is. A value a 1.0 is given for signals which are purely periodic. Lower values represent signals that are less periodic. The smallest value would be given for very noise-like signals.

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	22000/32500/#speed/#if	100	1
	max_delay			
nvoices	1	12	1	1
INT	number_voices			
steps	12	12	12	1
INT	scale_steps			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
AUDIOINPUT				
scale	0	18	0	1
			User,Ionian_(Maj),Dorian,Phrygian,Lydian,MixoLydian,Aeolian_(Min),Locrian,Major,Minor,Harmonic_Min,Melodic_Min,Chromatic,Whole-tone,Penta_Maj,Penta_Min,Enigmatic,Neapolitan,Hungarian	
CONTROLINPUT	scale			
key	0	16	0	1
			C,C#,Db,D,D#,Eb,E,F,F#,Gb,G,G#,Ab,A,A#,Bb,B	
CONTROLINPUT	key			
pitchtracking	0	1	0	1
CONTROLINPUT	pitchtracking			
hysteresis	0	50	10	1
	cents			
CONTROLINPUT	hysteresis			
correctrate	1	2000	1	1
	msec			
CONTROLINPUT	rate			
mincorrect	0	50	0	1
	cents			
CONTROLINPUT	min			
intervalglide	1	2000	1	1
	msec			
CONTROLINPUT	glide			
tuning	0	4	0	1
CONTROLINPUT	tuning			
tune	392	494	440	0.1
	Hz			
CONTROLINPUT	A4			
quantization	-2	2	0	1
CONTROLINPUT	quantize			
minpitch	0	47	21	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	minpitch			
gatelevel	-100	0	-60	1
	db			
CONTROLINPUT	gate			
xfadetime	0	100	10	1
	ms			
CONTROLINPUT	xfadetime			
mod				
AUDIOINPUT				
out				
AUDIOOUTPUT				
shift	-21	21	21	1
			-3_oct,-21st,-20th,-19th,-18th,-17th,-16th,-2_oct,-14th,-13th,-12th,-11th,-10th,-9th,-1_oct,-7th,-6th,-5th,-4th,-3rd,-2nd,unison,+2nd,+3rd,+4th,+5th,+6th,+7th,+1_oct,+9th,+10th,+11th,+12th,+13th,+14th,+2_oct,+16th,+17th,+18th,+19th,+20th,+21st,+3_oct	
CONTROLINPUT	shift~n			
pcent	-2400	2400	0	1
	cents			
CONTROLINPUT	pcent~n			
modamt	-2400	2400	1	1
	cents			
CONTROLINPUT	mod~n			
delayamt	0	@maxdelay	20	1
	ms			
CONTROLINPUT	delay~n			
userscale	0	12	0	1
CONTROLINPUT	Note~n			
pitch				
CONTROLOUTPUT	pitch out			
period				
CONTROLOUTPUT	period out			
freq				

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
CONTROLOUTPUT	frequency out			
amp				
CONTROLOUTPUT	amplitude out			
tonality				
CONTROLOUTPUT	tonality out			
timbre				
CONTROLOUTPUT	timbre out			
obj				
USEROBJECTPARENT	obj			

freqshift(fsh) - Audio Frequency Shifter

Description

Provides a frequency shift of all frequencies up to 1/4 nyquist from 'in' to 'out'. 'freq' is the amount of shift in Hz and can be positive or negative. This module use Single SideBand Modulation to accomplish its frequency shift. Note that frequency shifting is not the same as pitch shifting. In pitch shifting, audio frequencies are multiplied by a constant factor. In frequency shifting, a constant factor is ADDED to all frequencies. For this reason, frequency shifting audio will make it sound out of tune. However, small amount of frequency shifting (up to about 10 Hertz) can produce a very nice chorus effect.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The frequency shifter input.

Audio Outputs

Header	Description
out	The frequency shifted output.

Control Inputs

Header	Description
freq	-12 to +12 kiloHertz. Controls how much the input will be frequency shifted.

Header	Description

Control Outputs

Header	Description

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
out				
MONO				
AUDIOOUTPUT				
freq	-12000	12000	0	0.1
	hz			
CONTROLINPUT	%n			

h3000shiftlite(h3000shiftlite) - H3000 Style pitch shifter optimized for Micro Pitch Shift duties only

Description

This module emulates the autocorrelation H3000 pitch shifter cores, but removes exotic options.

Godlike Productions Comments

Specifiers

Header	Description
totalbuflen	Determines the total length of the audio buffer (predelay plus main)

Audio Inputs

Header	Description
in	The audio input.

Audio Outputs

Header	Description
out	The shifted output.
delayout	The delayed output

Control Inputs

Header	Description
pitchshift	Primary means of controlling pitch shift amount. This input is described in cents and added to 'pitchmod' to give overall pitch ratio.
pitchmodscale	Scales the value of pitchmod, from 0% to 100%
predelay1	Controls the amount of predelay in pitchshiting mode. predelay plus buffersublen equals the total delay length.
predelay2	Controls the delay time for the dry delay output.
buffersublen	Gives user dynamic control over the main buffer's length. Any length from 0 to 'bufferlen' is available. In pitch shifting mode, this allows control of the 'glitchiness' of shifting, and in delay mode this determines delay length.

Header	Description

Control Outputs

Header	Description

Mod Inputs

Header	Description
pitchmod	Secondary means of controlling pitch ratio. This input is added to 'pitchshift' to give overall pitch ratio.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
pitchmod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayout				
MONO				
AUDIOOUTPUT				
totalbuflen	5	3000	3000	0.1
FLOAT	bufferlen			
pitchshift	-3986	2400	0	1
CONTROLINPUT				
pitchmodscale	0	1200	0	0.001
CONTROLINPUT	pitchmodscale			
predelay1	0	@totalbuflen	0.1	0.1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	ms			
CONTROLINPUT	predelay1			
predelay2	0	@totalbuflen	0.1	0.1
	ms			
CONTROLINPUT	predelay2			
buffersublen	2	@totalbuflen	25	0.1
	ms			
CONTROLINPUT	buffersublen			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

h910alg(h910) - H910 pitch shifter algorithm

Description
This module emulates the Eventide H910 pitch shifter core.

Godlike Productions Comments

Specifiers

Header	Description
totalbuflen	Determines the total length of the audio buffer (predelay plus main). For H910 this should be 120 mS.
maxbuffersublen	Not documented. The maximum value of buffsublen.

Audio Inputs

Header	Description
in	The audio input.

Audio Outputs

Header	Description
out	The shifted output.
delayout	The delayed output

Control Inputs

Header	Description
mode	Determines whether module operates in pitch shifting mode or delay mode.
manual	Primary means of controlling pitch ratio. This input is added to 'pitchmod' to give overall pitch ratio.
pitchscale	Scales the value of pitchmod, from 0% to 100%
antifeedback	No documentation. Most likely phase inverted internal feedback amount.
predelay1	Controls the amount of predelay in pitchshiting mode. Predelay plus buffersublen equals the total delay length. In Delay Only mode this represents the full delay.
predelay2	Controls the amount of delay from the second, delay only, output.
buffersublen	Gives user dynamic control over the main buffer's length. Any length from 0 to 'bufferlen' is available. In pitch shifting mode, this allows control of the 'glitchiness' of shifting, and in delay mode this determines delay length. For the true H910 this should be set to 30 mS.
repeat	No documentation. Probably related to buffer looping or repeating, or possibly the number of repeats in delay mode.

Header	Description

Control Outputs

Header	Description

Mod Inputs

Header	Description
pitchmod	Secondary means of controlling pitch ratio. This input is added to 'manual' to give overall pitch ratio.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
pitchmod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayout				
MONO				
AUDIOOUTPUT				
totalbuflen	5	500	400	0.1
FLOAT	bufferlen			
maxbuffersublen	5	500	400	0.1
FLOAT	maxbuffersublen			

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
mode				
CONTROLINPUT	mode			
manual	0.1	4		
CONTROLINPUT				
pitchscale	0	1	1	0.001
CONTROLINPUT	pitchscale			
antifeedback				
			off,on	
CONTROLINPUT	antifeedback			
predelay1	0	@totalbuflen	0.1	0.1
	ms			
CONTROLINPUT	predelay1			
predelay2	0	@totalbuflen	0.1	0.1
	ms			
CONTROLINPUT	predelay2			
buffersublen	2	@totalbuflen	50	0.1
	ms			
CONTROLINPUT	buffersublen			
repeat				
CONTROLINPUT	repeat			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

h9xx(h9xx) - H910 & H949 pitch shifter

Description
This module emulates the Eventide H910 and H949 pitch shifter cores.

Godlike Productions Comments

Specifiers

Header	Description
totalbuflen	Determines the total length of the audio buffer (predelay plus main).

Audio Inputs

Header	Description
in	The audio input.

Audio Outputs

Header	Description
out	The shifted output.
delayout	The delayed output

Control Inputs

Header	Description
mode	Determines whether module operates in pitch shifting mode or delay mode.
manual	Primary means of controlling pitch ratio. This input is added to 'pitchmod' to give overall pitch ratio.
pitchscale	Scales the value of pitchmod, from 0% to 100%
algorithm	Controls which type of pitch shifting algorithm is used. 0 emulates the 910's method, 1 emulates the 949's 'algorithm 1,' and 2 emulates the 949's 'algorithm 2' (w/ autocorrelation). If mode is not set to pitch shifting, this control has no effect.
predelay1	Controls the amount of predelay in pitchshiting mode. Predelay plus buffersublen equals the total delay length. ??In Delay Only mode this represents the full delay??.
predelay2	No documentation. Controls the amount of delay from the second, delay only, output.
buffersublen	Gives user dynamic control over the main buffer's length. Any length from 0 to 'bufferlen' is available. In pitch shifting mode, this allows control of the 'glitchiness' of shifting, and in delay mode this determines delay length. For the true H910 this should be set to 30 mS.
repeat	No documentation. Probably related to buffer looping or repeating, or possibly the number of repeats in delay mode.

Header	Description

Control Outputs

Header	Description
pitchout	The pitch output. Presumably the amount of pitchshifting as a control signal.

Mod Inputs

Header	Description
pitchmod	Secondary means of controlling pitch ratio. This input is added to 'manual' to give overall pitch ratio.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
pitchmod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayout				
MONO				
AUDIOOUTPUT				
totalbuflen	5	500	400	0.1
FLOAT	bufferlen			
mode				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
			pitchshift,delay,reverse	
CONTROLINPUT	mode			
manual	0.1	4		
CONTROLINPUT				
pitchscale	0	1	1	0.001
CONTROLINPUT	pitchscale			
algo				
CONTROLINPUT	algo			
predelay1	0	@totalbuflen	0.1	0.1
	ms			
CONTROLINPUT	predelay1			
predelay2	0	@totalbuflen	0.1	0.1
	ms			
CONTROLINPUT	predelay2			
buffersublen	2	@totalbuflen	50	0.1
	ms			
CONTROLINPUT	buffersublen			
repeat				
CONTROLINPUT	repeat			
pitchout				
CONTROLOUTPUT	pitchout			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

moddetun(mtn) - Audio Signal Detuner with Modulation

Description

The detune module is used to add small amounts of pitch shift to an audio signal. This module is intended as an efficient (i.e. it doesn't use a lot of processing time) method of detuning a signal. In order to accomplish this, this module has traded off deglitch quality (glitches are a common artifact of pitch shifters) for processing efficiency. If totally glitch-free audio is needed, the pitch-shift module should be used instead of the detune module. This version of the detune module adds modulation to both the pitch and delay time.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	1 to 2000 milliseconds. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The audio input to be detuned.

Audio Outputs

Header	Description
out	A detuned version of the input signal.

Control Inputs

Header	Description
delay_ctl	1 to maxdelay milliseconds. Controls how much the audio will be delayed.
length_ctl	1 to maxdelay milliseconds. Controls the splice length for the detuning algorithm. Longer settings of this parameter will provide less "glitches" but will add more delay to the signal. Smaller settings may cause more glithes but will give a tighter sound (i.e. smaller delay). Extreme small settings may introduce modulation effects into the audio. Note also that the delay introduced as a result of this control will be variable, that is the audio delay will be continually changing from 0 thru the amount set by the length control.
pitch_ctl	-4800 to 1200 cents. Controls the amount of pitch detuning to be applied to the audio input. The adjustment is in "cents". A cent is one one-hundredth of a semitone. Positive values will shift the pitch upward and negative values will shift it downward.

Header	Description
xfadectl	0 to maxdelay milliseconds. Controls the xfade time of the splices and the sound of the glitches. Smaller values will make the glitches shorter but more obvious whereas longer values will make the glitches longer but will have a tendancy to smear the audio. Smaller xfade times are suggested for more transient input signals whereas longer xfade times are suggested for more tonal input signals.
delaymodamt	0 to maxdelay milliseconds. Controls how much affect the delay modulation has on the signal.
pitchmodamt	-1200 to 1200 cents. Controls how much affect the pitch modulation has on the signal.

Control Outputs

Header	Description

Mod Inputs

Header	Description
shiftmod	The modulation signal for the pitch shift amount.
delaymod	The modulation signal for the delay time.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	2000	20	1
	max_delay			
in				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
MONO				
AUDIOINPUT				
shiftmod				
MOD				
AUDIOINPUT				
delaymod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delayctl	1	@maxdelay	20	1
	ms			
CONTROLINPUT	delay			
pitchctl	-4800	1200	15	1
	cents			
CONTROLINPUT	pitch			
lengthctl	1	@maxdelay	20	1
	ms			
CONTROLINPUT	length			
xfadectl	0	@maxdelay	10	1
	ms			
CONTROLINPUT	xfade			
delaymodamt	0	@maxdelay	0	1
	ms			
CONTROLINPUT	delay_modamt			
pitchmodamt	-1200	1200	0	1
	cents			
CONTROLINPUT	pitch_modamt			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

modfreqshift(mfs) - Modulatable Audio Frequency Shifter

Description

Provides a frequency shift of all frequencies up to 1/4 nyquist from 'in' to 'out'. 'freq' is the amount of shift in Hz and can be positive or negative. This module use Single SideBand Modulation to accomplish its frequency shift. Note that frequency shifting is not the same as pitch shifting. In pitch shifting, audio frequencies are multiplied by a constant factor. In frequency shifting, a constant factor is ADDED to all frequencies. For this reason, frequency shifting audio will make it sound out of tune. However, small amount of frequency shifting (up to about 10 Hertz) can produce a very nice chorus effect.

Godlike Productions Comments

Specifiers

Header	Description

Audio Inputs

Header	Description
in	The frequency shifter input.
mod	This audio input will modulate the amount

Audio Outputs

Header	Description
out	The frequency shifted output.

Control Inputs

Header	Description
freq	-12 to +12 kiloHertz. Controls how much the input will be frequency shifted.
modamt	-12 to +12 kiloHertz. This controls the amount of modulation to be applied to the frequency shifter.

Header	Description

Control Outputs

Header	Description

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
in				
MONO				
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
freq	-12000	12000	0	0.1
	hz			
CONTROLINPUT	freqshift			
modamt	-12000	12000	0	0.1
	hz			
CONTROLINPUT	mod			

multishift(msh) - Multi-Output Pitch Shifter

Description

This module shifts the pitch of an input over a range of +/- four octaves. This pitch shifter can have from 1 to 16 independant pitch shift outputs. This can be useful for creating anything from two to five-part harmonies or very dense detuned chorus effects. The pitch shifter also has a built-in pitch detector whose results are made available through various control outputs.

Godlike Productions Comments

Specifiers

Header	Description
lowlatencymode	On Factor pedals, H9, and Eclipse the pitchshift module used a smaller splice buffer than on the H8k. This had two main effects - 1) A lower minimum delay value (@44.1k ~3ms H9 vs ~6ms H8K) and 2) A higher low note value for the pitch tracking (minpitch of ~9 for the H9 version and ~7 for the H8K). Another possible side effect would be slightly more artifacts in the lower latency version since a smaller splice buffer means the module xfades/splices more often. How noticeable these would be would be very situationally dependent. 0 - Higher latency, better low note tracking H8K version (default) 1 - Lower latency H9, Factor, and Eclipse version.
maxdelay	maximum delay
nvoices	1 to 16. This specifies how many independant output or "voices" this module will have.

Audio Inputs

Header	Description
in	The audio input to be pitch shifted.
mod(n)	This audio input will modulate the amount of pitch shift for a particular voice. This is useful to create vibrato effects. There is one mod input for each pitch shift voice.

Audio Outputs

Header	Description
out(n)	This is the output of the pitch shifter. There is one output per pitch shift voice.

Control Inputs

Header	Description
delayamt(n)	0 to 2000 milliseconds. These control the amount of delay for each pitch shift voice.
gatelevel	-100 to 0 dB. This control affects only the pitch detection output of this module. The gatelevel control determines at what level the pitch detector will output pitch readings. If the input signal level falls below the level set here, the pitch detect outputs will latch on to the old values.

Header	Description
minpitch	0 to 47. The minpitch control is used to optimize the pitch shifting algorithm. It sets the minimum pitch that pitch shifter is likely to detect. The values are as follows: 0 - C0, 1 - C0#, 2 - D0, ..., ..., 46 - A3#, 47 - B3
modamt(n)	-2400 to 2400 cents. These control the amount of modulation to be applied to each pitch shift voice. adjustment is in cents and it represents the amount of pitch shift that would be added to each voice if the mod input was fully on.
shift(n)	-4800 to 4800 cents. Controls the amount of pitch shift to be applied to the audio input. The adjustment is in "cents". A cent is one one-hundredth of a semitone. Positive value will shift the pitch upward and negative values will shift it downward.
xfadetime	0 to 100 milliseconds. This control signal is used to optimize the sound of the pitch shifters. Larger settings may result in smoother overall sound but may add a "flanged" sound to the audio. Smaller settings will result in a crisper sound but may allow more audible pitch shifting artifacts.

Control Outputs

Header	Description
amp	amp' is the r.m.s. amplitude relative to full scale ('amp' equal to 1 would be a square wave 'hitting the rail').
freq	The output of the pitch detector given as a frequency in Hertz.
period	The output of the pitch detector given as a period. The value is in milliseconds.
pitch	The output of the pitch detector given in cents relative to middle C.
timbre	is a measurement of the brightness of the tone independent of its pitch. A sine wave has a 'timbre' equal to 1, other wave shapes result in a higher 'timbre'.
tonality	A value representing how periodic the input signal is. A value a 1.0 is given for signals which are purely periodic. Lower values represent signals that are less periodic. The smallest value would be given for very noise-like signals.

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
lowlatencymode	0	1	0	1
	lower latency			
maxdelay	1	262144000/48000/#div	2000	1
	ms			
INT	max_delay			
nvoices	1	16	1	1
INT	number_voices			
in				
MONO				
AUDIOINPUT				
minpitch	0	47	21	
			c0,c#0,d0,d#0,e0,f0,f#0,g0,g#0,a0,a#0,b0,c1,c#1,d1,d#1,e1,f1,f#1,g1,g#1,a1,a#1,b1,c2,c#2,d2,d#2,e2,f2,f#2,g2,g#2,a2,a#2,b2,c3,c#3,d3,d#3,e3,f3,f#3,g3,g#3,a3,a#3,b3	
CONTROLINPUT	minpitch			
gatelevel	-100	0	-60	1
	db			
CONTROLINPUT	gate			
xfadetime	0	100	10	1
	ms			
CONTROLINPUT	xfadetime			
mod				
MOD		@nvoices		
AUDIOINPUT				
out				
MONO		@nvoices		
AUDIOOUTPUT				
shift	-4800	4800		1
	cents	@nvoices		
CONTROLINPUT	shift~n			
modamt	-2400	2400		1
	cents			
CONTROLINPUT	mod~n			
delayamt	0	@maxdelay		1

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	ms	@nvoices		
CONTROLINPUT	delay~n			
pitch				
CONTROLOUTPUT	pitch out			
period				
CONTROLOUTPUT	period out			
freq				
CONTROLOUTPUT	frequency out			
amp				
CONTROLOUTPUT	amplitude out			
tonality				
CONTROLOUTPUT	tonality out			
timbre				
CONTROLOUTPUT	timbre out			
obj				
USEROBJECTPARENT	obj			

pitchshift(psh) - Pitch Shifter

Description

This module shifts the pitch of an input over a range of +/- four octaves. If it is needed to have more than one pitch shifted output, use multishift. The pitch shifter also has a built-in pitch detector whose results are made available through various control outputs.

Godlike Productions Comments

Specifiers

Header	Description
lowlatencymode	On Factor pedals, H9, and Eclipse the pitchshift module used a smaller splice buffer than on the H8k. This had two main effects - 1) A lower minimum delay value (@44.1k ~3ms H9 vs ~6ms H8K) and 2) A higher low note value for the pitch tracking (minpitch of ~9 for the H9 version and ~7 for the H8K). Another possible side effect would be slightly more artifacts in the lower latency version since a smaller splice buffer means the module xfades/splices more often. How noticeable these would be would be very situationally dependent. 0 - Higher latency, better low note tracking H8K version (default) 1 - Lower latency H9, Factor, and Eclipse version.
maxdelay	maximum amount of delay memory used for the pitchshifter. Max 2000ms

Audio Inputs

Header	Description
in	The audio input to be pitch shifted.
mod	This audio input will modulate the amount

Audio Outputs

Header	Description
out	This is the output of the pitch shifter.

Control Inputs

Header	Description
delayamt	0 to 2000 milliseconds. This controls the amount of delay for the pitch shifter.
gatelevel	-100 to 0 dB. This control affects only the pitch detection output of this module. The gatelevel control determines at what level the pitch detector will output pitch readings. If the input signal level falls below the level set here, the pitch detect outputs will latch on to the old values.
minpitch	0 to 47. The minpitch control is used to optimize the pitch shifting algorithm. It sets the minimum pitch that pitch shifter is likely to detect. The values are as follows: 0 - C0 1 - C0# 2 - D0 ... ... 46 - A3# 47 - B3
modamt	-2400 to 2400 cents. This controls the amount of modulation to be applied to the pitch shifter. The adjustment is in cents and it represents the amount of pitch shift that would be added if the mod input was fully on.

Header	Description
shift	-4800 to 4800 cents. Controls the amount of pitch shift to be applied to the audio input. The adjustment is in "cents". A cent is one one-hundredth of a semitone. Positive value will shift the pitch upward and negative values will shift it downward.
xfadetime	0 to 100 milliseconds. This control signal is used to optimize the sound of the pitch shifters. Larger settings may result in smoother overall sound but may add a "flanged" sound to the audio. Smaller settings will result in a crisper sound but may allow more audible pitch shifting artifacts.

Control Outputs

Header	Description
amp	amp' is the r.m.s. amplitude relative to full scale ('amp' equal to 1 would be a square wave 'hitting the rail').
freq	The output of the pitch detector given as a frequency in Hertz.
period	The output of the pitch detector given as a period. The value is in milliseconds.
pitch	The output of the pitch detector given in cents relative to middle C.
timbre	is a measurement of the brightness of the tone independent of its pitch. A sine wave has a 'timbre' equal to 1, other wave shapes result in a higher 'timbre'.
tonality	A value representing how periodic the input signal is. A value a 1.0 is given for signals which are purely periodic. Lower values represent signals that are less periodic. The smallest value would be given for very noise-like signals.

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
lowlatencymode	0	1	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
	lower latency			
maxdelay	1	262144000/48000/#div	2000	1
	ms			
INT	max_delay			
in				
MONO				
AUDIOINPUT				
minpitch	0	47	21	
CONTROLINPUT	minpitch			
gatelevel	-100	0	-60	1
	db			
CONTROLINPUT	gate			
xfadetime	0	100	10	1
	ms			
CONTROLINPUT	xfadetime			
mod				
MOD				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
shift	-4800	4800		1
	cents			
CONTROLINPUT	shift			
modamt	-2400	2400		1
	cents			
CONTROLINPUT	mod			
delayamt	0	@maxdelay		1
	ms			
CONTROLINPUT	delay			
pitch				
CONTROLOUTPUT	pitch out			
period				
CONTROLOUTPUT	period out			
freq				

Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
CONTROLOUTPUT	frequency out			
amp				
CONTROLOUTPUT	amplitude out			
tonality				
CONTROLOUTPUT	tonality out			
timbre				
CONTROLOUTPUT	timbre out			
obj				
USEROBJECTPARENT	obj			

reverse(rev) - Reverse Pitch Shifter

Description

This module implements a version of reverse pitch- shifting, a fixture on Eventide Harmonizers(R). This module takes small segments of audio and plays them in reverse. At the same time, it can change the playback pitch. This module can also operate as a standard pitch shifter, albeit without deglitching, i.e., its sound like an H910.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	1 to 32500 milliseconds. Specifies the maximum delay this module will use.

Audio Inputs

Header	Description
in	The audio input to this module.

Audio Outputs

Header	Description
out	The reverse shifted output.

Control Inputs

Header	Description
delay_ctl	0 to maxdelay milliseconds. Controls how much the audio will be delayed.
length_ctl	1 to maxdelay milliseconds. Controls the splice length for the reverse algorithm. This controls the length of the audio segment that is played backwards.
pitch_ctl	-4800 to 4800 cents. Controls the amount of pitch pitch shift to be applied to the reverse shifted audio. The adjustment is in "cents". A cent is one one-hundredth of a semitone. Positive value will shift the pitch upward and negative values will
direction	Controls whether the shifter is operating as a reverse pitch shifter or as a standard pitch shifter (without de-glitching). The values are: 0 = reverse, 1 = normal pitch shift

Header	Description

Control Outputs

Header	Description

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	1	22000/32500/#speed/#if	20	1
	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
delay_ctl	0	@maxdelay	20	1
	ms			
CONTROLINPUT	delay			
length_ctl	0	@maxdelay	20	1
	ms			
CONTROLINPUT	length			
pitch_ctl	-4800	4800	0	1
	ms			
CONTROLINPUT	pitch			

reverse

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
direction	0	1		
CONTROLINPUT	direction			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

stereoshift(psh) - Stereo Pitch Shifter

Description

This module shifts the pitch of a stereo input over a range of +/- four octaves. The shifter maintains stereo imaging and mono compatibility. The pitch shifter also has a built-in pitch detector whose results are made available through various control outputs.

Godlike Productions Comments

Specifiers

Header	Description
lowlatencymode	On Factor pedals, H9, and Eclipse the pitchshift module used a smaller splice buffer than on the H8k. This had two main effects - 1) A lower minimum delay value (@44.1k ~3ms H9 vs ~6ms H8K) and 2) A higher low note value for the pitch tracking (minpitch of ~9 for the H9 version and ~7 for the H8K). Another possible side effect would be slightly more artifacts in the lower latency version since a smaller splice buffer means the module xfades/splices more often. How noticeable these would be would be very situationally dependent. 0 - Higher latency, better low note tracking H8K version (default) 1 - Lower latency H9, Factor, and Eclipse version.
maxdelay	maximum delay memory in msecs the module uses
nvoices	1 to 16. The number of stereo pitch shifted output voices. Each stereo voice has its own shift, delay, and modulation

Audio Inputs

Header	Description
leftin	The left audio input to be pitch shifted.
mod(n)	This audio input will modulate the amount of pitch shift. This is useful to create vibrato effects.
rightin	The right audio input to be pitch shifted.

Audio Outputs

Header	Description
left(n)	This is the left output of the pitch shifter.
right(n)	This is the right output of the pitch shifter.

Control Inputs

Header	Description
delayamt(n)	0 to 2000 milliseconds. This controls the amount of delay for the pitch shifter.
gatelevel	-100 to 0 dB. This control affects only the pitch detection output of this module. The gatelevel control determines at what level the pitch detector will output pitch readings. If the input signal level falls below the level set here, the pitch detect outputs will latch on to the old values.
minpitch	0 to 47. The minpitch control is used to optimize the pitch shifting algorithm. It sets the minimum pitch that pitch shifter is likely to detect. The values are as follows: 0 - C0, 1 - C0#, 2 - D0, ..., ..., 46 - A3#, 47 - B3

Header	Description
modamt(n)	-2400 to 2400 cents. This controls the amount of modulation to be applied to the pitch shifter. The adjustment is in cents and it represents the amount of pitch shift that would be added if the mod input was fully on.
processoff	0 or 1, turns off processing
shift(n)	-4800 to 4800 cents. Controls the amount of pitch shift to be applied to the audio input. The adjustment is in "cents". A cent is one one-hundredth of a semitone. Positive value will shift the pitch upward and negative values will shift it downward.
xfadetime	0 to 100 milliseconds. This control signal is used to optimize the sound of the pitch shifters. Larger settings may result in smoother overall sound but may add a "flanged" sound to the audio. Smaller settings will result in a crisper sound but may allow more audible pitch shifting artifacts.
pan	-1 to 1. Used to optimize the pitch shifting algorithm. It sets which channel is more important to get right. 1 is right, -1 is left.

Control Outputs

Header	Description
amp	amp' is the r.m.s. amplitude relative to full scale ('amp' equal to 1 would be a square wave 'hitting the rail').
freq	The output of the pitch detector given as a frequency in Hertz.
period	The output of the pitch detector given as a period. The value is in milliseconds.
pitch	The output of the pitch detector given in cents relative to middle C.
tonality	A value representing how periodic the input signal is. A value a 1.0 is given for signals which are purely periodic. Lower values represent signals that are less periodic. The smallest value would be given for very noise-like signals.

Mod Inputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
lowlatencymode	0	1	0	1
	lower latency			
maxdelay	1	262144000/48000/#div	100	1
	ms			
INT	max_delay			
nvoices	1	16	1	1
INT	number_voices			
leftin				
LEFT				
AUDIOINPUT				
rightin				
RIGHT				
AUDIOINPUT				
minpitch	0	47	21	
			c0,c#0,d0,d#0,e0,f0,f#0,g0,g#0,a0,a#0,b0,c1,c#1,d1,d#1,e1,f1,f#1,g1,g#1,a1,a#1,b1,c2,c#2,d2,d#2,e2,f2,f#2,g2,g#2,a2,a#2,b2,c3,c#3,d3,d#3,e3,f3,f#3,g3,g#3,a3,a#3,b3	
CONTROLINPUT	minpitch			
gatelevel	-100	0	-60	1
	db			
CONTROLINPUT	gate			
xfadetime	0	100	10	1
	ms			
CONTROLINPUT	xfadetime			
mod				
MOD		@nvoices		
AUDIOINPUT				
left				
LEFT		@nvoices		
AUDIOOUTPUT				
right				
AUDIOOUTPUT				
shift	-4800	4800		1
	cents	@nvoices		

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	shift			
modamt	-2400	2400		1
	cents	@nvoices		
CONTROLINPUT	mod			
delayamt	0	@maxdelay/2/#-div		1
	ms	@nvoices		
CONTROLINPUT	delay			
pan	-1	1		0.1
CONTROLINPUT	pan			
processoff	0	1	0	1
CONTROLINPUT	turn_off			
pitch				
CONTROLOUTPUT	pitch out			
period				
CONTROLOUTPUT	period out			
freq				
CONTROLOUTPUT	frequency out			
amp				
CONTROLOUTPUT	amplitude out			
tonality				
CONTROLOUTPUT	tonality out			
obj				
USEROBJECTPARENT	obj			

Chapter 15 - Reverb

diffchorus(dfc) - Diffusor with Chorus

Description

The diffusor module creates a dense field of delay repeats that is typically used to create reverberator structures. This particular flavor of diffusor has built-in chorusing; that is, delays are randomly swept so as to prevent a build-up of undesirable resonances in the reverb. Like the standard diffusor, the chorused diffusor is essentially a chain of series-connected allpass filters.

Godlike Productions Comments

Specifiers

Header	Description
declick_g	0 or 1. When set to 1 there is a 2 second slew applied to any changes in the gain.
nsections	2 to 32. This specifies how many allpass filters are to be used to construct this diffusor.
smooth_glide	0 or 1. Declicks changes in delay time.

Audio Inputs

Header	Description
in	The audio input to be diffused.

Audio Outputs

Header	Description
out	This is the output of the diffusor.

Control Inputs

Header	Description
delayamt(n)	These inputs control the amount of delay used for each of the allpass sections. The adjustment is in milliseconds.
diffusion	0 to 1.0. The amount of diffusion. A value of 1.0 yields maximum diffusion or recirculation in the allpass filter. A value of zero yields no diffusion.
g(n)	These control the amount of feedback for each of the individual allpass sections.
gliderate	This controls how fast the delays will respond to changes in their setting.
moddepth	Adjusts the range of delay modulation for the allpass filters. The adjustment is in milliseconds.
modrate	Determines how fast the delays will be modulated. The adjustment is in milliseconds per second. High settings will result in noticeable pitch shift of the audio.

Header	Description
modratespan	This control determines the difference in sweep rate for all of the internal delays. With a setting of zero, all delays will be swept at the same rate. At a setting of 1.0, the delay sweep rates will vary from 0 to the modrate setting.
processoff	0 or 1, turns off processing
size	0 to 1.0. The size control scales the delays of all of the allpass filters. A value of 1.0 gives the largest size and a value of zero gives the smallest.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nsections	2	32	2	1
	nsections			
totaldelay	100	2000	2000	1
	msec			
INT	total_delay			
declick_g	0	1	0	1
INT	g_declicking			
smooth_glide	0	1	0	1
INT	smoother_mods			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
diffusion	0	1	0.5	0.01
CONTROLINPUT	diffusion			
dsize	0	1	0.5	0.01

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	size			
moddepth	0	100	0	0.01
	msec			
CONTROLINPUT				
modrate	0	100000	0.5	0.01
	ms/sec			
CONTROLINPUT				
modratespan	0	1	0	0.01
CONTROLINPUT	span			
gliderate	0	1	0	0.01
CONTROLINPUT	gliderate			
delayamt	0	@totaldelay/@nsections/#-div/#trunc	30	0.01
	msec	@nsections		
CONTROLINPUT	delay~n			
g	-1	1	0	0.001
		@nsections		
CONTROLINPUT	coef~n			
processoff	0	1	0	1
CONTROLINPUT	turn_off			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

diffusor(dfr) - Diffusor

Description

The diffusor module creates a dense field of delay repeats that is typically used to create reverberator structures. This diffusor is essentially a chain of series-connected allpass filters.

Godlike Productions Comments

Specifiers

Header	Description
nsections	2 to 32. This specifies how many allpass filters are to be used to construct this diffusor.
maxdelay	Maximum delay

Audio Inputs

Header	Description
in	The audio input to be diffused.

Audio Outputs

Header	Description
out	This is the output of the diffusor.

Control Inputs

Header	Description
delayamt(n)	These inputs control the amount of delay used for each of the allpass sections. The adjustment is in milliseconds.
diffusion	0 to 1.0. The amount of diffusion. A value of 1.0 yields maximum diffusion or recirculation in the allpass filter. A value of zero yields no diffusion.
g(n)	These control the amount of feedback for each of the individual allpass sections.
size	0 to 1.0. The size control scales the delays of all of the allpass filters. A value of 1.0 gives the largest size and a value of zero gives the smallest.

Header	Description

Control Outputs

Header	Description

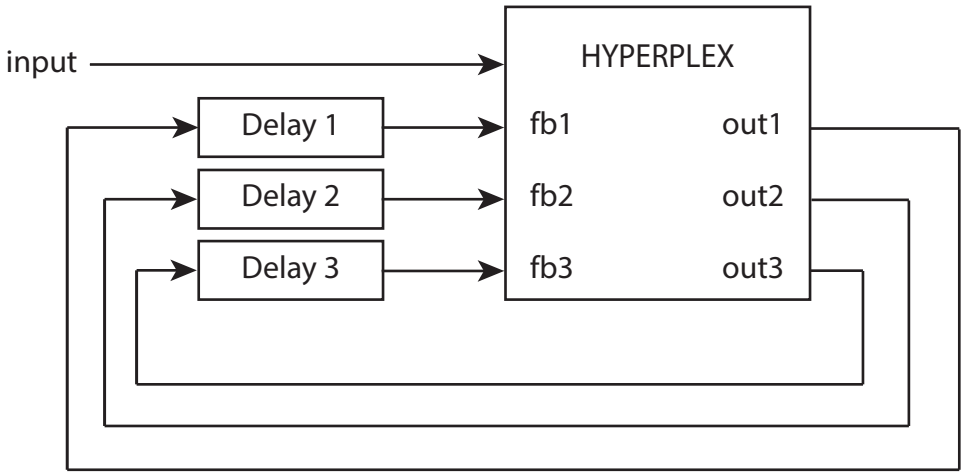
Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nsections	1	32	2	1
	#_sections			
maxdelay	1	22000/32500/#speed/#if	2000	1
	ms			
INT	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
diffusion	0	1	0.5	0.1
CONTROLINPUT	diffusion			
dsize	0	1	0.5	0.1
CONTROLINPUT	size			
delayamt	0	@maxdelay/@nsections/#-div/#trunc	30	0.01
	ms	@nsections		
CONTROLINPUT	delay~n			
g	-1	1	0	0.01
		@nsections		
CONTROLINPUT	coef~n			
obj				
USEROBJECTPARENT	obj			

hyperplex(hpx) - Reverberation Tool

Description

The hyperplex module provides a simple way of creating high quality reverberators. To create a reverberator, the outputs of a bunch of delay lines are fed into the hyperplex module and the outputs of the hyperplex module are fed back to the inputs of the delay line. The hyperplex module combines the delay outputs to produce an exponentially increasing density of echoes, hence a dense reverb. The diffusion of the reverb can be increased by rotating the coefficient matrix for each pairwise combination of elements in the feedback matrix. The total number of possible rotations is (numdelays*(numdelays-1))/2. Here is a typical patch:



You can make things even more interesting by experimenting with add other modules besides delay lines in the feedback structure, however, each output->input path must have a total gain less than 1.0, otherwise oscillation may occur.

Godlike Productions Comments

Specifiers

Header	Description
numdelays	2 through 32. Specifies how many feedback loops to create.
numthetas	The number of angles over which you want indepentent control. Will be a number between 1 and (numdelays-1)*numdelays/2 representing each pair of delays.

Audio Inputs

Header	Description
fb[n]	The audio input/feedback input. The connection to the delay line outputs.

Audio Outputs

Header	Description
out[n]	Plex outputs. The outs to be connected to the delay line inputs.

Header	Description

Control Inputs

Header	Description
diffusion	0.0 to 1.0. Scales each rotation angle
processoff	0 or 1, turns off processing
size	0.0 to 1.0. Scales the delay time output values to easily control Size.
decay	0.0 to 100.0 S. RT60 time of reverb, based on the delay time input values and size parameter.
delay[n]	0.0 to 65536 mS.
theta[m]	-180 to 180 degrees. Amount of influence the delay has on it's neighbors. The order is theta(1,2), theta(1,3), theta(2,3), theta(1,4), theta(2,4), theta(3,4), theta(1,5), ... , theta(order-1,order)

Control Outputs

Header	Description
delayout[n]	delay time scaled by size parameter

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
numdelays	2	32	8	1
	plex_size			
numthetas	1	@numde-lays/1/#sub/@ numdelays/2/#-div/#mul	1	1
INT	num_thetas			
fb				
MONO		@numdelays		
AUDIOINPUT				
out				
MONO		@numdelays		

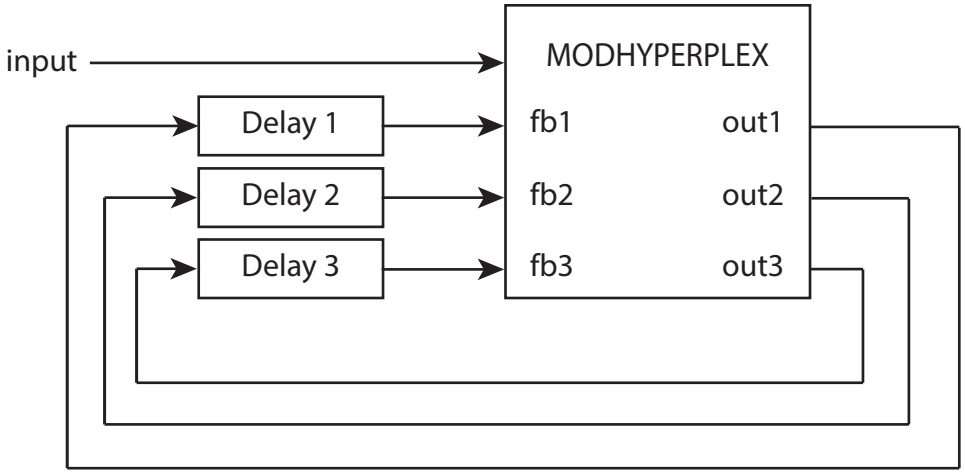
Name	Min	Max	Default	Resolution
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
AUDIOOUTPUT				
size	-1	1	0.9	0.0001
	%			
CONTROLINPUT				
decay	0	100	1	0.1
	S			
CONTROLINPUT	*100			
diffusion	-1	1	0.9	0.0001
	%			
CONTROLINPUT				
delay	0	65536	20	0.1
	ms	@numdelays		
CONTROLINPUT	delay~n			
theta	-180	180	0	1
	degrees	@numthetas		
CONTROLINPUT	theta~n			
processoff	0	1	0	1
CONTROLINPUT	turn_off			
delayout				
CONTROLOUTPUT	delayout~n			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

modhyperplex(hpx) - Reverberation Tool

Description

The modhyperplex module provides a simple way of creating high quality reverberators. To create a reverberator, the outputs of a bunch of delay lines are fed into the hyperplex module and the outputs of the hyperplex module are fed back to the inputs of the delay line. The hyperplex module combines the delay outputs to produce an exponentially increasing density of echoes, hence a dense reverb. Like the hyperplex module, the diffusion of the reverb can be increased by rotating the the coefficient matrix for each pairwise combination off elements in the feedback matrix. The total number of possible rotations is (numdelays*(numdelays-1))/2. Additionally, modhyperplex allows you to modulate each rotation angle via a dedicated mod input. Here is a typical patch :



You can make things even more interesting by experimenting with add other modules besides delay lines in the feedback structure, however, each output->input path must have a total gain less than 1.0, otherwise oscillation may occur.

Godlike Productions Comments

Specifiers

Header	Description
numdelays	2 through 32. Specifies how many feedback loops to create.
numthetas	The number of angles over which you want indepentent control. Will be a number between 1 and (numdelays-1)*numdelays/2 representing each pair of delays.

Audio Inputs

Header	Description
fb[n]	The audio input/feedback input. The connection to the delay line outputs.
thetamod[m]	a modulation signal for each theta.

Audio Outputs

Header	Description
out[n]	Plex outputs. The outs to be connected to the delay line inputs.

Header	Description

Control Inputs

Header	Description
diffusion	0.0 to 1.0. Scales each rotation angle
processoff	0 or 1, turns off processing
size	0.0 to 1.0. Scales the delay time output values to easily control Size.
decay	0.0 to 100.0 S. RT60 time of reverb, based on the delay time input values and size parameter.
delay[n]	0.0 to 65536 mS.
theta[m]	-180 to 180 degrees. Amount of influence the delay has on it's neighbors. The order is theta(1,2), theta(1,3), theta(2,3), theta(1,4), theta(2,4), theta(3,4), theta(1,5), ... , theta(order-1,order)
thetamodamt[m]	Not documented. The amount that thetamod modifies theta[m]

Control Outputs

Header	Description
delayout[n]	delay time scaled by size parameter

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
numdelays	2	16	2	1
	plex_size			
numthetas	1	@numde-lays/1/#sub/@ numdelays/2/#-div/#mul	1	1
INT	num_thetas			
fb				
MONO		@numdelays		
AUDIOINPUT				
thetamod				

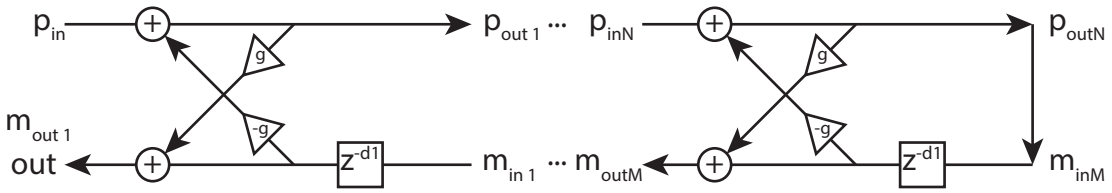
Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
MONO		@numthetas		
AUDIOINPUT				
out				
MONO		@numdelays		
AUDIOOUTPUT				
size	-1	1	0.9	0.0001
	%			
CONTROLINPUT	*100			
decay	0	100	1	0.1
	S			
CONTROLINPUT				
diffusion	-1	1	0.9	0.0001
	%			
CONTROLINPUT	*100			
delay	0	65536	20	0.1
	ms	@numdelays		
CONTROLINPUT	delay~n			
theta	-180	180	0	1
	degrees	@numthetas		
CONTROLINPUT	theta~n			
thetamodamt	-180	180	0	1
	degrees			
CONTROLINPUT	thetamodamt~n			
processoff	0	1	0	1
CONTROLINPUT	turn_off			
delayout				
		@numdelays		
CONTROLOUTPUT	delayout~n			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

nested_allpass - series lattice filters combined to

Description

create an allpass-like reverberant response. This chaining of lattice filters together creates allpass structures useful for quickly building up echo density in a larger reverb structure. Signal flow diagram: The nested allpass module implements 2 or more stages of a series connection of N lattice filters.



Further information: Audio input flows into in and through the top rail. Lattice instances are connected in series, pout feeding p_in of the next instance, until the end which links the top rail (pout) to its own bottom rail (m_in), wherein the audio flows in the opposite direction. The bottom rail output (mout) at the initial stage, should have an allpass transfer function.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	Specifies the maximum delay in this lattice stage in milliseconds.
nsections	2 to 32. This specifies how many lattice filters are to be used to construct this nested allpass.

Audio Inputs

Header	Description
in	The main audio input.

Audio Outputs

Header	Description
out	The main audio output

Control Inputs

Header	Description
g(n)	-1 to 1. The reflection coefficient, or gain, of this stage.
processoff	0 or 1, turns off processing
dly_amt(n)	0 to maxdelay/nsections msecs. The delay in this stage in milliseconds.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nsections	2	32	2	1
	nsections			
maxdelay	1	200	1	1
FLOAT	max_delay			
in				
MONO				
AUDIOINPUT				
out				
MONO				
AUDIOOUTPUT				
dly_amt	0	@maxdelay/@nsections/#-div/#trunc	1	0.01
	msec	@nsections		
CONTROLINPUT	delay~n			
g	-1	1	0.5	0.001
		@nsections		
CONTROLINPUT	reflection~n			
processoff	0	1	0	1
CONTROLINPUT	turn_off			

reverb_2016 - Multi-reverb module with algorithms from SP2016 and Reverb 2016

Description

Versatile module that encompasses the sound and features of SP2016 and Reverb 2016.

Godlike Productions Comments

Thanks to Eventide for additional information. You guys rock.

Specifiers

Header	Description
declick	Declicking on or off
no_dry_latency_compen-sation	Not documented. Turn latency compensation on and off for the dry output.

Audio Inputs

Header	Description
input_left	The left audio input to be processed
input_right	The right audio to be processed.

Audio Outputs

Header	Description
output_left	This is the left output of the current reverb algorithm
output_right	This is the right output of the current reverb algorithm
scaled_dry_L	The left dry signal with dry/wet mix gain applied. For a wet dry mix, the signals should be added together outside of the module. This allows a softclipper, etc. to be applied to the output without processing the dry signal.
scaled_dry_R	The right dry signal with dry/wet mix gain applied. For a wet dry mix, the signals should be added together outside of the module. This allows a softclipper, etc. to be applied to the output without processing the dry signal.

Control Inputs

Header	Description
input_level	Input gain in dB
output_level	Output gain in dB
wet_dry	Dry/wet mix using a taper from the Reverb 2016
pre_delay	Predelay in milliseconds
decay_time	Decay for tail (approximate)
position	Front/rear mix. Inactive for SP2016 Plate
diffusion	Sets g for allpass sections. The range of this unit (as well as its effect) depend on whether the algorithm comes from the SP2016 or the Reverb 2016. Inactive for SP2016 Plate. Additionally, the Reverb 2016 algorithms have eight diffusers on each output channel, and this also controls their values. For this reason, the Reverb 2016 algorithms are substantially more affected by the diffusion control. Diffusion control is inactive for SP2016 Plate.
low_eq_gain	Gain for low shelf filter in dB. Inactive for SP2016 Plate.

Header	Description
low_eq_cutoff	Corner frequency for low shelf filter in increments of 50 Hz from 50 to 500 Hz. Inactive for SP2016 Plate.
high_eq_gain	Gain for high shelf filter in dB. Inactive for SP2016 Plate.
high_eq_cutoff	Corner frequency for high shelf filter in increments of 50 Hz from 1000 to 8000 Hz. Inactive for SP2016 Plate.
kill	Kills the input while tail continues
is_bypassing	Activates bypassing
algorithm	Not documented. Determines the Reverb 2016 algorithm to be used. Range 0-5. 0: Vintage Stereo Room, 1: Vintage Room, 2: Vintage Plate, 3: Modern Stereo Room, 4: Modern Room, 5: Modern Plate
input_mode	stereo, mono L, mono R

Control Outputs

Header	Description
internal_clip	Peak amplitude
delay_compensation	Not documented. Likely the ms or samples of delay of this algorithm so that the output can be aligned with the input, if this is required.

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
declick	0	1	1	1
	declicking			
no_dry_latency_compen-sation	0	1	0	1
INT	no_dry_latency_compensation			
input_left				
AUDIOINPUT				
input_right				
AUDIOINPUT				
output_left				
AUDIOOUTPUT				
output_right				
AUDIOOUTPUT				
scaled_dry_left				
AUDIOOUTPUT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
scaled_dry_right				
AUDIOOUTPUT				
input_level	-100	12	0	0.001
CONTROLINPUT	input_level			
output_level	-100	12	0	0.001
CONTROLINPUT	output_level			
wet_dry_mix	0	1	1	0.001
CONTROLINPUT	wet_dry_mix			
pre_delay	0	1000	25	1
CONTROLINPUT	pre_delay			
decay_time	0.2	100	3	0.1
CONTROLINPUT	decay_time			
position	0	30	13	1
CONTROLINPUT	position			
diffusion	0	15	10	1
CONTROLINPUT	diffusion			
low_eq_gain	-8	4	-4	1
CONTROLINPUT	low_eq_gain			
low_eq_cutoff	50	500	250	50
CONTROLINPUT	low_eq_cutoff			
high_eq_gain	-8	0	-4	1
CONTROLINPUT	high_eq_gain			
high_eq_cutoff	1000	8000	5000	500
CONTROLINPUT	high_eq_cutoff			
kill	0	2	0	1
CONTROLINPUT	kill			
is_bypassing	0	1	0	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	is_bypassing			
algorithm	0	5	0	1
CONTROLINPUT	algorithm			
input_mode				
CONTROLINPUT	input_mode			
internal_clip				
CONTROLOUTPUT	internal_clip			
delay_compensation				
CONTROLOUTPUT	delay_compensation			

reverb_a(rva) - Reverberator (12 delays)

Description

This is the first in a family of four reverb modules, reverb_a,b and c. These modules are of high, medium, low, and adjustable densities respectively. These modules each make fairly respectable reverberators. When they are combined in a patch with delays, diffusors, EQs, etc., they can be part of very high quality reverbs. These modules are all stereo in/stereo out and have control over decay (RT60), roomsize, predelay, and equalization. They also have built-in delay randomization that helps to reduce flutter and resonances. The user is also given access to the internal delays that make up the reverberator module so that infinite varieties of rooms may be created. In order to create a high quality room simulation, it is usually desirable to connect a pair of diffusor modules before the input to the reverberator. This provides a smooth attack to the reverb and provides a more natural build-up of reverberant energy.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	maximum amount of delay memory used by the module. Range is 100 to 32500

Audio Inputs

Header	Description
leftin	The left audio input.
rightin	The right audio input.

Audio Outputs

Header	Description
left	Left output
right	Right output

Control Inputs

Header	Description
gliderate	0 to 1.0. Controls the rate at which changes to the delay values will be "glided". Affects roomsize settings as well. 1 mean very fast.
moddepth	0 to 10 milliseconds. Adjusts the amount of delay randomization. Too large values may produce an overly chorussed effect.
modrate	0 to 1.0. Controls the speed at which the delays will change while being randomized. Again, too high values may result in noticable pitch shift.
modratespan	0 to 1.0. Adjusts the degree to which the different internal delays will be swept at different rates. A setting of zero will result in all delays swept at the same rate. Higher settings will spread out the sweep rate reducing the possibility of a build-up of a noticeable pitch shift.
decay	0 to 1000 seconds. Controls the overall RT60 of the reverb.

Header	Description
roomsize	0 to 1.0. Adjusts the relative roomsize of the reverb. A value of 1.0 would be the largest. The actual roomsize depends on both this parameter and the individual delay settings.
predelay	0 to maxdelay/12 milliseconds. Provides an overall delay before the reverberant effect.
low_freq	20 to 1000 Hertz. Controls the frequency at which the low frequency attenuation works.
high_freq	1000 to 20000 Hertz. Controls the frequency at which the high frequency attenuation works.
low_atten	-20 to 0 dB. Conrols how much frequencies below low atten will be attenuated. This is used to quiet an overly rumbly reverb.
high_atten	-20 to 0 dB. Conrols how much frequencies above high atten will be attenuated. This is used to diminish the high sizzle of the reverb and to produce a warmer, more natural sound.
delay[n]	0 to maxdelay/12 milliseconds. The 12 internal delays of this reverb. Adjust these to create different room characters.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	100	22000/32500/#speed/#if	600	10
	max_delay			
leftin				
LEFT				
AUDIOINPUT				
rightin				
RIGHT				
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				
AUDIOOUTPUT				
decay	0	1000	1	0.1
	sec			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	decay_			
roomsize	0	1	0.5	0.01
	%%			
CONTROLINPUT	size			
predelay	0	@maxde-lay/12/#-div/#trunc	1	0.1
	ms			
CONTROLINPUT	predelay			
low_freq	20	1000	100	0.1
	hz			
CONTROLINPUT	low_freq_			
high_freq	1000	20000	2000	0.1
	hz			
CONTROLINPUT	high_freq			
low_atten	-20	0	-1	1
	db			
CONTROLINPUT	low_decay_			
high_atten	-20	0	-2	1
CONTROLINPUT	high_decay			
moddepth	0	10	0	0.01
	ms			
CONTROLINPUT	moddepth			
modrate	0	1	0	0.01
	%%			
CONTROLINPUT	modrate			
modratespan	0	1	0	0.01
CONTROLINPUT				
gliderate	0	1	1	0.01
	%%			
CONTROLINPUT	glide_rate			
delay1	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay1			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
delay2	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay2			
delay3	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay3			
delay4	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay4			
delay5	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay5			
delay6	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay6			
delay7	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay7			
delay8	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay8			
delay9	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay9			
delay10	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay10			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
delay11	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay11			
delay12	0	@maxde-lay/12/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay12			

reverb_a

reverb_b(rvb) - Reverberator (8 delays)

Description

This is the second in a family of four reverb modules, reverb_a,b,c, and d. These modules are of high, medium, low, and adjustable densities respectively. These modules each make fairly respectable reverberators. When they are combined in a patch with delays, diffusors, EQs, etc., they can be part of very high quality reverbs. These modules are all stereo in/stereo out and have control over decay (RT60), roomsize, predelay, and equalization. They also have built-in delay randomization that helps to reduce flutter and resonances. The user is also given access to the internal delays that make up the reverberator module so that infinite varieties of rooms may be created. In order to create a high quality room simulation, it is usually desirable to connect a pair of diffusor modules before the input to the reverberator. This provides a smooth attack to the reverb and provides a more natural build-up of reverberant energy.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	maximum amount of delay memory used by the module

Audio Inputs

Header	Description
leftin	The left audio input.
rightin	The right audio input.

Audio Outputs

Header	Description
left	Left output
right	Right output

Control Inputs

Header	Description
gliderate	0 to 1.0. Controls the rate at which changes to the delay values will be "glided". Affects roomsize
moddepth	0 to 10 milliseconds. Adjusts the amount of delay randomization. Too large values may
modrate	0 to 1.0. Controls the speed at which the delays will change while being randomized. Again, too high values may result in noticable pitch shift.
modratespan	0 to 1.0. Adjusts the degree to which the different internal delays will be swept at different rates. A setting of zero will result in all delays swept at the same rate. Higher settings will spread out the
decay	0 to 1000 seconds. Controls the overall RT60 of the reverb.

Header	Description
roomsize	0 to 1.0. Adjusts the relative roomsize of the reverb. A value of 1.0 would be the largest. The actual roomsize depends on both this parameter and the individual delay settings.
predelay	0 to maxdelay/8 milliseconds. Provides an overall delay before the reverberant effect.
low_freq	20 to 1000 Hertz. Controls the frequency at which the low frequency attenuation works.
high_freq	1000 to 20000 Hertz. Controls the frequency at which the high frequency attenuation works.
low_decay	-20 to 0 dB. Conrols how much frequencies below low atten will be attenuated. This is used to quiet an overly rumbly reverb.
high_decay	-20 to 0 dB. Conrols how much frequencies above high atten will be attenuated. This is used to diminish the high sizzle of the reverb and to produce a warmer, more natural sound.
delay[n]	0 to maxdelay/8 milliseconds. The 8 internal delays of this reverb. Adjust these to create different room characters.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	100	22000/32500/#speed/#if	640	10
	max_delay			
leftin				
LEFT				
AUDIOINPUT				
rightin				
RIGHT				
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				
AUDIOOUTPUT				
decay	0	1000	1	0.1
	sec			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	decay_			
roomsize	0	1	0.5	0.01
	%%			
CONTROLINPUT	size__			
predelay	0	@maxdelay/8/#-div/#trunc	1	0.1
	ms			
CONTROLINPUT	predelay			
low_freq	20	1000	100	0.1
	hz			
CONTROLINPUT	low_freq_			
high_freq	1000	20000	2000	0.1
	hz			
CONTROLINPUT	high_freq			
low_decay	-20	0	-1	1
	db			
CONTROLINPUT	low_decay_			
high_decay	-20	0	-2	1
CONTROLINPUT	high_decay			
moddepth	0	10	0	0.01
	ms			
CONTROLINPUT	moddepth_			
modrate	0	1	0	0.01
	%%			
CONTROLINPUT	modrate__			
modratespan	0	1	0	0.01
CONTROLINPUT				
gliderate	0	1	1	0.01
	%%			
CONTROLINPUT	glide_rate			
delay1	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay1			
delay2	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	delay2			
delay3	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay3			
delay4	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay4			
delay5	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay5			
delay6	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay6			
delay7	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay7			
delay8	0	@maxdelay/8/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay8			

reverb_c(rvc) - Reverberator (6 delays)

Description

This is the third in a family of four reverb modules, reverb_a,b,c, and d. These modules are of high, medium, low, and adjustable densities respectively. These modules each make fairly respectable reverberators. When they are combined in a patch with delays, diffusors, EQs, etc., they can be part of very high quality reverbs. These modules are all stereo in/stereo out and have control over decay (RT60), roomsize, predelay, and equalization. They also have built-in delay randomization that helps to reduce flutter and resonances. The user is also given access to the internal delays that make up the reverberator module so that infinite varieties of rooms may be created. In order to create a high quality room simulation, it is usually desirable to connect a pair of diffusor modules before the input to the reverberator. This provides a smooth attack to the reverb and provides a more natural build-up of reverberant energy.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	maximum amount of delay memory used by the module

Audio Inputs

Header	Description
leftin	The left audio input.
rightin	The right audio input.

Audio Outputs

Header	Description
left	Left output
right	Right output

Control Inputs

Header	Description
gliderate	0 to 1.0. Controls the rate at which changes to the delay values will be "glided". Affects roomsize
moddepth	0 to 10 milliseconds. Adjusts the amount of delay randomization. Too large values may
modrate	0 to 1.0. Controls the speed at which the delays will change while being randomized. Again, too high values may result in noticable pitch shift.
modratespan	0 to 1.0. Adjusts the degree to which the different internal delays will be swept at different rates. A setting of zero will result in all delays swept at the same rate. Higher settings will spread out the
decay	0 to 1000 seconds. Controls the overall RT60 of the reverb.

Header	Description
roomsize	0 to 1.0. Adjusts the relative roomsize of the reverb. A value of 1.0 would be the largest. The actual roomsize depends on both this parameter and the individual delay settings.
predelay	0 to maxdelay/6 milliseconds. Provides an overall delay before the reverberant effect.
low_freq	20 to 1000 Hertz. Controls the frequency at which the low frequency attenuation works.
high_freq	1000 to 20000 Hertz. Controls the frequency at which the high frequency attenuation works.
low_decay	-20 to 0 dB. Conrols how much frequencies below low atten will be attenuated. This is used to quiet an overly rumbly reverb.
high_decay	-20 to 0 dB. Conrols how much frequencies above high atten will be attenuated. This is used to diminish the high sizzle of the reverb and to produce a warmer, more natural sound.
delay[n]	0 to maxdelay/6 milliseconds. The 6 internal delays of this reverb. Adjust these to create different room characters.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	100	22000/32500/#speed/#if	600	10
	max_delay			
leftin				
LEFT				
AUDIOINPUT				
rightin				
RIGHT				
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				
AUDIOOUTPUT				
decay	0	1000	1	0.1
	sec			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	decay_			
roomsize	0	1	0.5	0.01
	%%			
CONTROLINPUT	size__			
predelay	0	@maxdelay/6/#-div/#trunc	1	0.1
	ms			
CONTROLINPUT	predelay			
low_freq	20	1000	100	0.1
	hz			
CONTROLINPUT	low_freq_			
high_freq	1000	20000	2000	0.1
	hz			
CONTROLINPUT	high_freq			
low_decay	-20	0	-1	1
	db			
CONTROLINPUT	low_decay_			
high_decay	-20	0	-2	1
CONTROLINPUT	high_decay			
moddepth	0	10	0	0.01
	ms			
CONTROLINPUT	moddepth_			
modrate	0	1	0	0.01
	%%			
CONTROLINPUT	modrate__			
modratespan	0	1	0	0.01
CONTROLINPUT				
gliderate	0	1	1	0.01
	%%			
CONTROLINPUT	glide_rate			
delay1	0	@maxdelay/6/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay1			
delay2	0	@maxdelay/6/#-div/#trunc	20	0.1
	ms			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	delay2			
delay3	0	@maxdelay/6/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay3			
delay4	0	@maxdelay/6/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay4			
delay5	0	@maxdelay/6/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay5			
delay6	0	@maxdelay/6/#-div/#trunc	20	0.1
	ms			
CONTROLINPUT	delay6			
obj				
USEROBJECTPARENT	obj			

reverb_d(rvd) - Reverberator (n delays)

Description

This is the last in a family of four reverb modules, reverb_a,b,c, and d. These modules are of high, medium, low, and adjustable densities respectively. These modules each make fairly respectable reverberators. When they are combined in a patch with delays, diffusors, EQs, etc., they can be part of very high quality reverbs. These modules are all stereo in/stereo out and have control over decay (RT60), roomsize, predelay, and equalization. They also have built-in delay randomization that helps to reduce flutter and resonances. The user is also given access to the internal delays that make up the reverberator module so that infinite varieties of rooms may be created. In order to create a high quality room simulation, it is usually desirable to connect a pair of diffusor modules before the input to the reverberator. This provides a smooth attack to the reverb and provides a more natural build-up of reverberant energy.

Godlike Productions Comments

Specifiers

Header	Description
maxdelay	maximum amount of delay memory used by the module
numdelays	0 to 32. Total number of delaylines, must be a multiple of 4

Audio Inputs

Header	Description
leftin	The left audio input.
rightin	The right audio input.

Audio Outputs

Header	Description
left	Left output
right	Right output

Control Inputs

Header	Description
gliderate	0 to 1.0. Controls the rate at which changes to the delay values will be "glided". Affects roomsize
moddepth	0 to 10 milliseconds. Adjusts the amount of delay randomization. Too large values may
modrate	0 to 1.0. Controls the speed at which the delays will change while being randomized. Again, too high values may result in noticable pitch shift.
modratespan	0 to 1.0. Adjusts the degree to which the different internal delays will be swept at different rates. A setting of zero will result in all delays swept at the same rate. Higher settings will spread out the
decay	0 to 1000 seconds. Controls the overall RT60 of the reverb.

Header	Description
roomsize	0 to 1.0. Adjusts the relative roomsize of the reverb. A value of 1.0 would be the largest. The actual roomsize depends on both this parameter and the individual delay settings.
predelay	0 to maxdelay/numdelays milliseconds. Provides an overall delay before the reverberant effect.
low_freq	20 to 1000 Hertz. Controls the frequency at which the low frequency attenuation works.
high_freq	1000 to 20000 Hertz. Controls the frequency at which the high frequency attenuation works.
low_decay	-20 to 0 dB. Conrols how much frequencies below low atten will be attenuated. This is used to quiet an overly rumbly reverb.
high_decay	-20 to 0 dB. Conrols how much frequencies above high atten will be attenuated. This is used to diminish the high sizzle of the reverb and to produce a warmer, more natural sound.
delay[n]	0 to maxdelay/numdelays milliseconds. The 6 internal delays of this reverb. Adjust these to create different room characters.

Control Outputs

Header	Description

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
maxdelay	100	22000/32500/#speed/#if	2000	10
	max_delay			
numdelays	4	32	12	4
INT	num_delays			
leftin				
LEFT				
AUDIOINPUT				
rightin				
RIGHT				
AUDIOINPUT				
left				
LEFT				
AUDIOOUTPUT				
right				
RIGHT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
AUDIOOUTPUT				
decay	0	1000	1	0.1
	sec			
CONTROLINPUT	decay_			
roomsize	0	1	0.5	0.01
	%%			
CONTROLINPUT	size__ *100			
predelay	0	@maxdelay/@numdelays/#-div/#trunc	1	0.1
	ms			
CONTROLINPUT	predelay			
low_freq	20	1000	100	0.1
	hz			
CONTROLINPUT	low_freq_			
high_freq	1000	20000	2000	0.1
	hz			
CONTROLINPUT	high_freq			
low_atten	-20	0	-1	1
	db			
CONTROLINPUT	low_decay_			
high_atten	-20	0	-2	1
CONTROLINPUT	high_decay			
moddepth	0	10	0	0.01
	ms			
CONTROLINPUT	moddepth_			
modrate	0	1	0	0.01
	%%			
CONTROLINPUT	modrate__			
modratespan	0	1	0	0.01
CONTROLINPUT				
gliderate	0	1	1	0.01
	%%			
CONTROLINPUT	glide_rate			
delay	0	@maxdelay/@numdelays/#-div/#trunc	20	0.1

Name	**Min**	**Max**	**Default**	**Resolution**
Type (A)	**Units**	**Size**	**Choices**	
Type	**Description**			
	ms			
CONTROLINPUT	delay~n_			
obj				
USEROBJECTPARENT	obj			

stereo_room:

Description

Implementation of the stereo_room algorithm from the Eventide SP2016 and Reverb 2016 hardware units. A popular and efficient self-contained stereo reverb with unique controls for diffusion and position and equalization.

Godlike Productions Comments

Specifiers

Header	Description
declick	Declicking on or off

Audio Inputs

Header	Description
input_left	The left audio input to be processed, see input mode control for more info.
input_right	The right audio input to be processed, see input mode control for more info.

Audio Outputs

Header	Description
output_left	left reverberant output
output_right	right reverberant output

Control Inputs

Header	Description
input_level	0 to 1, relative level
wet_dry_mix	0 to 1, Dry/wet mix using a taper from the Reverb 2016. 0 is all dry, 1 is all wet
pre_delay	0 to 125 msecs, Predelay in milliseconds
decay_time	0 to 30 secs, Decay for tail (approximate)
position	0 to 30, Front/rear mix. 0 simulates listener position in the "front" of the room or close the source postion, 30 simulates listener position in the "rear" of the room or far from the source position. This is all assumed on the wet channel, mixing in dry signal will of course always give the illusion of added proximity to the source.
diffusion	0 to 15, increases the echo density with the most density at 15 and least at 0.
low_eq_gain	Gain for low shelf filter in dB. Inactive for SP2016 Plate.
low_eq_cutoff	Corner frequency for low shelf filter in increments of 50 Hz from 50 to 500 Hz. Inactive for SP2016 Plate.
high_eq_gain	Gain for high shelf filter in dB. Inactive for SP2016 Plate.
high_eq_cutoff	Corner frequency for high shelf filter in increments of 50 Hz from 1000 to 8000 Hz. Inactive for SP2016 Plate.

Header	Description
kill	0 or 1, 1 Kills the input while the reverb tail continues
is_bypassing	0 or 1, 1 Activates bypassing
input_mode	0 to 3, 0: stereo input, 1: input left is main mono input, 2: input right is main mono input, 3. sum to mono from input left and right

Control Outputs

Header	Description
internal_clip	Peak amplitude

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
declick	0	1	1	1
	declicking			
input_left				
AUDIOINPUT				
input_right				
AUDIOINPUT				
output_left				
AUDIOOUTPUT				
output_right				
AUDIOOUTPUT				
input_level	0	1	1	0.001
CONTROLINPUT	input_level			
wet_dry_mix	0	1	1	0.001
CONTROLINPUT	wet_dry_mix			
pre_delay	0	125	12	1
CONTROLINPUT	pre_delay			
decay_time	0	30	10	1

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLINPUT	decay_time			
position	0	30	13	1
CONTROLINPUT	position			
diffusion	0	15	10	1
CONTROLINPUT	diffusion			
low_eq_gain	0	12	8	1
CONTROLINPUT	low_eq_gain			
low_eq_cutoff	0	9	6	1
CONTROLINPUT	low_eq_cutoff			
high_eq_gain	0	8	2	1
CONTROLINPUT	high_eq_gain			
high_eq_cutoff	0	14	5	1
CONTROLINPUT	high_eq_cutoff			
kill	0	1	0	1
CONTROLINPUT	kill			
is_bypassing	0	1	0	1
CONTROLINPUT	is_bypassing			
input_mode				
CONTROLINPUT	input_mode			
internal_clip				
CONTROLOUTPUT	internal_clip			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

Chapter 16 - System

headm(in) - Multichannel Header Module

Description

The head module is required to appear once and only once in every patch. It contains the inputs from the A/Ds (or digital ins) and the outputs to the D/A converters (and digital outs). A user is only aware of the existence of this module if he is creating a patch with a .sig file. In this case, the head module must be the first entry in the sigfile. In addition to the inputs and outputs, the head module contains the root collection under which all user objects appear.

Godlike Productions Comments

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
ninputs	2	32	2	1
	number_inputs			
noutputs	2	32	2	1
INT	number_outputs			
in				
MONO		@ninputs		
AUDIOOUTPUT				
out				
MONO		@noutputs		
AUDIOINPUT				
null				
AUDIOOUTPUT				
zero				
CONTROLOUTPUT				
global1				
CONTROLOUTPUT				
global2				
CONTROLOUTPUT				
global3				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
CONTROLOUTPUT				
global4				
CONTROLOUTPUT				
name				
STRING				
tag				
STRING				
nkids				
INT				
kids				
		@nkids		
USEROBJECTCHILD	kids			
nullobj				
USEROBJECTPARENT				

remoter(rem) - Remote control module for preset params

Description

Godlike Productions Comments

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
remspec_key				
remspec_key_low				
INT				
remspec1				
INT				
remspec2				
INT				
remspec3				
INT				
remspec4				
INT				
offset				
INT				
maxi				
INT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

remoter(rem) - Remote control module for preset params

Description

Godlike Productions Comments

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			
nentries				
in	0	@nentries/1/#sub	0	1
CONTROLINPUT				
offset			0	
		nentries		
INT				
scale			0	
		nentries		
INT				
remspec_key			0	
INT				
remspec_key_low			0	
INT				
remspec1			0	
INT				
remspec2			0	
INT				
remspec3			0	
INT				
remspec4			0	
INT				

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

tail

Description

A special case module for internal use only. Serves as the end of a sigfile. No specifiers, inputs or outputs

Godlike Productions Comments

Module Entries

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

Name	Min	Max	Default	Resolution
Type (A)	Units	Size	Choices	
Type	Description			

Chapter 17 - Change Log

Rev 1.11

Control Process -> c_count2
Changed the title (and updated TOC) and added comment.
Delay -> bucket_brigade_delay
Added new dsp block that came with VSIG 3.4.2
Detector -> multimeter
Added new dsp block that came with VSIG 3.4.2
Filter -> distort
Renamed to distort and added new Prevent Wave Folding parameter introduced in VSIG 3.4.2
Filter -> modiirq
Added a note about deprecation as of VSIG 3.4.2
Interface -> footswitch
Added new dsp block that came with VSIG 3.4.2
Interface -> hfader
Updated with changes in VSIG 3.4.2
Interface -> hmonitor
Updated to match VSIG 3.4.2. There are still bugs.
Interface -> Taperknob
Added new dsp block that came with VSIG 3.4.2
Interface -> tapknob
Added note about deprecation in VSIG 3.4.2
Interface -> vmonitor
Updated to match VSIG 3.4.2. There are still bugs.
Miscellaneous -> transientsplit
Updated comment noting that this dsp block is missing in 3.4.2.
Mixer -> stereopanner
Added new dsp block that came with VSIG 3.4.2

Rev 1.10

Detector -> envdetect
Added comment about crashes caused by this block.
Detector -> peak
Updated description to include the Type parameter and comment about linear state machine not being effective for envelope follower.

Rev 1.09

Control Math -> c_sincos
Added comment about expecting the input in turns.

Rev 1.08

Math -> xformer
Added links to papers on transformers which may help with utilizing this block. Added other notes.

Rev 1.07

Control Math -> c_ptof
Added minimum range and notes about block accepting negative values.
Filter -> distortion
Added distortion curves
Oscillator -> H3000 Oscillator

Updated notes about retriggering waveforms.

Rev 1.06

Interface -> hmonitor
Added comment about bug that will crash emote if a log function reads a 0.
Interface -> vmonitor
Added comment about bug that will crash emote if a log function reads a 0.
Control Math -> c_log
Added comment about bug that will crash emote if a log function reads a 0.
Control Math -> c_lin2db
Added comment about bug that will crash emote if a log function reads a 0.

Rev 1.05

Interface -> multiknob
Added comment about major bug that will crash your H9000. Don't use this knob for now.

Rev 1.04

Dynamics -> ducker
Added comment about potential bug with this block. Use compressor instead.

Rev 1.03

Interface -> hmonitor
Added comment about emote crashes caused by high monitor rates. Use peak detect to avoid crashes.
Interface -> vmonitor
Added comment about emote crashes caused by high monitor rates. Use peak detect to avoid crashes.

Rev 1.02

Math -> Introduction
Added introduction to explain the use of constant block.

Rev 1.01

Math -> Multiply
Updated Godlike Productions Comments
Math -> Comparatator 2
Updated Godlike Productions Comments

Rev 1.00

Initial Release