Converting SF2 soundfonts to X2 DreamBlaster soundbanks : a step by step guide. (SE Compiler)

SE Rev 2, May 25th 2024, Serdaco BV

1. Introduction

The DreamBlaster X2/X2GS is a synthesizer module based on the Dream SAM5704B 4 core DSP chip. The X2 module has 64 mbyte of NOR flash memory to which you can upload binary soundbanks in using the X2Upload utility. X2GS has the same flash size, but 16mbyte is reserved for a fixed ROM GS bank, so only maximum 48 mbyte can be uploaded.

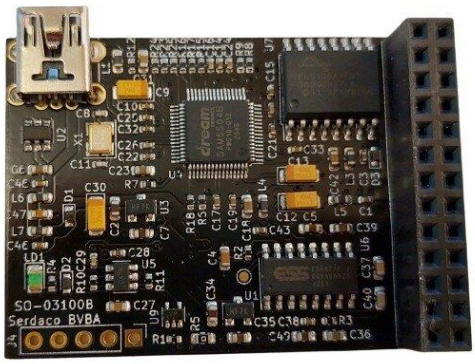

Figure 1: DreamBlaster X2

These binary soundbanks can be compiled using the proprietary dream bank compiler included in the professional Dream 5000 SDK. This 5000 SDK consists of :

- Hardware :
  - 5000 SDK hardware box : USB allows to edit and test instrument sounds and banks.
- Software :
  - Dream Instrument editor software : Allows to edit melodic instruments. (Dream .XDI XML source instrument format)
  - Dream Drumset editor software : Allows to edit drumsets (Dream .XDD XML collection of source drum set)
  - Dream Bank compiler software : Allows to import and configure dream soundbanks (Dream .XDB XML soundbank source format) and compile these to binary banks.

Figure 2: Dream 5000 SDK hardware module

In 2024, Serdaco and FMJ Software have teamed together to develop and test a solution to convert complete existing soundfonts (for example .sf2 format) into Dream compatible soundbanks. As a result, the new Awave Studio version 12.4, released in April 2024, now supports reading and writing the Dream 5000 SDK source formats. Awave Studio supports a huge number of file formats including .sf2, so this allows to convert from many current and legacy soundfont formats to Dream soundbank source.

This document explains how to convert a .sf2 soundfont into a soundbank for X2, and upload it to your X2.

Items needed to perform this workflow :

- Awave Studio software 12.4 or higher, registered version : for sale at fmjsoft.com
- Dream 5000 SDK Software and hardware bundle : for sale at Serdaco (price on request)
- X2upload utility : free download at serdaco.com/downloads
- DreamBlaster X2 or X2GS soundcard : for sale at serdashop.com

Serdaco is also planning to offer all the above together as a bundle.

2. Step by Step : Convert .SF2 into .XDB dream source file

As example, we will convert the [General User v1.471 sf2 soundfont] by Christian Collins.

We download unzip and store this file ‘GeneralUser GS v1.471.sf2’ in directory “C:\research\conversion\sf2soundfontfiles”. We will use Awave Studio to convert this into dream format.

Start up Awave Studio :

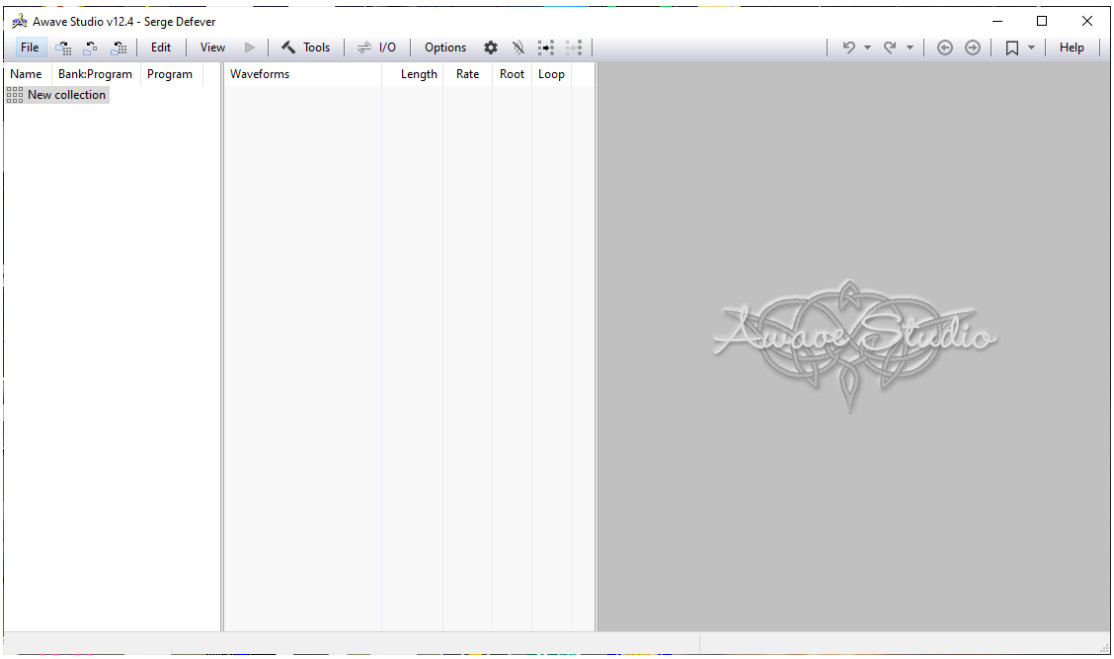

Figure 3: Awave Studio - freshly started

Using the menu File->Open File, select the .sf2 soundfont

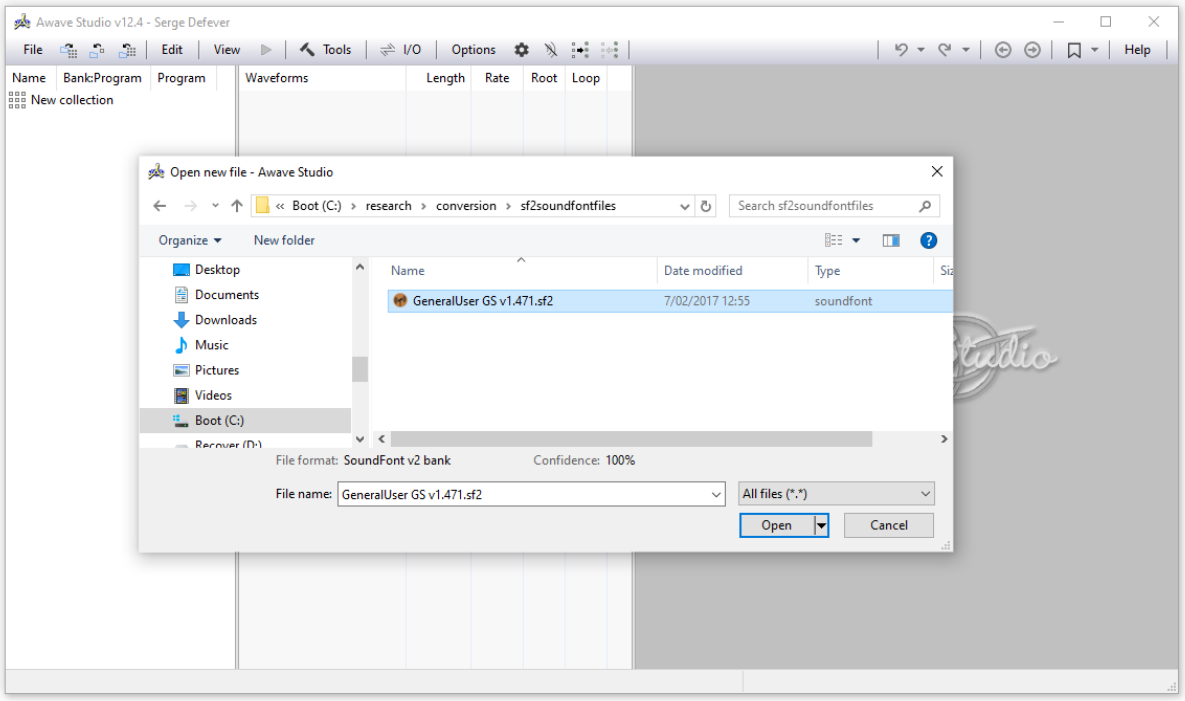

Figure 4: Awave Studio : open .sf2 soundfont

All the instruments of the soundfont will be visible in Awave Studio.

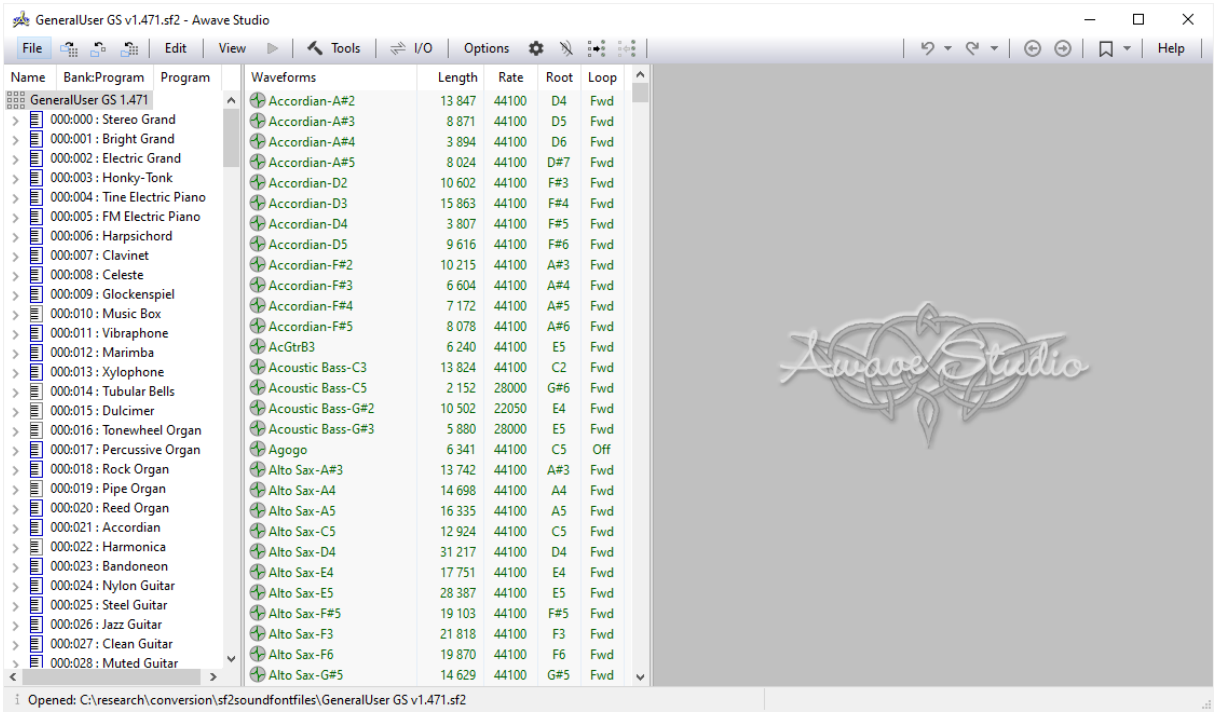

Figure 5: Awave Studio : .sf2 soundfont opened

Now go to menu File->Save Collection as… :

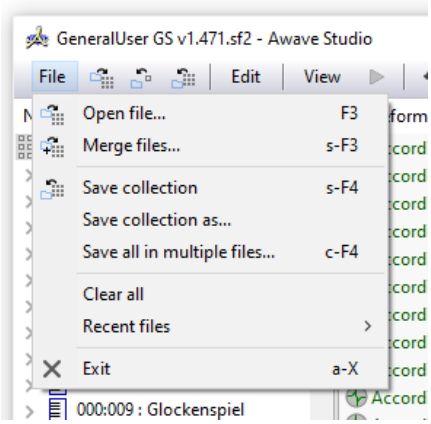

Figure 6: Awave Studio : Save Collection as...

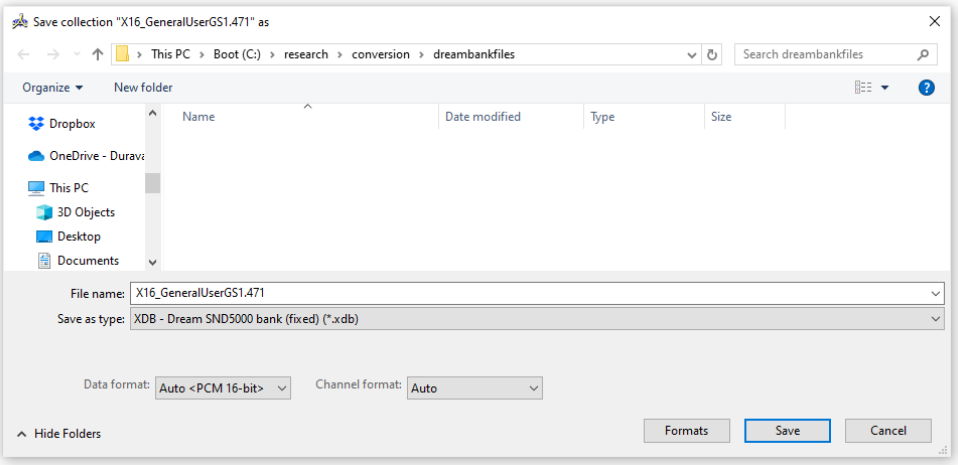

Figure 7 : Awave Studio : Save Collection as - selecting target format and path

Select the correct target path and file type : “Save as type” : XDB – Dream SND5000 bank (fixed) (*.xdb))”, and press ‘Save’.

If awave studio asks if waveforms may be modied, choose ‘modify always’ :

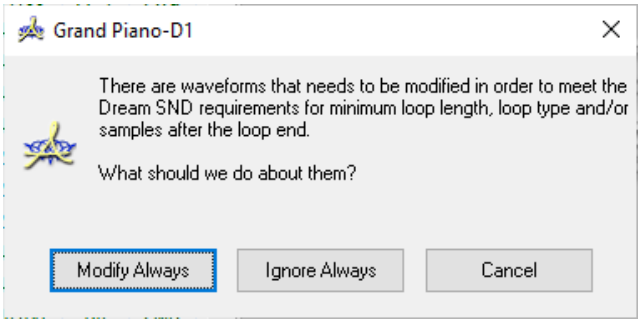

Figure 8 : Awave Studio : Modify always

Awave Studio will generate all needed Dream Source files in the target location.

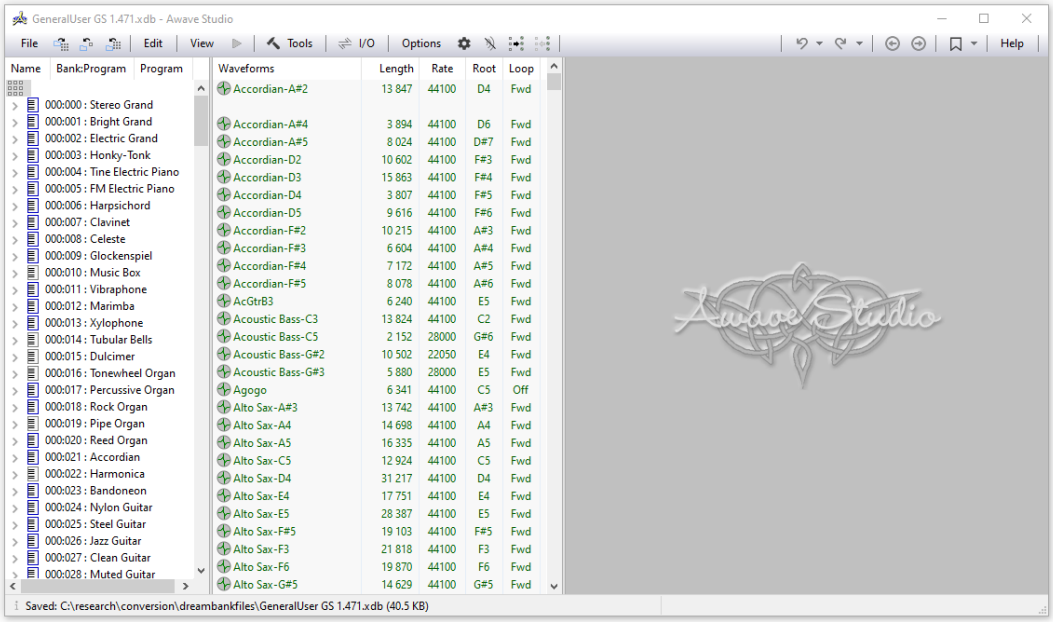

Figure 9: Awave Studio : saving Dream source files to target location

At the target location we now see the new files, nicely structured :

In the main directory, we see the .XDB file, which comprises the XML configuration of the dream bank :

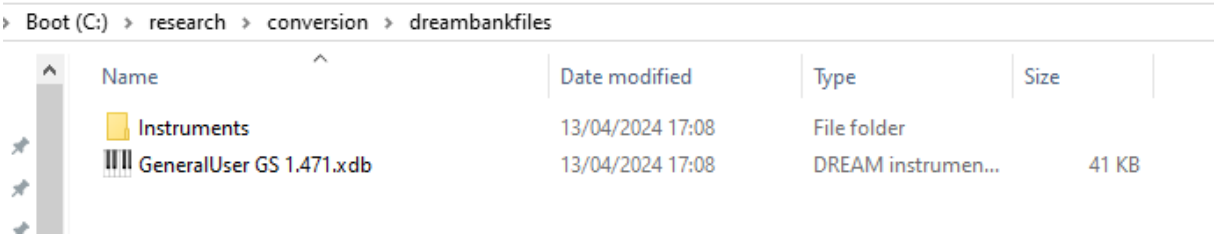

Figure 10: Dream source files :generated .XDB

In the subdirectory 'Instruments' we see all .XDI and .XDD dream instrument and drumset XML files.

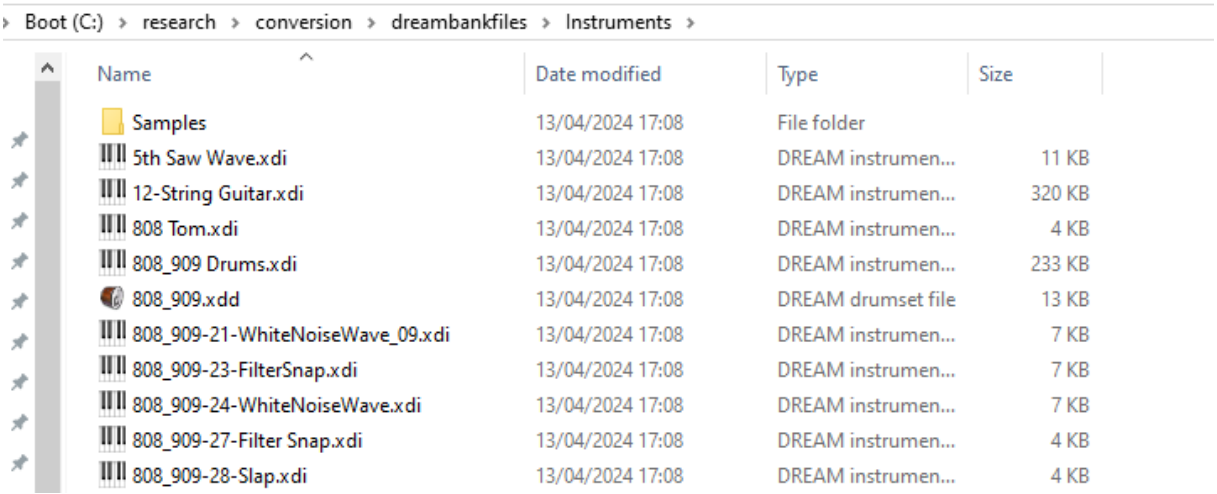

Figure 11 : Dream source files :generated .XDI and XDD files

Going one subdirectory deeper, in ‘Samples’, we see all related WAV files

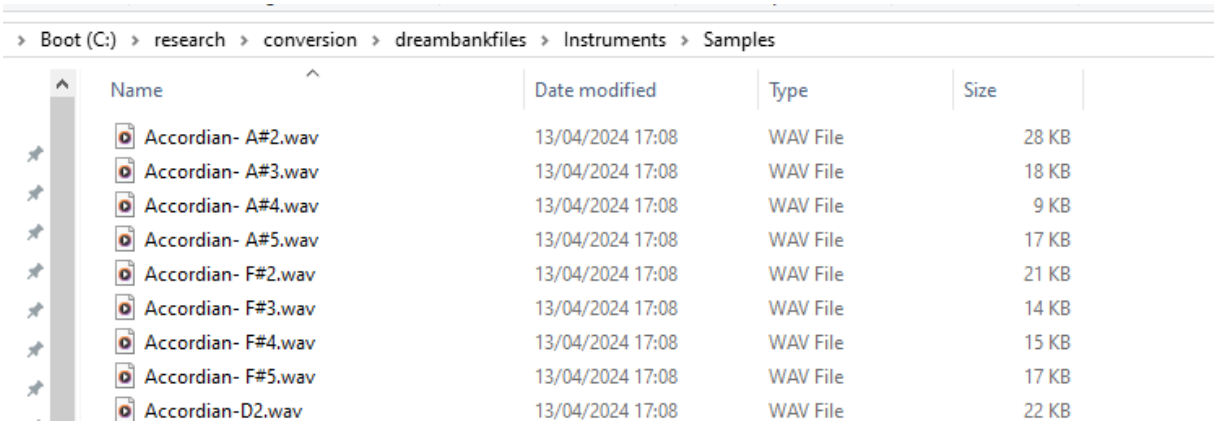

Figure 12 : Dream source files : WAV files

All these sources will be used by the 5000 SDK bank compiler, to compile into a binary file, compatible with DreamBlaster hardware.

Important Note:

Depending on the selected source bank and specific parameters it uses, the conversion result may sound slightly different from the original. Often it will be desired to further audit, manually edit and configure the instruments and bank using the 5000 SDK software and hardware. This is beyond the scope of this document. For simplicity, we will use the converted source files, without modification.

3. Step by Step : compiling the binary DreamBlaster Bank.

Start up the Dream Bank Compiler SE software

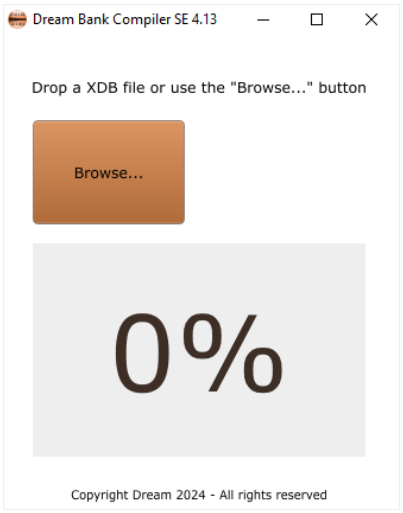

Figure 13: Dream Bank Compiler SE started

Press the 'Browse…' button and select the generated .xdb file.

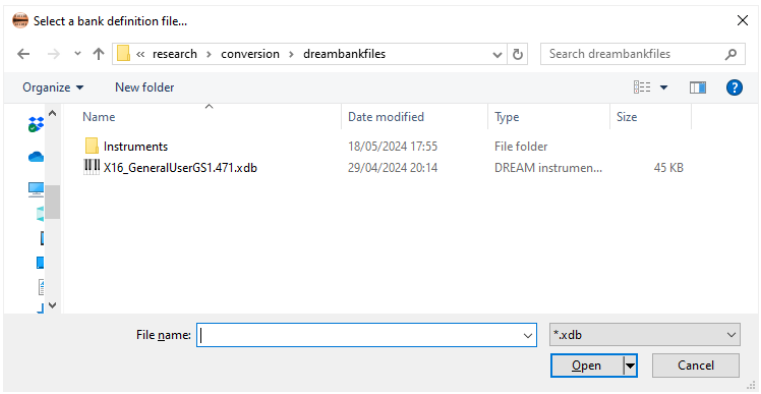

Figure 14: Dream Bank Compiler SE : select .XDB file

The bank compiler will immediatley start compiling your soundbank, this may take a few minutes.

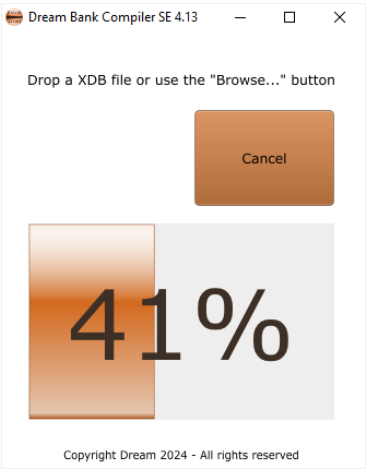

Figure 15: Dream Bank Compiler SE: Compiling bank

It takes a minute to compile the bank, the compiler goes through different compilation phases.

When the compiler is ready, it shows a popup saying ‘Compilation completed succesfully’.

Figure 16: Dream Bank Compiler SE : compilation succesful

If all went well, the bin file is ready at the target location.

At the target location, we see several new files generated by the compiler.

For example : GeneralUser GS 1.471_Midi.txt shows us the program change numbers (instruments) available :

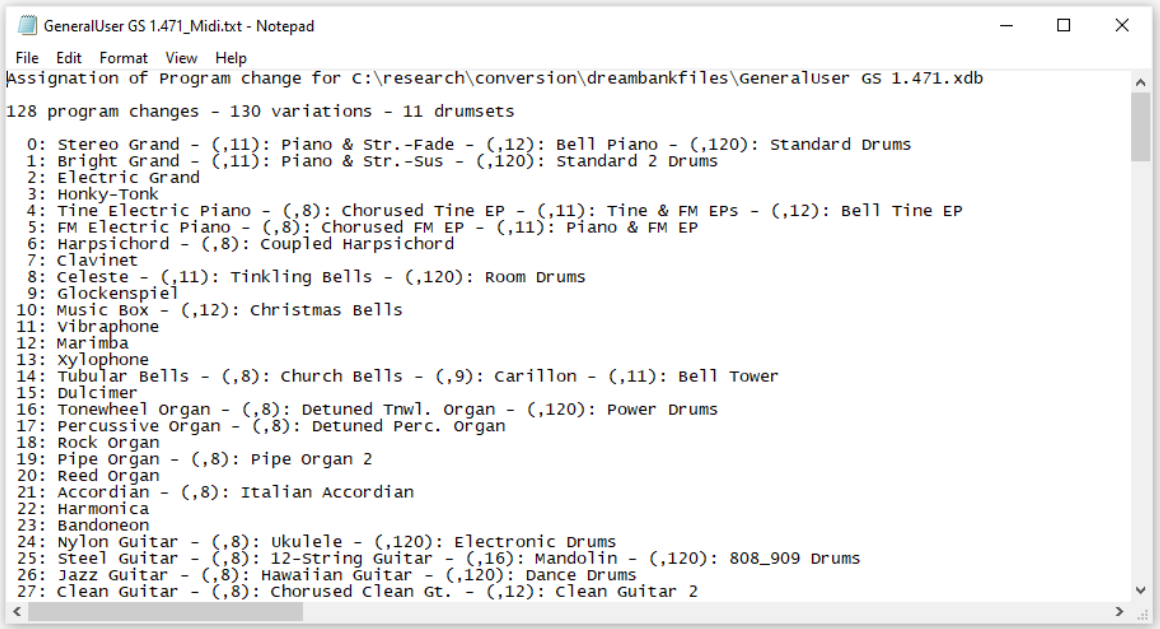

Figure 17: Dream Bank Compiler : Program change numbers

The most important file is the large .bin file (GeneralUser GS 1.471.bin in our example).
This file can be uploaded to the DreamBlaster X2, but it needs to be renamed to .dxb extension.

This is easy to achieve in windows explorer.

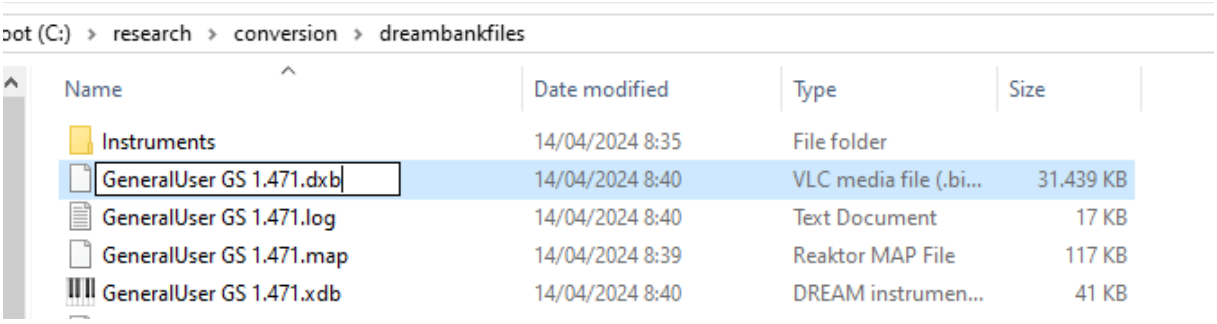

Figure 18: Renaming the .bin file to .dxb

This GeneralUser GS 1.471.dxb file now can be uploaded to DreamBlaster X2.

4. Uploading the compiled bank to DreamBlaster X2

Connect your DreamBlaster X2 (or X2GS) to your computer using an USB cable.

Start the X2Upload utility and select DreamBlaster X2 as OUT and IN device.

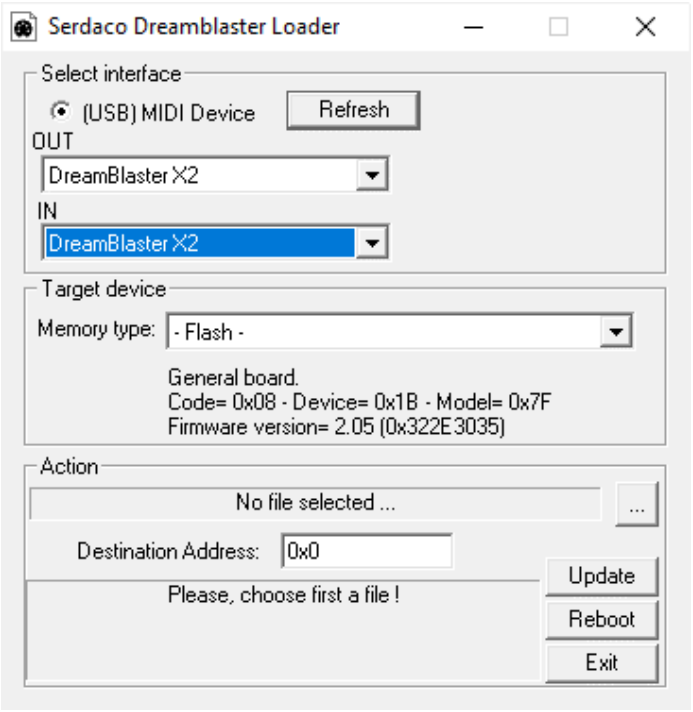

Figure 19: X2 upload – with DreamBlaster X2 selected as IN and OUT device

Press '…' to select the bin file of the Dream soundbank you wish to upload (GeneralUser GS 1.471.bin).

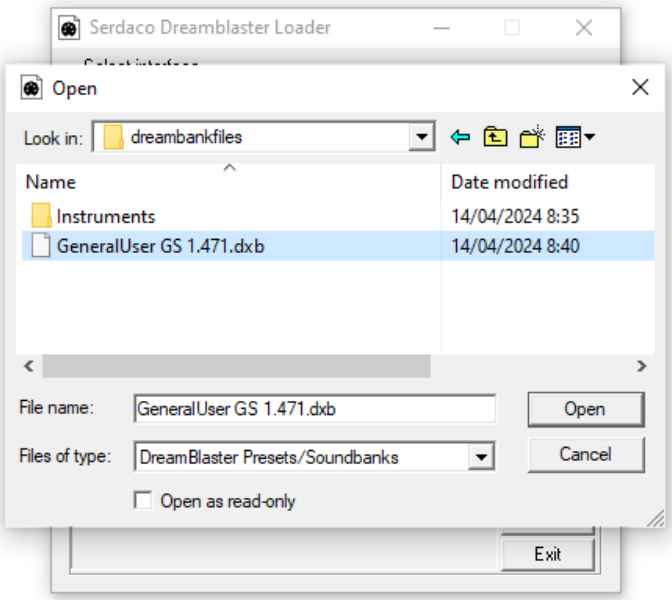

Figure 20: X2Upload : Selecting .dxb bank file

Press the 'Open' button. Now the correct bank is selected.

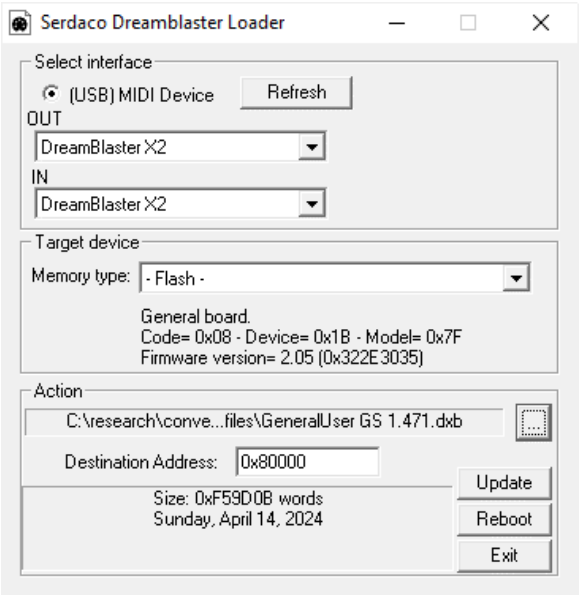

Figure 21: X2Upload : Bank file chosen.

Now press the 'update' button.

The software will first erase the flash, and then update to the new bank.

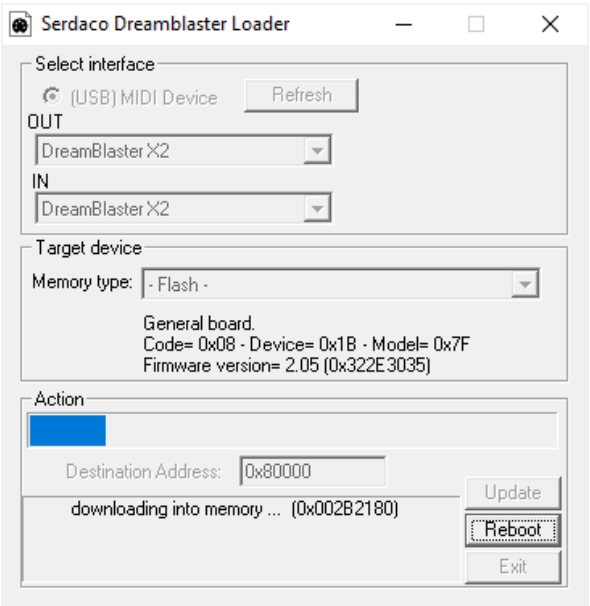

Figure 22: X2Upload : downloading into memory

This upload will take several minutes (depending on the size of the bank).

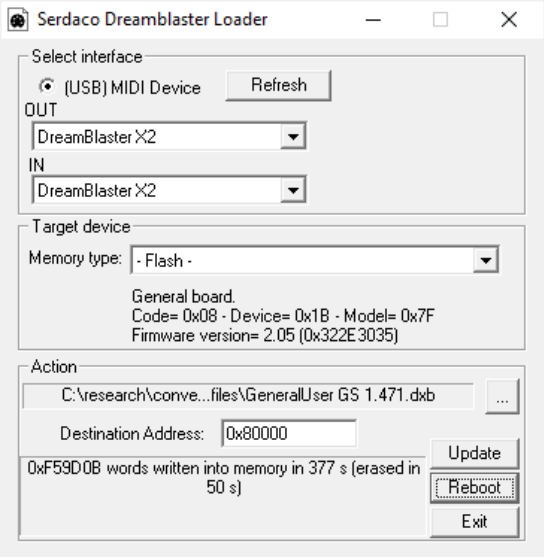

Figure 23: X2Upload : Upload ready

When the upload is ready, X2Upload will tell how much time it took. In this example (for a 34 mbyte bank), it took 50 seconds to erase the flash, and 377 seconds to program the new bank. So the total time was 427 seconds, or somewhat more than 7 minutes.

Your X2 is now updated to the new bank, and is ready to use. For example plugged on a CHiLL MIDI interface, you can play midi files through your MIDI interface.

Figure 24: X2 mounted on CHiLL interface

Notes :

- On X2GS it is needed to send a MIDI Command to switch from the GS bank to the user bank, after power cycle. X2GS also only has 48 mbyte of user programmable flash, as opposed to 64 mbyte on X2.
- If you want to upload faster, or have room for larger banks, it is recommended to have a look at the X16 or X8 products, which much have larger flash memory (1gbyte) and a much faster upload speed (seconds instead of minutes).

Good luck, and thanks for following this guide.

If you have any questions, you can contact me at https://www.serdashop.com/Contact