

MIDIFILE TO CSOUND CONVERTER V 0.92 R&U

MIDI2CS

shareware developed by
dipl.-ing ruediger bormann

(C) SONGLAB 1995
All Rights Reserved

Read this manual if the following topics seem to be interesting!

CSOUND, SCORE, ORCHESTRA, COMPOSER, COMPOSING,
MUSIC PRODUCTION, PRODUCER, ELECTRONIC MUSIC, MIDI, MIDIFILE,
SEQUENCER, COMPUTER MUSIC, HOME RECORDING,
SAMPLING, STUDIO, CD-AUDIO, PC, SOUND CARD, WAV, AIFF

Table of content

Introduction
Files
Composing and producing music
Installation
Usage
Options
Projectfiles
Tips & Tricks
Known Bugs & Additional Features
Tip
Plans for the future
License Agreement
Shareware
Registration
About the author

Introduction

Csound is one of the worlds most important audio processing systems. The system was developed at the Media Lab of the famous Massachusetts Institute of Technology by Barry L. Vercoe. The PC version is available by anonymous ftp in directory <ftp://ftp.maths.bath.ac.uk/pub/dream>. There you will find lots of informations about the Csound system and its use.

Nowadays you are able to run Csound on a PC to create high quality music productions, but using Csound is still difficult for starters. You need to know lots of things before any result is worth listening to.

MIDI2CS solves the problem for anyone who likes to work with sequencers and midifiles while composing music. MIDI2CS is the solution to create best documented Csound scores from any midifile on the fly. This will save lots of your time and money in producing music in professional quality. Using Csound does not seem to be a waste of time anymore.

MIDI2CS is fully independent from Csound and should run on any PC under DOS or in a DOS Windows under MS-Windows. Users of Csound on other than PC-platforms are able to run MIDI2CS on PCs without an installed Csound version.

MIDI2CS does not contain any part of Csound software and therefore does not touch any copyright.

Files

MIDI2CS	EXE	78,181	- the program
CS	BAT	51	- my csound batch file
091	BAT	281	- usage of version 0.91 just call it !
092	BAT	19	- usage of version 0.92 just call it !
CPS	BAT	19	- creating cps-scores just call it !
092	PRO	1,040	- used by 092.bat
FATIMA	PRO	841	- used by 092.bat
CPS	PRO	1,036	- used by cps.bat
FATIMA	MID	40,578	- an example midifile
FATIMA	TXT	413	- fatima.mid copyright
FATIMA	HDR	1,513	- fatima scoreheader
FATIMA	ORC	1,436	- fatima orchestra example
MIDI2CS	WRI	28,800	- manual etc.
MIDI2CS	TXT	23,287	- ascii manual

Composing and producing music

Composing and producing music in my view are two different things.

Composing music can be done with cheapest equipment with cheapest keyboards and sequencers. You really do not have to invest thousands of dollars to build your homestudio. The result will be a combination of cheap sounds and noone but you expects that it will be a powerful record someday. You are working in realtime and you will have no difficulties to compose your song in a reasonable time. The result of your work as a composer should be a midifile!

Producing music means much more than composing songs. You will have to look out for best sounds and effects to get the maximum out of your composition. You will have to find the best studio and the best equipment. You will have to produce the maximum of dynamics. If you do not have the knowledge you have to pay a good producer for doing this work. This time is over if you do not expect to make everything in realtime. Best effects will take a while for processing on a PC, but it's worth it. Following PC generations will make the processing faster and faster. In my view it really does not matter anymore.

MIDI2CS now gives you the opportunity to be your own producer. You need a computer with an installed Csound system and a program to play the created soundfiles. If you invest in a good soundcard or a digital output to your DAT noone will be able to beat the quality of your production!

Installation

You just have to copy the executable file of MIDI2CS.EXE into your utility directory. That should do.

Usage

The easiest command if you name your midifile midi2cs.mid is

```
midi2cs
```

Of course MIDI2CS supports any filename if you do not want to rename it.

```
midi2cs "name of your midifile"
```

These commands will create full MIDI2CS default scores from the first to the last bar of the midifile containing all tracks. MIDI2CS versions up from v0.92 are able to write variable parameters in variable successions.

Using only the name of the midifile means to write the default values which are the seven parameters instrument, start, duration, pitch, midinote, midivelocity and midirelease.

If you would like to change the succession of the default parameters you will have to configure your project file. Its default name is midi2cs.pro.

If you would like to make simple modifications to the resulting score, you can do that by using options.

```
midi2cs [options] "name of midifile"
```

If you are an experienced user you should configure the projectfile and type

```
midi2cs -r"name of the projectfile"
```

If your projectfile is named midi2cs.pro you just have to type

```
midi2cs
```


Options

-a

The output will not overwrite an existing scorefile. It will be appended.

-b#

If you do not want to convert the whole midifile you can specify the first track you'd like to convert e.g. -b25 means begin with track 25.

-d#,#[,#,#]...

If you have a single instrument that should have a fixed duration you are able to specify the duration by typing e.g. -d3,2500,5,350. This means that all notes of track 3 will have the length 2.5 seconds and all notes of track 5 will have a duration of 0.35 seconds.

-e#

If you do not want to convert the whole midifile you can specify the last track you'd like to convert e.g. -e30. This means that the last bar will be track 30.

-f"filename"

Csound scores often consist of a part with function declarations and other specific stuff. This part can be included as a file by default.
E.g. by typing -fdummy.hdr

-i#,#[,#,#]...

If you like to give each track a specific instrument number you should use this option to assign it. E.g. -i4,23,8,9 assigns instrument number 23 to track 4 and instrument number 9 to track 8.

-l

If you do not like too much chat and comments inside your scorefile use this option. It will suppress most of the unnecessary text.

-o"filename"

This option is useful to assign the name of the scorefile.
E.g. -odummy.sco creates the score dummy.sco

`-p#,#[,#,#]...`

Drumtracks sometimes need just the 3 parameters instr, start and duration. If you use this option you can suppress the writing of unnecessary parameters. E.g. `-p7,4,3,3` causes that track 7 will have 4 of the default parameters and track 3 just the minimum of 3.

`-r"filename"`

This option is one the most powerful option of MIDI2CS and new to v0.92. If you type `-rproject.pro` it will read the projectfile. Any syntax errors will be printed out to the screen. See the description of the syntax in chapter projectfile.

`-s#[,#]...`

If you do not want to write all tracks as score you can select some of them by typing e.g. `-s3,6,12`. Only tracks 3,6 and 12 will be written.

`-t#,#[,#,#]...`

If you like to transpose tracks this option helps you to do this easily. E.g. type `-t3,5,7,-7` transposes all notes of track 3 5 halftones and all notes of track 7 -7 halftones.

The following options will give you some online information if you do not have this manual by hand. You do not need to type the name of a midifile for the following options.

Online help is available by typing

```
midi2cs -h (elp)
midi2cs -?
```

Known bugs and additional features of the new version by typing

```
midi2cs -k (nown bugs)
```

The following option explains why MIDI2CS is shareware

```
midi2cs -w (hy is this shareware)
```


Project files

```
; --- MIDI2CS PROJECT FILE ---

; - this default projectfile is identical to default options -
; - written automatically if it does not exist by calling midi2cs -

; --- MIDIFILE GLOBAL VALUES SECTION ---

scoreheader midi2cs.hdr ; file to be included (readonly)
nameofscore midi2cs.sco ; score output (will be written)
; nameoforc midi2cs.orc ; future extension not realized yet
midifile      midi2cs.mid

; - all tracks will be initialized by the following values -

firstbar      0
lastbar       10000
; notrackselected
; duration 1000
; transpose 0
; lesscomments

; - define default parameters and succession -

p_pch
p_midinote
p_midivelocity
p_midirelease
; p_cps midi2cs.cps
; p_maxamplitude 32000
; p_fixeddB -6

; --- MIDITRACK SECTION ---
; - define individual values for each track -

miditrack      1
    ; instrument 1
    ; name dummy1
    score                ; all score parameters and order are optional

                        ; firstbar      0
                        ; lastbar       10000
                        ; duration 1000 ; fixed duration in milliseconds
                        ; transpose 0

                        ; --- parameters and succession ---
                        ; p_pch
                        ; p_midinote
                        ; p_midivelocity
                        ; p_midirelease
                        ; p_maxamplitude
```

```
        ; p_fixeddB -6
        ; p_cps midi2cs.cps

        ; - if you want to use default parameters set limit with
parameters -
        ; parameters 10 ; parameters has to be the last entry
                        ; in the score section (instr + start +
                        ; duration + the defined parameters)
        endscore
        ; separator
endtrack

endofproject
```

CPS-Files

You want your own scales ? No problem.

Create a file like the following example and configure your projectfile. Each track can have its own CPS-table!

```
; MIDI2CS - well tempered cycles per second
; uncomplete internal table of MIDI2CS v0.92
```

```
; midibyte  cyclespersecond  note
```

```
000          30          ;      C-2
001          30          ;      C#-2
002          30          ;      D-2
003          30          ;      D#-2
004          30          ;      E-2
005          30          ;      F-2
006          30          ;      F#-2
007          30          ;      G-2
008          30          ;      G#-2
009          30          ;      A-2
010          30          ;      A#-2
011          30          ;      B-2
012          30          ;      C-1
013          30          ;      C#-1
014          30          ;      D-1
015          30          ;      D#-1
016          30          ;      E-1
017          30          ;      F-1
018          30          ;      F#-1
019          30          ;      G-1
020          30          ;      G#-1
021          30          ;      A-1
022          30          ;      A#-1
023          30          ;      B-1
024          30          ;      C0
025          30          ;      C#0
026          30          ;      D0
027          30          ;      D#0
028          30          ;      E0
029          30          ;      F0
030          30          ;      F#0
031          30          ;      G0
032          30          ;      G#0
033          30          ;      A0
034          30          ;      A#0
035          30          ;      B0
036          65.41        ;      C1
037          69.29        ;      C#1
038          73.41        ;      D1
```

039	77.78	;	D#1
040	82.4	;	E1
041	87.31	;	F1
042	92.49	;	F#1
043	97.99	;	G1
044	103.8	;	G#1
045	110	;	A1
046	116.5	;	A#1
047	123.5	;	B1
048	130.8	;	C2
049	138.6	;	C#2
050	146.8	;	D2
051	155.6	;	D#2
052	164.8	;	E2
053	174.6	;	F2
054	185	;	F#2
055	196	;	G2
056	207.6	;	G#2
057	220	;	A2
058	233.1	;	A#2
059	246.9	;	B2
060	261.6	;	C3
061	277.2	;	C#3
062	293.7	;	D3
063	311.1	;	D#3
064	329.6	;	E3
065	349.2	;	F3
066	370	;	F#3
067	392	;	G3
068	415.3	;	G#3
069	440	;	A3
070	466.1	;	A#3
071	493.9	;	B3
072	523.3	;	C4
073	554.3	;	C#4
074	587.3	;	D4
075	622.3	;	D#4
076	659.2	;	E4
077	698.4	;	F4
078	739.9	;	F#4
079	783.9	;	G4
080	830.6	;	G#4
081	880	;	A4
082	932.3	;	A#4
083	987.7	;	B4
084	1047	;	C5
085	1109	;	C#5
086	1175	;	D5
087	1245	;	D#5
088	1318	;	E5
089	1397	;	F5
090	1480	;	F#5
091	1568	;	G5

092	1661	;	G#5
093	1760	;	A5
094	1865	;	A#5
095	1975	;	B5
096	2093	;	C6
097	3000	;	C#6
098	3000	;	D6
099	3000	;	D#6
100	3000	;	E6
101	3000	;	F6
102	3000	;	F#6
103	3000	;	G6
104	3000	;	G#6
105	3000	;	A6
106	3000	;	A#6
107	3000	;	B6
108	3000	;	C7
109	3000	;	C#7
110	3000	;	D7
111	3000	;	D#7
112	3000	;	E7
113	3000	;	F7
114	3000	;	F#7
115	3000	;	G7
116	3000	;	G#7
117	3000	;	A7
118	3000	;	A#7
119	3000	;	B7
120	3000	;	C8
121	3000	;	C#8
122	3000	;	D8
123	3000	;	D#8
124	3000	;	E8
125	3000	;	F8
126	3000	;	F#8
127	3000	;	G8

Tips&Tricks

TIP 1: Use version 0.92 and the powerful projectfile !!

Nothing to edit anymore afterwards. The score should be perfect. Otherwise your projectfile seems not to be configured right.

TIP 2: Using Csound

Csound commandlines can be complicated. So if you do not like to think of options and typing them you should write yourself a batch file.

My Csound batchfile is named cs.bat and contains the following two lines:

```
echo off
csound -d -H -W -o %1.wav %1.orc %1.sco
```

Csound writes the default WAV file into the sampledirectory. It will have the same prefix the score and the orchestra files have.

e.g. `cs "prefix of score and orcfile"`

TIP 3: Working with version 0.91

You can use lots of options if you e.g. are writing a batch file that does the following. You will have to edit the created score a little bit afterwards.

```
rem 6 Bass
midi2cs -b34 -e51 -l -s6 -i6,20 -p6,4 -ffatima.hdr
fatima.mid
rem 7 Bassdrum
midi2cs -a -b34 -e51 -l -s7 -d7,800 -i7,1 -p7,3
fatima.mid
rem 8 Snare
midi2cs -a -b34 -e51 -l -s8 -d8,1000 -i8,2 -p8,3
fatima.mid
rem 9 Hihat
midi2cs -a -b34 -e51 -l -s9 -d9,190 -i9,3 -p9,3
fatima.mid
rem 1 Juno
midi2cs -a -b34 -e51 -l -s1 -i1,21 -p1,4 fatima.mid
rem 3 Strings
```

```
midi2cs -a -b34 -e51 -l -s3 -i3,21 -p3,4 fatima.mid  
rem 5 Sequence  
midi2cs -a -b34 -e51 -l -s5 -i5,21 -p5,4 fatima.mid
```

Known bugs and additional features

New Features v0.92

- a) New power option -r is supported (project file midi2cs.pro) !!!
- b) Variable parameters and succession depending on project file
- c) Printing midinotes as cps depending on project file and cps file
- d) Option -k is working
- e) Extended General MIDI comments for sound switches
- f) Append function deletes last e (end of score)
- g) Time statement t is just a comment now, put it in your headerfile!
- h) All default files have the prefix midi2cs now (midi2cs.sco, midi2cs.orc)
- i) Reading from default midifile midi2cs.mid if exists
- j) Reading from default project file midi2cs.pro if exists
- k) Some other little things to make everything more comfortable
- l) Option 'n' - not supported anymore
Take option "notracksselected" in your default projectfile

Assumed Bugs

- a) The default table with midivalues to cps values is not o.k yet

Known Bugs

- a) Midi format 0 seems to be not as good supported as format 1
- b) The tempo statement (t 0 ...) seems to be not calculated o.k. yet
Therefore I write it as a comment. Put it into your headerfile.

Plans for the future

Maybe the final result of MIDI2CS v1.0r ...

a) will have the full functionality to write several scores and sample-orchestras with specified default effects automatically. There will be no necessity to edit something afterwards in scores or orchestras so that Csound is able to calculate the soundfile. When this result is reached the implementation of the following points begins.

b) there will be an additional windows desktop version.

c) the windows desktop version will support the configuration of projectfiles with a lot of configuration menues. These will be easy to handle.

d) the windows desktop version will be able to trigger Csound to create WAV files.

e) the windows desktop version will be able to play the created WAV-files.

f) If there is interest, I will create a Linux version of MIDI2CS of the last DOS version of MIDI2CS.

g) If there is interest, a MAC version will be created of the last DOS version of MIDI2CS.

I hope that lots of people are interested in the development of this music production tool. The development and release of new versions will continue in dependence of the number of registered users.

Send questions, comments, suggestions for further development, not working midifiles and anything useful to SONGLAB.

License Agreement

This is a legal agreement to which you are consenting to be bound. If you do not agree to all of the terms of this license, do not use the software.

1. SONGLAB grants to registered users a non-exclusive, non-sublicensable, unlimited license to use this version of MIDI2CS in binary executable form.

Unregistered users will have the right to use this software for trial and evaluation use purposes and for educational and research purposes only. If you want to become your own producer this is commercial usage and you should register MIDI2CS. The unregistered version of MIDI2CS will delete itself in december 1995.

2. SONGLAB makes no representations about the suitability of this software or about any content or information made accessible by the software, for any purpose. The software is provided 'as is' without express or implied warranties, including warranties of merchantability and fitness for a particular purpose or noninfringement. SONGLAB shall not be liable under any theory or any damages suffered by you or any user of the software.

3. All content accessed through the Software is the property of the applicable content owner and may be protected by applicable copyright law. This License gives you no rights to such content.

4. While SONGLAB intends to distribute a version 1.0 release of MIDI2CS SONGLAB reserves the right at any time not to release this or, if released, to alter prices, features, specifications, capabilities, functions, licensing terms, release dates, general availability or other characteristics of the release.

5. Title, ownership rights, and intellectual property rights in and to the Software shall remain in SONGLAB and/or its suppliers. You acknowledge that the Software in source code form remains a confidential trade secret of SONGLAB and/or its suppliers and therefore you agree not to modify the Software or attempt to decipher, decompile, disassemble or reverse engineer the Software, except to the extent applicable laws specifically prohibit such restriction.

6. This License is personal to you and you agree not to assign your rights herein.

SONGLAB is a trademark of SONGLAB
Copyright (c) 1995 SONGLAB
All rights reserved

Shareware

MIDI2CS will save lots of your time. Now you are able to compose your Csound works with a sequencer and map your midi files on the fly to best documented csound scores.

It gives me lots of motivation to see that people need this program as I do and are willing to participate in the process of development.

Registration of this program will give you a high influence of what midi2cs v1.0 will be able to do when it is released.

The use of v0.92u (u stands for unregistered version) is limited to the end of november 1995.

I think it's worth the price if I only charge you one hour of my work and that's US \$50 or DM 75. Please register.

Registration

Write a letter to SONGLAB, R.Borrmann, Bonhoefferufer 13, 10589 Berlin with your address and a cheque of \$50 or DM75.

I will send you a personal registered copy of the latest MIDI2CS.EXE on disc with current documentation and maybe other helpful tools that are in development now.

Registered users will receive all new versions of MIDI2CS that contain major changes until MIDI2CS v1.0r is finished.

About the author

The author of MIDI2CS Dipl.-Ing. Rüdiger Borrmann is an independent composer and music producer. Since 1989 he works as an employed software developer at Sietec Systemtechnik in the development of archive and information systems and WWW. He is married, has one child and lives in the middle of Berlin.

The thirty four years old musician began playing guitar when he was fourteen. He founded several bands. After school he studied at the Technical University of Berlin from 1982-88 to become an electrical engineer. During his studies his music changed from handmade rock music to electronic based music. He learned a lot about programming computers and electronic music and had the opportunity to work in the field of synthsizers and MIDI to finish his degree.

In 1991 he founded his own label SONGLAB which always seemed to be more a hobby than a real possibility of income. He spent more than DM 100.000 in studios to produce his music and studied the music producers work. Some records were produced that did not become hits but are good as hell as he would pretend.

In 1994 he decided to build an own studio and bought a PC based harddisk recording system for SONGLAB. To complete his home studio that should produce better results than any of the expensive studios in the city he began to work with Csound and the development of MIDI2CS.