

NAME

midi2abc – program to convert MIDI format files to abc notation

SYNOPSIS

```

midi2abc -f infile [-xa] [-ga] [-a acbeats] [-m time signature] [-ppu parts per unit] [-aul denominator of
unit length] [-gu] [-b bars] [-Q tempo] [-u pulses] [-k key] [-c channel] [-obpl] [-bpl bars] [-bps bars]
[-o filename] [-s] [-sr units] [-sum] [-nb] [-nt] [-splitvoices] [-midigram] [-mftext] [-mftextpulses]
[-nogr] [-title string] [-origin string][[-midinotes]

```

DESCRIPTION

midi2abc takes a MIDI format file and converts it to something as close as possible to abc text format. The user then has to add text fields not present in the MIDI header and possibly tidy up the abc note output.

The output of *midi2abc* is printed to the screen. To save it to a file, use the redirection operator, (e.g. *midi2abc -f file.mid > file.abc*) or specify the output file using the *-o* option.

Use only one or none of the options *-u* *-gu*, *-b* and *-Q*. *Midi2abc* normally converts the MIDI time units into quantum units normally corresponding to the abc 1/16th note or 1/32nd note. If none of these is present, the program will use the PPQN information in the MIDI header to compute the suitable conversion factor. For most MIDI files on the web, it is recommended to rely on the MIDI header information and not use any of the options other than the formatting options.

The program will extract the time signature information from the MIDI file if it is present. Otherwise it will assume 4/4 or you could specify it with *-m*. option.

If the tune has an anacrusis, you can use either the *-ga* or *-xa* option to estimate the its length. Alternatively, you can specify its value using the *-a* option. The anacrusis is specified in half unit lengths, where the unit length is defined by the L: field. For example if L: 1/8, then a quarter note would be indicated by the value 4, (4 1/16 units).

OPTIONS**-a *acbeats***

where *acbeats* specifies the anacrusis in half unit lengths.

-xa extract the anacrusis from file by finding the first strong note

-ga guess the anacrusis by minimizing the number of ties across bars

-m *time signature*

time signature

-b *bars* number of bars wanted in output

-Q *tempo*

tempo in quarter-notes per minute

-u pulses

Allows you to specify directly the number of midi pulses per abc time unit.

-ppu parts per abc unit length

Normally, the smallest note unit that *midi2abc* can extract is half the L: unit length. This is called the quantum unit. Thus for L: 1/8, *midi2abc* can extract 1/16 notes but not 1/32 notes. You can change this by specifying *-ppu 4* for example. The number of parts should be a power of 2.

-aul denominator of abc unit length

Normally *midi2abc* chooses a unit length of 1/8 or 1/16 depending upon the time signature. For time signatures smaller than 3/4 the L: 1/16 is used and for larger time signatures L: 1/8 is used. You can specify the unit length to be used using this parameter. Thus *-aul 32* will cause *midi2abc* to use a unit length of 1/32 nd note.

- gu** Tells midi2abc to estimate the number of midi pulses per abc time unit from the note duration or spacing in the MIDI file.
- gk** Tells midi2abc to guess the key signature by minimizing the number of accidentals even if the key signature is already specified in the MIDI file. By default the key signature is the one specified in the MIDI file. If it is not specified, then the program guesses the key signature by minimizing accidentals.
- k key** key signature: -6 to 6 sharps.
- c channel**
select only this midi channel.
- f infile** input file in midi format
- o output file**
specifies the output abc file name.
- s** do not discard very short notes.
- sr quantum units**
do not notate a short rest smaller than the specified size after a note. If the size (in quantum units) is zero, nothing is done. For larger values, the rest is absorbed into the preceding note. In other words, the preceding note is lengthened to include that rest.
- sum** print a short summary of the input midi file.
- nb** do not look for broken rhythms
- nt** do not look for triplets
- obpl** Print only one bar per line instead of 4. For complex music this improves the readability and avoids some problems with some abc to postscript converters. This option is deprecated.
- nogr** (No note grouping.) Inserts a space between all notes. It makes a less pretty postscript file but it is easier to edit.
- bpl nbars**
Print nbars of music on every line followed by a backslash.
- bps nbars**
When nbars have been printed (including those lines joined by a backslash continuation) go to a new line (with no backslash).
- splitvoices**
This parameter handles polyphonic chords by splitting an entire voice into multiple voices. (A polyphonic chord is a chord composed of notes that do not start or end at the same time.) Normally, midi2abc joins the longer notes to the notes in the following chord using ties. Here is an example: [Bd-]d [Bd-]d|. This should be separated into two voices ideally Bz Bz and d2 d2. However, the separation is not unique. Bz d2 and d2 Bz are also ok.
- midigram**
When this option appears, all other options are ignored and no abc file is produced. Instead a list of all notes in the MIDI file are printed in a fixed format. Each line represents a pair of MIDI note on/off event. The line contains the on/off time of the note, its track number, channel number, midi pitch and midi velocity. The last record indicates the duration of the MIDI file in MIDI pulse units. The output is designed to go into a graphical user interface which will produce a graphical representation (piano roll).
- mftext**
When this option appears, all other options are ignored and no abc file is produced. Instead a list of all the MIDI commands are printed. The output is designed to go into a graphical user interface provided by runabc.tcl.

-mftextpulses

Same as -mftext except the time units is in midi pulses.

-title string

Replaces the default title field following T: with the given string.

-origin string

Adds an O: field with the given string.

-midinotes

Like -mftext, this option outputs the midi pitches present in the midi file, where middle C is 60, the deviation of the note from the equal tempered scale in units of cents, and the pitch value in cents of the note relative to C. If the pitchbend is used, then its value follows cents in parentheses. There are 100 cents in the semitone interval. Thus a pitch half way between C and C# would have a cent value of 50. There are 1200 cents in an octave, so the pitch value of D natural would be 200. The pitch value of B natural is 1100. These values are useful for verifying the pitch values in an equaltempered scale divided into more than 12 tones. The pitch values are determined from the pitchbend value in the midi file, which is normally 8192. However, for some midi players the pitch can also be altered using sysex (System Exclusive) messages. In particular, Jef Moine's abc2svg uses the latter rather than pitchbends which act on all the notes in a channel.

FEATURES

- * The key is chosen so as to minimize the number of accidentals. Alternatively, the user can specify the key numerically (a positive number is the number of sharps, a negative number is minus the number of flats).
- * Note length can be set by specifying the total number of bars or the tempo of the piece. Alternatively the note length can be read from the file. However, by default it is deduced in a heuristic manner from the inter-note distances. This means that you do not have to use the MIDI clock as a metronome when playing in a tune from a keyboard.
- * Barlines are automatically inserted. The user specifies the number of measures in the anacrusis before the first barline and the time signature.
- * The program can guess how the length of the anacrusis, either by looking for the first strong note or minimizing the number of notes split by a tie across a barline.
- * Where a note extends beyond a bar break, it is split into two tied notes.
- * The output has 4 bars per line.
- * Enough accidental signs are put in the music to ensure that no pitch errors occur if a barline is added or deleted.
- * The program attempts to group notes sensibly in each bar.
- * Triplets and broken rhythm (a>b) are supported.
- * Chords are identified.
- * Text information from the original MIDI file is included as comments.
- * The -c option can be used to select only 1 MIDI channel. Events on other channels are ignored.

LIMITATIONS

midi2abc does not ...

- * Supply tune title, composer or any other field apart from X: , K:, Q:, M: and L: - these must be added by hand afterwards, though they may have been included in the text of the MIDI file.

- * Support duplets, quadruplets, other esoteric features.
- * Support mid-tune key or time signature changes.
- * Deduce repeats. The output is just the notes in the input file.
- * Recover an abc tune as supplied to abc2midi. However, if you want to do this, "midi2abc -xa -f file.mid" comes close.

SEE ALSO

abc2ps(1), abc2midi(1), abc2abc(1)

AUTHOR

James Allwright <J.R.Allwright@westminster.ac.uk>

SUPPORTED

Seymour Shlien <fy733@ncf.ca>

VERSION

This man page describes midi2abc version 2.91 from March 09 2008.

COPYRIGHT

Copyright 1999 James Allwright

midi2abc does not work correctly if lyrics are embedded in the same track as the notes. If you are producing the MIDI file using abc2midi, use the -STFW option to ensure that the lyrics are put in a separate track.

midi2abc is supplied "as is" without any warranty. It is free software and can be used, copied, modified and distributed without fee under the terms of the GNU General Public License.