AutoDrum4.2

Automatic Drum performance software

User's Manual

2016/06/26 The 42th edition publish

E-mail:ee65051@yahoo.co.jp(Temporary)
URL:http://www.openmidiproject.osdn.jp/

Thank you for downloading or receiving AutoDrum 4.2

First please read “readme_en.txt” before reading this manual.

This guide is written by using OpenOffice4.1.0 Writer. It is recommended to print to paper for reading.

Attention

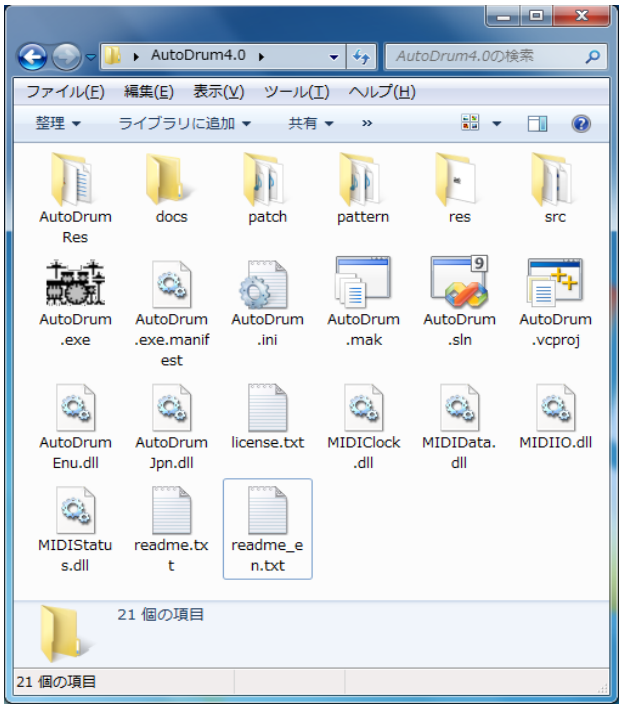
Index

1. Install and Execute

This software doesn't have an installer. You only need to extract zip file.

1-1. Install

(1) Please extract AutoDrum4.2.zip with folder. Following files and folders will be appeared. Please check all files and folders are exist.



* If “Hide hidden files and folders” is selected in My computer or Window explorer's folder option dialog, “*.dll” files are not shown. It is recommended to select “Show all files and folders” to check to exist “*.dll” files.

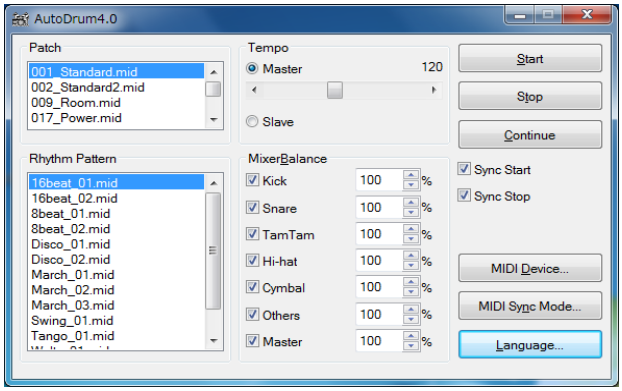
* Don't put AutoDrum in the “c:\program files” or “c:\ program files(x86)” or “c:\windows” folder. These folder is controled by Windows User Account Control (UAC) feature so writing configuration file (*.ini) is blocked.

FileName	Description
AutoDrum.exe	Main program.
AutoDrum.exe.manifest	Manifest file.
AutoDrum.ini	Configuration file.
AutoDrum.sln	Solution file for Microsoft Visual Studio 2008 Service Pack 1.
AutoDrum.vcproj	Project file for Microsoft Visual Studio 2008 Service Pack 1.
AutoDrum.mak	A make file for C/C++.
AutoDrumEnu.dll	AutoDrum Chinese language resource DLL.
AutoDrumEnu.dll	AutoDrum English language resource DLL.
AutoDrumJpn.dll	AutoDrum Japanese language resource DLL.
MIDIIO.dll	MIDI messeage input or output library.
MIDIClock.dll	MIDI clock measuring library.
MIDIData.dll	MIDI data creating / editing library.
MIDIStatus.dll	MIDI module's status keeping library.
readme.txt	Please read me first (Japanese).
readme_ch.txt	Please read me first (Chinese).
readme_en.txt	Please read me first (English).
license.txt	License (LGPL)
src	A folder for C source files (*.c), C header files (*.h), and resource script file (*rc).
res	A folder for resource files like *.bmp, *.ico, *.cur, and so on.
AutoDrumRes	A language depending resource script folder.
docs	A folder for documentations (*.odt) (*.pdf) .
patch	A folder for Patch MIDI data (*.mid).
pattern	A folder for Pattern MIDI data (*.mid)

I

1-2. Execute

Please double click AutoDrum.exe in the “my computer” or “explorer”. Following main window will appear.



* Don't forget to extract all files. Otherwise, AutoDrum will cause error.

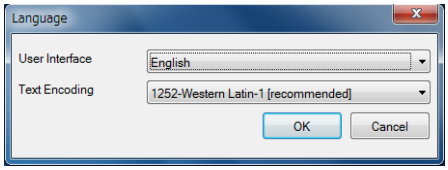
* Please see also 4. trouble shooting if AutoDrum doesn't execute normally.

* AutoDrum must be executed on the local computer that “AutoDrum.exe” is installed. Execution from the network computer will causes some troubles.

1-3. Setup of language

AutoDrum is made in Japan, therefore default GUI language is Japanese. You may select English language, there is two way to change language. (1) is from GUI, (2) is form text editor.

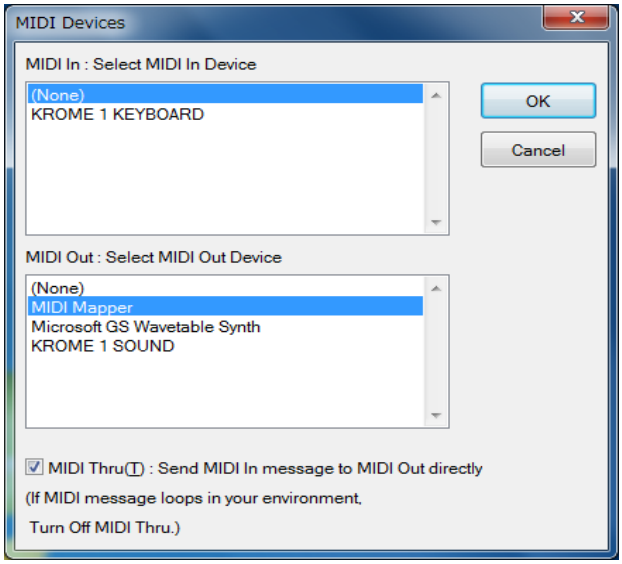
(1) Push "Language..." button and select language in the dialog, and restart AutoDrum again. If it is difficult to see the button because of character corruption, press [Alt]+[L] key, and you may open the dialog.



(2) Open "AutoDrum.ini" in your text editor, and change "Language=Japanese" into "Language=English" or "Language =Chinese", and then execute AutoDrum.

1-4. Setup of MIDI device and instrument

First you must select MIDI In device and MIDI Out device, so as to fit your using MIDI device in the MIDI Devices dialog from "MIDI Device...” button. In the default setting, “(None)” is selected for MIDI In device and “MIDI Mapper” is selected for MIDI Out device. If you select “(None)” as z MIDI Out Device, you will get no sound.



1-5. Exit

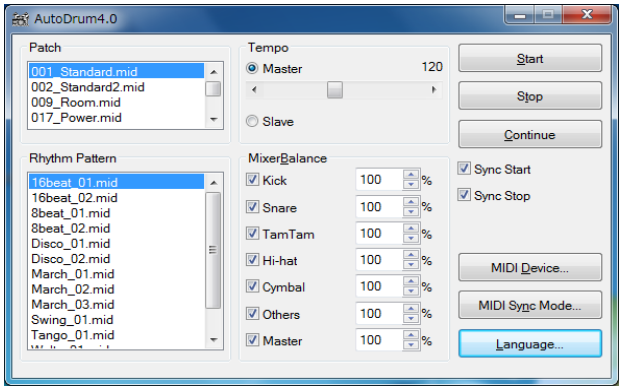
Push left-top button, or press [Alt] + [F4] key.

1-6. Uninstall

Delete “AutoDrum4.2” folder which contains “AutoDrum.exe”

2. Operations

2-1. Main window



Patch

Select type of drum set from the list box. This changes type of drum set immediately. You can use various type of drum set in one rhythm pattern.

AutoDrum has 9 type of preset drum set (patch). These patches are available at both GS module and XG module. If you double click the item, you can see the property of the MIDI data.

Patch Name	Description
001_Standard.mid	Standard drum set 1
002_Standard2.mid	Standard drum set 2
009_Room.mid	With room ambient
017_Power.mid	For powerful hard rock.
025_Electronic.mid	Electric drum set
026_TR808.mid	TR-808 drum set
033_Jazz.mid	For Jazz with stick.
041_Brush.mid	For Jazz with brush.
049_Orchestral.mid	For orchestral instrument.

* Patch data is provided as a standard MIDI file (*.mid). These MIDI data contains patch change(CC#0, CC#32, PC), volume(CC#7), Pan(CC#10), Expression(CC#111), Reverb send level(CC#91) and so on. These MIDI data does not contain any note on and note off event. A Tempo event including these MIDI data is ignored.

* You can add your original patch MIDI data for your MIDI module. See also 3. Create original MIDI data.

Rhythm Pattern

Auto Drum has following 13 preset rhythm patterns. Select your favorite pattern from the list box. If you double click the item, you can see the property of the MIDI data.

Pattern Name	Description
8beat_01.mid	Normal 8beat.
8beat_02.mid	Up beat 8beat.
16beat_01.mid	Normal 16beat.
16beat_02.mid	Up beat 16beat.
Disco_01.mid	Mainly bass drum and hi-hat.
Disco_02.mid	Mainly bass drum and hi-hat .
March_01.mid	Mainly snare.
March_02.mid	Mainly snare.
March_03.mid	Mainly snare.
Swing_01.mid	Mainly ride cymbal.
Tango_01.mid	For 4/4.
Waltz_01.mid	For 3/4.
Waltz_02.mid	For 3/4.

* Rhythm Pattern data is provided as a standard MIDI file (*.mid). These MIDI data contains note on and note off event for one loop. These MIDI data does not contain set up data like patch change (CC#0, CC#32, PC) and so on. A Tempo event including these MIDI data is ignored.

* You can add your original pattern MIDI data for your MIDI module. See also 3. Create original MIDI data.

Tempo

AutoDrum4.2 supports two type of tempo mode, one is master, the other is slave.

In master mode, AutoDrum uses internal timer and generete tempo. You can specify the tempo from 16 [BPM] to 256 [BPM] by moving the scroll bar at any time. If you click left or right button of the scroll bar, you can change the tempo at 1 [BPM] step.

In slave mode, AutoDrum uses external machine's signal and synchronize to the signal. To use slave mode normally, MIDI timing clock signal or SMPTE/MTC signal must be received from MIDI In device. You can specify which signal to slave in the sync mode dialog. Normally MIDI timing clock (0xF8) is used, which is sent 24 times per quarter note continuously, and the tempo is depend on the interval MIDI timing clock.

In each mode, a tempo event including MIDI data is ignored.

Mixer and Balance

In AutoDrum4.2, the note event is grouped to the following 6 groups, which is, Kick, Snare, Tamtam, Hi-hat, Cymbal, and Others.

Group	Note number : Instrument name
Kick	35(B1)：Standard Kick 1 36(C2)：Standard Kick 2
Snare	38(D2)：Acoustic Snare 40(E2)：Electronic Snare
Tamtam	41(F2)：Low Tom 2 43(G2)：Low Tom 1 45(A2)：Mid Tom 2 47(B2)：Mid Tom 1 48(C2)：High Tom 2 50(D2)：High Tom 1
Hi-hat	42(F#2)：Closed HiHat 44(G#2)：Pedal HiHat 46(A#2)：Open HiHat
Cymbal	49(C#3)：Crash Cymbal 1 55(A3)：Crash Cymbal 2
Others	Others

Each group's sound can be muted independently by checking off the each group's check box. If check box is checked, it sounds, and you can adjust the group's velocity level from 1% to 200% independently. Master's velocity level affects all sounds, the output velocity level is following formula.

$$\text{Output velocity} = \text{Original velocity} * \frac{\text{Group}}{100} * \frac{\text{Master}}{100}$$

* If the output velocity becomes over than 127, the output velocity becomes 127.

Start

Start playing the rhythm pattern from the beginning of the MIDI data, and repeat at the end of MIDI data automatically. If AutoDrum detect start command (0xFA) from MIDI In device, then also start playing.

Stop

Stop playing the rhythm pattern immediatelly. The current note on sound will be note offed except in case of the hold pedal is downed. If AutoDrum detect stop command (0xFC) from MIDI in device, then also stop playing.

Continue

Continue playing the rhythm pattern from the position that you've stopped, and repeat at the end of MIDI data automatically. If AutoDrum detect continue command (0xFB) from MIDI in device, then also continue playing.

Sync start

If this check box is checked, AutoDrum start playing the rhythm pattern automatically when your MIDI keyboard's key is pressed or the hold pedal is downed.

Sync stop

If this check box is checked, AutoDrum stop playing the rhythm pattern automatically when your MIDI keyboard's key is all released and the hold pedal is upped. If the hold pedal is kept downed, AutoDrum stop playing when the hold pedal is upped.

MIDI Device...

Open MIDI Device dialog.

MIIDI Sync Mode...

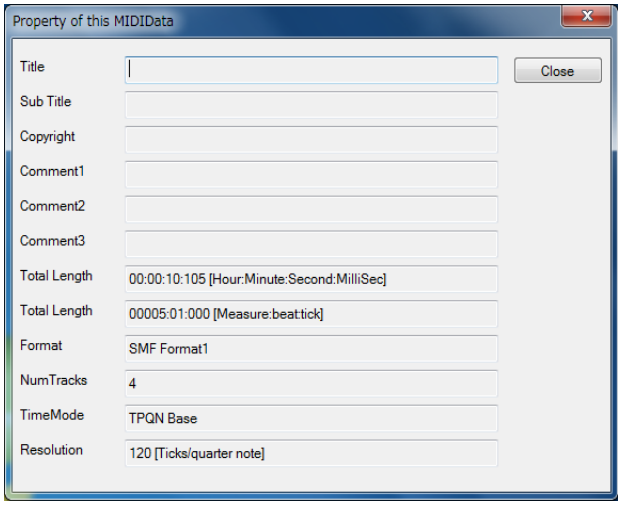
Open MIDI Sync dialog.

Language

Open Language dialog.

2-2. “Property of this MIDIData” Dialog

This dialog shows basic property of the MIDI data. This dialog can be opened by double-clicking list box's MIDI data.



Title

This shows the title of this MIDI data. This is related with the first track's the first track name event.

SubTitle

This shows the title of this MIDI data. This is related with the first track's the second track name event.

Copyrights

This shows the copyright of this MIDI data. This is related with the first track's the first copyright event.

Comment

This shows the comment of this MIDI data This is related with the first track's the first text event.

Total Length

This shows total length of this MIDI data as [hour : minute : second : millisec] and [measure : beat : tick](in TPQN base) or [frame : subframe] (in SMPTE base).

Format

This shows standard MIDI file's format 0 or 1. Format 0's MIDI data contains only one track and all events are included in the track. Format 1's MIDI data contains multiple track and tempo, time signature, key signature and so on' s events are included in the first track, which is called conductor track, and MIDI channel events like note on event are included in the second or following track.

NumTracks

This shows how many tracks are contained in this MIDI data. In format 0's MIDI data, this value is always 1. In format 1's MIDI data, this value is larger than 1 and the first track is a conductor track.

TimeMode

This shows the time mode of this MIDI data. Which is, TPQN Base, SMPTE24base, SMPTE25base, SMPTE29.97base, or SMPTE30base. Normally, MIDI data is TPQN (Ticks per quarter note) base. In AutoDrum, SMPTE base's MIDI data can't be used.

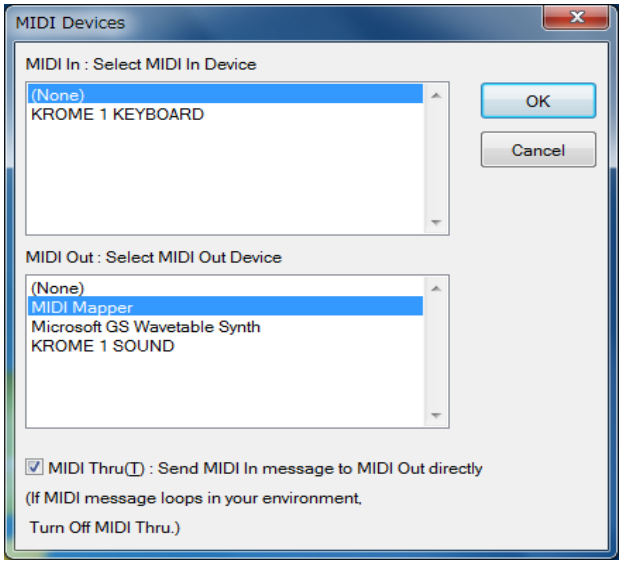
Resolution

This shows the time resolution of this MIDI data. If the MIDI data is TPQN base, this shows the resolution of quarter note [ticks per quarter note], which is normally 48, 72, 96, 120, 144, 168, 192, 216, 240, 360, 384, 480, or 960. If the MIDI data is SMPTE base,. This shows the resolution of 1 frame [subframes per 1 freame].

Close

Close this dialog.

2-3. “MIDI Device” Dialog



MIDI In

This shows the list of MIDI In Device which is installed to your Windows. Select one of them the your MIDI keyboard or MIDI controller is connected.

MIDI Out

This shows the list of MIDI Out Device which is installed to your Windows. Select one of them which your MIDI module or synthesizer which you want to play a sound is connected.

If you select “MIDI Mapper”, the default MIDI Out device which is selected in the windows control panel's “sound and multimedia”

MIDI Thru

If this check box is checked on, AutoDrum outputs the inputted MIDI message from MIDI in device into the MIDI Out Device directly.

If you connect your MIDI keyboard's input and MIDI module's output , MIDI message loops eternally and the sound will not stops. In this case, check off the MIDI Thru.

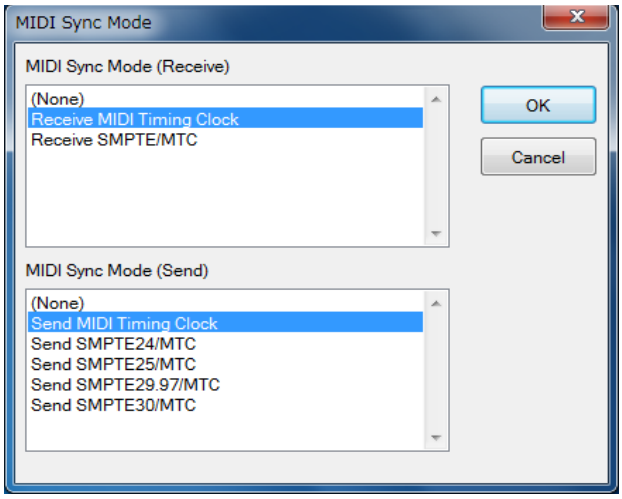
OK

Close this dialog and open specified MIDI devices.

Cancel

Close this dialog without change.

2-4. “MIDI Sync Mode” Dialog



MIDI Sync Mode (Receive)

Select the type of sync signal in the slave mode. If you select “(None)” here, time will not progress in the slave mode. Also, if you select “(None)” as a MIDI In Device, time will not progress in the slave mode.

a) Receive MIDI Timing Clock : AutoDrum slaves to the MIDI timing clock (0xF8). MIDI timing clock is sent 24 times per quarter note. The tempo will be changed by the interval of MIDI timing clock. This is the best mode for TPQN base MIDI data.

b) SMPTE/MTC : AutoDrum slaves to the MIDI time code quarter frame (0xF1). MIDI Time code is sent as a format including hour, minute, second, or frame, and generally it is sent 24 ~ 30 times per 1 second. The time will progress by the given hour : minute : second : frame.

MIDI Sync Mode (Send)

Select which MIDI sync signal to send to MIDI Out Device.

a) MIDI Timing Clock : AutoDrum sends MIDI timing clock (0xF8) 24 times per quarter note. If you change the tempo, the interval of MIDI timing clock will be changed. This is best mode for TPQN base MIDI data.

b) SMPTE24/MTC : AutoDrum sends MIDI time code quarter frame (0xF1) which contains hour, minute, second, or frame 24 times per 1 sec.

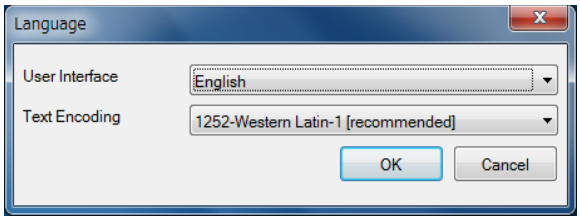
c) SMPTE25/MTC : AutoDrum sends MIDI time code quarter frame (0xF1) which contains hour, minute, second, or frame 25 times per 1 sec.

d) SMPTE29.97/MTC : AutoDrum sends MIDI time code quarter frame (0xF1) which contains hour, minute, second, or frame 29.97 times per 1 sec.

e) SMPTE30/MTC : AutoDrum sends MIDI time code quarter frame (0xF1) which contains hour, minute, second, or frame 30 times per 1 sec.

2-5. “Language” Dialog

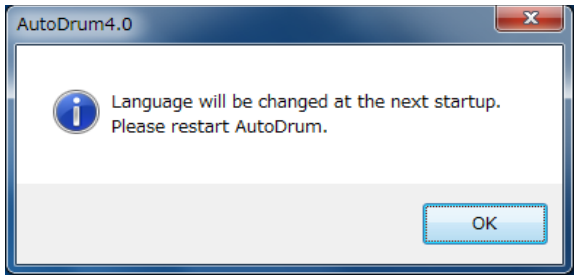
This dialog is used to select user inteference's language. This dialog can be opened from “Language...” button.



The language can be selected from Japanese or English or CHinese. In Japanese, MS P Gothic font is used as main GUI font. In English, MS Sans Serif font is used as main GUI font. (Except the part defined by OS, like title bar, menu, controls, and so on).

OK

Close this dialog and update language setup. If you click OK, following message box is shown.



Language will be changed at the next start up. Please execute AutoDrum again.

Cancel

Close this dialog.

Hint:

AutoDrum is made in Japan, therefore, default GUI language is Japanese. If you use non Japanese Windows, please select English language.

In non Japanese Windows, it may be difficult to open the “Setup” - “Language...” menu because of character corruption. Please use keyboard short cut, press [Alt] + [L] and you can open this dialog.

The GUI language can be changed by text editor, too. Open “AutoDrum.ini” in your text editor, and change change "Language=Japanese" into "Language=English", and execute AutoDrum.

3. Create user defined data

In AutoDrum, you can add user defined patch MIDI data and user defined rhythm pattern MIDI data. In AutoDrum4.2, patch MIDI data and rhythm pattern MIDI data are handled separately because to use one rhythm pattern in various tone.

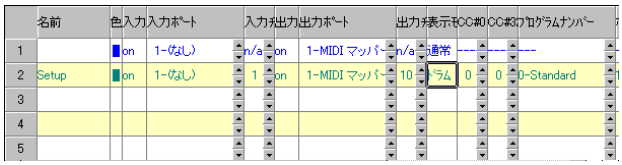
Patch data (*.mid)	Rhythm pattern data (*.mid)
Define the tone. It contains program change, control change, and pitch-bend event. It does not contain any note on note off event. These data is put in the patch folder.	Define the rhythm pattern .It contains only note on or note off event. And it is played as looped automatically in the AutoDrum. These data is put in the pattern folder.

Each data is standard MIDI file (*.mid), so you can create user defined data and add to tha AutoDrum. When restart AutoDrum, your data will be appear in the list box by saving patch or pattern folder.

Here shows an example how to create user defined Patch data or Rhythm pattern data by using Sekaiju midi sequencer.

3-1. create user defined patch data

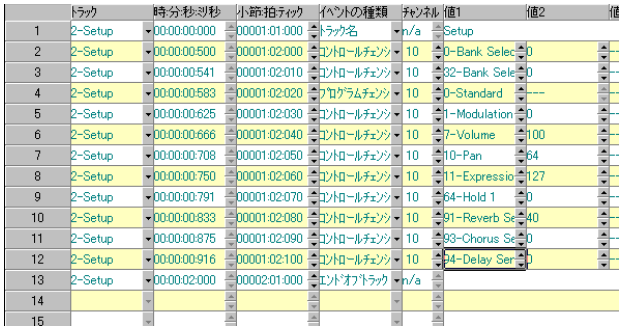
(1) By using track list window, put the first track (conductor track) and the second track (drum track). The second track's output channel must be 10.



* In the standard MIDI file format 1, the first track is conductor track, which can include only Track name, Tempo, Time signature, Key signature, and Marker event. MIDI channel event like control change or program change must be put in the second or following track. Other unnecessary track must be removed.

* Tempo event is ignored in AutoDrum.

(2) By using event list window, edit set up parameter like control change or program change in the second track. Each event's output channel must be 10.



* Don't put note on or note off event.

(3) Save as standard MIDI file (*.mid) and move it into the patch folder.

* It is recommended to make as SMF format 1.

* Don't put system exclusive data in the MIDI data.

3-2. Create user defined rhythm pattern data

(1) By using track list window, put the first track (conductor track) and some tracks for rhythm pattern.

	名前	色	入力	入力ポート	入力ch	出力	出力ポート	出力ch	表示モ	CC#0	CC#32	プログラムナンバー
1			on	1-(なし)	n/a	on	1-MIDI マッパー	n/a	通常	----	----	----
2	Kick		on	1-(なし)	1	on	1-MIDI マッパー	10	ドラム	----	----	----
3	Snare		on	1-(なし)	2	on	1-MIDI マッパー	10	ドラム	----	----	----
4	Drum		on	1-(なし)	3	on	1-MIDI マッパー	10	ドラム	----	----	----
5												

* In the standard MIDI file format 1, the first track is conductor track, which can include only Track name, Tempo, Time signature, Key signature, and Marker event. MIDI channel event like note on or note off must be put in the second or following track. Other unnecessary track must be removed.

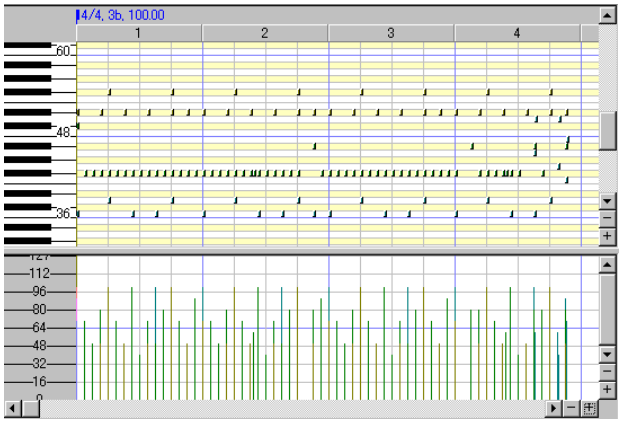
* Tempo event is ignored in AutoDrum.

* If need, you can put multiple tracks for rhythm pattern. But each track's output channel must be set to 10.

* Track name is free.

* Each track's set up event like control change, program change or pitch bend must be removed. Default set up events must be removed.

(2) By using piano roll window, insert note event in the second or following track.



* In AutoDrum's rhythm pattern, don't space the first measure (bar). Put note event from 1:1:000.

* The average note velocity must be about 64.

* The standard note duration must be about semiquaver or demisemiquaver in the drum track except long sound like snare roll.

* Each event's output channel must be set to 10.

(3) By using event list window, remove unnecessary event like control change or program change.

	トラック	時:分:秒:ミリ秒	小節:拍:ティック	イベントの種類	チャンネル	値1	値2
237	2-Drums	00:00:09:225	00004:04:045	ノートオン	10	43-Low Tom	0
238	2-Drums	00:00:09:250	00004:04:050	ノートオン	10	50-High Tom	0
239	2-Drums	00:00:09:300	00004:04:060	ノートオン	10	41-Low Tom	90
240	3-HiHat	00:00:09:300	00004:04:060	ノートオン	10	46-Open Hi-H	80
241	4-Perc	00:00:09:300	00004:04:060	ノートオン	10	51-Ride Cym	60
242	2-Drums	00:00:09:325	00004:04:065	ノートオン	10	47-Mid Tom	70
243	2-Drums	00:00:09:375	00004:04:075	ノートオン	10	41-Low Tom	0
244	3-HiHat	00:00:09:375	00004:04:075	ノートオン	10	46-Open Hi-H	0
245	4-Perc	00:00:09:375	00004:04:075	ノートオン	10	51-Ride Cym	0
246	2-Drums	00:00:09:400	00004:04:080	ノートオン	10	47-Mid Tom	0
247	2-Drums	00:00:09:600	00005:01:000	テキスト	n/a	End	
248	2-Drums	00:00:09:600	00005:01:000	エンドオブトラック	n/a		
249	3-HiHat	00:00:09:600	00005:01:000	テキスト	n/a	End	
250	3-HiHat	00:00:09:600	00005:01:000	エンドオブトラック	n/a		
251	4-Perc	00:00:09:600	00005:01:000	テキスト	n/a	End	
252	4-Perc	00:00:09:600	00005:01:000	エンドオブトラック	n/a		
253							

* In the second or following track for rhythm pattern, don't put non note event.

* Each note event's output channel must be set to 10.

(4) By using event list window, adjust each track's End of Track event's time.

(5) Save as a standard MIDI file (*.mid) and move it into pattern folder.

* It is recommended to make as SMF format 1.

* Don't put system exclusive data in the MIDI data.

4. Trouble Shooting

4-1. Error Message

MIDI In Device open failed.

AutoDrum can't open specified MIDI in device. If this error caused, “(None)” is selected automatically. Until AutoDrum can open some MIDI In Device, ”Slave”, “Sync Start”, “Sync Stop” can't be used. Check following list.

Reason	Solution
* Other application is using specified MIDI in device. *1	* End other application using specified MIDI In device.
* An application exited without closing specified MIDI in device.	* Reboot Windows.
* MIDI in device driver is broken.	* Uninstall the MIDI in device driver and then install it again.
* Nothing is connected to MIDI in terminal.	* Check cable connection.
* Your Windows doesn't have the device.	* Select the other device.

*1 : Generally, one MIDI in device can be opened from only one application, except multiple client MIDI interface.

MIDI Out Device open failed.

AutoDrum can't open specified MIDI out device. If this error caused, “(None)” is selected automatically. Until AutoDrum can open some MIDI out device, no sounds is played. Check following list.

Reason	Solution
* Other application is using specified MIDI out device.	* End other application using specified MIDI out device.
* An application exited without closing specified MIDI out device.	* Reboot Windows.
* MIDI out device driver is broken.	* Uninstall the MIDI out device driver and then install it again.
* Nothing is connected to MIDI out terminal.	* Check cable connection.
* Your Windows doesn't have the device.	* Select the other device.

*1 : Generally, one MIDI Out Device can be opened from only one application, except multiple client MIDI interface.

Insufficient memory, Insufficient resource.

Reason	Solution
* Too fast or slow tempo is specified. * Abnormal time base or time resolution is specified.	* Change the tempo value in the MIDI data. * Set time mode TPQN base and time resolution 120 or 480 in the MIDI data.
* Insufficient memory. * Insufficient resource.	* End other application. * Reboot Windows.

MIDI data open failed

AutoDrum can't open specified MIDI data. Check following list.

Reason	Solution
* Abnormal MIDI data.	* If it is saved in other sequencer, once open by the sequencer, fix bug, and then save again. Try saving in various file type, or SMF format 0 / 1, or the other time mode and time resolution.
* The disk storing the MIDI data is broken.	* Scan disk and repair disk. * Copy the file to other device.
* Specified MIDI data is used by other application.	* Close other applications..
* The MIDI data is removed or moved or renamed. *2	* Restart AutoDrum and refresh the list.
* Insufficient memory. * Insufficient resource.	* End other application. * Reboot Windows.

*1 : For detail of user defined MIDI data, see also 3. Create user defined data.

*2 : Patch MIDI data must be found in the patch folder. Pattern MIDI data must be found in the pattern folder.

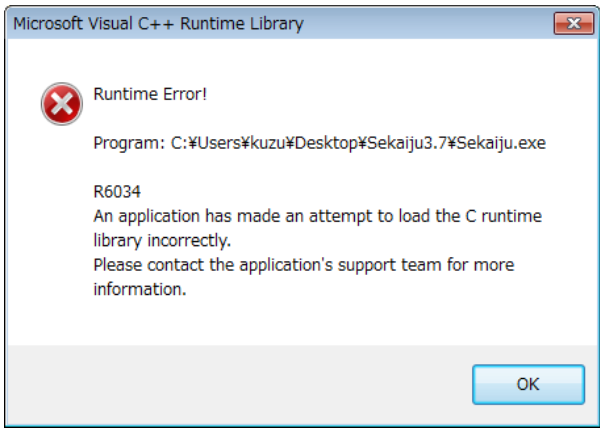
DLL File is not found.

AutoDrum.exe – System Error
***.DLL File is not found.**

AutoDrum.exe
Wrong Side by Side

Microsoft VisualC++ Runtime Library
Runtime Error!

AutoDrum
***.DLL Load failed!!**



This message may be shown when executing AutoDrum. To execute AutoDrum, Following DLL files are required.

AutoDrumJpn.dll	AutoDrum Japanese language resource DLL.
AutoDrumEnu.dll	AutoDrum English language resource DLL.
AutoDrumChs.dll	AutoDrum Chinese language resource DLL.
MIDIIO.dll	MIDI message input output library.
MIDIClock.dll	MIDI clock measureing library.
MIDIData.dll	MIDI data creating editing library.
MIDIStatus.dll	MIDI module status keeping library.

These DLLs are shipped with AutoDrum, and they must exist in the same folder as AutoDrum.exe folder. Please check following list.

Reason	Solution
* You forget to extract AutoDrum4.2.zip.	* Extract AutoDrum4.2.zip and then execute.
* DLL file is broken. * DLL's version is wrong.	* Download AutoDrum again.
* Manifest file is broken. * Manifest's versin is wrong.	* Download AutoDrum again.

4-2. If no sound is played

If no sound is played, check following list.

Reason	Solution
Amplifier's volume is 0 or amplifier has some trouble.	Set volume up or use a headphone.
Loudspeaker's cable has some trouble.	Check cable connection. Change cable.
MIDI cable or USB cable has some trouble.	Check cable connection. Change cable.
MIDI module's master volume is 0.	Send GM system off or GM Reset or GM2 reset or GS reset or XG reset.
MIDI module's channel volume level or expression level is 0.	Send GM system off or GM Reset or GM2 reset orGS reset or XG reset.
In case using software module, volume control is wrong.	Uncheck mute button and set volume higher in windows volume control.
“ (None)” is selected for MIDI out device.	In the MIDI device dialog, Select some MIDI out device.
Slave mode is selected but no sync message is detected from MIDI input port, so clock does not advance.	In the MIDI sync dialog, select MIDI In Sync mode (“Receive MIDI Clock” or “Receive SMPTE/MTC”), and send specified sync message to the input port.
There is no Note on event in the MIDI data, or too low note on velocity.	Write Note on event to the MIDI data and specify higher note on velocity.
You specified unavailable tone number.	Correct CC#0, CC#32, and program change value which is available in your MIDI module.
Lower volume level is specified by CC#7.	Specify higher volume level by CC#7.
Lower expression level is specified by CC#11.	Specify higher expression level by CC#11.
Patch MIDI data or Pattern MIDI data is wrong. (Wrong CC#0, CC#32, program change in patch data, or wrong note in pattern data).	Check patch data or pattern data's control change, program change, note event and so on. Each event's channel must be 10.
AutoDrum has crashed or caused some internal error.	Exit AutoDrum, and execute Sekaiju again.
Windows has crashed or caused some internal error.	Exit Windows, and reboot Windows.
MIDI module has crashed or caused some internal error..	Turn off the MIDI module, and then turn on MIDI module again.

5. MIDI Implementation

Model: AutoDrum4.2
Date: 2016/06/26

5-1. Receive data

Channel Voice Message

These message are recorded specified track whose input channel is the same as at real time recording.

*** Note Off**

Status	Second byte	Third byte
8nH	kkH	vvH
9nH	kkH	00H

n = MIDI Channel number : 0H-FH (ch.1~ch16)
kk = Note number : 00H-7FH (0~127)
vv = Note off velocity 01H-7FH (0 ~ 127)

* If sync stop is checked on, if all note is off and hold pedal is not downed, AutoDrum stop playing automatically.

*** Note On**

Status	Second byte	Third byte
9nH	k kH	vvH

n = MIDI Channel number : 0H-FH (ch.1~ch16)
kk = Note number : 00H-7FH (0~127)
vv= Note on velocity : 01H-7FH (1~127)

* If sync start is checked on, if a note is on or hold pedal is downed, AutoDrum start playing automatically.

*** Control Change**
Hold Pedal

Status	Second byte	Third byte
BnH	40H	vvH

n = MIDI Channel number : 0H-FH (ch.1~ch16)
vv= Control value : 00H-7FH (0~127)

* If sync start is checked on, if hold pedal is downed, AutoDrum start playing automatically.

* If sync stop is checked on, if hold pedal is upped and any key is not pressed, AutoDrum stop playing automatically.

System Real Time Message

* MIDI Timing Clock
Status
F8H

This message is sent 24 times per quarter note. In slave mode, if "Receive MIDI timing clock" is selected, Autodrum slaves to this message.

*** Start**
Status
FAH

Whenever receiving this message, AutoDrum start playing from the beginning of the MIDI data. If while playing, this message is ignored.

*** Continue**
Status
FBH

Whenever receiving this message, AutoDrum start playing from current playing position. If while playing, this message is ignored.

*** Stop**
Status
FCH

Whenever receiving this message, AutoDrum stop playing. If while not playing, this message is ignored.

5-2. Send data

Channel Voice Message

These message are recorded specified track whose input channel is the same as at real time recording.

*** Note Off**

Status	Second byte	Third byte
8nH	kkH	vvH
9nH	kkH	00H

n = MIDI Channel number : 0H-FH (ch.1~ch16)
kk = Note number : 00H-7FH (0~127)
vv = Note off velocity 01H-7FH (0 ~ 127)

*** Note On**

Status	Second byte	Third byte
9nH	k kH	vvH

n = MIDI Channel number : 0H-FH (ch.1~ch16)
kk = Note number : 00H-7FH (0~127)
vv= Note on velocity : 01H-7FH (1~127)

*** Key After Touch**

Status	Second byte	Third byte
AnH	kkH	vvH

n = MIDI Channel number : 0H-FH (ch.1~ch16)
kk = Note number : 00H-7FH (0~127)
vv= Key after touch : 00H-7FH (0~127)

*** Control Change**

Status	Second byte	Third byte
BnH	ccH	vvH

n = MIDI Channel number : 0H-FH (ch.1~ch16)
cc = Control change number : 00H-7FH (0~127)
vv= Control value : 00H-7FH (0~127)

*** Program Change**

Status	Second byte
CnH	ppH

n = MIDI Channel number : 0H-FH (ch.1~ch16)
pp = Program number : 00H-7FH (0~127)

*** Channel After Touch**

Status	Second byte
DnH	ppH

n = MIDI Channel number : 0H-FH (ch.1~ch16)
vv= Channel after touch : 00H-7FH (0~127)

*** Pitch Bend Change**

Status	Second byte	Third byte
EnH	llH	mmH

n = MIDI Channel number : 0H-FH (ch.1～ch16)
mm, ll = Pitch bend value : 00 00H - 40 00H - 7F 7FH (-8192~0~+8191)

System Exclusive Message

Status	Second byte	Last byte
F0H	vvH	F7H

vv = arbitrary value : 00H-7FH (0~127)

System Common Message

These message is not recorded to the MIDI data.

*** MIDI Time Code Quarter Frame**

Status	Second byte
F1H	tvH

t = Frame type : 0H-7H (0~7)
v = 4 bit value : 0H-FH (0~127)

t	v
0	Frame number (00~29) lower 4 bit
1	Frame number (00~29) higher 4 bit
2	Second (00～59) lower 4 bit
3	Second (00～59) higher 4 bit
4	Minute (00～59) lower 4 bit
5	Minute (00～59) higher 4 bit
6	Hour (00～23) lower 4 bit
7	Constant 0 1bit, Frame rate (00~03) 2bit, Hour (00~23) higher 1bit

MIDI Time Code Quarter Frame is sent to notify current playing position at every frame while playing if “Send SMPTE/MTC” is selected.

*** Song Position Selector**

Status	Second byte	Third byte
F2H	llH	mmH

mm, ll = Position value : 00 00H - 40 00H - 7F 7FH(0～16383)

Song Position Selector let move current playing position as semiquaver note = 1 unit. This message is sent when current playing position is changed if “Send MIDI Timing Clock” is selected

System Real Time Message

*** MIDI Timing Clock**

Status
F8H

This message is sent 24 times per quarter note. This message is sent while playing if “Send MIDI Timing Clock” is selected .

*** Start**

Status
FAH

This message let start playing from beginning of the MIDI data. This message is sent when you start playing from the beginning of the MIDI data if “Send MIDI Timing Clock” or “Send SMPTE/MTC” is selected .

*** Continue**

Status
FBH

This message let start playing from current playing position. This message is sent when you start playing from the middle of the MIDI data if “Send MIDI Timing Clock” or “Send SMPTE/MTC” is selected .

*** Stop**

Status
FCH

This message let stop playing. Current playing position is not modified. This message is sent when you stop playing if “Send MIDI Timing Clock” or “Send SMPTE/MTC” is selected.

5-3. MIDI Implementation Chart

Function		Send	Receive	Notice
Note	Key number	O	O	
	velocity	O	X	
Key after touch		O	X	
Channel after touch		O	X	
Control change	64	O	O	Hold Pedal
	others	O	X	
Program change		O	X	
System exclusive		O	O	
System common	MIDI Time code quarter frame (0xF1)	O	O	*2
	Song position (0xF2)	O	O	*1
	Song select(0xF3)	X	X	
	Tune request (0xF6)	X	X	
System real time	MIDI Timing clock (0xF8)	O	O	*1
	Start (0xFA)	O	O	*3
	Continue (0xFB)	O	O	*3
	Stop (0xFC)	O	O	*3
	Active sensing (0xFE)	X	X	
	System reset (0xFF)	X	X	

*1 : Send is available if “Send MIDI Timing Clock” is selected.
*1 : Receive is available if “Receive MIDI Timing Clock” is selected.
*2 : Send is available if “Send SMPTE/MTC” is selected.
*2 : Receive is available if “Receive SMPTE/MTC” is selected.
*3 : Send is available if “Send MIDI Timing Clock” or “Send SMPTE/MTC” is selected.

6. Specification

AutoDrum4.2 : automatic drum performance software.

6-1. Specification of software.

Programming language and Compiler	C/C++ language / Win32API / MFC / Microsoft Visual C++ 2008 Standard Edition SP1
Threads	Multiple thread (Main thread, patch playing thread, pattern playing thread, recording thread.)
MIDI Device I/O capacity	Input : 1 port (WMME) Output : 1 port (WMME) Thru : on / off
File I/O capacity	Standard MIDI file (*mid) load
Clock method	Master : Internal clock (Windows multimedia timer) Slave ; MIDI Timing Clock or SMPTE/MTC (MIDI time code quarter frame)
Tempo	16～255bpm、1bpm step

6-2. Required Environment

OS	WindowsXP/Vista/7/8.1/10
CPU	Core2Duo or higher
Memory	2GB
Hard disk	Empty of 10MB or larger
Monitor	640 x 480 pixel or larger and 16 colors or more monitor
MIDI module or Synthesizer	External MIDI module or Synthesizer is required. GM / GM2 / GS / XG module is better. Internal module (like “Microsoft GS Wavetable Synth”) or software module (like VSC-88) is available. VSTi is NOT available.
MIDI controller or Keyboard	Optional. Those with MMC/MTC send function is better.

6-3. Required Dynamic Link Library (*.dll)

Made by Open MIDI Project (shipped with AutoDrum)

AutoDrumJpn.dll	AutoDrum English language resource DLL.
AutoDrumEnu.dll	AutoDrum Japanese language resource DLL.
MIDIIO.dll	A library for MIDI input or output.
MIDIClock.dll	A library for measuring timing.
MIDIData.dll	A library for creating or editing MIDI data.
MIDIStatus.dll	A library for keeping MIDI module's status in real time.