

BladeEnc Manual

BladeEnc

Blade's MP3 Encoder

version 0.94.0

Release Date: March 9, 2001

written by Tord Jansson,
with the help of numerous contributors,
based on source code from ISO

E-mail: tord@bladeenc.mp3.no

Homepage: <http://bladeenc.mp3.no>

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

Table of Contents

[Disclaimer](#)

[Short Description](#)

[Quick-Start Guide](#)

[The More Advanced Users Guide](#)

[The Configuration File](#)

[The Structure of The Commandline Arguments](#)

[Commandline Switches](#)

[Gapless Encoding](#)

[L3Enc Emulation Mode](#)

[Updates, Questions and Information](#)

[Copyright and Distribution](#)

[Credits](#)

Disclaimer

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

Short Description

BladeEnc is a program to generate MP3 files from WAV, AIFF or RAW sound files. BladeEnc is available for most modern computer platforms, including Windows, Linux, BeOS, MacOS, 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 [the 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 text based 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 MacOS. 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 BladeEnc's 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 [the homepage](#) if you want some more information about MP3 in general or needs an MP3 Player.

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.

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. The commandline is where you enter text commands to your computer, the DOS-Prompt and "Start->Run.." -menu on a Windows system are examples of commandlines.

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).

To run BladeEnc from the commandline you should place BladeEnc.exe where your system automatically finds it, for example your Windows directory.

The Configuration File

The default settings of BladeEnc can be changed in the configuration file. On Windows and BeOS systems this configuration file is called bladeenc.cfg and **should always be placed in the same directory as BladeEnc.exe**. On a Unix system it is called .bladeencrc and should be placed in your home directory.

The config file contains the same kind of switches as you enter on the commandline, separated by one or more spaces, tabs or newlines.

Anything placed after a '#' until the end of the line is believed to be a comment and is ignored.

The switches in the configuration file are simply appended to the beginning of the commandline before it is parsed.

Anything typed on the commandline has higher priority than what is in the configuration file, so the default settings can easily be overridden from the commandline.

If the -nocfg switch is entered on the commandline the configuration file will be ignored. The -nocfg switch is not allowed in the configuration file.

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:

1. 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.*

2. 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.*

3. 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.*

4. 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).*

5. 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.*

6. 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.*

7. 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.*

8. 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.*

9. 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]	<p>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.</p> <p>Allowed bitrates are: 32, 40, 48, 56, 64, 80, 96, 112, 128, 160, 192, 224, 256 and 320.</p>
-concat	<p>Concatenates the output generated from this file to the last mp3 generated, thus making one mp3 file from two or more sample files. This option also automatically enables -nogap.</p>

-crc	Adds checksum data to each frame in the MP3 file. The checksum data is useful for error-correction when streaming the MP3 in realtime over internet (as done by internet radio stations). 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 drive space).
-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	<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.</p> <p>Setting this flag doesn't affect the encoding time, file size or quality in any way, so set it if it makes you happy.</p>
-copyright, -c	<p>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.</p> <p>Setting this flag doesn't affect the encoding time, filesize or quality in any way, so set it if it makes you happy.</p>
-copy	<p>This switch clears the MP3 file's original-flag that is set by default from version 0.50 of BladeEnc.</p> <p>Clearing the original-flag doesn't affect the encoding time, filesize or quality in any way, so do it if it makes you happy.</p>
-quiet	Prevents BladeEnc from doing any screen output, except for error messages. This is mostly useful 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. Linux/Unix users can also specify prio in the normal unix range of -20 to +20. This switch is global only. <i>This setting is not available in all ports.</i>
-rawfreq=[frequency]	Specifies the frequency for raw samples in hertz. Default is 44100.
-rawbits=[samplebits]	Specifies the number of bits for each sample in a raw sample file. Allowed values are 8 and 16. Default is 16.
-rawmono	Specifies that raw samples are mono, not stereo.
-rawstereo	Specifies that raw samples are stereo, not mono. This is default, so you won't have to use this switch unless you want to override a global -rawmono switch.
-rawsigned	Specifies that raw samples are signed, i.e. contains values in the range -32768 to +32767 (16-bit samples) or -128 to +127 (8-bit samples). This is default, so you won't have to use this switch unless you want to override a global -rawunsigned switch.
-rawunsigned	Specifies that raw samples are unsigned, i.e. contains values in the range 0 to 65535 (16-bit samples) or 0 to 255 (8-bit samples).
-rawbyteorder=[LITTLE/BIG]	Specifies the byteorder of raw samples. LITTLE gives LITTLE_ENDIAN (Intel x86 style) and BIG gives BIG_ENDIAN (Motorola style). Default is whatever is the native byteorder for your system, so hopefully you won't have to use this switch.
-rawchannels=[1/2]	Specifies the number of channels in a raw sample. -rawchannels=1 gives the same result as -rawmono and -rawchannels=2 give the same result as -rawstereo. Default is 2.
-nocfg	Ignore whatever is specified in the config file. This switch is global only.
-refresh=[1+]	<i>This switch is not supported anymore. It became superfluous due to internal changes. BladeEnc still understands (but ignores) it for compatibility reasons.</i>
-progress=[0-8]	Specifies which of the 7 available progress indicators to use or turns the progress indicator off if set to 0. The default progress indicator is number 1. Changing the progress indicator doesn't affect the encoding in any way it just lets you configure the look of BladeEnc a little bit according to your own taste: <ol style="list-style-type: none"> 1. Percentage and ETA shown for file and batch (default). 2. Long bar for file progress. 3. Long bar for batch progress. 4. Short bars for file and batch progress. 5. Samples done/total for file and batch. 6. ETA shown for file and batch. 7. Seconds of MP3 generated from file (default for RAW samples). 8. Alternative long bar for file progress. Doesn't rewind the cursor so it produces less output and works well when redirecting program output to a file.

	If a RAW sample is included in the batch you automatically gets progress indicator 7 for technical reasons. This switch is global only.
-nogap	Specifies that there should be no audible gap between this and the next sample if they are played after each other. Useful when encoding records where the tracks blend into each other, but can cause unwanted side effects otherwise. Please see the section "Gapless encoding" below for more information.
-sort	Sorts the specified input files according to filename before encoding them. Useful when using the -nogap switch combined with wildcards since it's critical that the tracks are encoded in the right order when doing gapless encoding. This switch is global only.

Gapless Encoding

From version 0.92.5 BladeEnc supports what we refer to as gapless encoding.

You might have noticed that you normally get a few milliseconds of silence in the beginning and end of each encoded track. This is normally not a problem, but when you have encode a record where the songs blends into each other (like Enigma and most live albums) and listen to the mp3s you can hear a short pause or click between the tracks which can be quite anoying.

There are some technical reasons for this which has made it quite hard to solve, but by using a few tricks and allowing a few milliseconds of music to be moved from the end of one track and onto the beginning of the next, we can make sure that we don't get any gaps.

The downside of this is that if you encode two totally unrelated tracks where the first one ends very sharply and the next one begins with silence, we can unwantedly get a short click in the beginning of the second track since it starts with some milliseconds of music that isn't intended to be there.

The sollution was therefore to by default disable gapless encoding and let the user enable it where wanted by using the -nogap switch.

The -nogap switch can either be placed as a global switch, specifying that all tracks should blend into each other or as a local switch if just two of the tracks should blend into each other.

Example:

1. bladeenc track01.wav track02.wav track03.wav -nogap track04.wav track05.wav

Only the transition between track03 and track04 is made gapless, only track04 'steals' a few milliseconds from track03.

2. bladeenc -nogap track01.wav track02.wav track03.wav track04.wav track05.wav

All the tracks are made to blend into each other, taking a few milliseconds of music from the previous one.

Please note the following!:

1. The tracks must be specified in the correct play order when using -nogap, otherwise you will get a few milliseconds of the wrong track, probably causing a loud click between the tracks.

2. All the tracks that should blend into each other needs to be encoded in the same session. You can't encode them one by one with the -nogap switch.
3. Some records have pre- and post-gaps between the songs. This normally causes a few milliseconds of music being lost between the tracks already under the CD extraction process and can't be undone by BladeEnc. The .mp3 files won't blend better into each other than the .wav/.aiff files.
4. **Many mp3 players do NOT support gapless replay!** Even if there is no silence in the beginning or end of the tracks they still pause briefly between the tracks. The only player that I know of which doesn't pause at all is mpg123 (Unix and Windows commandline player) but there are probably more and hopefully will many players be fixed once more gapless mp3 files start to circulate. There is also supposed to be a gapless output plugin for WinAmp.

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:

1. If the specified bitrate is over 1000 it is divided by 1000 in order to convert it from bits/sec to kBit/sec.
2. 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.

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. Please only e-mail me if you can't find the info you want on my homepage.

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.

Credits

This credits list might be out of date. Please see the file named AUTHORS in the source code distribution of BladeEnc for a more up to date credits list.

BladeEnc was started by Tord Jansson as a project to make a decent, free encoder for high-bitrate MP3 encoding, heavily based on the ISO source code for a reference encoder. Since then it has developed into a free software project (Open Source) with many contributors:

Project initiator and maintainer

Tord Jansson <tord.jansson@swipnet.se>

Major contributors

Andre' Piotrowski	Numerous improvements to the MP3 encoding routines.
Jeff Squyres	Maintains MPI port, various contributions (MPI port is now discontinued).
Petteri Kamppuri	Maintains MacOS port.

Contributors

Mikael Kjellström
Kevin Doherty
Milan Hodoscek
Wesley Hosking
Jon Watte
Dmitri Kalintsev
William Thompson
Olaf Pueschel
Marc Schefer
Lennart Börjeson
John DuBois
Jeremy Prior
Frans E. van Dorsselaer
Roger Fujii
Robert Schlabbach
Marco Amrein
Vitor Sessak
Serg 'Ice' Tsyganenko
Rich West
Ben Slusky
Simon Burge
Brad
Matt Craven
Paul Cantrell
Petter Reinholdtsen
Paul Martin
Troy Engel
Chris DeLise
Sadrudin Rejeb
Mauro Manfrin
Rob Ekl
Jeffrey Hawk
Stephen from Minnesota

Andrea Vallinotto
Chris Rempel
Trevor Taylor
Jean Robertson
Michael Langner

People who helped port BladeEnc to different architectures before it went Open Source

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

The ISO reference sourcecode credits the following people as its developers and contributors

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, Munsif 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.
