Intro

In order to play music, sm64 uses a proprietary file format similar to midi called Music Macro Language, or commonly referred to as m64 (the filetype). In order to create music that is playable in game, we will need to create midi files, rearrange the songs to use the sm64 soundfont, and then convert the midi to sm64's file format (m64). This tutorial will go over all of that, and extra steps on how to make your music sound as good as possible.

Tools

In order to port any music over, we need a program that will allow us to edit midis, and a program to convert them to m64. Most people who simply edit existing midis will be fine with a program called Anvil Studio. This is a free program with an old school interface with a relatively good feature set. If you want to compose your own music, or really edit and remix songs, I recommend you pick something else like FL studio or a DAW you're familiar with.
In order to convert the midi files into m64 files, we use a tool called seq64. There are two commonly used versions of seq64. I recommend you download both, but mostly stick with the older version as a beginner because it has a more intuitive GUI.

- Version 1.5 Best for new users
- Latest release Best for experienced users or more involved porting

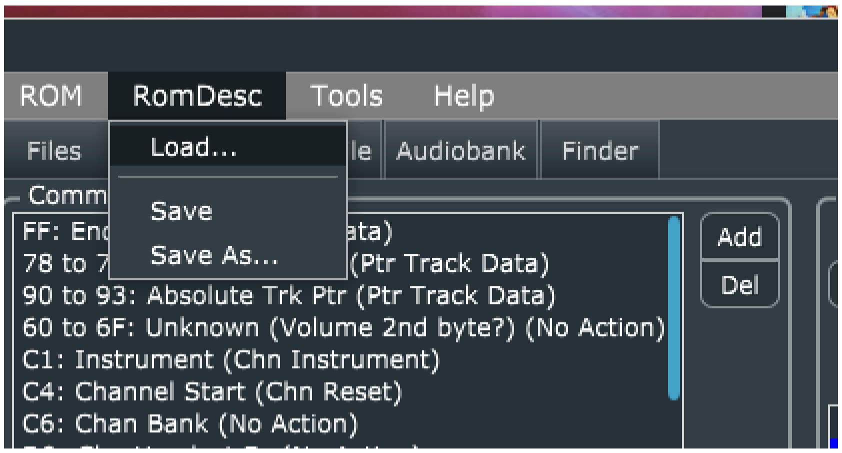
Midi and M64 Sourcing

If you are new, I recommend you start by using existing midis and looking at existing m64s as a reference if needed. You can find tons of midis online for most games over at vgmusic.com. You can navigate the games pages to look for things that have been reviewed and are known to be good, but to find anything recent, you will have to click the "Newly Submitted Files" link on the side and do some searches.
If you want to find some m64s to see what has been done already or need something extra for your hack, here is a list of places to find some:

- Smwcentral public music repository
- My personal m64 repository, has recordings and m64 files
- Sm64pie music ports
- Music ports by Helium
- Music ports by Prakxo

Creating a port

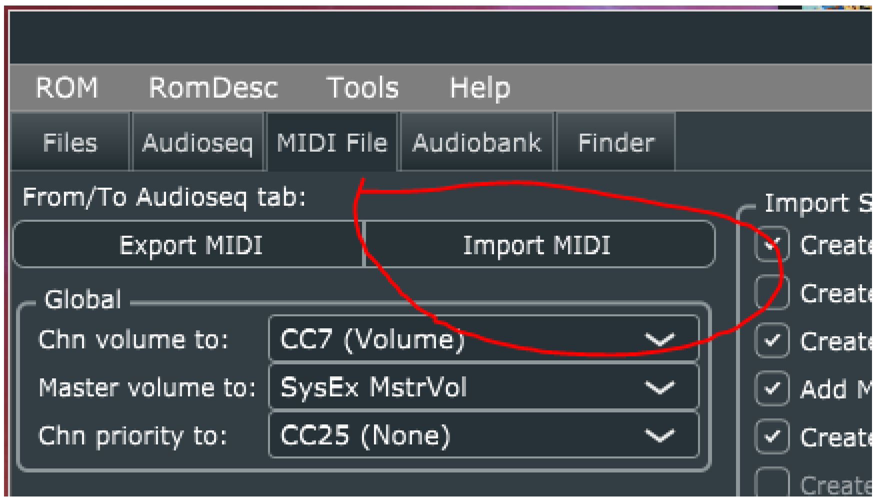
In order to make a port of a midi, you need to open up seq64 (I will use 1.6 for all examples in this tutorial), and load a RomDesc. Doing this will load the game specific information into the program, we want to choose sm64_info.xml.



Comm
FF: End ... ata)
78 to 7 ... (Ptr Track Data)
90 to 93: Absolute Trk Ptr (Ptr Track Data)
60 to 6F: Unknown (Volume 2nd byte?) (No Action)
C1: Instrument (Chn Instrument)
C4: Channel Start (Chn Reset)
C6: Chan Bank (No Action)

This file should be in the same location as your exe file for seq64, it comes with the program and is inside the zip folder.

Once you have done this, you want to load your midi into the program, you do this by going to the MIDI File tab and clicking Import MIDI.



From/To Audioseq tab:

Global
Chn volume to: CC7 (Volume)
Master volume to: SysEx MstrVol
Chn priority to: CC25 (None)

Import S
✓ Create
Create
✓ Create
✓ Add M
✓ Create
Create

After importing, you should click the Audioseq tab and click Save Raw to save your m64 file. When you save the file, type the file extension, so if you want to call it "test", then you should type "test.m64" into the file dialog.

Now your port is done and you can load it into Rom Manager or your decomp repository and listen to it. To load it into Rom Manager, you need to go to the "Music Table" tab and then click the "Add Sequence" button. In decomp, you will need to rename your file to have the sequence number at the start, then place the .m64 into the folder */sound/sequences/us/*. Once you do this, you also need to open */sound/sequences.json*

and add the filename of your sequence to the dictionary, which is really just a matter of repeating what all the other lines are, just with your .m64 filename.
Once you have imported your file and go to play it, chances are it will sound really bad or not even play anything at all. There are a few things we need to learn about and fix in the midi before we can get a good sounding port, but what you did above is basically the full conversion process.

Audio Banks

Sm64 uses audio banks to determine what instruments are available to play your notes. A bank is a collection of instruments that get loaded into memory together. If you look at your midi, you can choose between 128 different instruments, but most banks in sm64 have less than ten to choose from. In order to make this work out, we need to pick what bank we are going to use, and then limit the instruments in our midi to the instruments in that bank.
You can find a list of all the banks and what instruments they have in this document, but you can also choose to read the decomp source code. The source code is a bit more complicated, so I won't cover that here, but you can search for it yourself by looking at *./sound/sound_banks/* and reading the JSONs inside of decomp.
You can also choose to use a custom sound bank, which has access to more instruments. If you want to do this, I recommend you follow this guide and use ext bank 0xC with Luna's Fix, made by aglab2 & patched by ShiN3. If you just want to use the original extended bank 0xC, follow aglab's original guide.
In order to select the bank you want to use for your sequence, you will need to select it in your editor of choice. For rom manager, you will go to the "Music Table" tab, select the *NInst* column, and then choose your music bank. For decomp, you need to open up the file, /sound/*sequences.json* and then next to the name of your sequence, edit the number inside the braces. The names in the braces correspond to the filenames of banks inside of */sound/sound_banks*.

Percussion

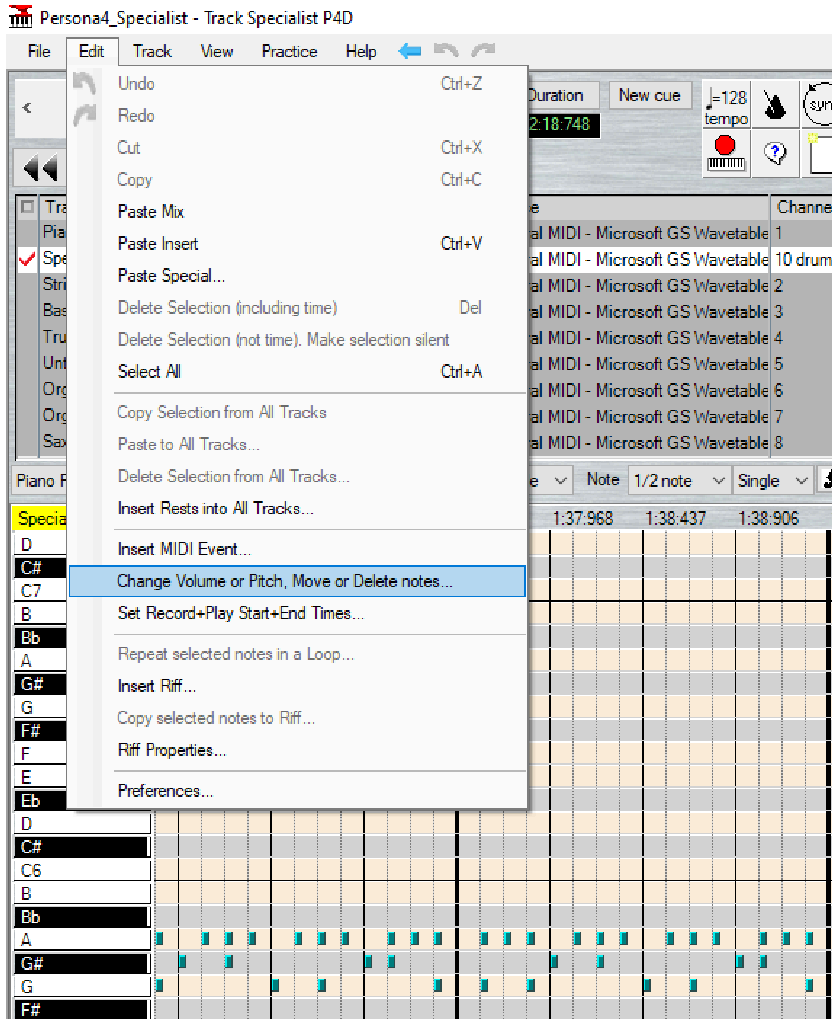
Percussion is separated from normal audio banks, just as it is in midi. Percussion has many more samples than a normal instrument does, so it needs to have special rules. Percussion is always instrument 127 (0x7F) and it has a custom mapping which you can find information about in this guide.

Arranging a Midi for sm64

The first thing you will need to do in order to arrange your midi to play well in sm64 is pick a bank. To make things simple for this tutorial, I will suggest you use bank 37 to start with. I will be making an example port with a typical midi in this tutorial, you can download the midi I am using on vgmusic. I will be using Anvil Studio to do this, but you should be able to do the same things I am doing with other programs.

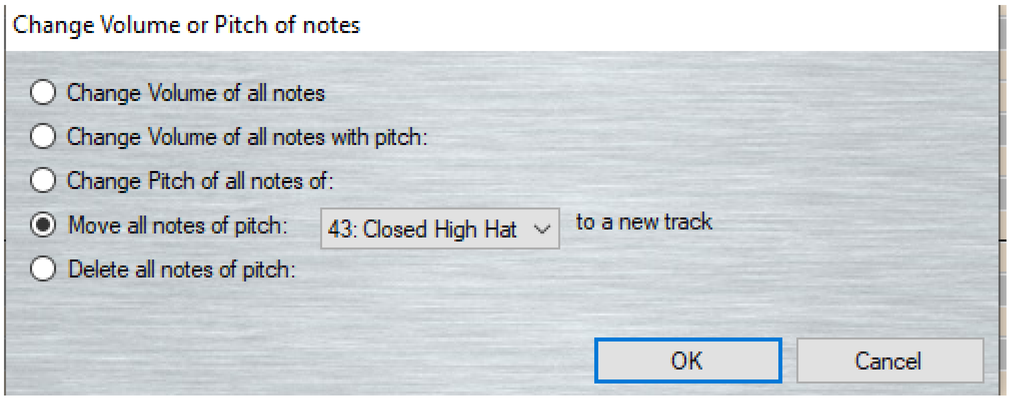
Fixing percussion

Fixing percussion is always the first step when arranging for sm64 from general midi. Gen mid has a lot more percussion sounds than sm64 does, so we need to replace or remove them for it to sound good. Gen midi always puts percussion in channel 10 with the instrument called "drums". Our goal is to move non percussion sounds outside of channel 10, and move the remaining notes to the ones sm64 uses. First you should try to move cymbals and hi hats out. Hi hats are notes (43-45) and (47) in midi. We need to open piano roll with our channel selected, and then we can move the notes by using Edit ->Change Volume or Pitch, move or delete notes.



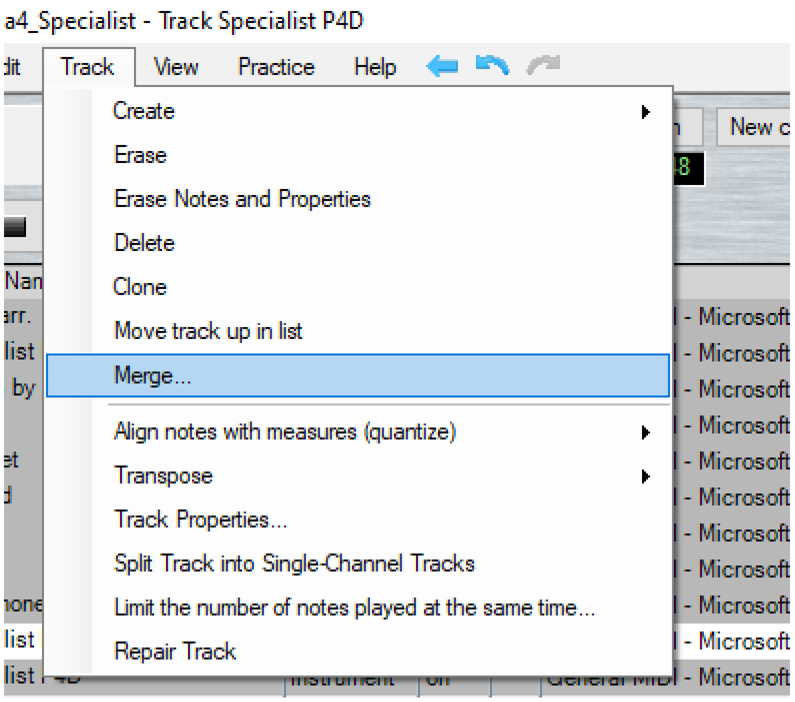
Persona4_Specialist - Track Specialist P4D
File
Edit
Track
View
Practice
Help
Undo Ctrl+Z
Redo
Cut Ctrl+X
Copy Ctrl+C
Paste Mix
Paste Insert Ctrl+V
Paste Special...
Delete Selection (including time) Del
Delete Selection (not time). Make selection silent
Select All Ctrl+A
Copy Selection from All Tracks
Paste to All Tracks...
Delete Selection from All Tracks...
Insert Rests into All Tracks...
Insert MIDI Event...
Change Volume or Pitch, Move or Delete notes...
Set Record+Play Start+End Times...
Repeat selected notes in a Loop...
Insert Riff...
Copy selected notes to Riff...
Riff Properties...
Preferences...

We will move all notes (43-45) and (47) to a new track by repeating this action for each note.


Change Volume or Pitch of notes

( ) Change Volume of all notes
( ) Change Volume of all notes with pitch:
( ) Change Pitch of all notes of:
(●) Move all notes of pitch: 43: Closed High Hat to a new track
( ) Delete all notes of pitch:

Once we do this, we will merge all of the new channels into a single channel, so that all hi hats are in the same place. Do this by selecting one of the new channels, and then clicking Track ->Merge...


a4_Specialist - Track Specialist P4D
Track
View
Practice
Help
Create
Erase
Erase Notes and Properties
Delete
Clone
Move track up in list
Merge...
Align notes with measures (quantize)
Transpose
Track Properties...
Split Track into Single-Channel Tracks
Limit the number of notes played at the same time...
Repair Track

Once a dialog pops up, choose another of the new channels. Once you merge the new track, delete the extra one by right clicking them, and hitting delete. We do not want repeated data, so we need to get rid of these extra channels. Repeat this for all new hi hat channels until one channel has all the data.

We will use this general procedure for all of the notes we do not want in percussion. In our midi, this is triangle (82), cabasa (70), crash cymbal (50), High and Low Agogo (68,69). We will merge the channels based on what goes together. If you look at the instrument list, you can see triangle and cabasa share an instrument, so we can merge them. We can also merge the agogos as they have a similar sound.

In order to verify that we have everything perfect, we need to look at the percussion list and see if all the instruments we have remaining in channel 10 are in the right spots. We can also do this to check if we need to remove anything else from the channel.
One thing that stands out is the notes at (87) in this midi. It has no label in Anvil Studio, and no equivalent in our percussion list. When this happens, the best thing you can do is left click the note in the piano roll and try to get a sound equivalent to it. I suggest putting this note on a low floor tom (42) based on the sound preview we get.
We also have some notes that do not match in sound, but have a label. Bongos and congas exist in both sm64 and gen midi, but not at the same notes, so we will use our percussion list to move the notes to the correct spot. In our midi I am moving Low Bongo (62) to (63). High Bongo (61) to (71), and High Timbale (66) to Tom Drum + 12 (54).

Special Note Instruments

Now that percussion works, we need to fix the instruments for the other tracks. It is simple enough to change what we start with to an instrument on the list, but certain instruments have special mappings we need to watch out for. One of these is hi hats, hi hats are closed up to note 44, and open afterwards. Luckily this matches the midi mapping, but others do not.
One of the ones we look out for is the cabasa. Cabasa works on note (83) and onwards, while triangle notes are before then. In our cabasa/triangle channel, move the cabasa notes (70) to (83) and I recommend you lower the triangle from its default (82) to something like (54). Finally, our crash cymbal will have to be on note (50) in order to work.

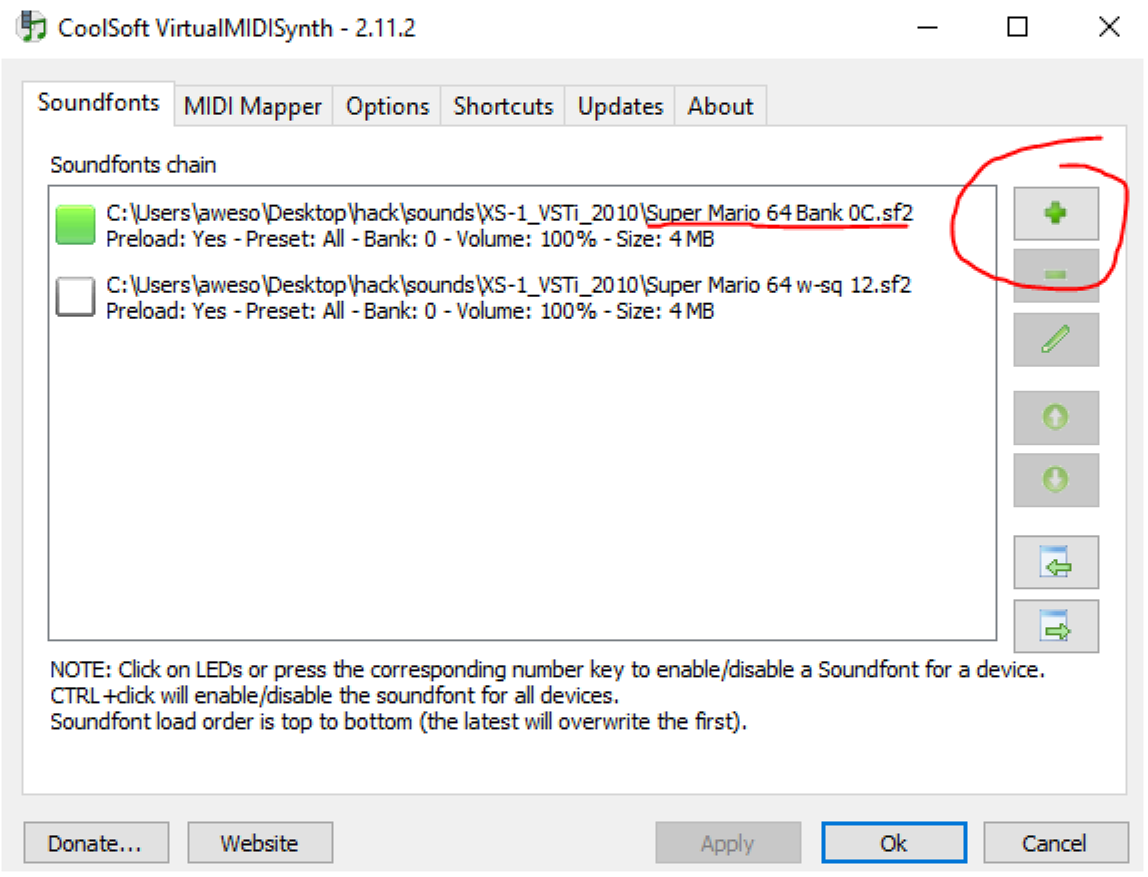
Picking Good Instruments

We have fixed up most of our instruments now, but we need to still choose instruments for channels other than 10. This is something you have to do by ear, but the basic process is that you choose an instrument in the bank that sounds close to, or can replace what is currently chosen, and then you try it out. Try to stay within the same class of instrument (like keep brass as brass, woods as wood etc.) and if you have to pick something really different and it sounds strange, try to transpose the track to see if it makes a difference.
I chose electric piano to replace piano, kept strings, hi hats, crash and triangle as their original instruments, made brass a french horn, tenor sax a pan flute, used crash cymbals for the agogos and made the organ into strings. Generally I recommend transposing channels down by one octave when swapping instruments to high pitch tunings like accordion or pan flute, but this is something else you need to test by ear.
Instruments in sm64 will also have different volumes than the instruments you hear in gen midi. You will have to lower and raise certain ones by ear. Porting songs is a lot of

trial and error, you can reduce this by using a sound font so you can preview how the instruments should sound in game.

Using sm64 Soundfont for Previewing

You can get a custom soundfont to use in my google drive. This soundfont was made by Edark, and edited by me to have some extra instruments in it. Download the soundfont, and then you can load it into your machine by using a program called virtual midi synth. Once you launch that program, click the plus button to choose the soundfont you want to load, and then hit the “Ok” button. I recommend you use the one titled “Super Mario 64 w-sq 12 pie fix.sf2” if you are using extended bank 0xC as that has the most up to date instruments and uses the same instrument numbers as extended bank 12, with the luna fix. If you just use the normal extended bank 12 this soundfont should work just fine as well. If you are using vanilla banks, use “Super Mario 64 w-sq.sf2” and the gen midi instruments should map to their SM64 counterpart.


CoolSoft VirtualMIDISynth - 2.11.2

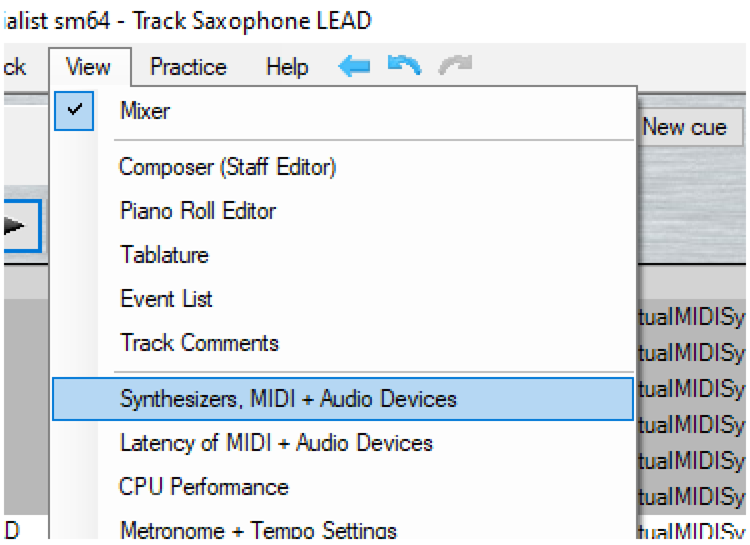
Soundfonts chain

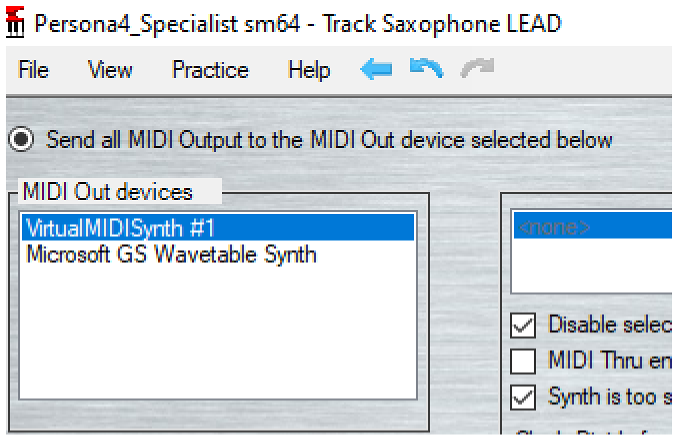
C:\Users\aweso\Desktop\hack\sounds\XS-1_VSTi_2010\Super Mario 64 Bank 0C.sf2
Preload: Yes - Preset: All - Bank: 0 - Volume: 100% - Size: 4 MB

C:\Users\aweso\Desktop\hack\sounds\XS-1_VSTi_2010\Super Mario 64 w-sq 12.sf2
Preload: Yes - Preset: All - Bank: 0 - Volume: 100% - Size: 4 MB

NOTE: Click on LEDs or press the corresponding number key to enable/disable a Soundfont for a device.
CTRL+click will enable/disable the soundfont for all devices.
Soundfont load order is top to bottom (the latest will overwrite the first).

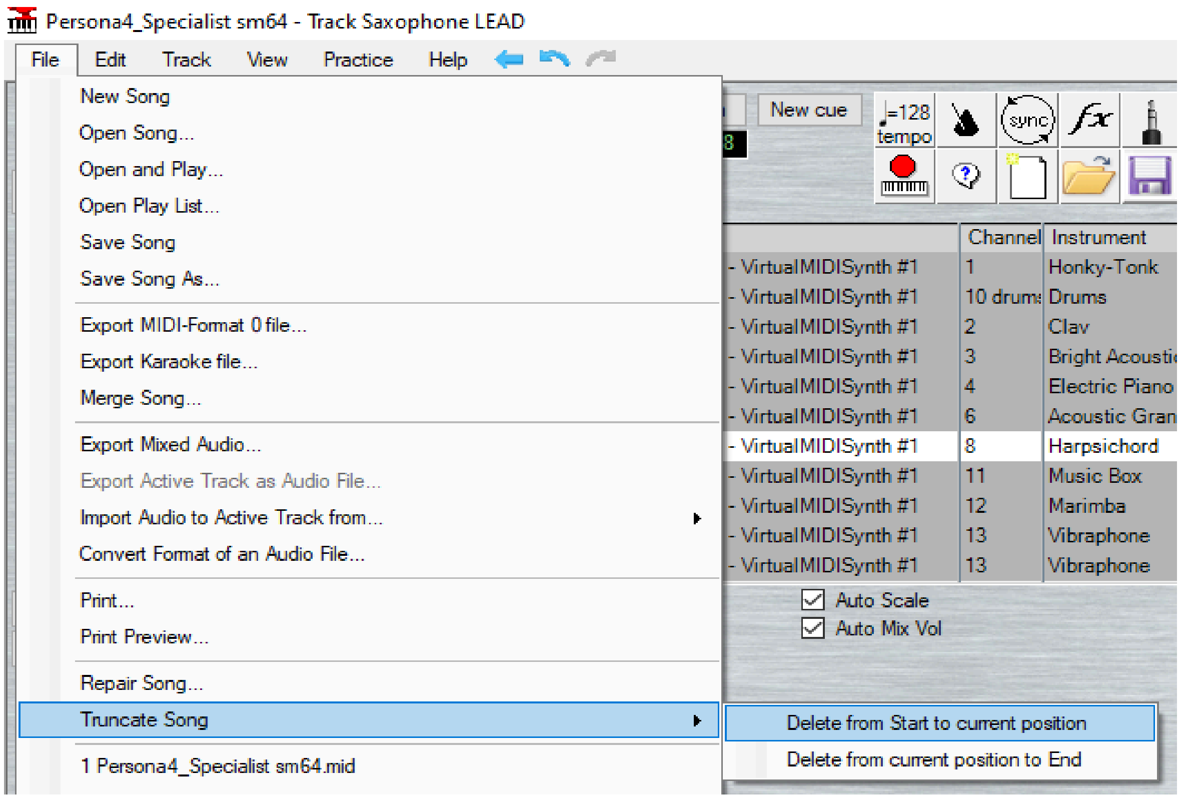
Now if you relaunch Anvil Studio, you can select your sm64 soundfont by clicking View -> Synthesizers, MIDI + Audio Devices.


ialist sm64 - Track Saxophone LEAD
ck
View
Practice
Help
Mixer
Composer (Staff Editor)
Piano Roll Editor
Tablature
Event List
Track Comments
Synthesizers, MIDI + Audio Devices
Latency of MIDI + Audio Devices
CPU Performance
Metronome + Tempo Settings
New cue
Then in the new window, click VirtualMIDISynth #1 as the output.


Persona4_Specialist sm64 - Track Saxophone LEAD
File
View
Practice
Help
Send all MIDI Output to the MIDI Out device selected below
MIDI Out devices
VirtualMIDISynth #1
Microsoft GS Wavetable Synth
<none>
Disable selec
MIDI Thru en
Synth is too s
Once you do all of this, you need to use the mapping text file inside the drive to see what instruments are what. In the text file, you will see a number on the left and an instrument name on the right. That number is what you select in Anvil Studio in order to play the sm64 instrument listed.

Looping

In order to loop, we just need to end the song at a loop point, and seq64 will automatically start the song back at the beginning. Some songs will have empty space at the start, we can get rid of that by navigating to the first note, or spot we want to start and clicking File -> Truncate Song -> Delete from Start to current position.

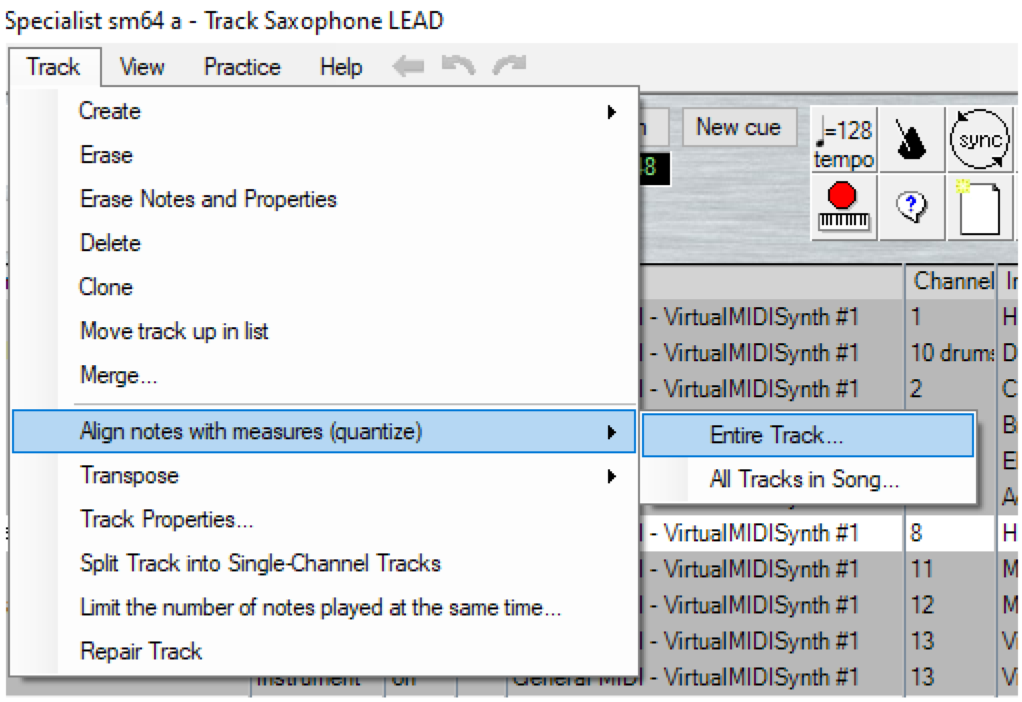

Persona4_Specialist sm64 - Track Saxophone LEAD
File
Edit
Track
View
Practice
Help
New Song
Open Song...
Open and Play...
Open Play List...
Save Song
Save Song As...
Export MIDI-Format 0 file...
Export Karaoke file...
Merge Song...
Export Mixed Audio...
Export Active Track as Audio File...
Import Audio to Active Track from...
Convert Format of an Audio File...
Print...
Print Preview...
Repair Song...
Truncate Song
1 Persona4_Specialist sm64.mid
Delete from Start to current position
Delete from current position to End
New cue
♩=128 tempo
sync
Auto Scale
Auto Mix Vol

	Channel	Instrument
- VirtualMIDISynth #1	1	Honky-Tonk
- VirtualMIDISynth #1	10 drums	Drums
- VirtualMIDISynth #1	2	Clav
- VirtualMIDISynth #1	3	Bright Acousti
- VirtualMIDISynth #1	4	Electric Piano
- VirtualMIDISynth #1	6	Acoustic Gran
- VirtualMIDISynth #1	8	Harpsichord
- VirtualMIDISynth #1	11	Music Box
- VirtualMIDISynth #1	12	Marimba
- VirtualMIDISynth #1	13	Vibraphone
- VirtualMIDISynth #1	13	Vibraphone

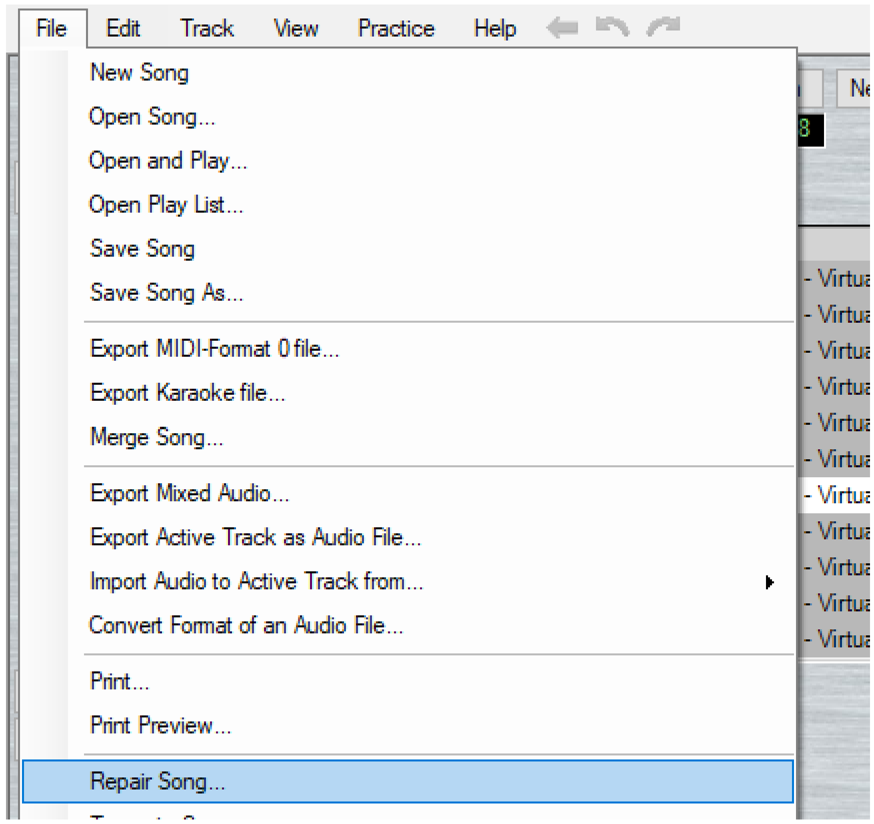
In order to get the proper end point in the song, we will do the same thing, but with the other Truncate song option. In order to know where to truncate so that we loop properly, you just need to listen to the song and look for the point where it repeats,or the point where it can return to the start without sounding bad.

Final Checks in MIDI

Finally, once you have a loop and your instruments set up, there are some final checks you can do to iron out some potential issues and lower the file size of your sequence. Make a back up before running any of these checks.
One optional check you can choose is to quantize your music. This will make sure all notes align on a certain interval. This can help things sound more consistent, and if notes are too quickly chained together, it can fix those as well.

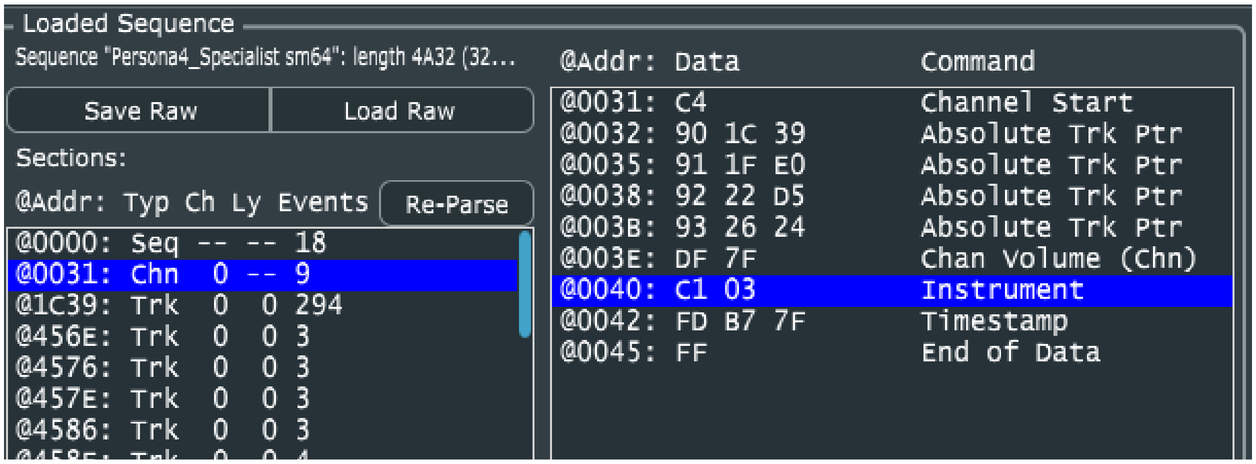

Specialist sm64 a - Track Saxophone LEAD
Track
View
Practice
Help
Create
Erase
Erase Notes and Properties
Delete
Clone
Move track up in list
Merge...
Align notes with measures (quantize)
Transpose
Track Properties...
Split Track into Single-Channel Tracks
Limit the number of notes played at the same time...
Repair Track
Entire Track...
All Tracks in Song...
New cue
♩=128 tempo
sync
Channel

Another tool to run is File -> Repair Song. This tool will run through the midi and remove all of the unnecessary events inside and optimize it automatically. This tool is aggressive, and can remove some necessary information, so double check if the song sounds the same afterwards before saving and when you do save, save it as a copy.



Final Touches Porting with seq64

If you repeat what I covered earlier in the Creating a port section, you will still likely have a bad sounding port. This is because we have a few fixes we need to do inside of seq64 before exporting this song properly.
After you import your midi and are on the Audioseq tab, instead of exporting, we need to edit some values. The ones we are going to focus on are in the channel instruments. In the middle list, inside the box called Loaded Sequence, we need to edit our channel data.

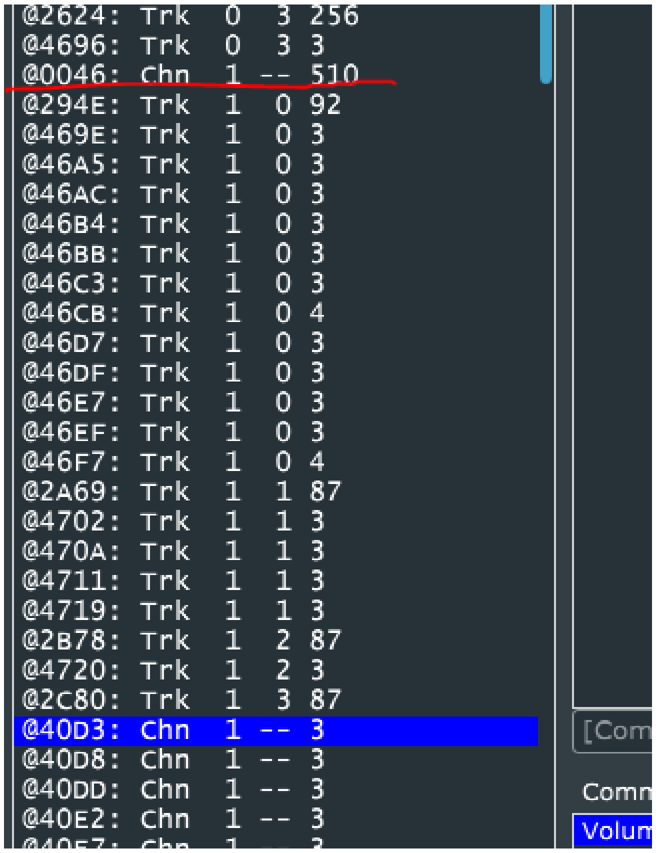


Sequence "Persona4_Specialist sm64": length 4A32 (32...

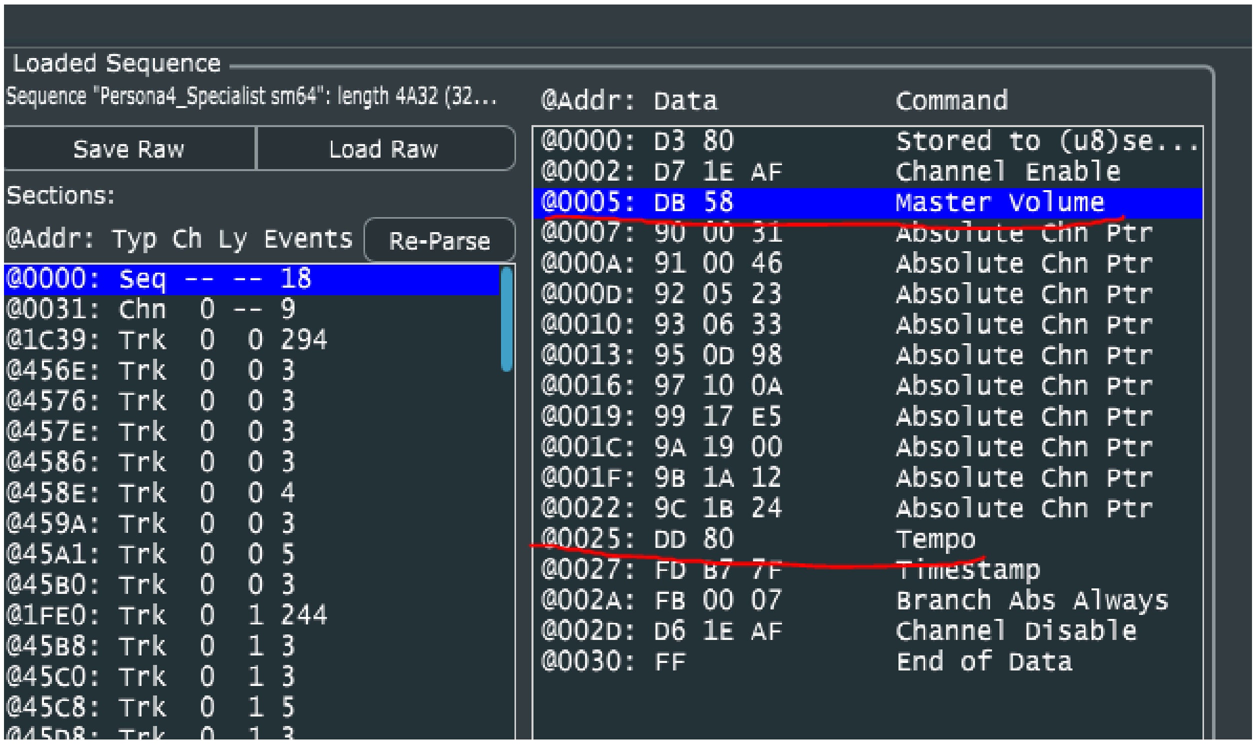
```
@Addr: Typ Ch Ly Events
@0000: Seq -- -- 18
@0031: Chn  0 -- 9
@1C39: Trk  0  0 294
@456E: Trk  0  0 3
@4576: Trk  0  0 3
@457E: Trk  0  0 3
@4586: Trk  0  0 3
```

```
@Addr: Data          Command
@0031: C4            Channel Start
@0032: 90 1C 39      Absolute Trk Ptr
@0035: 91 1F E0      Absolute Trk Ptr
@0038: 92 22 D5      Absolute Trk Ptr
@003B: 93 26 24      Absolute Trk Ptr
@003E: DF 7F         Chan Volume (Chn)
@0040: C1 03         Instrument
@0042: FD B7 7F      Timestamp
@0045: FF            End of Data
```

We need to make sure that the Value box at the bottom when we select Instrument here matches the instrument we want to play. I have Chn 0 selected inside my Sections table, which corresponds to channel 1 data in my midi. Channel 1 in my midi is supposed to be an electric piano, so I will find the electric piano in the instrument list for bank 37, and make sure that it matches the input to Value.
I have to do this for all my channels. You can find all of them by scrolling down in the middle box called Sections and clicking every row that says Chn X. The number where X is in Chn X tells you the channel number. The numbers here are one less than what you see in Anvil Studio.
While you are looking, you may see multiple Chn X rows with the same number, in general you only need to care about the first row. The other rows all represent something happening in the channel, but later in the song, which is after instruments are changed.


```
@2624: Trk  0  3 256
@4696: Trk  0  3 3
@0046: Chn  1 -- 510
@294E: Trk  1  0 92
@469E: Trk  1  0 3
@46A5: Trk  1  0 3
@46AC: Trk  1  0 3
@46B4: Trk  1  0 3
@46BB: Trk  1  0 3
@46C3: Trk  1  0 3
@46CB: Trk  1  0 4
@46D7: Trk  1  0 3
@46DF: Trk  1  0 3
@46E7: Trk  1  0 3
@46EF: Trk  1  0 3
@46F7: Trk  1  0 4
@2A69: Trk  1  1 87
@4702: Trk  1  1 3
@470A: Trk  1  1 3
@4711: Trk  1  1 3
@4719: Trk  1  1 3
@2B78: Trk  1  2 87
@4720: Trk  1  2 3
@2C80: Trk  1  3 87
@40D3: Chn  1 -- 3
@40D8: Chn  1 -- 3
@40DD: Chn  1 -- 3
@40E2: Chn  1 -- 3
```

Once you verify all your instrument numbers, you can also edit some other values. If you click the top row, you can edit the Master Volume and Tempo.



Sequence "Persona4_Specialist sm64": length 4A32 (32...

Sections:

```
@Addr: Typ Ch Ly Events
@0000: Seq -- -- 18
@0031: Chn  0 -- 9
@1C39: Trk  0  0 294
@456E: Trk  0  0 3
@4576: Trk  0  0 3
@457E: Trk  0  0 3
@4586: Trk  0  0 3
@458E: Trk  0  0 4
@459A: Trk  0  0 3
@45A1: Trk  0  0 5
@45B0: Trk  0  0 3
@1FE0: Trk  0  1 244
@45B8: Trk  0  1 3
@45C0: Trk  0  1 3
@45C8: Trk  0  1 5
```

```
@Addr: Data          Command
@0000: D3 80         Stored to (u8)se...
@0002: D7 1E AF      Channel Enable
@0005: DB 58         Master Volume
@0007: 90 00 31      Absolute Chn Ptr
@000A: 91 00 46      Absolute Chn Ptr
@000D: 92 05 23      Absolute Chn Ptr
@0010: 93 06 33      Absolute Chn Ptr
@0013: 95 0D 98      Absolute Chn Ptr
@0016: 97 10 0A      Absolute Chn Ptr
@0019: 99 17 E5      Absolute Chn Ptr
@001C: 9A 19 00      Absolute Chn Ptr
@001F: 9B 1A 12      Absolute Chn Ptr
@0022: 9C 1B 24      Absolute Chn Ptr
@0025: DD 80         Tempo
@0027: FD B7 7F      Timestamp
@002A: FB 00 07      Branch Abs Always
@002D: D6 1E AF      Channel Disable
@0030: FF            End of Data
```

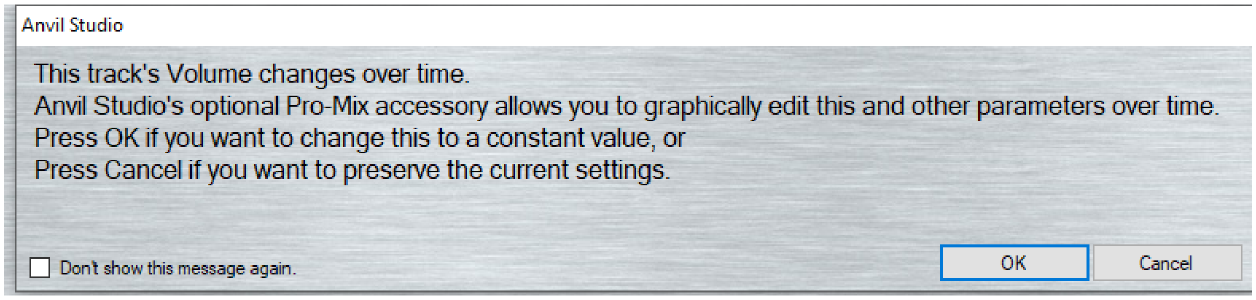
The Chn X rows may also have volumes of their own. You can choose to edit these for a quick re-import if you need a slight edit to a channel's volume, or if you are downloading some else's m64.
Once you are satisfied with your work, hit Save Raw just as before and make sure you choose bank 37 (0x25) for your m64 and it should import properly.
Your recording should sound similar to this: <my recording>. You can find the exact midi I used to create this here: <link>.

Fixing Extra Uncommon Issues

There are some uncommon issues you may come across that need involved steps to fix in your midi. Depending on how the midi is made, these issues can prevent your song from sounding good, or just cause issues during playback. The song I chose to port in this example has several of these issues and I will go over how to fix them, but most of these require some sort of sacrifice or a lot of extra work.

Channel Volume Bloat/Blocking

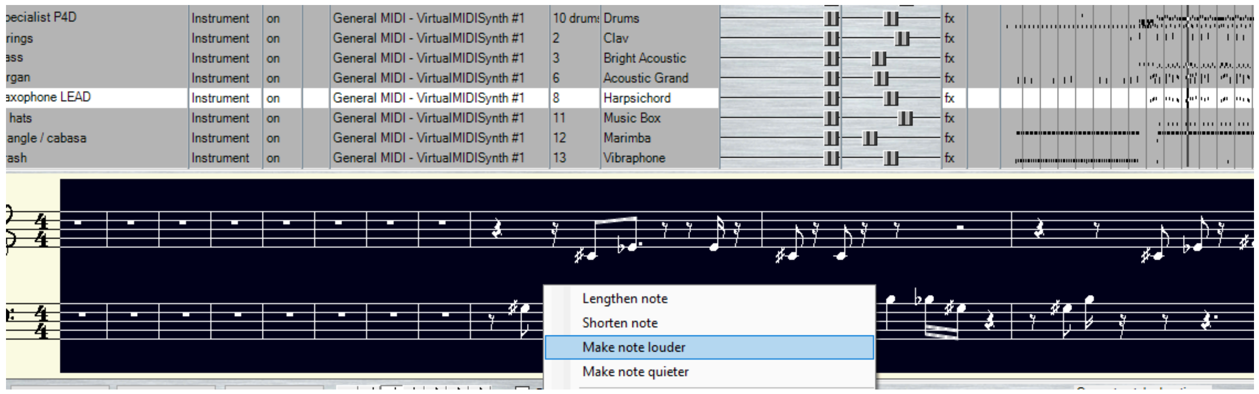
If you're using anvil studio, and you attempt to edit the volume in certain channels on certain midis, you may come across this message:



This track's Volume changes over time.
Anvil Studio's optional Pro-Mix accessory allows you to graphically edit this and other parameters over time.
Press OK if you want to change this to a constant value, or
Press Cancel if you want to preserve the current settings.

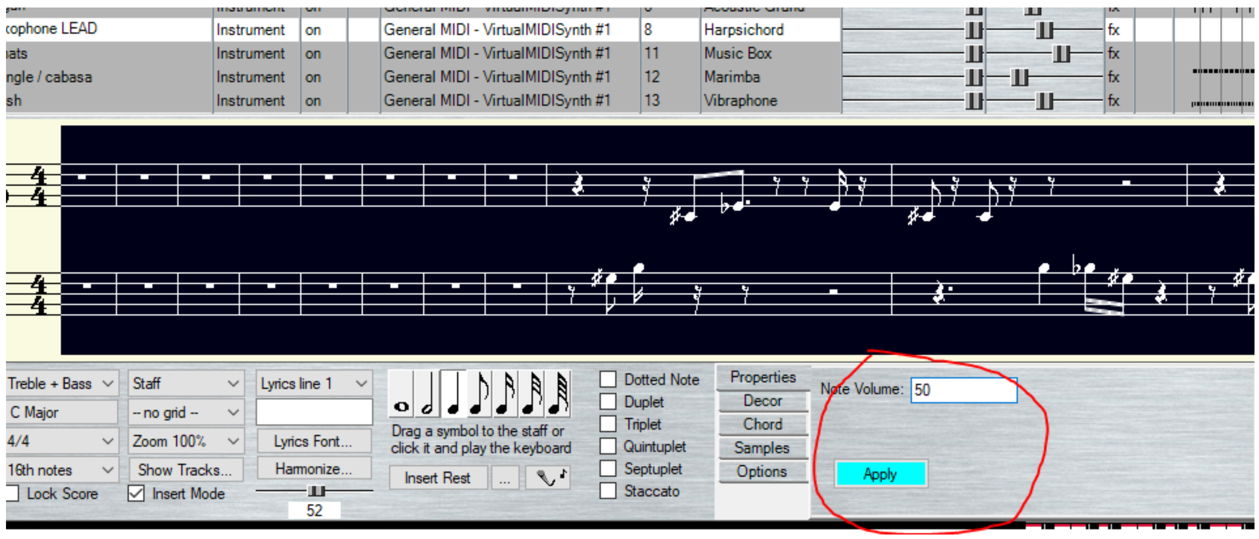
This is basically just letting you know that Anvil Studio sucks, and there is no good way to edit volume that is not one constant volume. There are multiple potential solutions to this, but the easiest and most effective is to hit the "ok" button, and then change the volume with the slider. This will make the volume in this channel constant, which sometimes is not really a big deal. For our example, this is the solution I chose. The added benefit of this is it reduces the file size of the midi, often considerably.
You can also fix this by instead opting to edit note volumes. This is more cumbersome, but lets you get the job done in Anvil Studio. I find the easiest way to do this is in Composer view. You select the notes you want to edit with your mouse, or select all with

ctrl + a, and then if you right click, you can lower or raise the volume with a button.



pecialist P4D	Instrument	on	General MIDI - VirtualMIDISynth #1	10 drum:	Drums	fx
rings	Instrument	on	General MIDI - VirtualMIDISynth #1	2	Clav	fx
ass	Instrument	on	General MIDI - VirtualMIDISynth #1	3	Bright Acoustic	fx
rgan	Instrument	on	General MIDI - VirtualMIDISynth #1	6	Acoustic Grand	fx
axophone LEAD	Instrument	on	General MIDI - VirtualMIDISynth #1	8	Harpsichord	fx
hats	Instrument	on	General MIDI - VirtualMIDISynth #1	11	Music Box	fx
angle / cabasa	Instrument	on	General MIDI - VirtualMIDISynth #1	12	Marimba	fx
ash	Instrument	on	General MIDI - VirtualMIDISynth #1	13	Vibraphone	fx

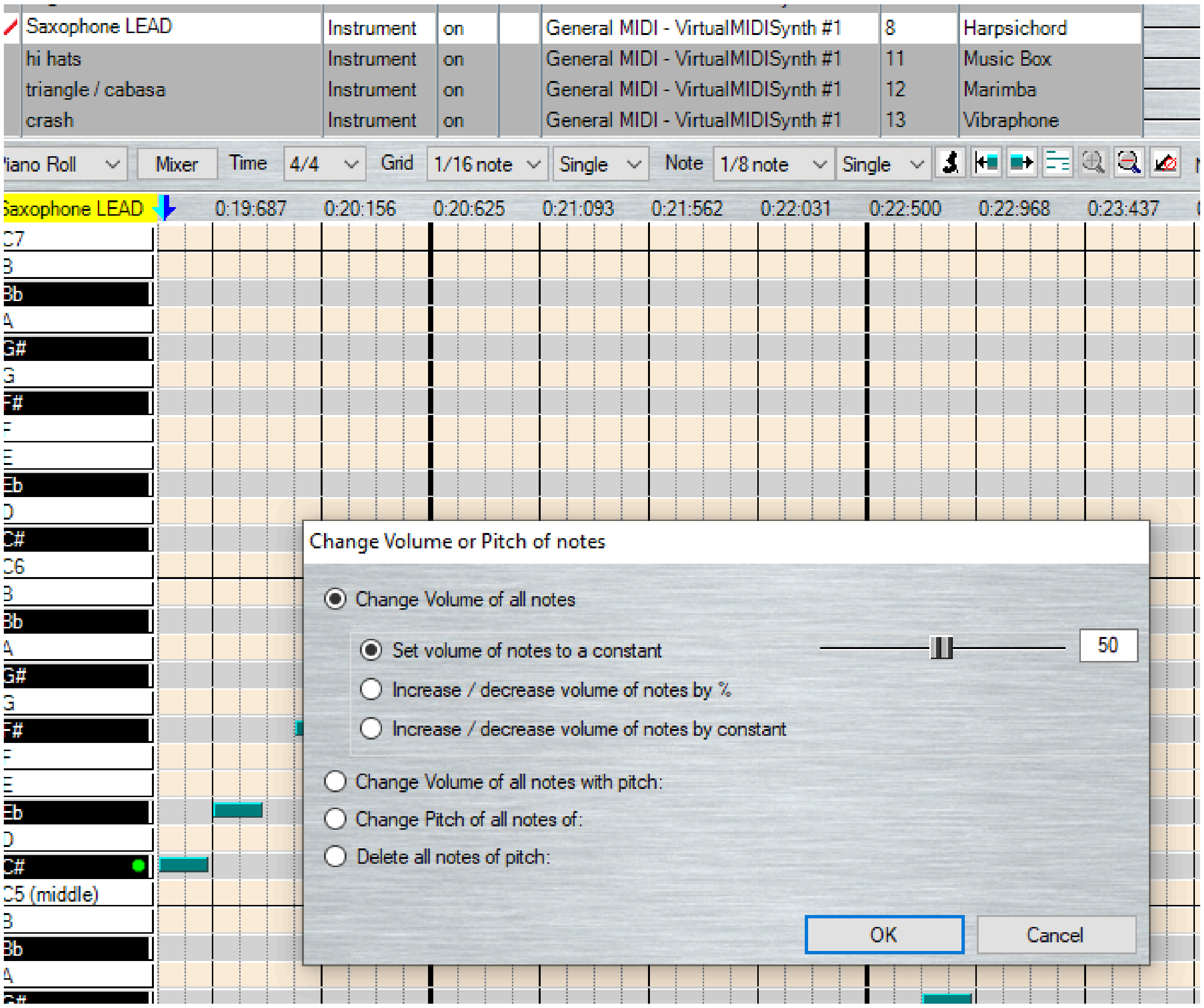
You can set a specific volume by typing in the note volume box and hitting apply.



cophone LEAD	Instrument	on	General MIDI - VirtualMIDISynth #1	8	Harpsichord	fx
ats	Instrument	on	General MIDI - VirtualMIDISynth #1	11	Music Box	fx
ngle / cabasa	Instrument	on	General MIDI - VirtualMIDISynth #1	12	Marimba	fx
sh	Instrument	on	General MIDI - VirtualMIDISynth #1	13	Vibraphone	fx

Treble + Bass
Staff
Lyrics line 1
C Major
-- no grid --
4/4
Zoom 100%
16th notes
[✓ Insert Mode]
52
Drag a symbol to the staff or click it and play the keyboard
Note Volume: 50

You can do the same for only some notes in the piano roll editor. Right click the keyboard on the left and you have options to lower by a specific amount, set to an amount, or lower by a percentage. You can do this for all notes, or just the one you right clicked.



Saxophone LEAD	Instrument	on	General MIDI - VirtualMIDISynth #1	8	Harpsichord
hi hats	Instrument	on	General MIDI - VirtualMIDISynth #1	11	Music Box
triangle / cabasa	Instrument	on	General MIDI - VirtualMIDISynth #1	12	Marimba
crash	Instrument	on	General MIDI - VirtualMIDISynth #1	13	Vibraphone

Change Volume or Pitch of notes

[✓ Change Volume of all notes]

[✓ Set volume of notes to a constant] 50

Increase / decrease volume of notes by %

Increase / decrease volume of notes by constant

Change Volume of all notes with pitch:

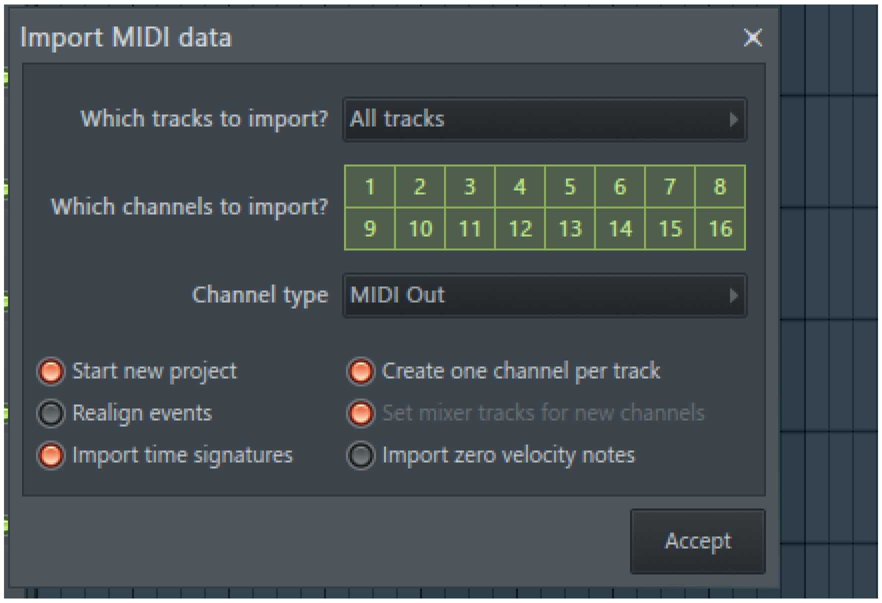
Change Pitch of all notes of:

Delete all notes of pitch:

If you don’t want to do these broad sweeps, you can also just edit the channel volumes in a more natural way, without destroying any of the note information. You do this by downloading a different program. I like to use FL Studio because it is free for midi editing (the free trial gives you everything you need) and useful for other issues.

Using FL Studio

Once you start up FL Studio, import with the file menu, file -> import -> import MIDI, and then after selecting your file, choose MIDI Out as your channel type.


Import MIDI data

Which tracks to import? All tracks

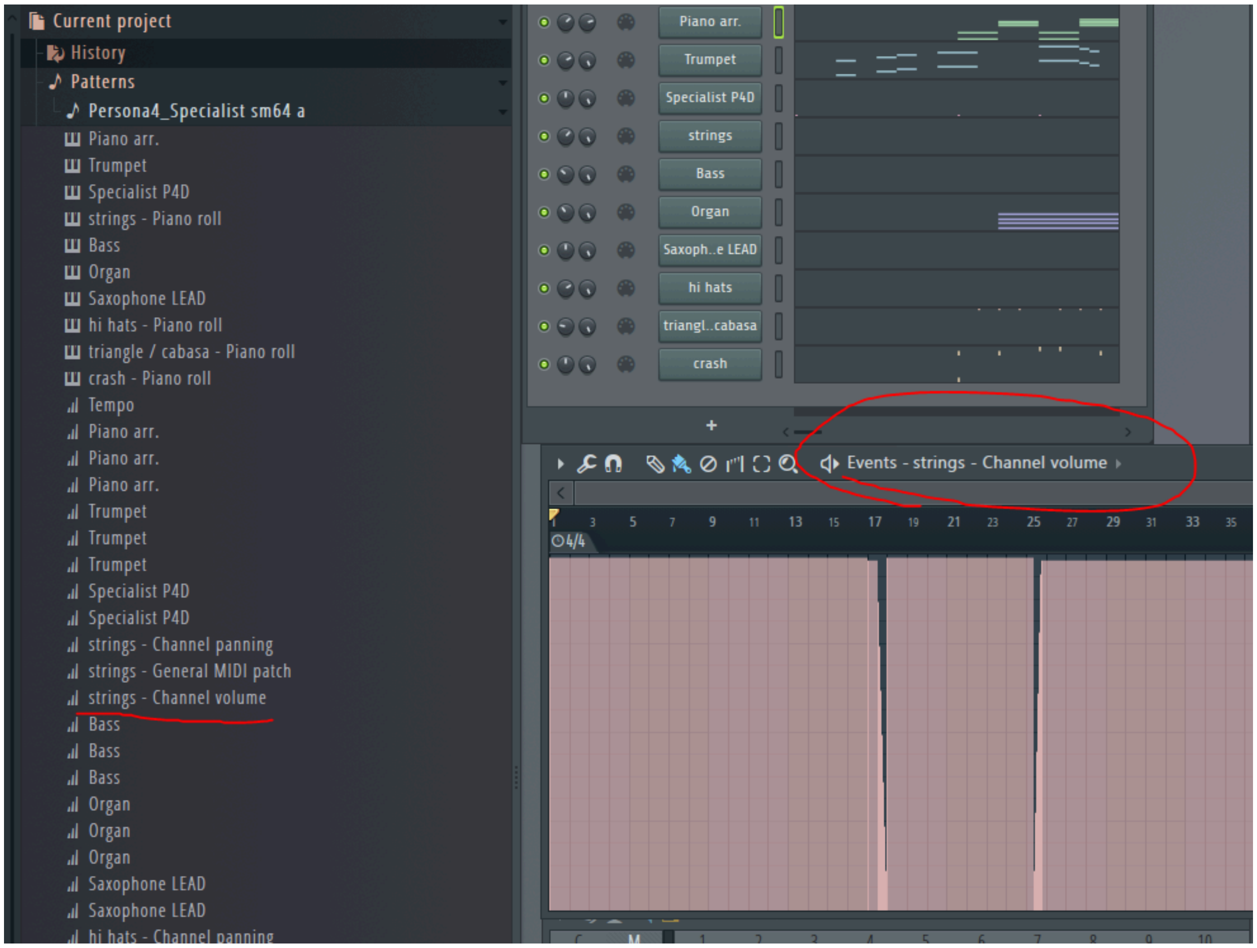
Which channels to import? 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Channel type MIDI Out

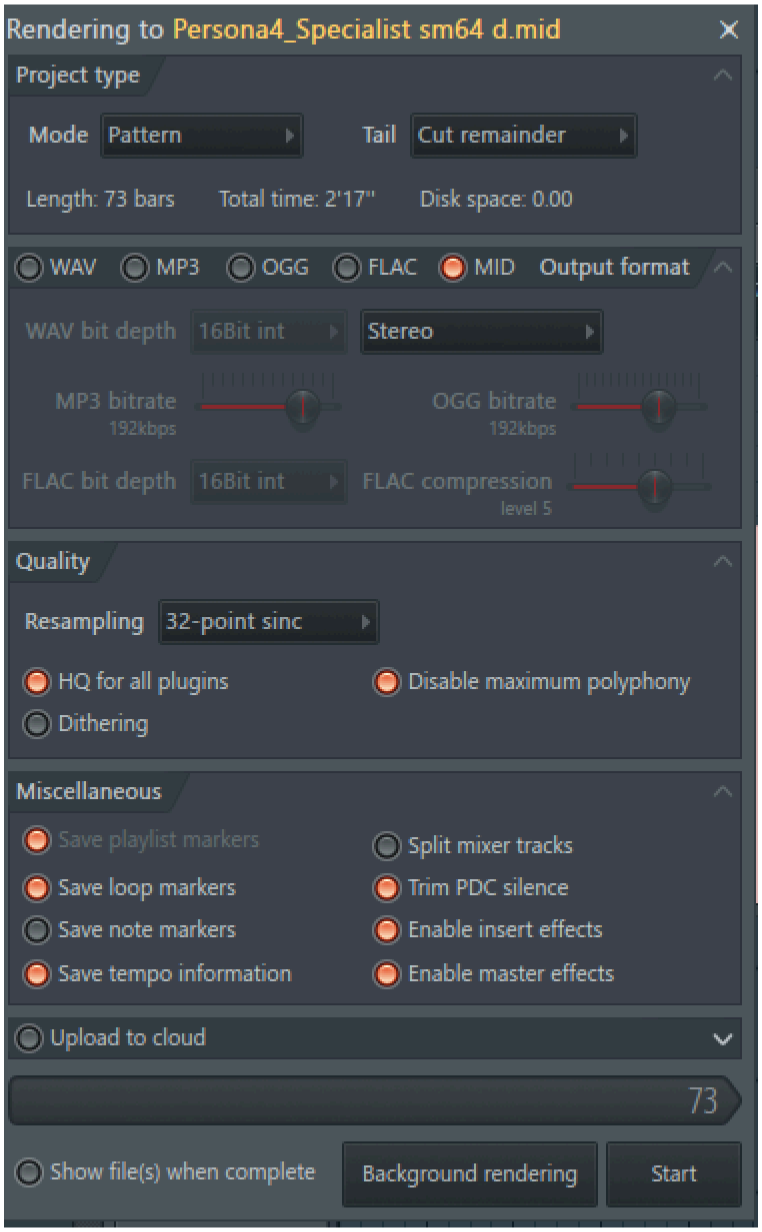
[✓ Start new project]
Realign events
[✓ Import time signatures]
[✓ Create one channel per track]
[✓ Set mixer tracks for new channels]
Import zero velocity notes

Accept

To find your channel data, you want to navigate the project tree on the left side of the screen to get to your channel's volume. The path for this is Current Project -> Patterns -> MIDI Name -> channel name - Channel Volume.


Current project
History
Patterns
Persona4_Specialist sm64 a
Piano arr.
Trumpet
Specialist P4D
strings - Piano roll
Bass
Organ
Saxophone LEAD
hi hats - Piano roll
triangle / cabasa - Piano roll
crash - Piano roll
Tempo
Piano arr.
Piano arr.
Piano arr.
Trumpet
Trumpet
Trumpet
Specialist P4D
Specialist P4D
strings - Channel panning
strings - General MIDI patch
strings - Channel volume
Bass
Bass
Bass
Organ
Organ
Organ
Saxophone LEAD
Saxophone LEAD
hi hats - Channel panning

Now you can edit the volume with the paintbrush. To save, export the file as MIDI and use these settings.



Rendering to Persona4_Specialist sm64 d.mid

Project type

Mode: Pattern — Tail: Cut remainder

Length: 73 bars — Total time: 2'17" — Disk space: 0.00

Output format: WAV MP3 OGG FLAC [✓ MID]

WAV bit depth: 16Bit int — Stereo

MP3 bitrate: 192kbps — OGG bitrate: 192kbps

FLAC bit depth: 16Bit int — FLAC compression: level 5

Quality

Resampling: 32-point sinc

[✓ HQ for all plugins] — [✓ Disable maximum polyphony]

Dithering

Miscellaneous

[✓ Save playlist markers] — Split mixer tracks

[✓ Save loop markers] — [✓ Trim PDC silence]

Save note markers — [✓ Enable insert effects]

[✓ Save tempo information] — [✓ Enable master effects]

Upload to cloud

73

Show file(s) when complete — Background rendering — Start

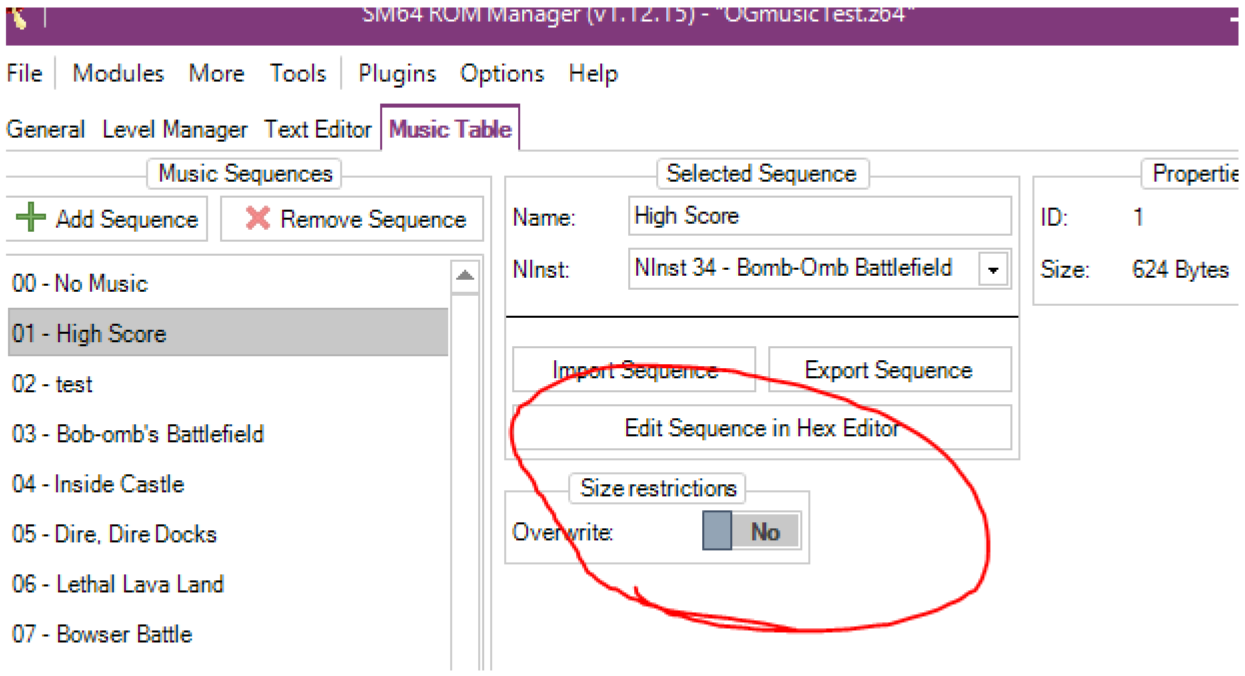
Bad Pitch Bends

Due to the inconsistencies of the MIDI format, sometimes channels with pitch bends do not sound correct when imported, even though they sound perfectly fine in your MIDI editor. There is no 100% tried method to fix this, but the most consistent way I have found to do this is to import the MIDI into FL Studio, and then export it immediately back out to Anvil Studio and then import again into sm64. The reason this works is because FL Studio Exports are generally aligned with the pitch bend scaling of sm64, so it will rewrite the MIDI with all the correct values just by doing a quick import/export.

I’ve gone over how to import/export with FL Studio in the above section, so refer to that on how to do this.

Notes Cutting Out or Seq Stopping

This is a common problem with MIDIs that have a lot of channels or notes, or with the settings you have in your sm64 file. The basic cause is that you’re running out of audio memory or hit the concurrent note limit which is 16 notes at once. There are advanced ways to deal with this, but the basic solutions are to increase the memory in sm64, and to reduce the amount of notes playing.
In general, if the entire sequence plays but notes end early, you hit a note limit, but if the entire sequence stops or things sound very weird, it's probably a memory issue.
Usually if you have this problem, you can afford to remove some notes. Test out what you can afford to lose by muting a channel or two and previewing it, and then delete the channels and save a copy and try to see if that works.
If you want to try and increase memory then you have a few options. In Rom Manager, your only choice is with the Overwrite Size Restrictions button.



00 - No Music
01 - High Score
02 - test
03 - Bob-omb's Battlefield
04 - Inside Castle
05 - Dire, Dire Docks
06 - Lethal Lava Land
07 - Bowser Battle

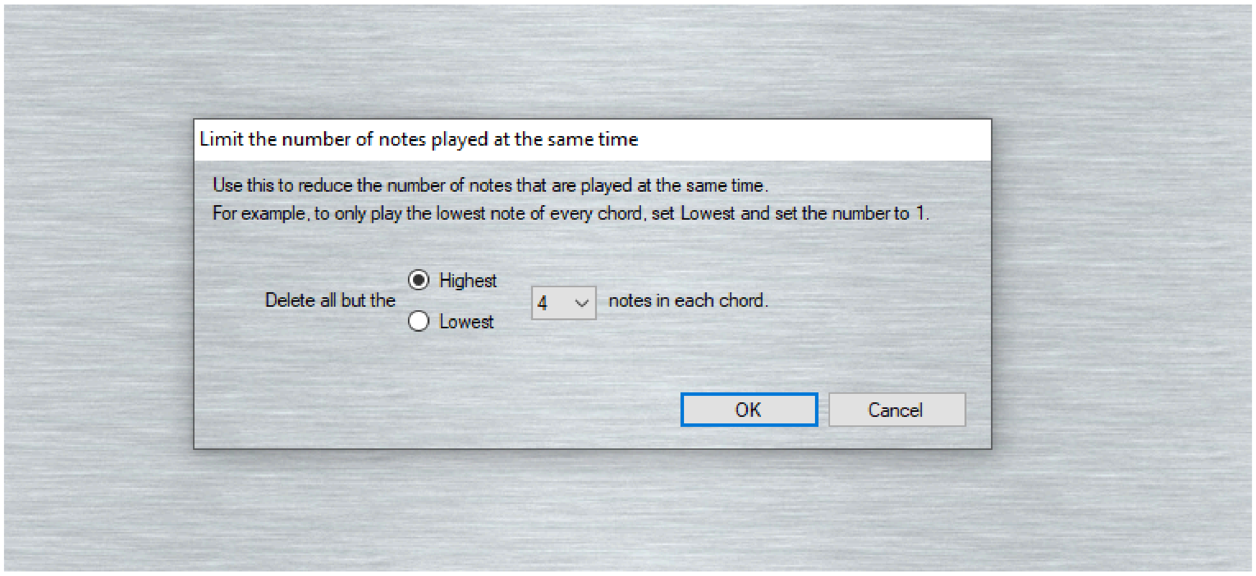
Name: High Score
NInst: NInst 34 - Bomb-Omb Battlefield
ID: 1
Size: 624 Bytes
Overwrite: No

This will increase your memory size. Different sequences use different memory sizes, and this will make them all use the largest one (and maybe increase it more??). If you’re putting this over a cap theme or some other vanilla theme, this could help solve your problem.
If you’re in decomp, you want to edit the variable *gAudioSessionPresets*. You can find this by going through */src/audio/data.c*. This is a file table that has all the memory sizes for sequences. You can read the comment in the file for what each member does, but to

not get too technical on all of them, you just increase the memory values and max number of simultaneous notes.

Too Many Simultaneous Notes

If you're playing too many notes at the same time in one channel, you will get an error message in seq64, and things will be cut out. This may not be a big deal, and oftentimes you can just ignore this, but it is good practice to deal with this before finalizing the midi.
The easiest way to deal with it is to remove the extra notes. You can automate this in Anvil Studio with a tool in the Track menu. Limit the notes to 4 and I would recommend keeping the highest 4 notes.

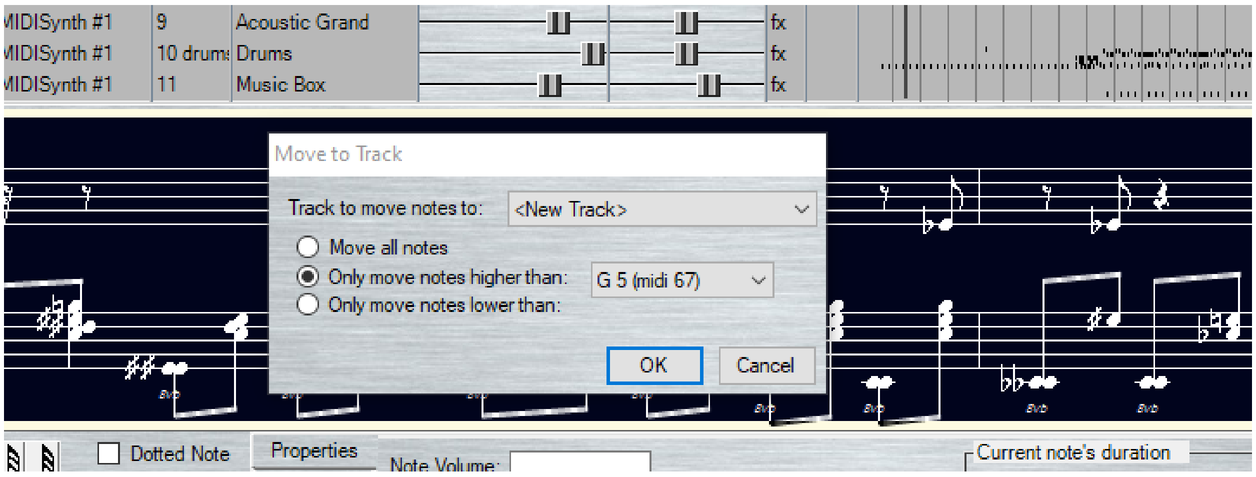


Limit the number of notes played at the same time

Use this to reduce the number of notes that are played at the same time.
For example, to only play the lowest note of every chord, set Lowest and set the number to 1.

Delete all but the [✓ Highest] Lowest 4 notes in each chord.

The other way to deal with this is to move notes to new tracks. You can do this the same way we did it earlier with specific notes, or you can choose to move an entire section in the Composer view. Select all the notes in a track in the composer, and then right click and hit the move to a new track option. You can then move all the notes above or below a certain note. Choose a note that will split the track cleanly and leave less than 4 notes in each one.



When you do this, make sure you move the settings on the original track to the new track. You can also clone settings by cloning a track and then deleting all the notes, which may be necessary if your track has a lot of things like volume changes you want to be certain are the same.

Extra Credit Tuning

If you've done all the above properly, you might have a pretty good port already, but there is some extra midi editing we can do to improve it besides just picking good instruments, volumes and panning.
These edits come at the cost of a larger sequence size, so I would not recommend always doing these because it can be more expensive and end up not running well in game, always do these with a copy of your original file.

Chorus

We can add a chorus effect to a track by doubling or tripling the track. This can be done for any track, bassline, melody or anything in between, just as long as the track would improve from sounding fuller.
Find the track you want to thicken sound in, right click it and hit clone track. This will create a new track with all the notes and same instrument. You need to have a spare track for this to work.
Now you need to edit the new and old track a bit to make it sound good. I like to lower the volume on both, and give the new track a slightly lower volume than the old. Then I usually also give them opposite panning. You can also try to change the octave, add vibrato or reverb for more depth. If you're feeling experimental you can also change the instrument as well, though that usually does not work as well.

Echo

We can add echo very easily by using multiple tracks. Clone the track you want to create an echo on (right click and hit clone track), and then enter the composer view. Now at the beginning of the new track at time zero, insert a rest. The length of the rest determines how the echo feels. I like to use a 16th note or 32nd note.
Now on the new track, reduce the volume a substantial amount and if you want, edit the panning to make it sound like it's moving, but don't go too hard on the panning or it will lose its echo effect.

Chords and octaving

Chords are formed when you play multiple notes at the same time. Sm64 supports 4 notes max at the same time on one channel. Oftentimes midis are created without using chords, or the sm64 instruments would work better with them than the original did. Depending on the instrument, adding chords or octaves to a channel can improve the sound, especially in higher registers.
To add chord effects, clone a track and transpose it. How you transpose depends on the effect you want, but I always start by going one octave lower. Now you need to edit the volume of these notes. Generally you want these notes to be quieter. To do this, enter the composer view, select all the notes (ctrl + a or right click -> select all) and then right click and hit lower note volume. Repeat until your desired volume is reached.
Now we want to put this on the same channel as our original channel. In order to be safe, start by changing the channel number to the same as the original instead of merging. After testing that you like the sound, merge the channels.
You can look up chords online, or use your own music knowledge to pick good ones. I usually just stick to a duplicate octave, but experiment and find what fits your song best.

Conclusion

I have gone over everything I would do in a basic process to port songs. You can find the ports I made for this tutorial at this link. I have both the original basic port, and the second port where I applied fixes to common issues you may come across.
With this tutorial, you should have a basic understanding on how to port songs. There are more advanced tutorials and video tutorials you can watch if you're still confused or need more help, but the best way to learn is to do the work yourself. If you have any questions you can reach me where I can be reached.

- sm64romhackswebsite@gmail.com
- https://romhacking.com/user/jesusyoshi54
- https://www.youtube.com/c/jesusyoshi54
- https://gitlab.com/scuttlebugraiser