

NAME

opusdec – decode audio from Opus format to WAV (or simple audio output)

SYNOPSIS

opusdec [**-hV**] [**--quiet**] [**--rate Hz**] [**--gain dB**] [**--no-dither**] [**--float**] [**--force-wav**] [**--packet-loss pct**] [**--save-range file**] **input.opus** [**output.wav**]

DESCRIPTION

opusdec decodes Opus files into PCM Wave (uncompressed) files.

If the input file is specified as **-**, then **opusdec** will read from stdin. Likewise, an output filename of **-** will cause output to be to stdout.

If no output is specified **opusdec** will attempt to play the audio in realtime if it supports audio playback on your system.

OPTIONS

-h, --help
Print help message

-V, --version
Display version information

--quiet Suppresses program output

--rate
Force decoding at sampling rate *n* Hz

--gain
Adjust the output volume *n* dB, negative values make the signal quieter.

--no-dither
Do not dither 16-bit output

--float 32-bit floating-point files instead of 16-bit files

--force-wav
Force including a wav header on output (e.g. for non-wav extensions and stdout)

--packet-loss
Simulate *n* % random Opus packet loss

--save-range
Saves check values for every frame to a file

EXAMPLES

Decode a file **input.opus** to **output.wav**
opusdec input.opus output.wav

Play a file **input.opus** and force output at 48000 regardless of the original sampling rate
(48kHz output may be faster, due to avoiding resampling and some sound hardware produces higher quality output when run at 48kHz)
opusdec --rate 48000 input.opus

Re-encode a high bitrate Opus file to a lower rate
opusdec --force-wav input.opus - | opusenc --bitrate 64 - output.opus

Play an http stream **http://iccast.somewhere.org:8000/stream.opus** with the help of curl on a system with

```
pulseaudio  
(press ctrl-c to quit)  
curl http://icecast.somewhere.org:8000/stream.opus | padsp opusdec -
```

AUTHORS

Jean-Marc Valin <jmvalin@jmvalin.ca>
Gregory Maxwell <greg@xiph.org>

BUGS

Opusdec does not currently reject all invalid files which it should reject. It also doesn't provide very helpful output for the corrupted files it does reject. Use **opusinfo**(1) for somewhat better diagnostics.

SEE ALSO

opusenc(1), **opusinfo**(1)