BASS_Encode_OPUS_GetVersion

Retrieves the version of BASSenc_OPUS that is loaded.

DWORD BASS_Encode_OPUS_GetVersion();
Return value
The BASSenc_OPUS version. For example, 0x02040103 (hex), would be version 2.4.1.3
BASS_Encode_OPUS_Start

Sets up an Opus encoder on a channel.

```c
HENCODE BASS_Encode_OPUS_Start(
    DWORD handle,
    char *options,
    DWORD flags,
    ENCODEPROC *proc,
    void *user
);
```
Parameters

handle  The channel handle... a HSTREAM, HMUSIC, or HRECORD.

options Encoder options... NULL = use defaults. The following OPUSENC style options are supported: --bitrate, --vbr, --cvbr, --hard-cbr, --comp / --complexity, --framesize, --expect-loss, --max-delay, --serial, --comment, --artist, --title, --album, --date, --genre, --padding. Anything else that is included will be ignored. See the OPUSENC documentation for details on the aforementioned options and defaults.

flags A combination of these flags.

BASS_ENCODE_QUEUE  Queue data to feed the encoder asynchronously. This prevents the data source (DSP system or BASS_Encode_Write call) getting blocked by the encoder, but if data is queued more quickly than the encoder can process it, that could result in lost data.

BASS_ENCODE_LIMIT  Limit the encoding rate to real-time speed, by introducing a delay when the rate is too high. With BASS 2.4.6 or above, this flag is ignored when the encoder is fed in a playback buffer update cycle (including BASS_Update and BASS_ChannelUpdate calls), to avoid possibly causing playback buffer underruns. Except for in those instances, this flag is applied automatically when the encoder is feeding a Shoutcast
or Icecast server.

**BASS_ENCODE_CAST_NOLIMIT**

Don't limit the encoding rate to real-time speed when feeding a Shoutcast or Icecast server. This flag overrides the BASS_ENCODE_LIMIT flag.

**BASS_ENCODE_PAUSE**

Start the encoder in a paused state.

**BASS_ENCODE_AUTOFREE**

Automatically free the encoder when the source channel is freed. If queuing is enabled, any remaining queued data will be sent to the encoder before it is freed.

**BASS_UNICODE**

Options is in UTF-16 form. Otherwise it should be UTF-8.

*proc*

Optional callback function to receive the encoded data... NULL = no callback.

*user*

User instance data to pass to the callback function.
Return value
The encoder handle is returned if the encoder is successfully started, else 0 is returned. Use `BASS_ErrorGetCode` to get the error code.
**Error codes**

BASS_ERROR_HANDLE   \textit{handle} is not valid.

BASS_ERROR_FORMAT   The channel's sample format is not supported by the encoder.

BASS_ERROR_UNKNOWN   Some other mystery problem!
Remarks

*BASS_Encode_StartUser* is used internally to apply the encoder to the source channel, so the remarks in its documentation also apply to this function.

Tags/comments should be in UTF-8 form. This function will take care of that when the BASS_UNICODE flag is used, but otherwise you will need to make sure that any tags included in *options* are UTF-8 (not ISO-8859-1/etc).
Platform-specific
On Windows and Linux, an SSE supporting CPU is required for sample rates other than 48000/24000/16000/12000/8000 Hz.
See also
BASS_Encode_OPUS_StartFile
BASS_Encode_CastInit, BASS_Encode_IsActive, BASS_Encode_ServerInit,
BASS_Encode_SetNotify, BASS_Encode_SetPaused, BASS_Encode_Stop,
BASS_Encode_Write, ENCODEPROC callback,
BASS_CONFIG_ENCODE_PRIORITY
Sets up an Opus encoder on a channel, writing the output to a file.

```
HENCODE BASS_Encode_OPUS_StartFile(
    DWORD handle,
    char *options,
    DWORD flags,
    char *filename
);
```
Parameters

handle  The channel handle... a HSTREAM, HMUSIC, or HRECORD.

options  Encoder options... NULL = use defaults. The following OPUSENC style options are supported: --bitrate, --vbr, --cvbr, --hard-cbr, --comp / --complexity, --framesize, --expect-loss, --max-delay, --serial, --comment, --artist, --title, --album, --date, --genre, --padding. Anything else that is included will be ignored. See the OPUSENC documentation for details on the aforementioned options and defaults.

flags  A combination of these flags.

BASS_ENCODE_QUEUE  Queue data to feed the encoder asynchronously. This prevents the data source (DSP system or BASS_Encode_Write call) getting blocked by the encoder, but if data is queued more quickly than the encoder can process it, that could result in lost data.

BASS_ENCODE_LIMIT  Limit the encoding rate to real-time speed, by introducing a delay when the rate is too high. With BASS 2.4.6 or above, this flag is ignored when the encoder is fed in a playback buffer update cycle (including BASS_Update and BASS_ChannelUpdate calls), to avoid possibly causing playback buffer underruns. Except for in those instances, this flag is applied automatically when
the encoder is feeding a Shoutcast or Icecast server.

BASS_ENCODE_CAST_NOLIMIT Don't limit the encoding rate to real-time speed when feeding a Shoutcast or Icecast server. This flag overrides the BASS_ENCODE_LIMIT flag.

BASS.Encode_PAUSE Start the encoder in a paused state.

BASS.ENCODE_AUTOFREE Automatically free the encoder when the source channel is freed. If queuing is enabled, any remaining queued data will be sent to the encoder before it is freed.

BASS_UNICODE options and filename are in UTF-16 form. Otherwise options should be UTF-8 on all platforms, and filename should be ANSI on Windows and UTF-8 on other platforms.

filename Output filename... NULL = no output file.
**Return value**
The encoder handle is returned if the encoder is successfully started, else 0 is returned. Use `BASS_ErrorGetCode` to get the error code.
Error codes
BASS_ERROR_HANDLE  handle is not valid.
BASS_ERROR_FORMAT   The channel's sample format is not supported by the encoder.
BASS_ERROR_CREATE   The file could not be created.
BASS_ERROR_UNKNOWN  Some other mystery problem!
Remarks

BASS_Encode_StartUser is used internally to apply the encoder to the source channel, so the remarks in its documentation also apply to this function.

Tags/comments should be in UTF-8 form. This function will take care of that when the BASS_UNICODE flag is used, but otherwise you will need to make sure that any tags included in options are UTF-8 (not ISO-8859-1/etc).
**Platform-specific**
On Windows and Linux, an SSE supporting CPU is required for sample rates other than 48000/24000/16000/12000/8000 Hz.
See also

BASS_Encode_OPUS_Start
BASS_Encode_IsActive, BASS_Encode_SetNotify, BASS_Encode_SetPaused,
BASS_Encode_Stop, BASS_Encode_Write,
BASS_CONFIG_ENCODE_PRIORITY