

bladeenc

Benjamin Niemann

| |
|----------------------|
| COLLABORATORS |
|----------------------|

| | | | |
|---------------|----------------------------|------------------|------------------|
| | <i>TITLE :</i> bladeenc | | |
| <i>ACTION</i> | <i>NAME</i> | <i>DATE</i> | <i>SIGNATURE</i> |
| WRITTEN BY | Benjamin Niemann | January 19, 2025 | |

| |
|-------------------------|
| REVISION HISTORY |
|-------------------------|

| NUMBER | DATE | DESCRIPTION | NAME |
|--------|------|-------------|------|
| | | | |

Contents

| | | |
|----------|--|----------|
| 1 | bladeenc | 1 |
| 1.1 | Welcome | 1 |
| 1.2 | Disclaimer | 1 |
| 1.3 | Short description | 2 |
| 1.4 | Quick-Start Guide | 2 |
| 1.5 | The More Advanced Users Guide | 2 |
| 1.6 | L3Enc Emulation Mode | 7 |
| 1.7 | Updates, Questions and Information | 7 |
| 1.8 | Copyright and Distribution | 7 |
| 1.9 | Credits | 8 |

Chapter 1

bladeenc

1.1 Welcome

BladeEnc Manual

BladeEnc
Blade's MP3 Encoder
version 0.82
Release Date: July 22, 1999
written by Tord Jansson,
based on sourcecode from ISO
E-mail: tord.jansson@swipnet.se
Homepage: <http://bladeenc.cjb.net>

This manual is written for the Windows version of BladeEnc.
Various ports may differ slightly in some aspects (but normally
they don't).

Disclaimer
Short Description
Quick-Start Guide
The More Advanced Users Guide
L3Enc Emulation Mode
Updates, Questions and Information
Copyright and Distribution
Credits

1.2 Disclaimer

Disclaimer

BladeEnc is distributed 'as is' with no warranty of any kind. The Author
is not to be held responsible for any use or missuse of this product or
the result thereof.

1.3 Short description

Short Description

BladeEnc is a program to generate MP3 files from WAV or AIFF sound files. BladeEnc is available for most modern computer platforms, including Windows, Linux, BeOS, UnixWare, Solaris, FreeBSD, NetBSD, OpenBSD, OS/2 and many more.

BladeEnc is free software distributed under the LGPL license and the sourcecode is available from my homepage for those who might be interested.

Windows users will find that it looks like a DOS-program, but it is in fact a true 32-bit Windows application that supports long filenames etc. This textbased interface might be a bit harder for the beginner, but makes it easy to make automated scripts (like BAT-files) to perform your tasks and makes BladeEnc a doodle to port to all modern operating systems except Mac. A large selection of graphical frontends are available from my homepage for those who prefer to do their encoding just by moving and clicking their mouse.

On my homepage you can also find some more detailed information about BladeEnc, including a FAQ (answers to Frequently Asked Questions), speed comparisons with other encoders and information about the quality of MP3s generated by BladeEnc.

You can also go to the link section of my homepage, if you want some more information about MP3 in general or needs an MP3 Player.

1.4 Quick-Start Guide

Quick-Start Guide

If you just want to turn your WAV/AIFF-files into standard 128 kBit stereo MP3 files, just drag and drop your files onto BladeEnc.exe (that is the BladeEnc ICON and not the window that opens if you double click it).

Once started, BladeEnc will encode the files one by one until finished, which is a process that can take several hours.

BladeEnc can be stopped any time during the process by pressing CTRL-C (all systems) or ESC (Windows & OS/2 only).

The resulting MP3-files are put in the same directory as their corresponding WAV/AIFF-files.

1.5 The More Advanced Users Guide

The More Advanced Users Guide

BladeEnc is despite its primitive appearance both powerful and easy to use. It is a console application that takes its parameters from the commandline.

Any number of WAV/AIFF-files can be specified on the commandline and you can even use wildcards to specify more than one file at the same time. For example will the command "BladeEnc *.wav" compress all WAV-files in the current directory. Long filenames are supported when entering them on the commandline, but if they include space-characters you will have to enclose the names with quotation-marks (").

Switches can be entered on the commandline together with the filenames. In earlier versions of BladeEnc it didn't matter where you put the switches since they always affected all files anyway, but from BladeEnc 0.80 this has changed in order to give you more flexibility.

You can get a list of all valid arguments by running BladeEnc without any commandline arguments (for example by double-clicking on its icon).

If you feel that you need a temporary performance boost to your system you can pause the process by pressing ESC (Windows & OS/2) or CTRL-Z (Linux and other UNICES).

The Structure of The Commandline Arguments

The way the commandline is interpreted by BladeEnc has changed in version 0.80 in order to make it all more flexible and powerful and should work nicely with most older frontends. Also, the l3enc emulation mode has been removed, so it doesn't distinguish between l3enc- and BladeEnc commandlines anymore, but should be able to understand both anyway.

On the commandline you specify the files you want to encode, together with any switches to control the process and optionally the name of the output files. Most switches can be applied both globally, which makes them affect all the files, or locally, which makes them affect only one file.

Switches that only affects the general behaviour of BladeEnc and don't affect the output of the encoding process, can only be specified globally (like -quiet).

The commandline is read by BladeEnc following these simple rules:

- * Files to be encoded are specified on the commandline.

Example: bladeenc track1.wav track2.wav
will first encode the file track1.wav and then track2.wav.

- * If an input file is immediately followed by a filename ending with ".mp3" that will be the output name of the encoded file. If it isn't followed by a filename ending with ".mp3", the output file will get the same name as the input file, but substituting the extension with ".mp3".

Example: bladeenc track1.wav track2.wav output2.mp3

will first encode track1.wav into track1.mp3 and then encode track2.wav into output2.mp3.

- * Switches that are placed before the first input file are considered global and will affect all the specified files.

Example: bladeenc -crc -160 track1.wav track2.wav
will generate 160 kBit MP3 files with checksums of both input files.

- * Switches that are placed anywhere else will only affect the preceding input file.

Example: bladeenc track1.wav -crc -160 track2.wav -mono
will generate a 160 kBit MP3 file with embedded checksum of track1.wav and a 128 kBit mono MP3 file of track2.wav (128 kBit is the default bitrate).

- * Local switches have higher priority than global switches.

Example: bladeenc -160 track1.wav -256 track2.wav
will generate a 256 kBit mp3 file of track1.wav and a 160 kBit mp3 file of track2.wav.

- * Local switches must be placed after any specified output file. They may not be placed between the input and output file.

Example: bladeenc track1.wav output1.mp3 -256
will work just fine, but bladeenc track1.wav -256 output1.mp3
will just confuse bladeenc and generate an error message.

- * Stdin can be specified as input by simply substituting the name of the input file with stdin. This will make bladeenc read the input from a redirection or pipe instead.

Example: bladeenc stdin output.mp3
will encode the file coming through stdin and put the result in output.mp3.

- * Stdout can be specified as output by simply substituting the name of the output file with stdout. This will make bladeenc send the generated MP3 file through the standard output stream.

Example: bladeenc track1.wav stdout
will encode track1.wav and send the mp3 file through the standard output stream. ↩

- * And all this can be combined into one big operation.

Example:
bladeenc -160 track1.wav stdout -mono track2.wav track3.wav output.mp3 -crc ↩
stdin stdout -112
will do all the following:
- Encode track1.wav into a 160 kBit mono mp3 that is sent to stdout.
- Encode track2.wav into a 160 kBit mp3 named track2.mp3.
- Encode track3.wav into a 160 kBit mp3 with CRC named output.mp3.
- Encode what comes through the stdin into a 112 kBit mp3 that is appended to the data that is already sent to stdout.

Commandline Switches

All commandline switches are case-insensitive, so you can enter them in either upper- or lower-case.

-[bitrate]

-br [bitrate]

Defines the bitrate for the MP3-file. Higher bitrates gives better quality, but also bigger files. Most people prefer to generate 128 kBit MP3s. Please note that it's the total bitrate that is specified, so if you're generating a stereo MP3-file at 128 kBit you get 64 kBit for left channel and 64 kBit for the right channel. The default setting is 128 kBit for stereo files and 64 kBit for mono files.

Allowed bitrates are: 32, 40, 48, 56, 64, 80, 96, 112, 128, 160, 192, 224, 256 and 320.

-crc

Adds checksum data to each frame in the MP3 file. The checksum data is needed for error-correction when streaming the MP3 in realtime over internet (as done by internet radiostations). It lowers the quality of the sound slightly since the checksum data also needs to fit in the specified bitrate and is not needed for normal use.

-delete

-del

The WAV-file is automatically deleted after having been encoded. Be careful with this switch because the WAV-file is deleted even if the encoding process failed (for example due to insufficient drivespace).

-mono

-dm

Downmixes stereo input files to mono before they are compressed, thus generating mono MP3 files. This switch does nothing if the input file already is mono.

-leftmono

-lm

Takes only the left channel of the input file and encodes it into a mono MP3. If the input file is mono it will be compressed as it is.

-rightmono

-rm

Takes only the right channel of the input file and encodes it into a mono MP3. If the input file is mono it will be compressed as it is.

-swap

Swaps the left and right channels of stereo input files before encoding them.

-private

-p

Sets the private-flag in the MP3-file, specifying that this is a private MP3. As far as I know there is no program that treats private MP3's differently from normal ones so it is practically useless.

Setting this flag doesn't affect the encoding time, filesize or quality in any way, so set it if it makes you happy.

-copyright

-c

Sets the copyright-flag in the MP3-file, specifying that this is a copyrighted MP3. As far as I know there is no program that treats copyrighted MP3's differently from normal ones so it is practically useless.

Setting this flag doesn't affect the encoding time, filesize or quality in any way, so set it if it makes you happy.

-copy

This switch clears the MP3 file's original-flag that is set by default from version 0.50 of BladeEnc.

Clearing the original-flag doesn't affect the encoding time, filesize or quality in any way, so do it if it makes you happy.

-quiet

Prevents BladeEnc from doing any screen output, except for error messages. This is mostly usefull if you want to run bladeenc as a background process on UNIX systems. The screen output takes nearly no time at all, so you won't even save 1% by disabling screen output. This switch is global only.

-quit

-q

Makes BladeEnc quit automatically when all files have been encoded. Normally BladeEnc waits for someone to press RETURN before quitting. This switch is global only.

-outdir=[path]

Specifies an output path for the encoded files. Normally the MP3 files ends up in the same directory as their corresponding WAV-files resides in, but using this switch you can get them to end up wherever you like. This switch is global only.

-prio=[setting]

Changes the task priority of BladeEnc. Valid settings are HIGHEST, HIGHER, NORMAL, LOWER, LOWEST and IDLE. BladeEnc is by default set to LOWEST priority which basically means that BladeEnc functions fine in the background

without disturbing or slowing down any other program. This switch is global only.

This setting is not available in all ports.

1.6 L3Enc Emulation Mode

L3Enc Emulation Mode

The l3enc emulation mode of older versions of BladeEnc has been removed in BladeEnc 0.80 since the new way of reading the commandline in this version makes it understand l3enc commandlines anyway.

Just remember that the default bitrate of BladeEnc is 128 kBit, while l3enc uses 112 kBit.

However, two small tricks were necessary to make BladeEnc compatible with l3enc:

- * If the specified bitrate is over 1000 it is divided by 1000 in order to convert it from bits/sec to kBit/sec.
- * If l3enc's -hq switch is placed on the commandline or a bitrate over 1000 is specified locally using the -br switch, BladeEnc automatically activates the -quit switch.

1.7 Updates, Questions and Information

Updates, Questions and Information

The latest version of BladeEnc is available at my homepage together with an FAQ and some other information. See the top of this document for the URL and my E-mail address.

1.8 Copyright and Distribution

Copyright and Distribution

BladeEnc is copyright Tord Jansson.

BladeEnc is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

You should have received a copy of the GNU Lesser General Public License with BladeEnc. If not, please see BladeEnc's homepage or www.fsf.org for

more details.

1.9 Credits

Credits

BladeEnc includes code from distribution 10 of ISO's reference code.
The contributors to that reference code have been:

Bill Aspromonte, Shaun Astarabadi, R. Bittner, Karlheinz
Brandenburg, W. Joseph Carter, Jack Chang, Mike Coleman,
Johnathan Devine, Ernst Eberlein, Dan Ellis, Peter Farrett,
Jean-Georges Fritsch, Vlad Fruchter, Hendrik Fuchs, Bernhard
Grill, Amit Gulati, Munsu Haque, Chuck Hsiao, Toshiyuki
Ishino, Masahiro Iwadare, Earl Jennings, James Johnston,
Leon v.d. Kerkhof, Don Lee, Mike Li, Yu-Tang Lin,
Soren Neilsen, Simao F. Campos Neto, Mark Paley, Davis
Pan, Tan Ah Peng, Kevin Peterson, Juan Pineda,
Ernst F. Schroeder, Peter Siebert, Jens Spille, Sam Stewart,
Al Tabayoyon, Kathy Wang, Franz-Otto Witte, Douglas Wong.

All modifications of the reference code in order to create all versions
of BladeEnc up to 0.80 have been made by Tord Jansson.

Thanks to the following people for contributing code and bugfixes to
BladeEnc:

Mikael Kjellström, Kevin Doherty, Milan Hodoscek, Wesley Hosking,
Jon Watte, Dmitri Kalintsev, William Thompson, Olaf Pueschel, Marc Schefer,
Lennart Börjeson, John DuBois.

Thanks to the following people for providing binaries of BladeEnc for
various formats:

Trevor Phillips, Zac Livingston, Jon Coyle, Alexey Marinichev, Mikael
Kjellström, Steve Burns, Markus Ridinger, Giao Nguyen, Joel Fredrikson,
Marca Registrada, Rob Braun.