==
Voyetra Sequencer Esoteric INI Options (EIO)
Last Modified: May 3, 1996

This file explains how you can edit the text in the initialization (INI) files. By editing the INI file you can customize the way the applications work and look. The procedures described are optional. For normal operation it isn't necessary to edit the INI files.

The applications and the INI files that this text refers to are:
MIDI Orchestrator which uses ORCH.INI.
MIDI Orchestrator Plus which uses ORCHPLUS.INI, SYSEX.INI and VSEQPTCH.INI.
Digital Orchestrator Plus which uses DOP.INI, SYSEX.INI and VSEQPTCH.INI.

If a setting is used only in MIDI Orchestrator Plus it will be marked as (+).
If a setting is used only in Digital Orchestrator Plus it will be marked as (DA).
If the settings can be changed in MIDI Orchestrator, MIDI Orchestrator Plus, and Digital Orchestrator Plus, there will be no markings.

The INI file stores a number of user settings and preferences, in a manner similar to most Windows programs. VSEQPTCH.INI stores the names of patches and controllers that are displayed in various windows and dialog boxes. (+,DA Only)

All of these files except VSEQPTCH.INI and SYSEX.INI will be found in the Windows directory.

General Notes and Precautions
The techniques described, for editing INI files, are recommended for advanced users only. Changing settings in an improper manner may cause the application to run incorrectly.
Make backup copies before editing these files.

The settings discussed can be added or changed only by editing the INI file with a text editor, such as Windows' Notepad or DOS Edit. Other settings in the INI file should only be changed from within the program.

Recovering from errors
If you make a mistake editing the INI file, delete the entire line containing the error. The deleted setting will be restored to its default value. You can also delete the INI file entirely. The application will automatically create a new INI file the next time it is run. All preferences and settings will be restored to default values.

The [Options] section affects the way this application handles MIDI data

```
[Options]
MaxTracks=nnn
WriteMutedTracks=x
ppq=nnn
undosize=nnn              (+, DA)
MIDIOutBackup=nnn
DragScrollRate=nnn        (+, DA)
UseMidiTimeStamps=x
InstrumentDir=            (+, DA)
DefPatchMap=              (+, DA)
DefDrumMap=               (+, DA)
DefDrumChannel=           (+, DA)
```

MaxTracks=nnn

nnn can be an integer from 1 to 16000
Default is 1000
Sets the maximum number of tracks you can edit.
To save memory, choose the smallest number that meets your needs.
Be sure to allow a few extra tracks for copying and pasting.

WriteMutedTracks=x
  x may be 1 or 0
  Default is 0
  If x = 1, files saved in MIDI (.MID) format will include tracks that are
  temporarily silenced by the Mute and/or Solo buttons.
  This guards against accidental loss of data when saving files in MIDI format.
  However, the file may not play back exactly as it was when you last listened to it.
  Files saved in this application's .ORC file format always include all tracks,
  including those that are temporarily silenced by the Mute and/or Solo buttons.

ppq=nnn
  Default is 480.
  Sets value for PPQ (parts per quarter note).
  this application subdivides each quarter note into nnn parts.
  If you open a file that has a different PPQ value, this application will
  change the file's PPQ setting to the one shown here.
  There is rarely any reason to change this value.

undosize=nnn
  nnn may range from 1 - 4000
  Default is 12.
  Sets the maximum number of operations you can reverse with the undo command.
  For example, if nnn is set to 10, your ten most recent operations will be temporarily saved
  and each Undo command cancels one of them in turn.
  When you save a file, load a new file, or exit this application, the undo buffer is cleared.
  To save memory, choose the smallest number you can work comfortably with.

SynchPlaybackRate=nnn
  nnn may range from 0 to 100.
  Default is 4.
  The may be used to adjust the amount of time spent updating windows during playback.
  The higher the number, the less time will be spent updating inactive views, leaving
  more time for actual playback and updating the active view.
  If 0, all windows will be updated at equal time intervals.

MIDIOutBackup=nnn
  nnn may range from 0 to 1024
  Default is 100.
  This controls the amount of internal buffering used by the application on playback.
  If the MIDI Out LED ever turns red, you should increase this value.
  Setting this value to zero will cause all buffering to be left up to the individual driver.
  NOTE: If you are using the MIDI Mapper as your output port, this option may not be effective.

DragScrollRate=nnn
  nnn may range from 100 to 2000
  Default is 250.
  This controls the speed of the scrolling while dragging objects or selecting a range. It is the
  number of milliseconds to wait before scrolling another measure.

UseMIDITimeStamps=x

x may be 0 or 1
Default is 1
This controls whether the application uses Driver Time Stamps while recording MIDI or uses its own internal counter.
There is rarely any reason to change this value.

InstrumentDir=
The directory where VSEQPTCH.INI and other PATCHMAP.INI files may be found.
Default is in the application directory.

DefPatchMap=
The name of the patch map (as found in the top of VSEQPTCH.INI) which should be used as the default. Note that if you have previously set patch maps for a particular driver using the Patch Map Setup dialog, this setting will have no effect. You can clear existing patch settings by deleting all the lines in the [PatchSetup] section of the INI file.

DefDrumMap=
The name of the patch map (as found in the top of VSEQPTCH.INI) which should be used as the default on the drum channel. Note that if you have previously set patch maps for a particular driver using the Patch Map Setup dialog, this setting will have no effect. You can clear existing patch settings by deleting all the lines in the [PatchSetup] section of the INI file.

DefDrumChannel=
Which channel should be used as the drum channel when deciding whether to use the DefDrumMap or the DefPatchMap. Note that if you have previously set patch maps for a particular driver using the Patch Map Setup dialog, this setting will have no effect. You can clear existing patch settings by deleting all the lines in the [PatchSetup] section of the INI file.

The [View] section can change how the Bar View looks

```
[View]
BarWidth=nnn                    (+, DA)
xinset=nnn                      (+, DA)
yinset=nnn                      (+, DA)
BarSkip=nnn                     (+, DA)
hashskip=nnn                    (+, DA)
Color0=rrr ggg bbb              (+, DA)
Color1=rrr ggg bbb              (+, DA)
Color2=rrr ggg bbb              (+, DA)
Color3=rrr ggg bbb              (+, DA)
Color4=rrr ggg bbb              (+, DA)
Color5=rrr ggg bbb              (+, DA)
Color6=rrr ggg bbb              (+, DA)
Color7=rrr ggg bbb              (+, DA)
Color8=rrr ggg bbb              (+, DA)
Color9=rrr ggg bbb              (+, DA)
NumColors=nnn                   (+, DA)
DenSen=nnn                      (+, DA)
Mono=x                          (+, DA)
```

BarWidth=nnn
Default is 10.
Sets the width in pixels of one bar in the Track/View window.
nnn must equal at least twice xinset + 1.

xinset=nnn
　Default is 2.
　Sets the horizontal margin for each MIDI data block
　(the rectangle that represents a measure of MIDI data) within its bar.
　If you set nnn to 0, each block fills the entire width of the bar,
　and all the blocks meet along their left and right edges.

yinset=nnn
　Default is 2.
　Sets the vertical margin for each data block within its bar.
　If you set nnn to 0, each block fills the entire height of the bar,
　and all the blocks meet along their top and bottom edges.

BarSkip=nnn
　Default is 8.
　Sets the interval between numbered bars in the bar ruler.
　A value of 8 means every eighth bar is labeled with a number.
　Must be greater than or equal to hashskip.

hashskip=nnn
　Default is 2.
　Sets the interval in bars between vertical hash lines in the bar ruler.

Color0=rrr ggg bbb
　to
Color9=rrr ggg bbb
　Sets the colors of the data blocks used to represent each degree MIDI density.
　The letters "rrr," "ggg" and "bbb" represent values from 0 to 255 for red, green and blue.
　Normally, colors range from Color0 =255 255 255 (white) to Color9=0 0 0 (black).

NumColors=nnn
　Value may range from 2 - 10.
　Default is 10.
　Sets the number of different shades or colors available to draw data blocks.
　Color0 is always the first color used.

DenSen=nnn
　Default is 1.
　Sets the number of MIDI events which trigger a jump to the next color.

Mono=x
　x may be 1 or 0
　Default is 0
　If x=1, this application uses monochrome bitmaps to simulate colors of data blocks.
　This option is useful for monochrome systems, such as portables and laptops.

The [Thru] section controls the operation of MIDI Thru

```
[Thru]
DefPort=nnn                 (+, DA)
DefChan=nnn
DefFollowTrack=nnn
ThruPoolSize=nnn
```

DefPort=nnn
    Possible values depend on MIDI interface hardware.
    Default is 0.
    Sets the port to which MIDI Thru sends data on startup.

DefChan=nnn
    May range from 0 - 15.
    Default is 1
    Sets Channel to which MIDI Thru assigns data on startup.
    MIDI data is thru'ed on the same channel it was received on.

DefFollowTrack=nnn
    Default is 0 (Track # 1).
    Selects which track's settings will be used for MIDI for Thru,
    until a track is chosen for recording.
    NOTE: Once a track is selected for recording, Thru will use that track until
    you select another track or exit the program.

ThruPoolSize=nnn
    Default is 127
    Sets the maximum number of notes that this application can receive
    simultaneously as MIDI Thru data.

The [Transcription] section controls how musical notation is transcribed and printed

```
[Transcription]
MaxChordRange=nnn          (+, DA)
SuppressRestsRatio=nnn     (+, DA)
DrumChannel=nnn            (+, DA)
```

MaxChordRange=nnn
    nnn may range from 3 - 127
    Default is 14 (Major 9th).
    Sets the maximum pitch range which will produce a single voice within a track.
    This option is useful when you want to render certain notes as part of a different chord,
    for example, to render left and right hand parts separately in a piano transcription.

SuppressRestsRatio=nnn
    nnn may range from 0 - 100
    Default value is 70.
    Sets the "Normal" ratio used in the Suppress Rests option.
    The Notation and Piano Roll windows maintain rest suppression settings independently.

DrumChannel=x
    Notation uses this in order to process the auto clef for drums. The default is 10. Valid values
    are 0-16 where 0 turns this feature off and 1-16 specify the MIDI channel which contains
    drums.

The [PianoRoll] section controls how MIDI data is transcribed in the Piano Roll window

```
[PianoRoll]
```

```
SuppressRestsRatio=nnn
```
   nnn may range from 0 - 100
   Default value is 70.
   Sets the "Normal" ratio used in the Suppress Rests option.
   The Notation and Piano Roll windows maintain rest suppression settings independently.

The [DigitalAudio] section controls some digital audio playback and drawing options

```
[DigitalAudio]

DrawDataBinSize=nnn        (DA)
11025=nnn                  (DA)
22050=nnn                  (DA)
44100=nnn                  (DA)
WaveSyncOutOnly=x          (DA)
WaveSyncToLast=x           (DA)
WaveSyncToIgnoreDupes=x    (DA)
WaveSyncMsPosOffset=nnn    (DA)
SilenceColor=rrr ggg bbb   (DA)
WaveColor=rrr ggg bbb      (DA)
ForceBufferSize=           (DA)
```


DrawDataBinSize=nnn
   nnn may range from 0 - 10000
   Default Value is 100.
   DOP stores information about the Digital Audio data in order to refresh the screen more
   quickly. The DrawDataBinSize describes how many samples are kept in each bin.
   For most accurate drawing and least amount of memory usage set this field to 0 and
   the data will be read from disk each time. The optimum number would be equal to the
   approximate number of samples drawn per pixel.

11025=nnn
22050=nnn
44100=nnn
   nnn is the actual rate that your card plays and records 11K, 22K, 44K files.
   This is useful if you attempt to start from the middle of a song and digital audio tracks appear
   to start from the wrong location.   You can use SoundCheck to calculate this value
   Defaults are 11025, 22050, and 44100.

WaveSyncOutOnly=x
   x may be 0 or 1 - default is 0
   Note: This option requires the WaveSync option to be enabled in the Digital Audio Options
   Dialog.
   If x = 1, this will automatically turn off WaveSync if there is no Digital Audio being played
   back.   You may want to add this option if MIDI playback seems erratic while recording Digital
   Audio with WaveSync enabled (but is fine while only playing Digital Audio)

WaveSyncToLast=x
   x may be 0 or 1 - default is 0
   Note: This option requires the WaveSync option to be enabled in the Digital Audio Options
   Dialog.
   This option has no effect unless "Device can play and record simultaneously" is enabled in

the Digital Audio Options Dialog.

If x = 1, this option forces WaveSync to use the wave input instead of wave output when using the FullDuplex feature of your sound card.

You may want to add this option if (with WaveSync enabled) MIDI playback is erratic while recording AND playing DigitalAudio at the same time.

WaveSyncIgnoreDupes=x

x may be 0 or 1 - default is 0

Note: This option requires the WaveSync option to be enabled in the Digital Audio Options Dialog.

If x = 1, this option will put DOP into a "compatibility mode" with some incorrect wave drivers.

You may want to add this option if playback is erratic with WaveSync Enabled.

WaveSyncMsPosOffset=nnn

nnn may range from -1000 - 1000 - default is 0

Note: This option requires the WaveSync option to be enabled in the Digital Audio Options Dialog.

This options specifies the number of milliseconds by which MIDI playback is ahead of Audio playback.

SilenceColor=rrr ggg bbb

Defaults to 255 0 0 (red)

WaveColor=rrr ggg bbb

Defaults to using the "Window Text" color set in control panel.

Sets the colors of the waveform.

The letters "rrr," "ggg" and "bbb" represent values from 0 to 255 for red, green and blue.

Normally, colors range from Color0 =255 255 255 (white) to Color9=0 0 0 (black).

ForceBufferSize=

Sets the size (in Kbytes) of the buffers used for Audio Playback and Recording. There is usually no reason to set this value.

The [xfDAScale] section controls the Digital Audio Crescendo Transform

```
[xfDAScale]
XFDAZero=1...32000 (DA)
```

This entry sets the Digital Audio Crescendo Transform residue value for 0% setting. The minimum value should be 1, which would result in a very steep decay curve. A high value would give rise to a decay curve that deviates from the exponential function and resembles more like a straight line. If this entry is set to 0, however, the default value, 100, will be used instead.

The [SysEx] section controls all the System Exclusive settings

```
[SysEx]
SxNumOfBuffers=nnn                              (+, DA)
SxBufferSize=nnnn                               (+, DA)
SxMaxBanks=nnn                                  (+, DA)
SxDelayMS=nnn                                   (+, DA)
SxTimeOutSec=n                                  (+, DA)
SysExIniDir=<drive><path><filename.ext>   (+, DA)
```

```
AutoSxTimeZero=x                                    (+, DA)
```

SxNumOfBuffers=nnn
    nnn is the number of buffers that is set when receiving SysEx messages
    Default is 128

SxBufferSize=nnnn
    nnnn is the number size of each buffer (in Bytes) that is set when receiving SysEx messages
    Default is 512

SxMaxBanks=nnnn
    nnnn is the maximum number of banks that you see in the SysEx window
    Default is 1000

SxDelayMS=nnnn
    Time between sending statements in milliseconds

SxTimeOutSec=n
    n is the number of seconds to keep trying if there was a driver problem.
    Default is 5 seconds

SysExIniDir=
    This is the path and file name of the file that contains the Macros
    for communicating with the Synthesizers.
    Default is SYSEX.INI in the same directory as the executable.

AutoSXTimeZero=x
    Note: This options affects the reading of MIDI files only.
    if x is 1 then SysEx banks at time zero (bar 1, beat 1, click 0) will be marked as auto and sent
    accordingly. This is the default setting. Otherwise, (x = 0) the banks will be treated the same
    as other banks and sent during playback.


```
*********************************************************
```
Patch Name File   (+, DA)

The Patch Name file stores the text that appears in various windows
and dialog boxes to identify patches and controllers.
This file already contains listings for a number of popular synthesizers.
If patch names for your synthesizer are not listed, you can create your own
list of patch names for your synthesizer.
You can also customize the controller names.

Though the text in the patch name file affects what you see on the screen,
it does not affect the operation of the program. The actual patches
and controllers are determined by your hardware. The settings discussed
below can be changed only by editing the patch name file with a text editor.
VSEQPTCH.INI must either be in the same directory as the executable file or in the directory
referenced by InstrumentDir= (InstrumentDir= is under [Options] in VSEQPTCH.INI).

To add (or edit) a patch map:
First, create a name for your patch map and add it to the list of Patch Maps.
In the example below, we've created an entry for "My Patch Map" at position #2.

```
[PatchMaps]
General MIDI=gm.ini
```

```
0..127=
My Patch Map=
```

If there is nothing to the right of the equal sign, it is assumed that the patch map is in
VSEQPTCH.INI. Otherwise, it is assumed that a file is named which is in the same directory as
VSEQPTCH.INI. It is suggested that if you add a new patch map, add to a separate file.

Create a new section header for your patch map anywhere below the
[Patch Maps] section. Copy exactly the name you chose for your patch map
in the previous step, and enclose it in square brackets.
Follow this with your numbered list of patches, as shown below:

```
[My Patch Map]
0=My Piano
1=My Flute
2=My Violin
```

and so on, for up to 128 patches. Numbers to the left of the equal
sign may range from 0 - 127. You don't need to name all 128 patches.
If you choose not to name a patch you may omit its number. The program
will display an integer instead.

The patch map always uses 0 for the lowest patch number. If your synth
uses patch numbers beginning with 1, you can instruct this application
to add 1 to each patch number for display. Add a "base=1"
line to your patch map just below the bracketed patch map name, as
shown below:

```
[My Patch Map]
base=1
0=My Piano
1=My Flute
2=My Violin
```

and so on.

VSEQPTCH.INI also supports
data=nnn
where nnn may be 0,1,2, or 3 (default)
This tells the program what the data represents.
   0 - Bank MSB - Controller 0
   1 - Bank LSB - Controller 31
   2 - a Logical Grouping - no associated data
   3 - Patch Change

and
picture=nnn
where nnn may be 0-19
This associates an icon with an entry in the patch map.
Note: the lowest level always uses the Patch Cable Icon.
The use of these icons are demonstrated in GM.INI

Editing Controller Names:

Each controller name has a full version and an abbreviated version.

The full controller names appear in the Event Window and in some dialog boxes. These names appear in the [Controllers] section.

The abbreviated controller names are used in screen displays where space is at a premium, such as in the Mixer screen and in the header in the Track/View Window. These names appear in the [Ctrls] section. Simply add or edit the text in these two sections to match the controller names in your device.

The format is:
n=text
   n may be any number from 0 - 127
   The text may be anything you would like.
   Controllers names may appear in any order, or be omitted altogether.
   Try to limit the abbreviated versions to 6 or 7 characters.

****

SYSEX.INI   (+, DA)

This file has one section: [SysEx Macros]

The purpose of this file is to store the Dump Request Macros.
Each Macro must be on a single line with this format.

```
<Macro name>=F0 <data> F7 [n<number of buffers>] [s<buffer size>]
```

<Macro name> is the name that appears in the SysEx menu of Macro choices.
If the request macro requires the user input, place "..." after
the name. E.g., "Yamaha TX81Z Tune 442 ...".

<data> is any series of hexadecimal bytes.

[n<number of buffers>] sets the number of allocated buffers to receive the SysEx message.
[s<buffer size>] sets the buffer size that receive the SysEx message.
For example, appending n32 s2048 would allocate 32 buffers of 2KB each.
The default values are set by the SXNumOfBuffers and SXBufferSize defined in MOPDA.INI

NOTE: Do not put any spaces between the "n" and the number of buffers,
and between the "s" and the buffer size.

Note that normal MIDI data bytes must have the high bit clear: The value
must be less than 128d, 80h. Since values over 128 can never be part of a
System Exclusive message.

OPCODES: Special <data> values that are not sent:

FD, FE : Prompt user to "Input Patch Voice Channel number"
   and send what they input.
FC : Arithmetic-OR of next two bytes. The next two bytes--or results
    of other opcodes--are OR'd together and the result is sent.