Insert Manually a Song Available on the Forum

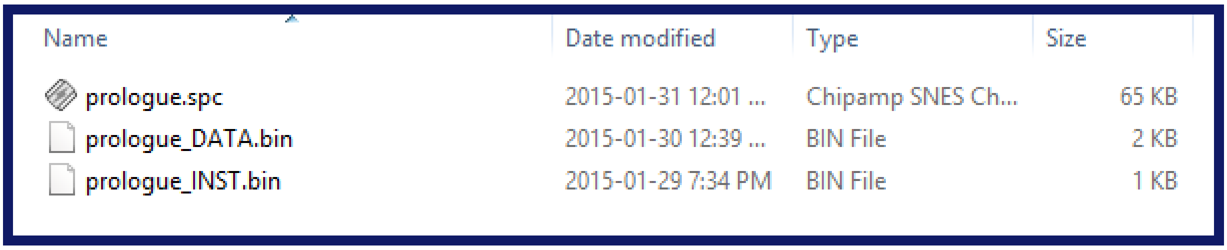
In this tutorial we'll be importing in FF3us a song that is already in FF6 music format. There is no need for musical knowledge to be able to complete the following steps. The only things that will help is being familiar with the hexadecimal system as well as some basic ROM hacking knowledge like what is an offset, the difference between an absolute and hirom offset and how to use an hex editor.

1. Getting the file and tool

We'll be using the FFIV song "The Prologue" from our Song Database. The other thing you will need is a hex editor. There are many you can choose from but I'd suggest one that has a copy selection, paste-write and paste-insert functionalities. One good all purpose hex editor is HxD and this is what has been used to take the screenshots below.

2. Files we will be importing

Extract the files from **FF4_prologue.7z** and you will see the following files:



Name	Date modified	Type	Size
prologue.spc	2015-01-31 12:01 ...	Chipamp SNES Ch...	65 KB
prologue_DATA.bin	2015-01-30 12:39 ...	BIN File	2 KB
prologue_INST.bin	2015-01-29 7:34 PM	BIN File	1 KB

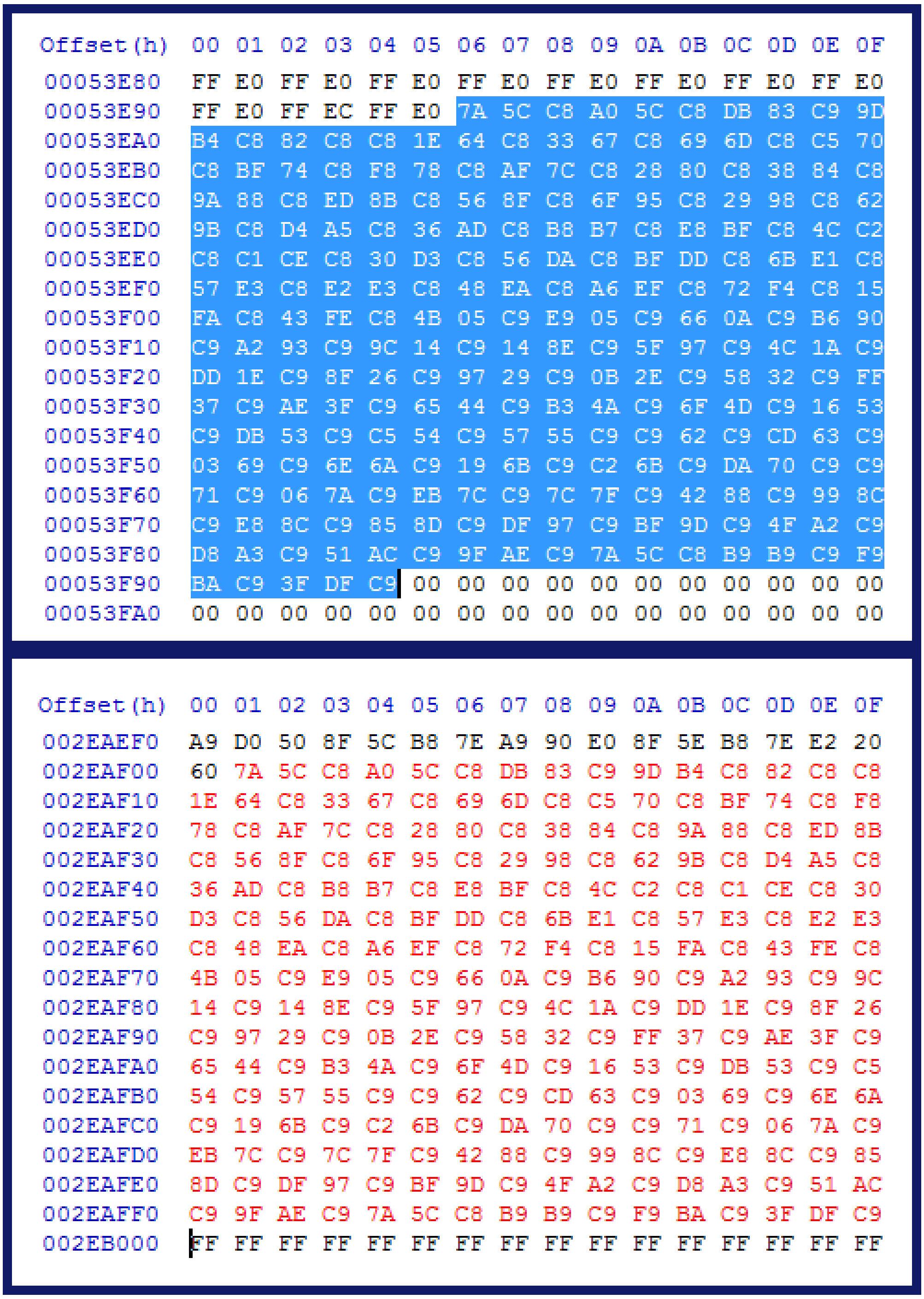
prologue.spc is the song in SNES format. Those can be played with a SPC player or with winamp and a plug-in. For more info on SPC file and how to play them, use Google or check the great extracted music tutorial at FantasyAnime.com. **prologue_DATA.bin** and **prologue_INST.bin** are two binary files, the first has the music data in FF6 format and the other the instruments used in the song. The DATA and INST files do not contain instrument samples, those are in the game.

All songs in our song database have the three same files. Some songs, mostly done by tsushiy and having a samll “p” next to their title need the instrument patch, available in the same thread. This patch install new BRR samples in the game, giving access to a wider range of instruments to use in songs. Some songs instrument file use those added instruments.This is not the case of our prologue song.

3. Choosing the right spot

The first thing that you have to ask yourself is *“do I want to replace an existing song or expand the number of existing songs?”* If you only want to replace a existing song you can skip to section 4. To expand the number of songs there are few thing to do. First you have to move the $C53E96-$C53F94 song pointers block because there is no room to add an extra one. To find a spot in non-expanded ROM, you can look here. The offsets in this list take in account the ROM header though while my whole tutorial assume you have a headerless ROM, so substract 0x200 from the offset you choose. You can also expand your ROM and put the pointers in the $FX banks.

I decided for the example to take the $EEAF01 spot, which has 767 free bytes, which is more than enough. As shown in the image below, I select with the mouse the pointer block, I do **Ctrl+c**, **Ctrl+g** to enter my destination (you can scroll a long time for the same result) and then **Right-Click→Paste write**. Now let's say I would put my song at $F30540 in expanded ROM, I would type **40 05 F3** at $EEB0000 (see right picture). As you have guessed pointers are always inverted whether they are two or three bytes long. As for the old pointer data copied, you can blank with 00 or FF all the data since you moved it, giving you room for other things if needed.



```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

00053E80  FF E0 FF E0 FF E0 FF E0 FF E0 FF E0 FF E0 FF E0
00053E90  FF E0 FF EC FF E0 7A 5C C8 A0 5C C8 DB 83 C9 9D
00053EA0  B4 C8 82 C8 C8 1E 64 C8 33 67 C8 69 6D C8 C5 70
00053EB0  C8 BF 74 C8 F8 78 C8 AF 7C C8 28 80 C8 38 84 C8
00053EC0  9A 88 C8 ED 8B C8 56 8F C8 6F 95 C8 29 98 C8 62
00053ED0  9B C8 D4 A5 C8 36 AD C8 B8 B7 C8 E8 BF C8 4C C2
00053EE0  C8 C1 CE C8 30 D3 C8 56 DA C8 BF DD C8 6B E1 C8
00053EF0  57 E3 C8 E2 E3 C8 48 EA C8 A6 EF C8 72 F4 C8 15
00053F00  FA C8 43 FE C8 4B 05 C9 E9 05 C9 66 0A C9 B6 90
00053F10  C9 A2 93 C9 9C 14 C9 14 8E C9 5F 97 C9 4C 1A C9
00053F20  DD 1E C9 8F 26 C9 97 29 C9 0B 2E C9 58 32 C9 FF
00053F30  37 C9 AE 3F C9 65 44 C9 B3 4A C9 6F 4D C9 16 53
00053F40  C9 DB 53 C9 C5 54 C9 57 55 C9 C9 62 C9 CD 63 C9
00053F50  03 69 C9 6E 6A C9 19 6B C9 C2 6B C9 DA 70 C9 C9
00053F60  71 C9 06 7A C9 EB 7C C9 7C 7F C9 42 88 C9 99 8C
00053F70  C9 E8 8C C9 85 8D C9 DF 97 C9 BF 9D C9 4F A2 C9
00053F80  D8 A3 C9 51 AC C9 9F AE C9 7A 5C C8 B9 B9 C9 F9
00053F90  BA C9 3F DF C9 00 00 00 00 00 00 00 00 00 00 00
00053FA0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

002EAEF0  A9 D0 50 8F 5C B8 7E A9 90 E0 8F 5E B8 7E E2 20
002EAF00  60 7A 5C C8 A0 5C C8 DB 83 C9 9D B4 C8 82 C8 C8
002EAF10  1E 64 C8 33 67 C8 69 6D C8 C5 70 C8 BF 74 C8 F8
002EAF20  78 C8 AF 7C C8 28 80 C8 38 84 C8 9A 88 C8 ED 8B
002EAF30  C8 56 8F C8 6F 95 C8 29 98 C8 62 9B C8 D4 A5 C8
002EAF40  36 AD C8 B8 B7 C8 E8 BF C8 4C C2 C8 C1 CE C8 30
002EAF50  D3 C8 56 DA C8 BF DD C8 6B E1 C8 57 E3 C8 E2 E3
002EAF60  C8 48 EA C8 A6 EF C8 72 F4 C8 15 FA C8 43 FE C8
002EAF70  4B 05 C9 E9 05 C9 66 0A C9 B6 90 C9 A2 93 C9 9C
002EAF80  14 C9 14 8E C9 5F 97 C9 4C 1A C9 DD 1E C9 8F 26
002EAF90  C9 97 29 C9 0B 2E C9 58 32 C9 FF 37 C9 AE 3F C9
002EAFA0  65 44 C9 B3 4A C9 6F 4D C9 16 53 C9 DB 53 C9 C5
002EAFB0  54 C9 57 55 C9 C9 62 C9 CD 63 C9 03 69 C9 6E 6A
002EAFC0  C9 19 6B C9 C2 6B C9 DA 70 C9 C9 71 C9 06 7A C9
002EAFD0  EB 7C C9 7C 7F C9 42 88 C9 99 8C C9 E8 8C C9 85
002EAFE0  8D C9 DF 97 C9 BF 9D C9 4F A2 C9 D8 A3 C9 51 AC
002EAFF0  C9 9F AE C9 7A 5C C8 B9 B9 C9 F9 BA C9 3F DF C9
002EB000  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

The next thing to do is to change the following code. It's the only place pointers are read and you have to modify each LDA instruction so new base offset, new base offset + 1, new base offset + 2. This is not an ASM course so I'm not explaining further except only mentioning you have a total of 9 bytes to change.

Original code

Modified code

Last thing to do is changing the total number of songs. The offset is $C53C5E. It's a single byte that should be $55. Increase it of one each time you add a new song (replacing an existing one does not count). So congratulation you have not expanded pointer and can add as much songs as you want up to a limit of 255!

Result in-game

sd-result.mp4