

**A Complete Guide to
Plogue chipcrusher®**

Copyright © 2013 Plogue Art et Technologie, Inc. All rights reserved.



Including the ARIA™ Engine

Copyright © 2005-2013 by MakeMusic and Plogue Art et Technologie, Inc. All rights reserved.

| | |
|----------------------------------|---|
| Produced by: | David Viens |
| ARIA Engine Design/Development: | Plogue Art et Technologie, Inc. |
| Sound Programming: | David Viens Eric Patenaude |
| Hardware research and modeling : | David Viens |
| In house Testing and QA: | Eric Patenaude |
| Zero Latency Convolution | Jeff Hurchalla |
| UI Design | Eric Patenaude, Flavours of Lime |
| Logo | http://kikencorp.com |
| Original ARIA Manual Template: | Gary Garritan |
| Operations: | Max Deland |

The information in this document is subject to change without notice and does not represent a commitment on the part of Plogue Art et Technologie, Inc. No part of this publication may be copied, reproduced or otherwise transmitted or recorded, for any purpose, without prior written permission by Plogue Art et Technologie, Inc.

Plogue chipcrusher® is a registered trademark of Plogue Art et Technologie, Inc. Use of Plogue chipcrusher® software instrument and the contents herein are subject to the terms and conditions of the license agreement distributed with the program. You should carefully read the license agreement before using this product. The sounds presented in Plogue chipcrusher® are protected by copyright and cannot be distributed, whether modified or unmodified. The Guide to Plogue chipcrusher contained herein are also covered by copyright. ARIA™ is a trademark of MakeMusic Inc and Plogue Art et Technologie Inc . The ARIA Engine is copyrighted by Plogue Art et Technologie and MakeMusic Inc.

Any other trademarks of third-party programs are trademarks of their respective owners. The information contained herein may change without notice and does not represent a commitment on the part of Plogue Art et Technologie, Inc.

Plogue Art et Technologie, Inc.
Montreal QC, Canada.
e-mail: info@plogue.com
Visit us on the World Wide Web: www.plogue.com

Welcome to Plogue chipcrusher

chipcrusher has been in the pipes for nearly as long as chipsounds itself.

It was a challenge to fit its concepts into a musically useful effect, and many iterations of features were considered.

Over the years some features were included directly into chipsounds updates (the Amiga DAC really shines in the context of a synth) and others were put on ice until we could have the time to do them right (analog non-linearities).

I kind of forget now how long and tedious some of those captures were. Running test tones through old speakers and devices required dismantling, soldering, placing into an anechoic space, setting up microphones, record, deconvolve, crop, and finally build the presets.

DAC Emulation needed oscilloscope and logic analyzer dumps, custom code and fair amount of head scratching over obscure long-expired patents.

I don't think there is anything like this quite honestly. There are other 'Bit Crushers' out there, but nearly all of these only emulate LPCM encodings. There are other Speaker IR libraries, but nothing that covers classic gaming hardware, monitors, and vintage synthesizer speakers.

Please enjoy, and thanks for supporting our crazy research, which BTW is more alive than ever!

... well I'm not going to go over details of the four other retro digital products I have in mind right now right?

David Viens

June 2013

What is chipcrusher?

Plogue chipcrusher is an audio effect which plays back input audio through a simulation of early lofi digital audio codecs (**DAC Encoding**), adds grit (**Background Noise**), and outputs through a selection of speaker and filter impulse responses (**Post-Processing**).

chipcrusher can work as a plug-in for most major sequencing audio programs.

Since there is a wide spectrum of results achievable with chipcrusher, we have targeted a few use cases:

- 1) You just want to uniquely destroy/mangle a beat, a guitar or any other audio track.
- 2) You want to play single hits and emulate the sound of old samplers.
- 3) You want to add 'accurate dirt' to chipsounds's output.(*)

(*) Quite a bit of users have told us that the outputs of chipsounds were too 'clean' and would have liked some of the characteristic dirt present in recordings of original consoles. For this use particular case, make sure chipsounds dry audio output is sent to chipcrusher, and turn **DAC Encoding** off.

There is a misconception that if you send any audio track into a bit crusher that it magically makes it '8bit'. This is neither true, nor the purpose of this plugin.

The R&D behind this product was quite significant. We had to modify each and every device we have included one way or another to get the required impulse responses, noise, and test recordings.

What's Included

This 1.0 version of chipcrusher includes the following:

- Download file containing the complete chipcrusher software.
- The unique graphical license key for the product.
- This digital PDF manual.

Before you begin installation, make sure you have read the End User Licensing Agreement in the pages which follow. By installing the software you are indicating you agree to the terms of the license.

How to Use This Manual

The goal of this manual is to help you learn how to use chipcrusher. Although many of us dislike reading manuals, if you wish to get the most out of this new plug-in it is absolutely essential that you read this manual. Doing so will help you understand how to use this effect. The operation of many of the essential features is not obvious in casual use and we realize many users are not music technologists. We'll do our best to make this easy for you in this manual and to explain technical concepts.

You can refer to this manual whenever you wish. This manual is provided in digital form as an Adobe Acrobat document file (also known as a PDF) which can be viewed on a computer monitor or printed. If you do not have the Adobe Acrobat Reader, it is available for free from www.adobe.com.

Further Documentation and Resources

For the latest information including updated documentation, visit our support pages at: www.plogue.com. There you can find updated information provided after the manual was written, corrections or additions to this manual FAQ pages answering common questions suggestions from the users of Plogue software and news about upcoming Plogue releases. You can also visit the Plogue Forums for up-to-date information.

The address is: www.plogue.com/phpBB3/ .

Please send any reports of errors in this manual or suggestions for improvement to :
chipcrusher.support@plogue.com

End User License Agreement

Please read the terms of the following software licensing agreement before using this software. By installing and loading this product on your computer you acknowledge that you have read this license agreement, understand the agreement, and agree to its terms and conditions. If you do not agree to these terms and conditions, do not install or use the sounds contained herein. This is the complete agreement between you and Plogue Art et Technologie, Inc that supersedes any other representations or prior agreements, whether oral or in writing.

An important thing to understand is that YOU ARE OBTAINING A LICENSE FOR YOUR USE ONLY—THE NOISE RECORDINGS and IMPULSE RESPONSES do NOT belong to you The implications are described below. The sounds, samples and programming in the Plogue chipcrusher program remain the sole property of Plogue Art et Technologie, Inc. and are licensed (not sold) to you.

What You Can Do: You can use these effects in music productions, public performances, and other reasonable musical purposes within musical compositions. You can use these effects in your own musical compositions as much as you like without any need to pay Plogue Art et Technologie Inc or obtain further permission. If you do use these effects, we kindly ask that in any written materials or credits accompanying your music that utilizes material from Plogue chipcrusher (CD booklet, film credits, etc), that you include the following courtesy credits: "Some sounds were processed through Plogue chipcrusher", or a similar credit where practicable. If you can't disclose your use of chipcrusher for any reason to the general public, we would at least love to be aware of the usage of chipcrusher in a commercial recording for our own PRIVATE records.

What You Cannot Do: The enclosed noise recordings and impulse responses cannot be re-used in any other commercial sample library or any competitive product. You are absolutely forbidden to duplicate, copy, distribute, transfer, upload or download, trade, loan, reissue or resell this product or any of the contents in any way to anyone. You cannot redistribute them through an archive, nor a collection, nor through the Internet, nor binaries group, nor newsgroup, nor any type of removable media, nor through a network. The sounds and samples contained herein cannot be edited, modified, digitally altered, re-synthesized, or manipulated without direct written consent of Plogue Art et Technologie Inc.

Disclaimers and Conditions: *A right to use Plogue chipcrusher is granted to the original end-user only, and this license is not transferable unless there is written consent from Plogue Art et Technologie, Inc and payment of an additional fee. The sounds of Plogue chipcrusher will only work with the bundled Plogue ARIA Engine and will not work with any other engine. Licensor will not be responsible if the content does not fit the particular purpose of the Licensee. Please make sure before installing this item that it meets your needs. Information contained herein is subject to change without notice and does not represent a commitment on the part of Plogue Art et Technologie, Inc. The sounds are licensed "as is" without warranties of any kind. Neither Plogue Art et Technologie, Inc, nor any agent or distributor can be held responsible for any direct or indirect or consequential loss arising from the use of this product in whatever form. The Aria Engine is covered by the installer's End User License Agreement and is incorporated by reference. Plogue chipcrusher may not be returned for any reason. The terms of this license shall be construed in accordance with the substantive laws of Canada and the province of Quebec.. The user agrees to read the manual before seeking technical support and to make sure his or her system meets the recommended requirements.*

Specifications & Computer System Requirements

The following table lists the computer and hardware requirements for using chipcrusher. You can use chipcrusher on most any modern personal computer that meets the specifications listed below. The specifications provide the minimum standards. For optimal functioning, it is recommended you have a powerful enough computer with a fast CPU processor, Please also observe the systems requirements of your host application, tracker program and/or sequencing program if applicable. Please see the Plogue forum or website if you are looking for recommendations or for more information.

| Computer System Requirements | | |
|-------------------------------------|---|--|
| Computer | Operating System | Hardware |
| Windows PC | Microsoft Windows XP (SP2 or SP3 recommended) Microsoft Windows Vista 32 or 64 bits Microsoft Windows 7 32 or 64 bits | <ul style="list-style-type: none"> ◆ 2.4 GHz CPU Pentium 4 or better, 2.0 GHz Core Duo or better recommended ◆ 32 MB of free hard drive space ◆ Internet connection for download version, CD ROM drive required for boxed version installation ◆ Monitor with 1,024x768 resolution or better for usage of the default skin. ◆ A suitable VST host, or Pro Tools 8.0.5 or later ◆ Keyboard: A MIDI interface may be required if you are using a MIDI keyboard. Some keyboards use USB. ◆ High quality speakers and amplifier, or high quality headphones ◆ Internet connection for download, updates. |
| Mac | Mac OS X 10.6 minimum | <ul style="list-style-type: none"> ◆ Mac Intel; Core Duo or better recommended ◆ 32 MB of free hard drive space ◆ Internet connection for download version, CD ROM drive required for boxed version installation ◆ Monitor with 1,024x768 resolution or better for usage of the default skin. ◆ A suitable AU or VST host, or Pro Tools 8.0.5 or later ◆ High quality speakers and amplifier, or high quality headphones ◆ Internet connection for download, updates and online registration |

The stated requirements are an average. Certain music program may have additional resource requirements. Please also observe the system requirements of your host application, tracker program and/or sequencing program if applicable. The demands of various other processing software (including the sequencer, audio and effects processors, other plug-ins, and so on) can affect functionality.

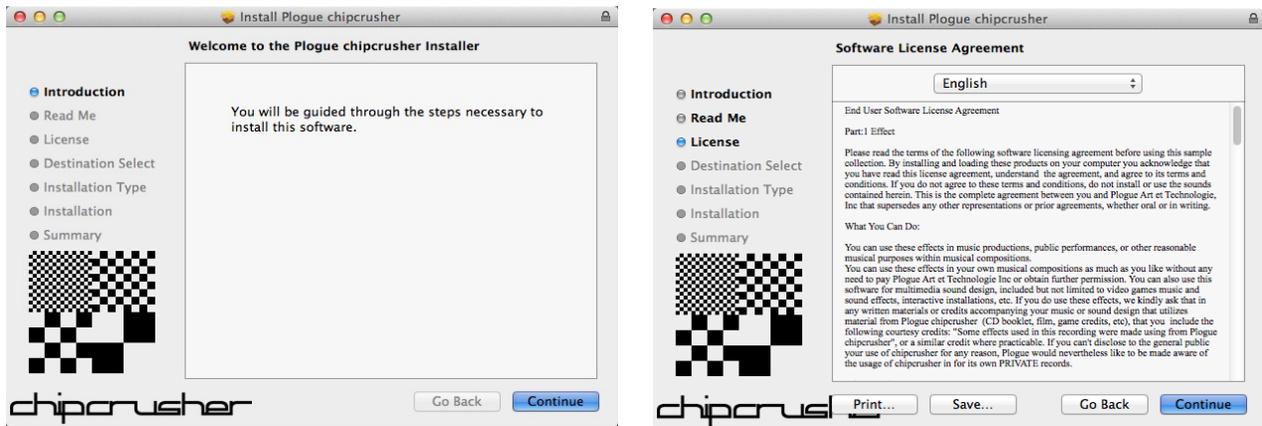
INSTALLATION

Installing Plogue chipcrusher

Installing Plogue chipcrusher is easy. Before you begin, make sure you meet the system requirements. The full installation requires approximately 32MB of free system hard disk space. Installation involves installing the chipcrusher plugin and also the advanced ARIA engine. A setup program will guide you through the process step-by-step.

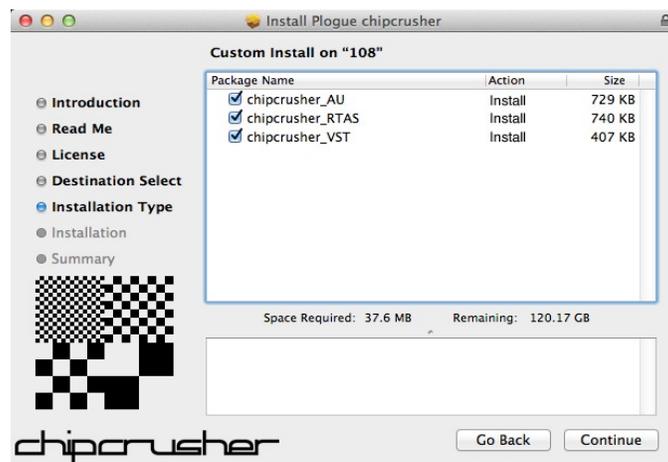
Mac Setup

- ◆ First, make sure your audio and MIDI hardware is set up and working with your computer.
- ◆ Close any programs you are running.
- ◆ Double click on MAC_chipcrusher_v1.0.pkg (Accept the defaults on the installer. You should see a Welcome Screen.
- ◆ You will then be asked to accept the End User License Agreement before proceeding with installation.

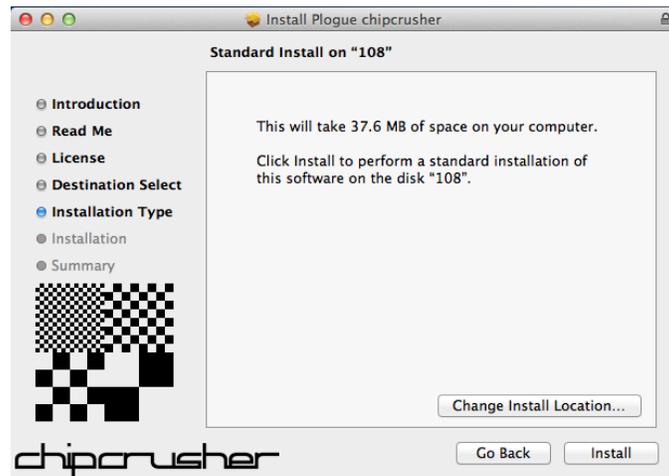


- ◆ The setup lets you select which plug-in formats you require. These plug-in formats allow Plogue chipcrusher to run as a virtual effect plug-in that seamlessly integrates into your favorite music software program or sequencer (assuming that it accepts such plug-ins).

For Mac, the Choices are Standalone, Audio Units (AU), RTAS and VST at least ONE needs to be chosen



- ◆ Setup is now ready to begin installing Plogue chipcrusher. Click on “Install” to begin installation and the program will begin installing. This process should not take more than a minute.



- ◆ After a few moments a dialog box will notify you that the setup wizard has finished installing Plogue chipcrusher on your computer.



Important:

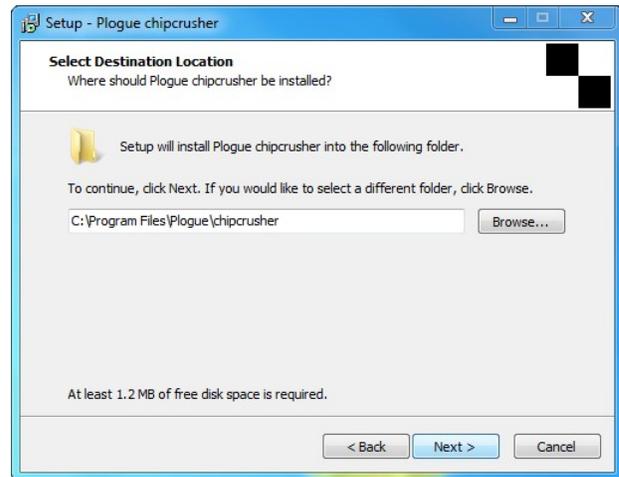
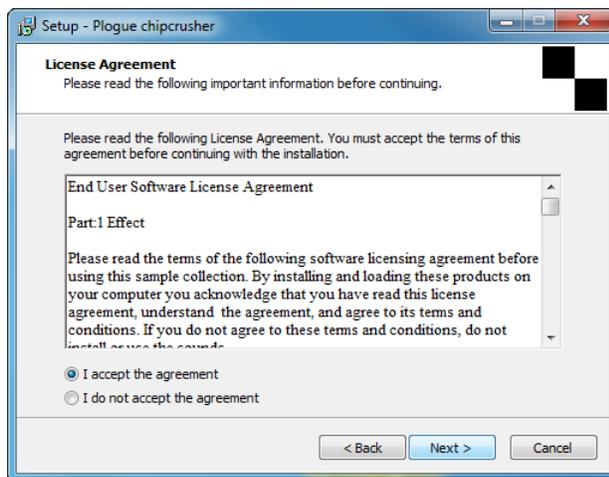
Please do not cancel setup after installation begins, otherwise a partial, broken installation may result.

Windows Setup

- ◆ Close any programs you are running.
- ◆ Double click on WIN_chipcrusher_v1.0.exe (Accept the defaults on the installer. You should see a Welcome Screen.

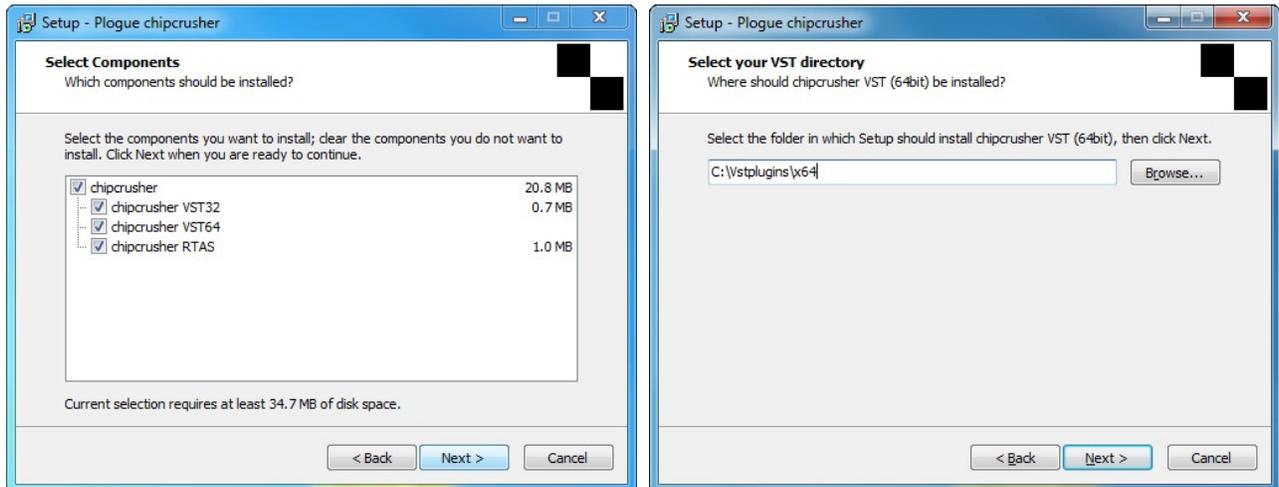


- ◆ You will then be asked to accept the End User License Agreement before proceeding with installation.
- ◆ And choose which location to install it on:

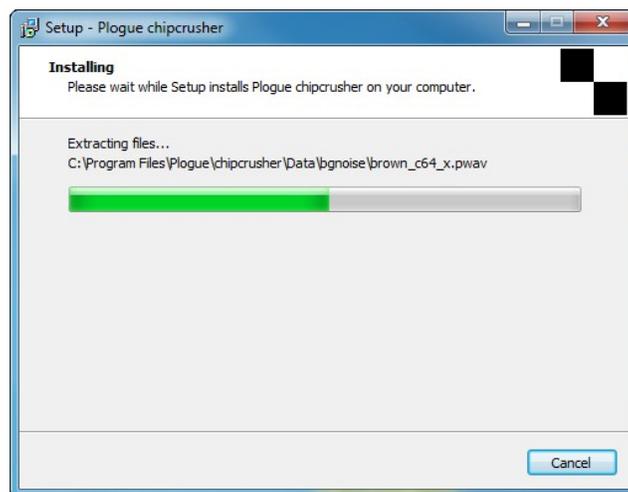


- ◆ The setup lets you select which plug-in formats you require. These plug-in formats allow Plogue chipcrusher to run as a virtual effect unit that seamlessly integrates into your favorite music software program or sequencer (assuming that it accepts such plug-ins).

For windows, the choices are RTAS and VST (32 and 64 bit). At least ONE needs to be chosen. You'll also be asked to select your VST plugins folder separately for 32 and 64 bit versions, if applicable.



- ◆ Setup is now ready to begin installing Plogue chipcrusher. Click on "Install" to begin installation and the program will begin installing. This process should not take more than a minute.



- ◆ After a few moments a dialog box will notify you that the setup wizard has finished installing Plogue chipcrusher on your computer.



Important:

Please do not cancel setup after installation begins, otherwise a partial, broken installation may result.

Authorizing chipcrusher

chipcrusher needs to be authorized in order to be fully functional, otherwise it will run in **DEMO** mode. When you order a license from Share*it, you receive a personal Activation Key card named **aria_key_1015.png** as an email.

The Activation Key card is an image resembling a typical credit card. This image contains your registration and details encoded within the Key card image. It will look like this:

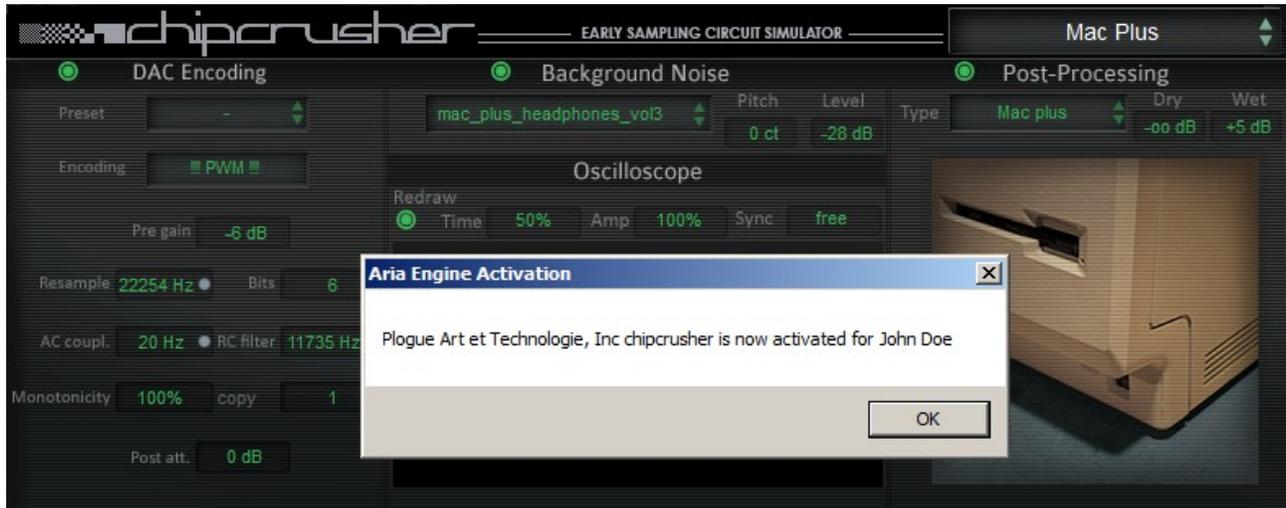


You should save the **aria_key_1015.png** image file to your hard drive (keep it in a safe place). For convenience, we recommend that you initially save the .png file to your desktop. You will also receive a copy of the license card in your email inbox.

- 1) Locate the "license card" image where you saved it on your hard drive. (**aria_key_1015.png**)
- 2) Launch your favorite host and make sure you see chipcrusher interface.
- 3) Simply click and hold on the file, drag the "license image" or file icon directly onto the application's UI itself, and release it.

If you don't get any message (or are not able to drop the key in that host), try to **import** the png file from the snapshot load menu at the top right of the interface (use *.* as file filter)

You should be presented with a message such as this one:



In the rare case when all this isn't working, use the **Import License** button located on the **Settings** page. See the settings page section of this manual (p.27).

If none of those methods work, please contact chipcrusher.support@plogue.com and attach your key.

Extremely Important!!

The *aria_key_1015.png* file contains your **sensitive personal information**, encrypted inside, including your **full name and address** taken from the online shop. Carefully protect this file. DO NOT GIVE THIS FILE TO ANYONE OR DISTRIBUTE IT IN ANY WAY OR YOUR PERSONAL INFORMATION WILL BE COMPROMISED. IF THE FILE BECOMES PUBLIC THE CARD NUMBER WILL BE BLACKLISTED AND THE CARD REVOKED. WE ARE NOT RESPONSIBLE IF YOU GIVE YOUR PERSONAL DETAILS TO A THIRD PARTY. IF THE CARD IS STOLEN, CONTACT US IMMEDIATELY. Without a valid card you will also not be able to obtain critical updates to the program.

Important Note:

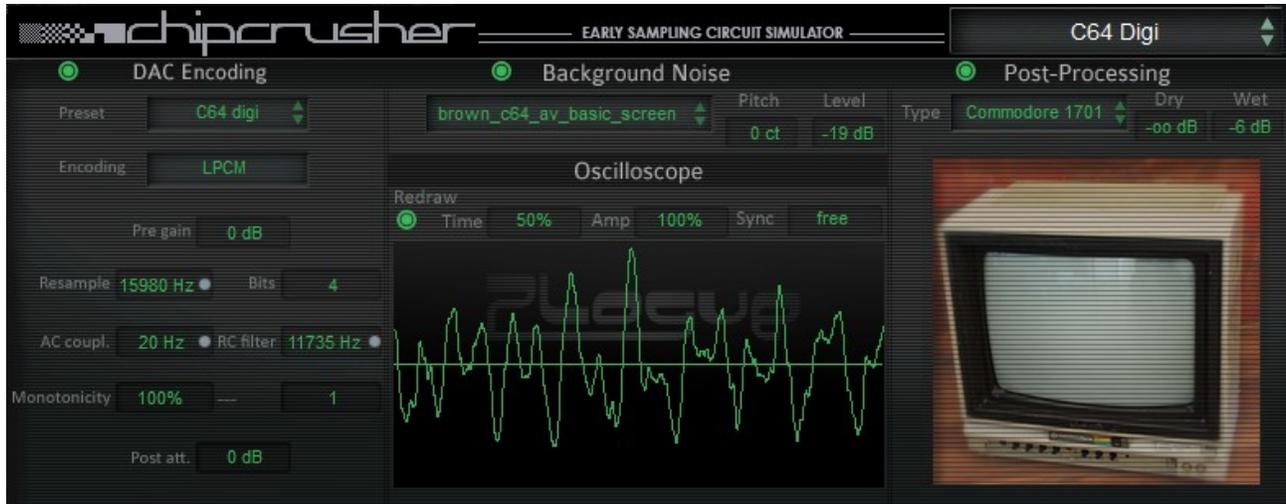
If you have special circumstances or require site licensing, please contact us.

Updating to the Latest Version

Be sure to check the Plogue Web sites for any possible updates that have occurred since the time your version of the software was released. Software is frequently updated and a more recent version may be available.

How to Use Plogue chipcrusher

Once installed and authorized, it's time to get started with Plogue chipcrusher. Just choose the chipcrusher plug-in in your favorite host, typically on a stereo track as an insert:



There are three main components (**DAC Encoding**, **Background Noise** and **Post-Processing**)

The audio inputs first go to the DAC Encoding. Then it get mixed with the Background Noise to finally get sent to the Post-Processing stage. Of course, each component can be bypassed on demand without muting the audio. Here's the details of those 3 components:

DAC Encoding:



chipcrusher's **DAC Encoding** does the following passes on the input sound:

- 1) Resamples(*1) the input audio input from the host's samplerate to a user specified 'virtual' samplerate.
- 2) Encodes/decodes that data (See **CODECs explained** section)
- 3) Change the weighting of bits in the DAC to emulate non-monotonic behavior.
- 4) Oversamples the decoded data to mimic the behavior of specific DACs. (Zero-Order Hold, PWM, etc)
- 5) Applies basic filtering: one pole low pass (RC Filter) and high pass (AC Coupling)
- 6) Finally it downsamples the result back into the output host samplerate.(see **Latency Section**)

(*1) The initial downsampling uses a **brick wall filter** currently. Future additions would allow to make this filter **less** perfect, this would allow to better emulate some legacy samplers like the SP12/SP1200

Monotonicity:

When monotonicity=100% all the bits of the DAC influence the output as perfectly as you'd expect. When you start to drift from that, you then get weird waveshaping artifacts due to bits slightly changing 'weight', or electronically speaking if you think of a 'Resistor ladder' made DAC, that the resistances are progressively going out of tolerance.

Values of 95-100% should be used to stay in the 'realistic range', though realism sometimes gets in the way of FSU, so you can do what you want.

CODECs explained:

LPCM (Linear pulse-code modulation)

Interesting trick: lower sample rate and bit depth just like any other bit crusher. Yay!

Bits : Resolution [1;16]
Parameter : No effect

FPCM (Floating-point pulse-code modulation)

A variant on LPCM that uses a mantissa and an exponent to represent a value. Its purpose (like many other old encoders) was an attempt to preserve sonic details even in the lower bits in a time when the technology to drive more than 10 bits in a DAC was too prohibitive to include in cheap consumer devices.

The trick was to use various forms of external analog gain stages to expand the overall voltage with the use of exponent bits.

Example the YM301X series DAC used for Yamaha's 4OP FM synthesizers encoded audio on 10.3 bits

Bits : mantissa bits [2;15]
Parameter : exponent bits [1;4] (else the results will mostly overflow the 16 bit range)

This allows for 50 distinct combinations of mantissa/exponent bits From (2.1, 2.3, 2.4 until 15.1)

NOTE: Sum of mantissa+exp needs to be no bigger than 16 (exp will be internally lowered as needed)

Adaptive differential pulse-code modulation Category

DPCM (Differential pulse-code modulation)

Encoding based on the difference between two successive samples, this difference can be stored on one or more bits. One bit encoding is a special case and is simply called 'Delta Modulation'

Bits : Resolution [1;16]
Parameter : delta encoding bits [1;16] number of bits to represent the difference between two samples

Interesting Trick: changing Resampling live can add a carrier frequency that you could tune.

Example: Delta modulation channel provided by the RP2A03/RP2A07 CPUs used in the Big N's 8bit console uses Bits=6 and Parameter=1

A-Law and μ -Law

Part of the G.711 standard for telephony. It compresses the input PCM adaptively into 8bits.

μ -Law was also used as encoding for Early US samplers and drum machines which used AM6070/AM6072 ICs like the LINN Drum machines, and the E-MU Emulator I/II/Drumulator.

Bits : no effect

Parameter: no effect

DIALOGIC:

an improved form of 4->12 bit compression

Example: MSM(OKI)6295 Used in LOTS of late 80's and early 90's arcade titles (All non Q-sound CPS1 titles)

These are fixed encodings, bits and extra has no effect.

Bits : no effect

Parameter: no effect

BRR (Bit Rate Reduction)

S_ny's specific ADPCM Encoding used in the S-DSP of the SNES. This implementation always uses the 'no IIR filter' block type.

Bits : Allows downresing the input of the encoder.

Parameter: no effect

YADPCM :

a Yamaha-specific ADPCM variant.

Bits : no effect

Parameter: no effect

Example N's 'Oui-mote' and various Yamaha FM chips for sample playback:

LPC10:

Use Linear prediction encoding to compress input audio.

Bits: (no effect)

Parameter: Mode

- 1) Voiced and Unvoiced
- 2) Voiced Only
- 3) Unvoiced Only

Example use in the Speak & Spell.

Encoders based on 1bit DACs at (usually) very high samplersates: !!!! 11kHz TONE WARNING !!!

PWM (Pulse-width modulation)

1bit PWM Audio encoding works by using the input signal amplitude value to change the instantaneous pulse width of a carrier frequency, itself running at the desired target samplerate frequency.

Since this carrier wave's amplitude reaches 0 dBFS, we have raised the lowest resample value you can use from 1kHz to 11kHz to spare your hearing somewhat, and we *only* added this possibility to emulate some legacy software on the Apple II (SoftDAC 3 and 4 bit versions), you will find that they are NASTY sounding.

You have been warned.

Bits : Resolution [1;6] bits (we could add more, but then the CPU use/latency would be too high)

Parameter: Copy [1:X] Duplicates each sample to raise the Carrier frequency, at the expense of losing one bit of resolution for each each copy. 1: no change, 2: two identical samples, 3: four identical copies, etc.

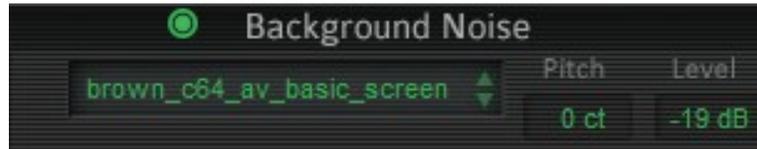
PDM (Pulse-density modulation)

A 1bit DAC which instead of using pulse width as a direct representation of the input audio, keeps an internal averaging/density logic and only outputs a 0 or a 1 at the target samplerate.

This is usually run at VERY high sample rates, (SACD/DSD uses 2.8224 MHz, but since here we only have straight PCM for input, whose rate depend on the host rate (typically 44/48/96kHz), it is quite useless to use PDM at higher frequency, since we don't have extra information to encode in time.

NOTE: monotonicity should be kept at 100% to be true to a 1bit DAC.

Background Noise



Noises were carefully recorded and looped from our collection of gear (consoles and arcade boards) which were also used for the research behind chipsounds. Just select the preset and adjust volume to taste. Note: default volumes should be used if you want to keep it subtle.

You can adjust the pitch of the noise slightly (± 400 cents, or 4 semitones), either to simulate PAL 50Hz noise or to tune it along with your music.

Some units generate different noises depending on external factors, for instance, it is well known that the Commodore 64 and Vectrex background noises were directly related to what was displayed at a particular time on the screen. In this case various noises are available.

A quick look at the available selection:

- Atari 2600
- Atari STf
- Atari 800XL
- Bally Astrocade
- C64's (brown breadbox)
- C64C with 8580,
- C64 (through the RF)
- C128
- GameBoy (DMG-1)
- GameBoyColor (CGB)
- Gemini (Atari 2600 clone)
- Genesis 1 (using headphones – two levels), using the audio out, and the RF modulator
- Intellivision I (RF)
- Intellivision III (RF)
- Leisure Vision
- NES and VS arcade PCB
- Odyssey2
- Casio PV1000 RF (good luck finding your own :)
- Sega Master System (RCA and RF)
- Super PacMan arcade PCB
- Vectrex (4 different display patterns)
- VIC-20

Post Processing



A very efficient *zero latency*(*3) convolution engine allows chipcrusher's sound (which can be very harsh at times) to go through a few of the most well loved vintage gaming devices, monitors. Each impulse gives a totally different tonal quality to the sound. You can dial in the unaffected sound as well for added flexibility.

(*3)Note all impulse responses were trimmed with the maximum amplitude at 32 samples from the start of the file at 96kHz. You can consider this little delay to be the 'response time' of the speakers

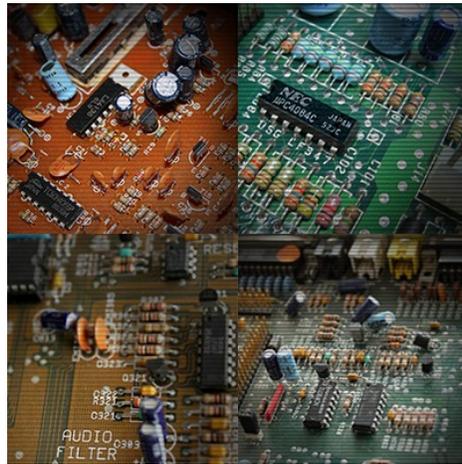
We have split the available impulses into the following categories:

Computing:



Speaker and casings impulses of various PC clones but also three Apple classics, the Apple IIe and the Apple IIGS

Filters:



Mostly filled with the 3 variants of AMIGA Original Chip Set machines (A500, A1000 and A2000), with the infamous LED filters on and off. We also included the extreme lowpass filter from the SK1.

Monitors:



From the classic Commodore monitors used with the C64, to the Mac Plus and even a 50s Television set

GAMING:



Most portable gaming devices, or speaker-bearing devices of the day including a Coleco Mini-Arcade, and a *fullsize* Galaxian Arcade Cabinet and Nintendo VS cabinet. A Oui-Mote, a Vectrex, 4 different gameboy's (DMG-1, CGB, GBA and GBA clam shell) and more!

Instruments:



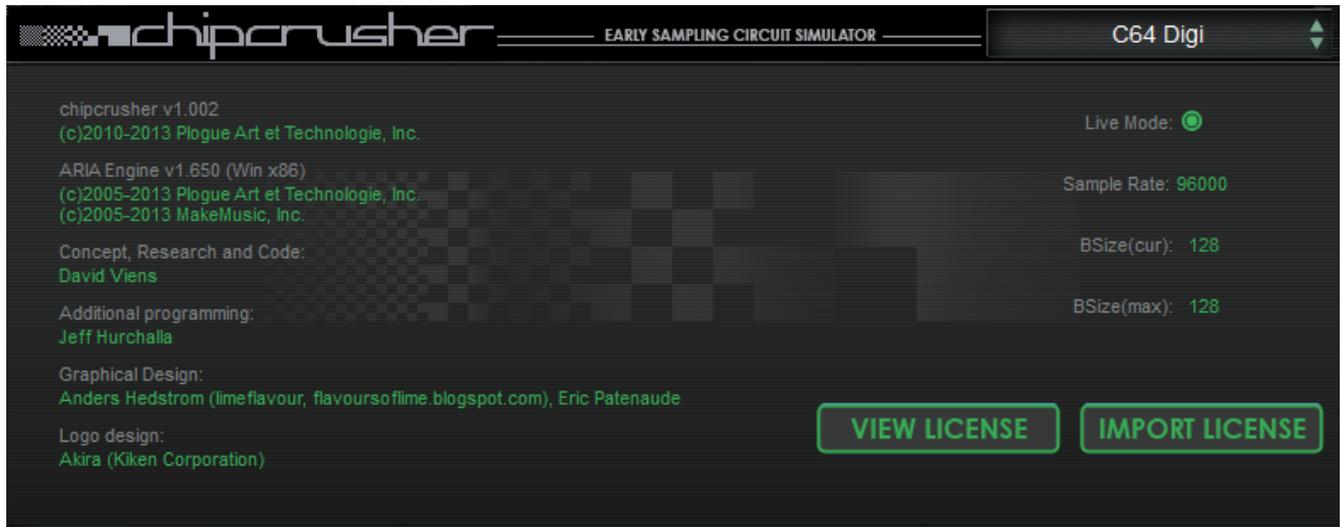
Various classic drum machines, keyboards which came with a speaker.

IMPORTANT NOTE ON PRESETS:

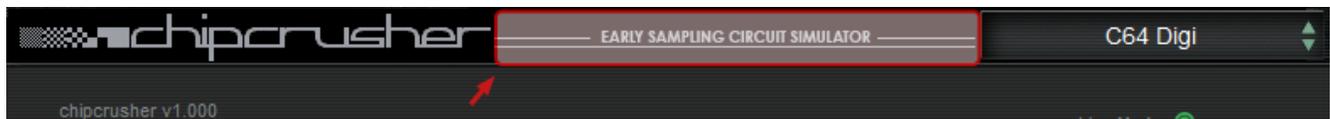
.ariax files : full content of the UI is saved in xml form, including values of parameters's at the time of saving. (VST hosts fxb/fxp files are similar in that they contain the same data)

.ariap files : presets to sub portions of the UI, - presets for the DAC Encoding, the Background Noise, and the Post-Processing.

The Settings Page



You can access the Settings section of chipcrusher by clicking on the middle area of the top section, between the logo and the Presets box:



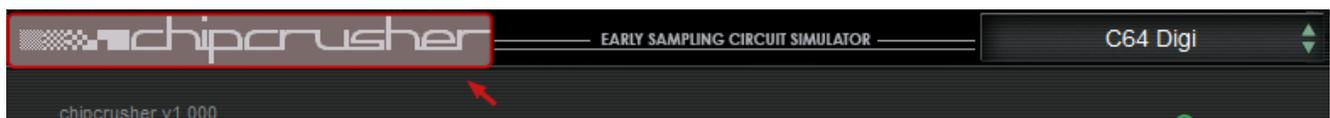
This will show you the credits and some vital info on the host's audio engine setting. This info can be useful if you ever need technical support.

The **Live Mode** button disables latency compensation data sent to the host program as well as some internal overhead. That way, chipcrusher can be used in a live setting as an effect without causing additional latency (More on that in the **Latency** section).

Clicking on the **License** button will display your license Keycard. Click again anywhere on the interface to close the image.

As an alternate way of authorizing chipcrusher, the **Import License** button lets you find your keycard in order to authorize chipcrusher if your host doesn't allow the drag and drop method.

To go back to the main controls, click on the logo in the top left.



Latency

Since the DAC Emulator juggles between 3 distinct samplerates, introducing latency is sadly inevitable.

Upon a session start chipcrusher will warn the host (VST/AU only) about a worst case latency for the current host samplerate setting, so that changing the internal 'virtual' samplerates does not desynchronise the outputs with regards to the other tracks.

However since the DAC Encoding section can be bypassed, we have added internal delay as *well* to compensate for the possibility of changing the section to bypass while the host plays.

Since PDC (plugin delay compensation) behavior varies greatly from one host to the other, your tracks might need to be rendered and offsetted for the best results.

Note: Turning Live Mode On in the settings tab will force reported host latency to be nil AND internal delay as well. This is useful not only for live situations, but also if you only use the Post Processing section.

Getting Help

The first place to look for a solution to any problem you may be experiencing is in this manual. Please read the manual before contacting support. Next, check the readme files (if any) which contain important information and all last-minute changes that haven't been available when creating this guide.

Plogue chipcrusher is evolving and growing. Please check the support area of our website at www.wlogue.com for the latest up-to-date information products, troubleshooting, FAQs, helpful hints and tutorials. Another resource is the support forums.

Whenever you encounter problems, you should also check if you have installed the latest updates. The version number of your software is displayed in the **About** dialog. (click on center-top of the interface) Updates are released regularly to fix known problems and to improve the software.

If you can't find a solution to your problem please email us at chipcrusher.support@plogue.com. The best way to get the help you need is by giving us plenty of detailed information about the problem you are having. We do ask you to read this guide thoroughly and exhaust the other avenues of support before contacting us.

The Plogue forum can be accessed at: <http://www.plogue.com/phpBB3/>

You don't have to register to browse posts, but before you can post, you will have to sign up.

Acknowledgments

Producing Plogue chipcrusher would not have been possible without the combined help, talent and support of many extraordinary people. I am grateful to those who have contributed and would like to thank them all.

Jeff Hurchalla: for your incredible zero latency and efficient convolver and deconvolver utilities which were of great help.

Previous research (no particular order):

Lord Nightmare, balrog, kevtris, Blargg, Antti Lankila, Brad Taylor, Sebastian Tomczak, and the MAME/MESS team.

Testing and suggestions (no particular order):

XC3N, 8GB, Chupathingy, 4Mat, sink, Peter Swimm, Bill Hicks, Bill Prokopow, Brady Leo, Chad Beckwith,

Last but far from least:

The Plogue team for believing in my second crazy project.. don't worry there's more to come :)

Martine for enduring the hoarding of all that "crap" in the basement, and my long technical gibberish rants. And my two wonderful sons, for sometimes being a d___ when my c___ doesn't work.

David Viens, June 2013

Appendix A: Quick Parameter Reference Guide for chipcrusher

DAC Encoding parameters:

| | | | |
|---|------------|----------|-----------|
| 00) DAC Resample | unit=Hz | min=1024 | max=44100 |
| 01) DAC A/C Coupling | unit=Hz | min=0 | max=120 |
| 02) DAC RC Filter | unit=Hz | min=1000 | max=20000 |
| 03) DAC Pre Gain | unit=float | min=0 | max=8 |
| 04) DAC Encoding | unit=list | | |
| list=LPCM,DPCM,FPCM,PDM,PWM,BRR,ALAW,MULAW,DIALOGIC,LPC-10,YADPCM, Future... | | | |
| 05) DAC Bits | unit=int | min=1 | max=16 |
| 06) DAC Parameter | unit=int | min=1 | max=16 |
| 07) DAC Monotonicity | unit=float | min=0 | max=1 |

Post-Processing parameters:

| | | |
|--------------|------------|-------|
| 20) Post Dry | unit=float | max=4 |
| 22) Post Wet | unit=float | max=4 |

StereoMixer parameters:

| | | |
|--------------------|------------|-------|
| 30) DAC Post Atten | unit=float | |
| 31) BG Noise Level | unit=float | max=2 |

Processing Modes parameters:

| | | |
|-------------------|-----------|-------------------------------|
| 41) DAC ProcMode | unit=list | list=Process Mute Passthrough |
| 42) BG Noise Mute | unit=bool | |
| 48) Post ProcMode | unit=list | list=Process Mute Passthrough |

Oscilloscope parameters:

| | | |
|-----------------------|------------|--|
| 51) Scope Mode | unit=float | |
| 54) Scope Time Window | unit=float | |
| 55) Scope Amp Window | unit=float | |
| 56) Scope Sync Draw | unit=bool | |

Background Noise parameters:

| | | | |
|-------------------|------------|----------|---------|
| 60) BG Noise Tune | unit=float | min=-400 | max=400 |
|-------------------|------------|----------|---------|

Appendix B: Host Specific Issues

Renoise versions 2.1 and prior:

You HAVE to untick "Auto Suspend" in the bottom right of the chipcrusher instrument box for the UI to be usable. This WON'T be changed, if you use Renoise get used to that. :)

Sonar versions prior to 8.5:

You may get audio dropouts when the chipcrusher UI is shown, please visit our support pages for a registry based fix.

»