Instructions for Use.

Introduction.

General.

'Klean Boy' is a tool that can be used to reduce broadband noise and hum from stereo digital recordings. Typical applications include restoration of audio from tape recordings and vinyl. Where most other similar tools require to learn the noise floor before correction can begin, 'Klean Boy' provides a choice between manual sampling of the noise floor or fully automatic continuous and adaptive extraction of the noise floor. The automatic extraction mode provides the advantage that it can be used on material where no easy sampling of the noise is possible. It is common in such applications that the noise reduction process also removes a considerable portion of the signal of interest. 'Klean Boy' provides a number of proprietary algorithms that are designed to ensure the noise is reduced while leaving as much of the signal as possible. Control of key parameters enables optimization to a broad range of material. Above all, 'Klean Boy' is designed to be easy and intuitive to use.

Features.

- Manual or fully Automatic high quality broadband noise and hum reduction from any stereo digital audio material.
- Floating point signal processing with Over-sampling.
- Manual mode for use when a noise floor sample is available or preferred.
- Fully Automatic mode learns and adapts the noise floor in real time, extracting the noise floor even from the signal of interest.
- A choice of 4 noise reduction algorithms.
- Elevayta's unique 7 band panning EQ.
- Full flexibility over key processing parameters enables easy optimization to suite material.
- Tools tips and context sensitive GUI.
- Full PDC Compensation (in compatible hosts).

This brief guide is intended to provide some further insight into the usage and options available in 'Klean Boy'.

Document Release Information.

Product:	Klean Boy
Product Versions:	V5.04
Doc. Version:	1.2
Doc. Release Date:	January 3, 2008
Author:	P.R.H.

The 'Manual' Page.

Figure 1 – The 'Manual' Page.

This page is focused on **manual mode** operation. In **manual mode** it is necessary to manually capture a noise floor sample before noise reduction can be performed. Manual capture of the noise floor is performed by playing a portion of noise-only audio (if available from the audio source) and pressing the 'Sample Noise' button once to initiate capture and once again to stop capture. The portion of noise-only audio should be, preferably, a few seconds long and can usually be found at the beginning, or between songs, of vinyl or tape recordings. If such a portion is not available, then automatic mode is more appropriate (see below).

The parameters.

'Master Pan'	Controls the over-all audio panning.
'Master Gain'	Controls the over-all audio gain.
'Tone NR'	'Tone NR' provides noise reduction for specific tones as detected during the noise sampling. To be useful, a noise curve must already have been captured (either manually or automatically). Activating this parameter reveals a list of the 6 most dominant tones detected. These tones may be suppressed by selecting the displayed checkbox (see below).
'Bypass'	When activated, all processing is bypassed leaving only the input signal.
'Noise Listen'	Listen only to the noise that is removed by 'Klean Boy'.
'Sample Noise'	Available in manual mode. This parameter activates noise floor capturing when a portion of noise-only audio data is available. Use this to capture the noise floor from a portion of noise-only audio data.

'Left Vertical Slider'	Adjust the noise floor baseline for **manual mode**. Allows to fine tune the manually captured noise floor by adding or subtracting a baseline. Value appears in status display when mouse is over.
'Right Vertical Slider'	Adjusts the noise reduction level parameter for **manual mode**. Can be used to fine tune the performance of the noise reduction algorithm. Normally this does not require modification from default (60%). Value appears in status display when mouse is over.

The 'Auto' Page.

Figure 2 – The 'Auto' page.

This page is focused on **automatic mode** operation. In **automatic mode** the noise floor is determine automatically and continuously as long as the 'Learn Noise' button is pressed. If desired, an automatically determined noise floor can be frozen and used in **manual mode** by pressing the 'Freeze' button. For **automatic mode** it is not required to have a separate portion of noise-only audio data since the noise floor is determined directly from the playing audio. As long as the 'Learn Noise' button is pressed, 'Klean Boy' will operate in **automatic mode** until either the button is depressed or the 'Freeze' button is pressed. The horizontal dashed green line indicates the mean noise level baseline.

The parameters.

'Analysis Window' — In automatic mode, 'Klean Boy' uses an adaptive algorithm to determine the baseline noise level. The 'Analysis Window' parameter determines the duration of the time averaging window that is used to determine the noise level.

'Sensitivity' — The adaptive noise detection algorithm uses a time averaging process. The sensitivity parameter determines the sensitivity with which the algorithm reacts to changes in the noise curve. At the 0% setting (most sensitive), the noise detection algorithm reacts to every small change in noise level. At the 100% setting, the noise detection algorithm is reacting more slowly (least sensitive).

'Smooth' — When activated, the noise floor curve is smoothed.

'Freeze' — This parameter is available in **automatic mode**. When activated, 'Freeze' stores the currently displayed noise floor and switches to 'Klean Boy' to **manual mode**. Processing continues in manual mode using the captured noise floor. This features allows to use **automatic mode** to continuously capture a noise floor. When the user is satisfied that the automatic noise floor is good enough, they can

	freeze (store) this noise floor and use it in **manual mode** (i.e. so that it doesn't change).
'Process…'	The process box allows to select from 4 different noise reduction algorithms. Each algorithm has slightly different characteristics and varying CPU penalty. The result in most cases are similar. 'Process C' is a general all purpose algorithm.
'Learn Noise'	This parameter activates **automatic mode** and initiates the noise floor learning process. When activated, 'Klean Boy' continuously learns the noise floor directly from the audio signal (i.e. in place of the manually recorded noise floor).
'Left Vertical Slider'	Adjust the noise floor baseline for **automatic mode**. Allows to fine tune the automatically determined noise floor by adding or subtracting a baseline. Value appears in status display when mouse is over.
'Right Vertical Slider'	Adjusts the noise reduction level parameter for **automatic mode**. Can be used to fine tune the performance of the noise reduction algorithm. Normally this does not require modification from default (60%). Value appears in status display when mouse is over.

The 'EQ' Page.

Figure 3 – The 'EQ' page.

The unique 'EQ' functionality of 'Klean Boy' provides 7 frequency bands for adjusting, and panning, the EQ on the processed (wet) signal. The dry signal is unaffected by the EQ settings. This enables to adjust the timbre of the processed signal.

The vertical sliders control the gain/attenuation for each band and the horizontal sliders enable to pan the frequencies from that band to the left or right as desired.

The EQ can be enabled or disabled by a single button press for comparison purposes. When enabled, a green display is visible in the button bar (can be seen from all windows).

Note: To reset any EQ slider to its center position, just click the mouse on the numerical display of that slider.

All EQ settings can be saved and recalled as usual.

The parameters.

'Enable EQ' Enable/disable the EQ. This can also be done by clicking on the green 'EQ' in the status window.

The 'About' Page.

Figure 4 – The 'About' page.

Shows the latency in ms, credits, and the registration information. The 'About' page also contains a number of parameters.

The parameters.

'Reset'	Use this to reset curves and parameters.
'Oversample'	Enable/disable over-sampling in the frequency domain processing stage. Over-sampling can optimize sound quality but at the expense of higher CPU usage.
'8192…'	Select the length of the processing chunk (in samples). Smaller uses lower CPU but can lead to increased artifact level.
'Humidity'	Adjusts the mix of dry to wet (processed to unprocessed) audio signal in the output. The default (and normal) setting should be 100%. Under normal use, there is no need to adjust this parameter.

The Basic Audio Cleaning Process.

The audio clean process involves the two basic steps of:

1. Learn the noise characteristics of the audio signal – the noise floor.

2. Reduce the noise from the audio signal.

'Klean Boy' provides the means to perform both steps in as simple or sophisticated manner as desired. The first step can be performed either manually or fully automatically depending on the availability of a portion of the audio material that contains only noise. The second step is performed automatically once the noise floor is known.

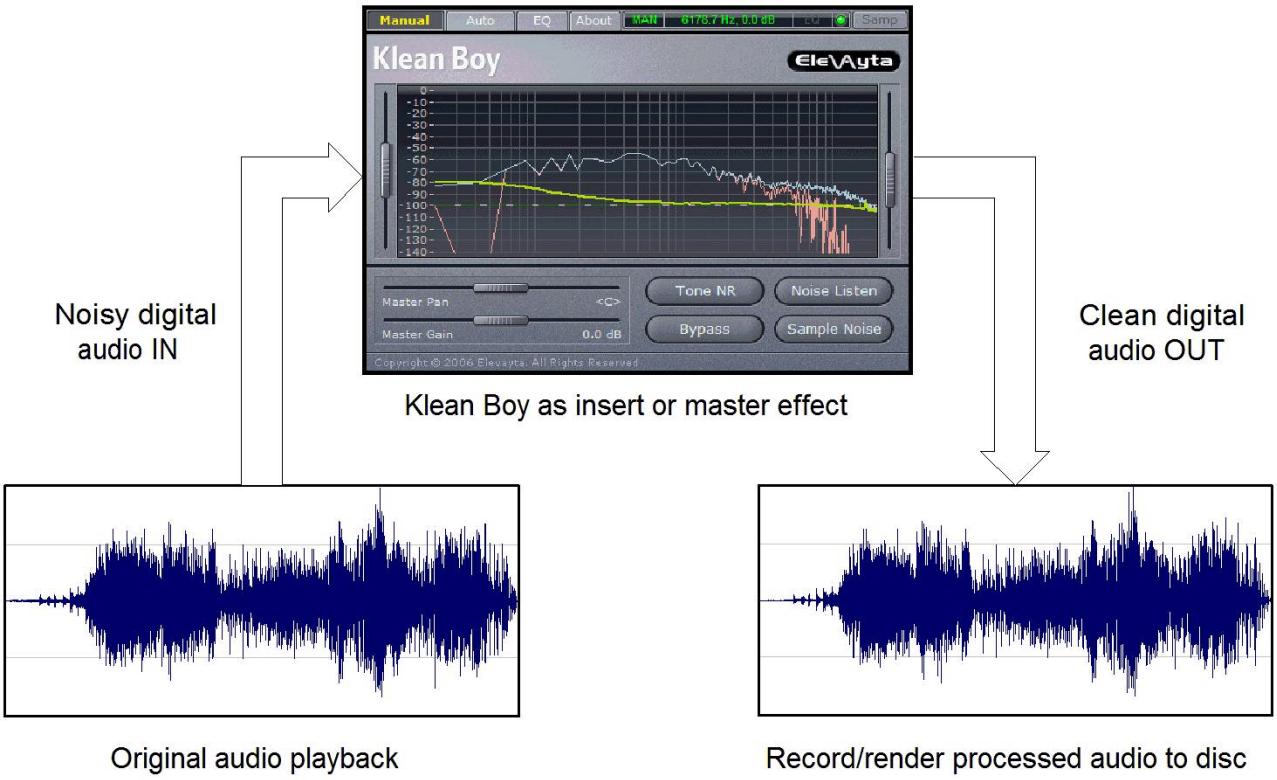

Figure 5 – The basic audio cleaning process.

The basic **manual** audio cleaning process involves the following steps:

1. Load the audio track that you wish to process into your VST compatible Digital Audio Workstation (DAW).
2. Assign 'Klean Boy' as an insert or master effect on the same track.
3. Select the desired quality preset.
4. Select a region of the audio track that contains just noise.
5. Playback the noise while the 'Sample Noise' button is pressed and, after some averaging time, capture the recorded noise floor by de-selecting the 'Sample Noise' button.
6. With the noise floor captured, continue playback or render/record the whole audio track to a new file on disc.

The basic **automatic** audio cleaning process involves the following steps:

1. Load the audio track that you wish to process into your VST compatible Digital Audio Workstation (DAW).
2. Assign 'Klean Boy' as an insert or master effect on the same track.
3. Select the desired quality preset.
4. Playback the track while the 'Learn Noise' button is pressed. The noise floor will be determined automatically and continuously.
5. Choose to freeze the noise floor or use the dynamically updated noise floor.
6. With the noise floor determined, continue playback or render/record the whole audio track to a new file on disc.

For more details, see the following sections.

How to Perform Manual Noise Reduction.

Figure 6 – Starting the noise floor capture.

This chapter describes how to manually capture the noise floor from a digital recording and use the captured noise floor to reduce the broadband noise from the whole recording. It is assumed that the digital recording that you wish to clean contains a section (at beginning, end or between songs) that contains only noise. This section of noise should last for a duration of 1 or 2 seconds and is used to determine the noise floor.

It is assumed that your VST compatible Digital Audio Workstation (DAW) is already loaded with the chosen audio material and that 'Klean Boy' is loaded as an insert or master effect in such a way that the audio signal passes through 'Klean Boy' and can be subsequently recorded back to disc (after processing by 'Klean Boy').

Figure 7 – After a successful noise floor capture.

Step 1: The first step is to identify the region of noise-only in the audio material. When this region has been found, start audio playback from the start of the noise-only

region and press the 'Sample Noise' button to begin capturing the noise floor. When all is working as required, the display will look as in figure 6 above.

Step 2: Before the noise-only region finishes, press the 'Sample Noise' button again to terminate noise floor capture and store the noise floor. When all is working as required, a thick green line will appear indicating the shape of the captured noise floor. See figure 7.

After the noise floor has been captured manually, simply continue to play the audio material through 'Klean Boy'. The captured noise floor will remain fixed and is used as the basis for reducing the broadband noise.

Note: Once the 'Sample Noise' button is switched off (and a noise floor curve exists) 'Klean Boy' will now be operating in noise reduction mode.

No other actions are required. You should hear a dramatic reduction in the level of noise without significant impact on the quality of the desired audio material.

Figure 8 – The normal noise removal process.

Figure 8 shows the noise removal process in action. The thick green line is the captured noise floor. The blue line is the original audio signal and the pink line is the audio signal after noise removal.

With the noise floor determined, continue playback or render/record the whole audio track to a new file on disc.

If you wish to compare with and without noise removal, press the 'Bypass' button. If you wish to hear just the noise that is being removed, press the 'Noise Listen' button.

How to Perform Automatic Noise Reduction.

Figure 9 – Automatic noise reduction. Simply select the 'Auto' page and press 'Learn Noise'.

This chapter describes how to automatically reduce the broadband noise from a digital recording. Because the noise floor is determined in an automatic fashion there is no requirement that the audio material has a noise-only region for noise floor sampling. In automatic mode, the noise floor is determined continuously and dynamically from the audio material of interest. If decided, the automatically determined noise floor can be allowed to change dynamically throughout the whole audio track or can be frozen and used just as would a manually captured noise floor.

Figure 10 – Automatic noise reduction.

It is assumed that your VST compatible Digital Audio Workstation (DAW) is already loaded with the chosen audio material and that 'Klean Boy' is loaded as an insert or master effect in such a way that the audio signal passes through 'Klean Boy' and can be subsequently recorded back to disc (after processing by 'Klean Boy').

For automatic noise reduction the second (Auto) page of 'Klean Boy' should be selected. There is just one step for automatic noise reduction.

Step 1: The only step required to perform fully automatic noise reduction is to press the 'Learn Noise' button, as shown in figure 9. This can be done before or during audio playback.

After activating the 'Learn Noise' mode and starting audio playback, 'Klean Boy' will begin to learn, in real time, the noise floor from the audio track. As soon as there is enough information to construct a representative noise floor, the thick green line will appear and noise reduction will commence as shown in figure 10.

Note: Once the noise floor (thick green line) is displayed 'Klean Boy' will now be operating in noise reduction mode.

As long as the 'Learn Noise' button is pressed, 'Klean Boy' will continue to dynamically update the noise floor as it receives new input from the audio material. It is possible to allow continuous learning throughout the whole noise reduction process. However, if, after some time, the noise floor estimate looks stable and the audio results are satisfactory, the user can choose to freeze the noise floor. When the noise floor is frozen (by pressing the 'Freeze' button) 'Klean Boy' reverts to manual mode using the frozen noise floor, as shown figure 11.

Figure 11 – Freezing the noise floor and reverting to manual mode.

With the noise floor determined, continue playback or render/record the whole audio track to a new file on disc.

If you wish to compare with and without noise removal, press the 'Bypass' button. If you wish to hear just the noise that is being removed, press the 'Noise Listen' button.

Using The 'Tone NR' Feature.

The 'Tone NR' function provides automatic detection, and reduction, of constant tone noise in a recording i.e. 50 Hz hum or tape hum.

Figure 12 – Activating the 'Tone NR' function.

The 'Tone NR' function is available only after a noise floor has been captured. Once a captured noise floor is available, pressing the 'Tone NR' button will reveal a list of the six most prominent frequencies in the audio material being played through 'Klean Boy'. In the example, shown figure 12, we can see 21.5 Hz and 48.4 Hz. To suppress/reduce these frequencies simply select the tick box next to the frequency, as shown in figure 13.

Figure 13 – Selecting frequencies to suppress.

The selected suppressed frequencies are indicated on the frequency plot by the vertical blue dashed lines.

The 'Klean Boy' Presets.

Figure 14 – The Presets.

'Klean Boy' is supplied with 12 presets. The presets cover a range of qualities and CPU loads. It is best to experiment to find the preset that delivers the preferred quality at the minimum CPU.

For general use, the 'Standard – MAN – Oversample' or 'Standard – AUTO – Oversample' should yield acceptable results.

FAQs.

1) Why is there such a long latency when using 'Klean Boy' in a host that doesn't support PDC?

A: 'Klean Boy' uses sophisticated processing that cannot be performed without some latency. In a PDC compliant host, this is not a problem.

Summary.

This guide was intended to provided a brief introduction to the features of 'Klean Boy'. As you, the user, gain experience using this plug-in, many more possibilities will become apparent. Your own imagination is the only restriction.

Constructive feedback is always appreciated. Please send comments and suggestions to:

info@elevayta.com

Enjoy this product and look out for more to come from Elevayta.

Disclaimer.

This software 'Klean Boy' is provided 'as is' without warranty of any kind. The author makes no guarantee of correctness, accuracy, reliability, safety or performance. You alone are responsible for determining if this software is safe for use in your environment. Neither the author, Elevayta, nor anyone else who has been involved in the creation or delivery of this product shall be liable for any direct, indirect, consequential, or incidental damages arising from the use or inability to use such product even if the author has been advised of the possibility of such damages.

Copyright Statement.

A Message for the Software Pirates.

Some elements of the code used in this software have been bought, by me, from small scale developers in Russia, Chile and China. Some software is expensive, I agree – but not this. It's a challenge to crack the code and show off – if that's what turns you on. But, come on now, what's so impressive about cracking $20 worth of code? Why not spend that time being a bit more creative and generate some useful software for somebody else. Who knows, if you go about it in the right way, perhaps somebody will want to buy your software and you can earn some money for yourself! That would be much more constructive than using your talents to destroy the business of others. I put money back in to the world software development pool, and that directly benefits the small scale developers in an environment dominated by large corporations – don't destroy my business – use your imagination more constructively – don't take that opportunity away from yourself – I might want to buy your software also one day!