

Converting SF2 soundfonts to X16 DreamBlaster soundbanks : a step by step guide.

Rev 1, April 13th 2024, Serdaco BV

1. Introduction

The DreamBlaster [X16/X16GS](#) is an advanced synthesizer module based on the Dream SAM5716B 16 core DSP chip. This module has 1 Gbyte of flash memory to which you can upload binary soundbanks in using the X16Manager utility.



Figure 1: DreamBlaster X16

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 X16, and upload it to your X16.

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)
- X16Manager upload utility : free download at serdaco.com/downloads
- DreamBlaster [X16](#), [X16GS](#) or [X8](#) 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\sfontfiles". We will use Awave Studio to convert this into dream format.

Start up Awave Studio :

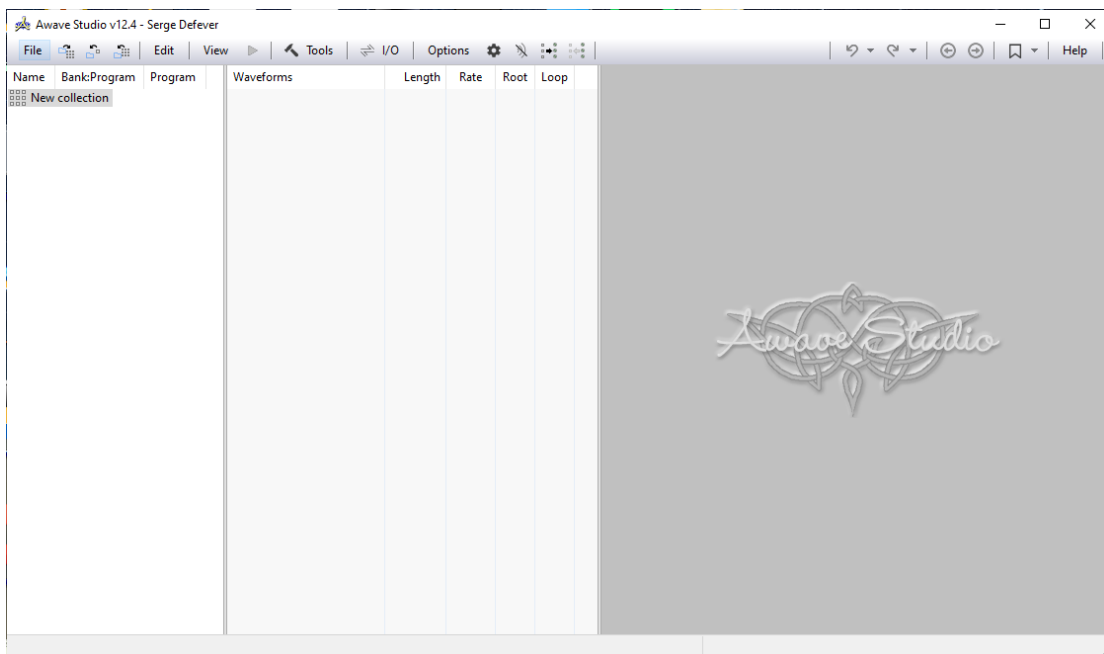


Figure 3: Awave Studio - freshly started

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

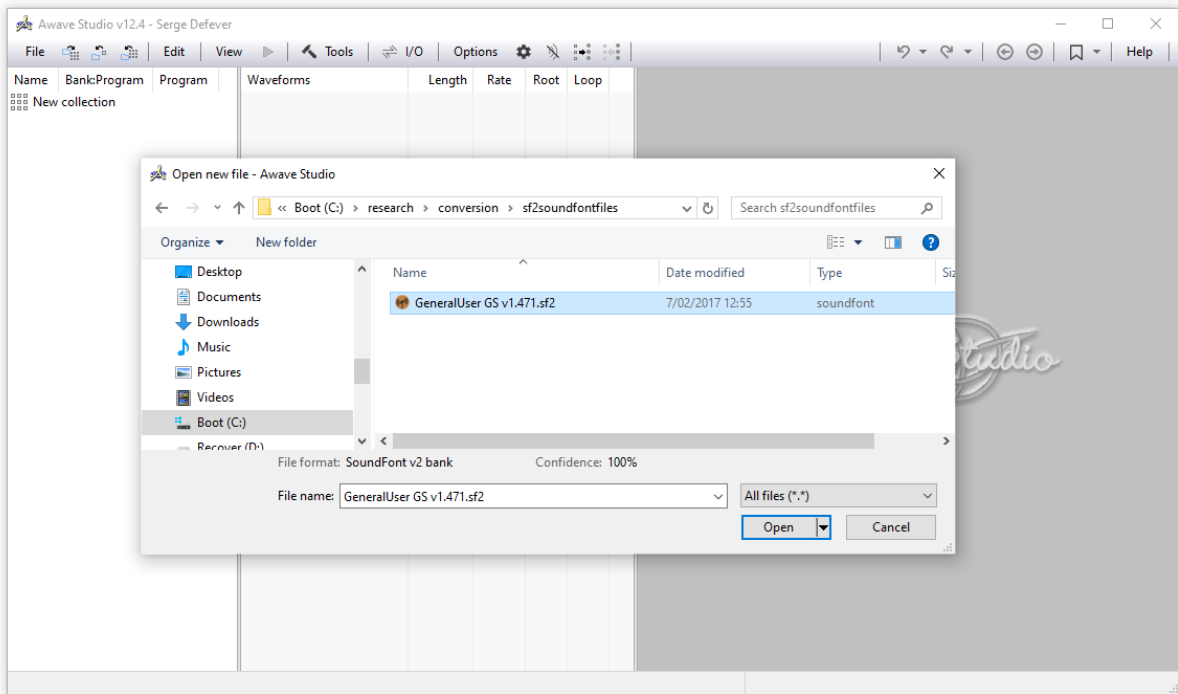


Figure 4: Awave Studio : open .sf2 soundfont

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

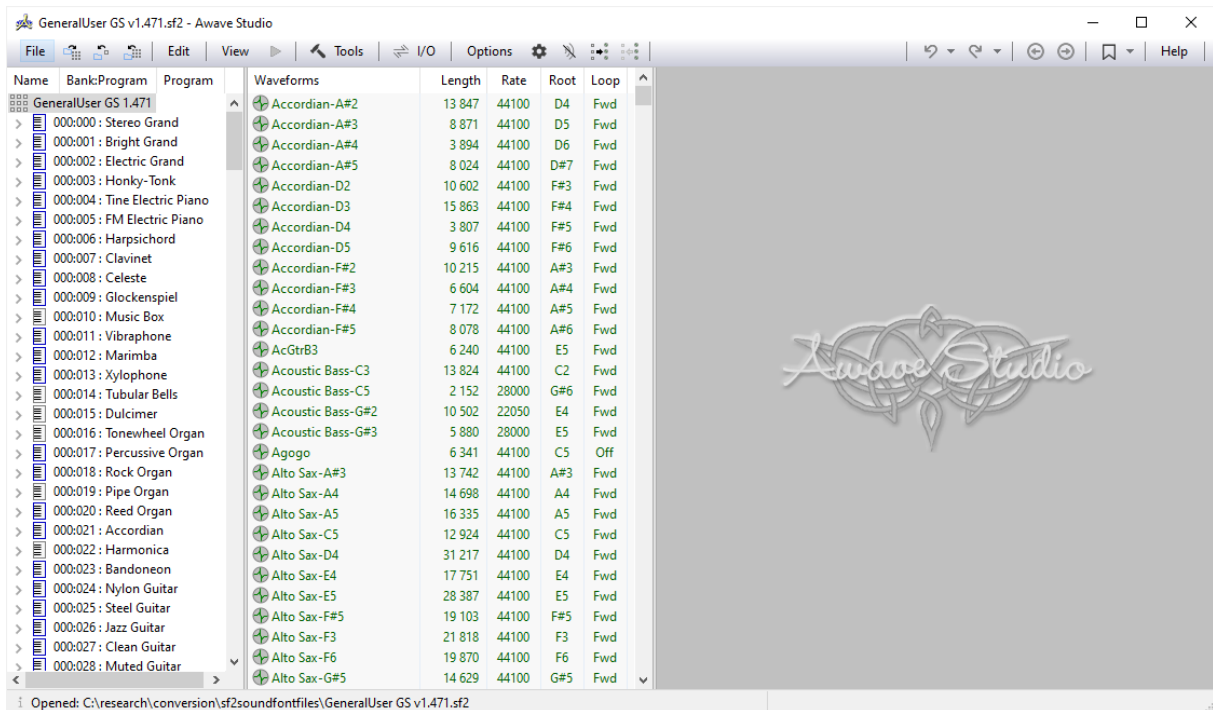


Figure 5: Awave Studio : .sf2 soundfont opened

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

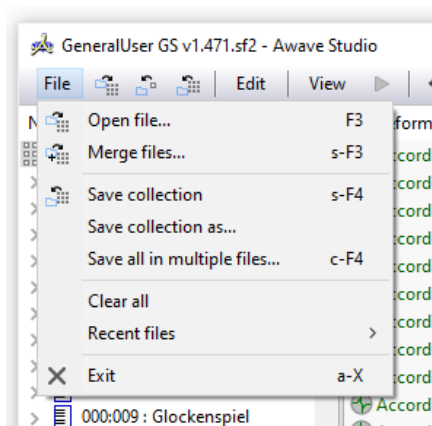


Figure 6: Awave Studio : Save Collection as...

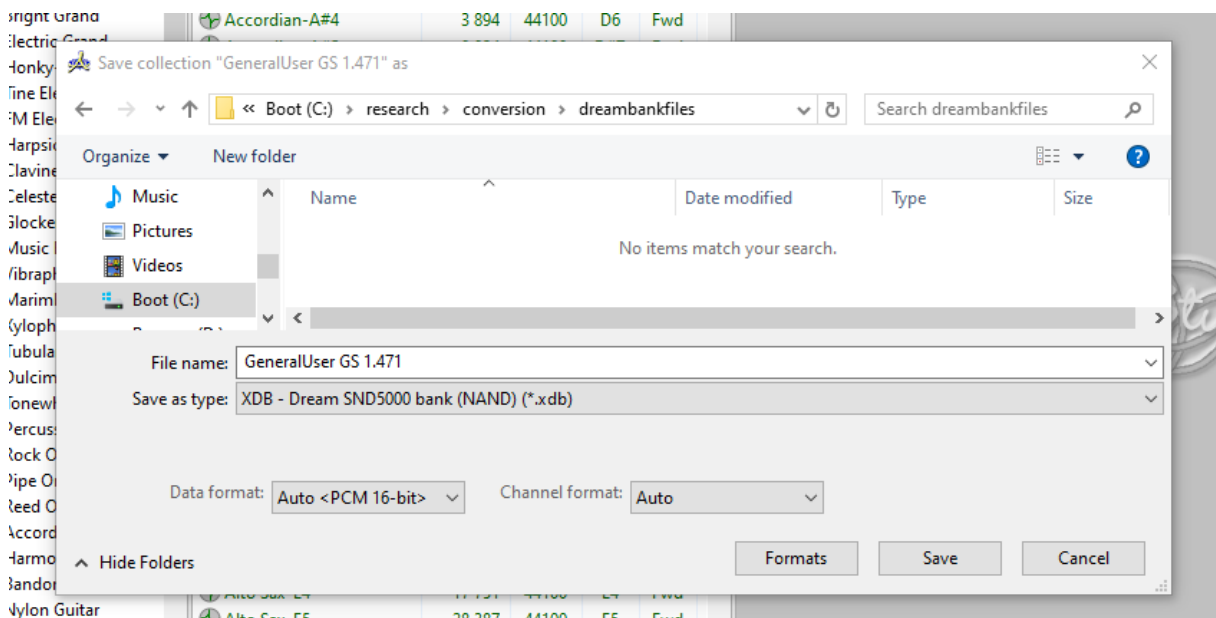


Figure 7: Awave Studio : Save Collection as - selecting target format and path

Select the correct target path and file type : "Save as type" : XDB – Dream SND5000 bank (NAND) (*.xdb)", and press 'Save'.

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

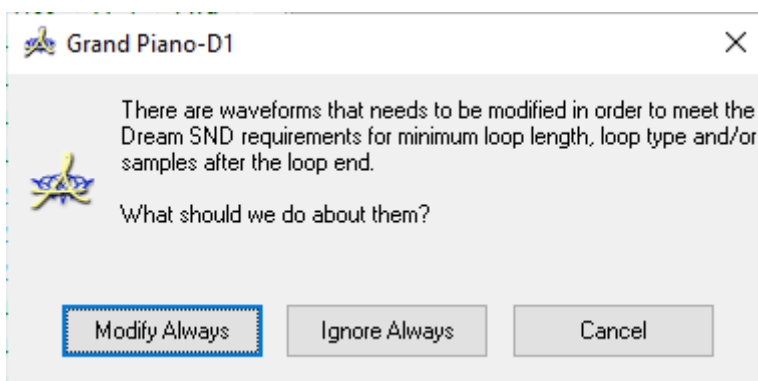


Figure 8 : Awave Studio : Modify always

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

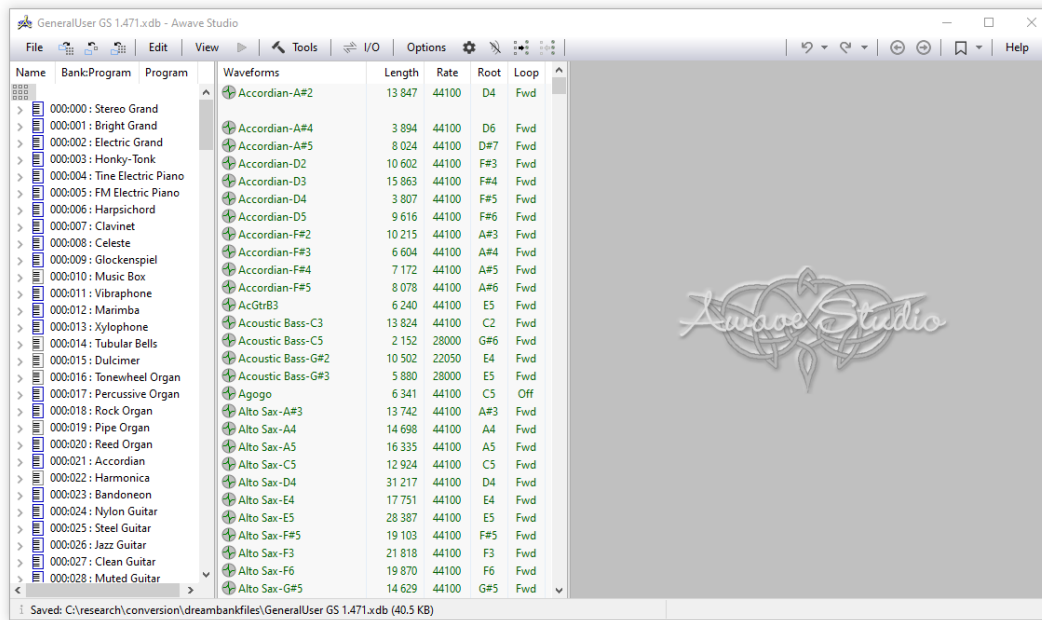


Figure 9: Awave Studio : saving Dream source files to target location

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 :

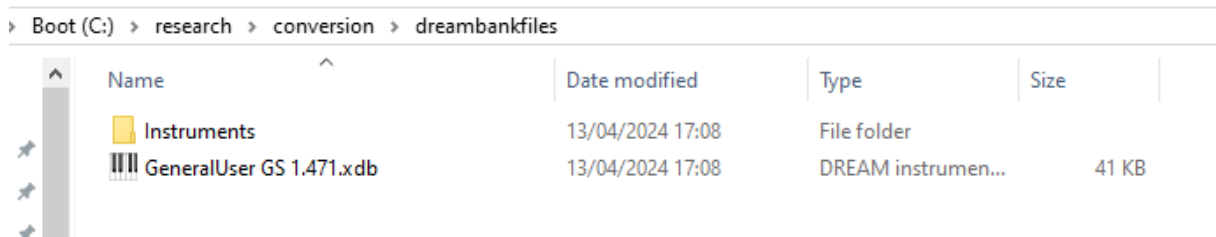


Figure 10: Dream source files :generated .XDB

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

» Boot (C:) » research » conversion » dreambankfiles » Instruments »

Name	Date modified	Type	Size
Samples	13/04/2024 17:08	File folder	
5th Saw Wave.xdi	13/04/2024 17:08	DREAM instrumen...	11 KB
12-String Guitar.xdi	13/04/2024 17:08	DREAM instrumen...	320 KB
808 Tom.xdi	13/04/2024 17:08	DREAM instrumen...	4 KB
808_909 Drums.xdi	13/04/2024 17:08	DREAM instrumen...	233 KB
808_909.xdd	13/04/2024 17:08	DREAM drumset file	13 KB
808_909-21-WhiteNoiseWave_09.xdi	13/04/2024 17:08	DREAM instrumen...	7 KB
808_909-23-FilterSnap.xdi	13/04/2024 17:08	DREAM instrumen...	7 KB
808_909-24-WhiteNoiseWave.xdi	13/04/2024 17:08	DREAM instrumen...	7 KB
808_909-27-Filter Snap.xdi	13/04/2024 17:08	DREAM instrumen...	4 KB
808_909-28-Slap.xdi	13/04/2024 17:08	DREAM instrumen...	4 KB

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

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

» Boot (C:) » research » conversion » dreambankfiles » Instruments » Samples

Name	Date modified	Type	Size
Accordian- A#2.wav	13/04/2024 17:08	WAV File	28 KB
Accordian- A#3.wav	13/04/2024 17:08	WAV File	18 KB
Accordian- A#4.wav	13/04/2024 17:08	WAV File	9 KB
Accordian- A#5.wav	13/04/2024 17:08	WAV File	17 KB
Accordian- F#2.wav	13/04/2024 17:08	WAV File	21 KB
Accordian- F#3.wav	13/04/2024 17:08	WAV File	14 KB
Accordian- F#4.wav	13/04/2024 17:08	WAV File	15 KB
Accordian- F#5.wav	13/04/2024 17:08	WAV File	17 KB
Accordian-D2.wav	13/04/2024 17:08	WAV File	22 KB

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 software (from Dream 5000 SDK).

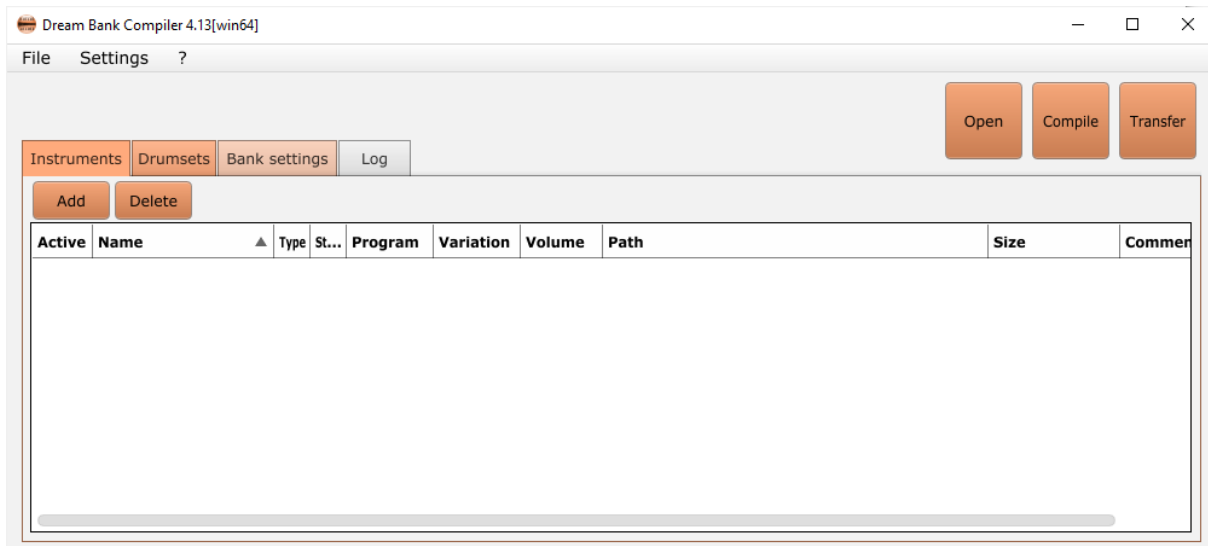


Figure 13: Dream Bank Compiler started (5000 SDK)

In the File menu, select 'open bank definition file' and select the generated .xdb file and press Open

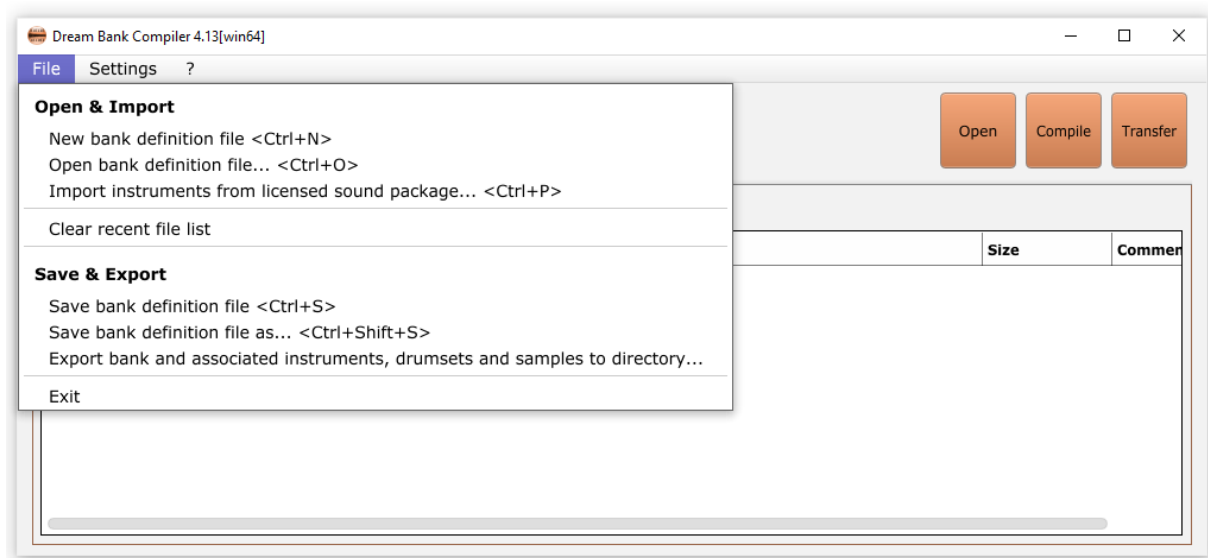


Figure 14 : Dream Bank Compiler : Open bank definition file

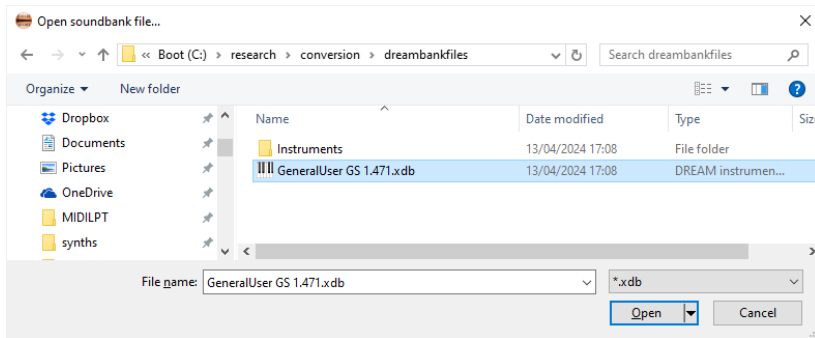


Figure 15: Dream Bank Compiler : Select .xdb file

The bank compiler will show the instruments in your bank. The other tabs show the Drumsets and Bank settings.

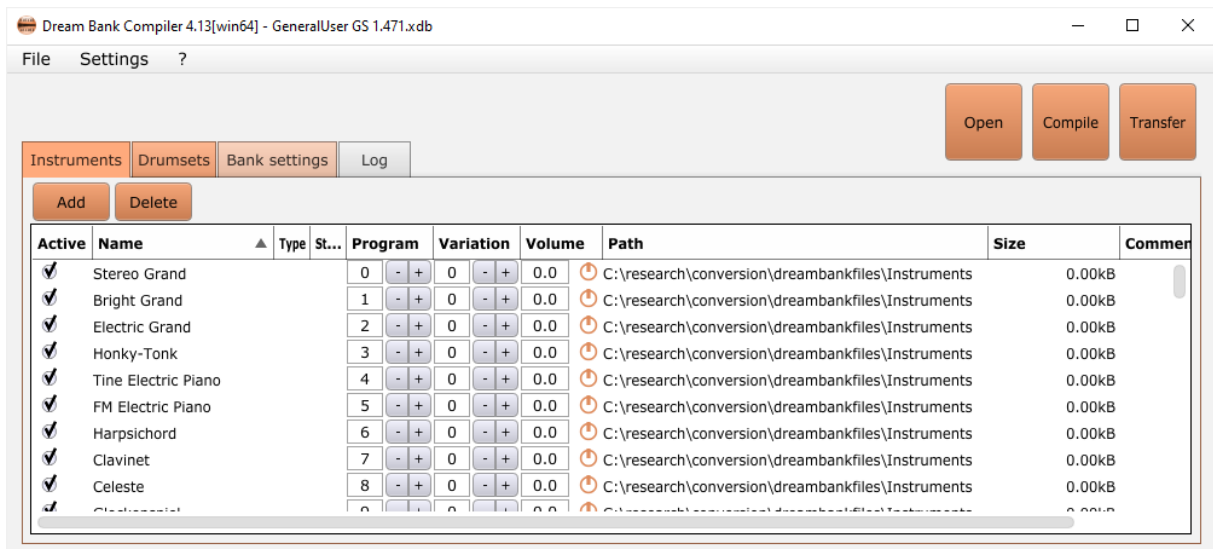


Figure 16: Dream Bank Compiler : Instruments

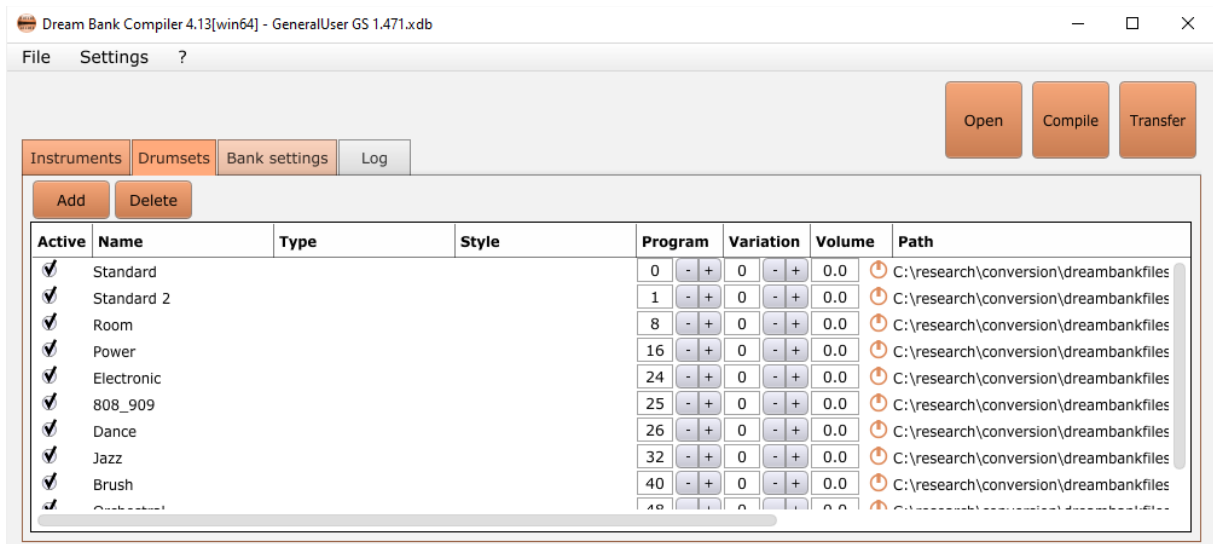


Figure 17: Dream Bank Compiler : Drumsets

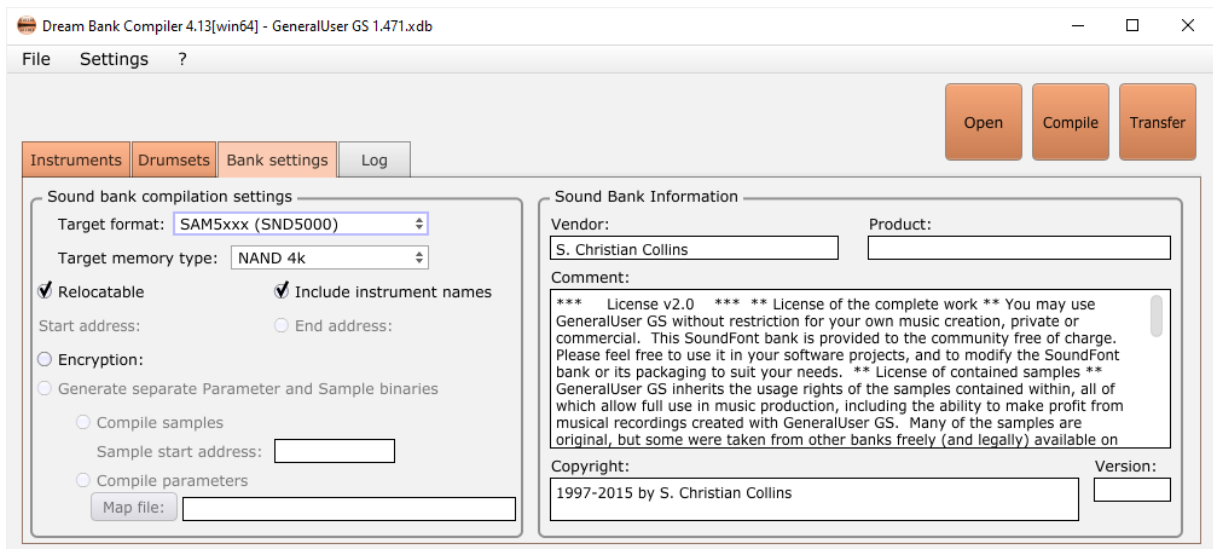


Figure 18: Dream Bank Compiler : Bank Settings

All bank settings are ready to use for DreamBlaster X16. Press 'Compile' to compile the bank.

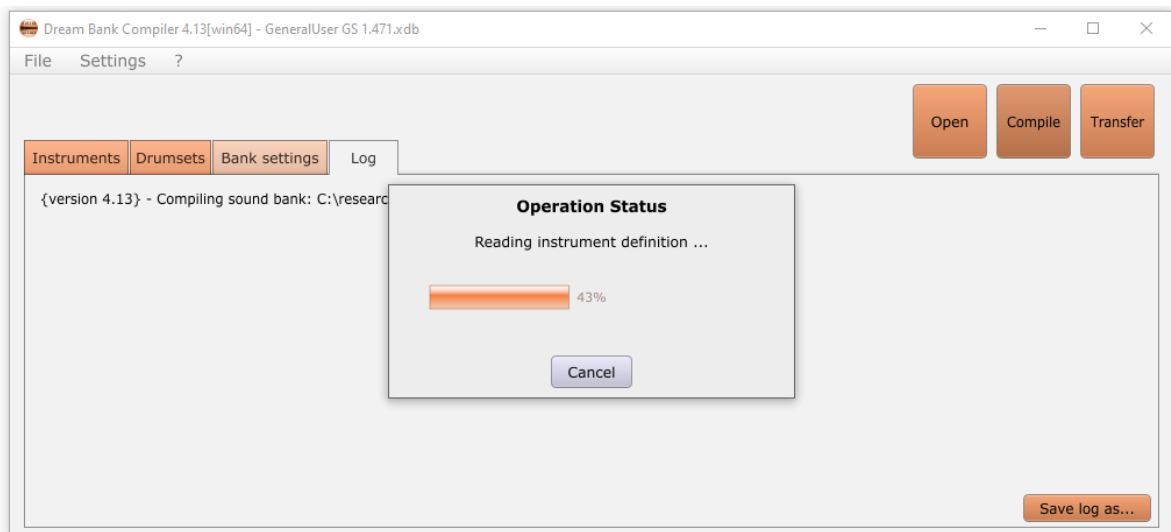


Figure 19: Dream Bank Compiler : Compiling bank

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

When the compiler is ready, it shows the compilation log. If all went well, the bin file is ready at the target location.



Figure 20: Dream Bank Compiler : compilation ready

At the target location, we see several new files : compiler log files, and other interesting information.

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

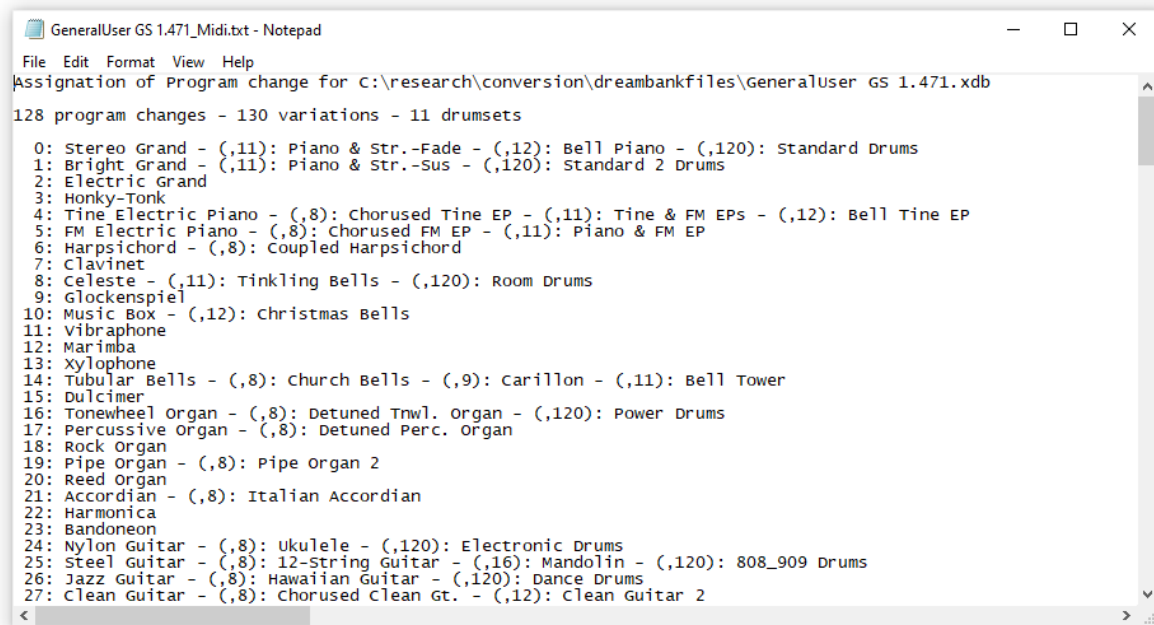


Figure 21: Dream Bank Compiler : Program change numbers

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 X16.

4. Uploading the compiled bank to DreamBlaster X16

Connect your DreamBlaster X16 (or X16GS or X8) to your computer using a USB cable.

Start the X16Manager utility.

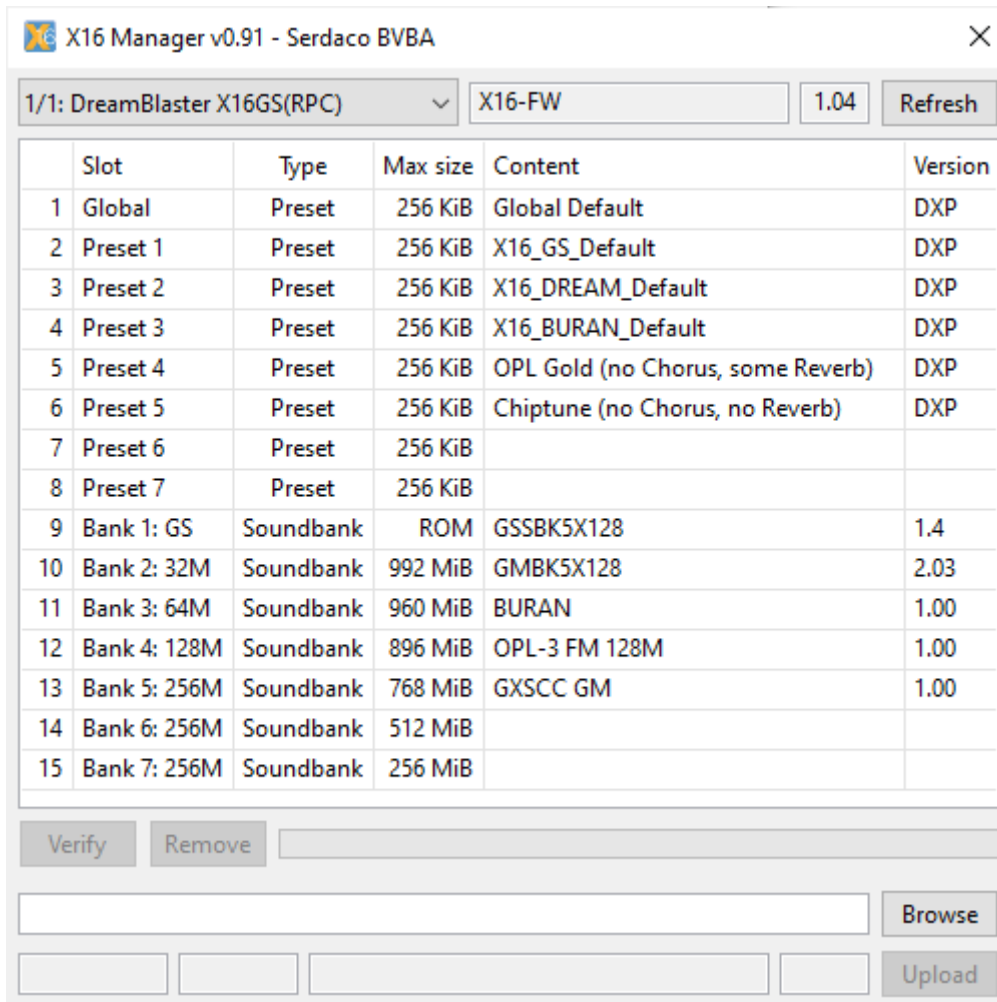


Figure 22: X16 manager – started

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

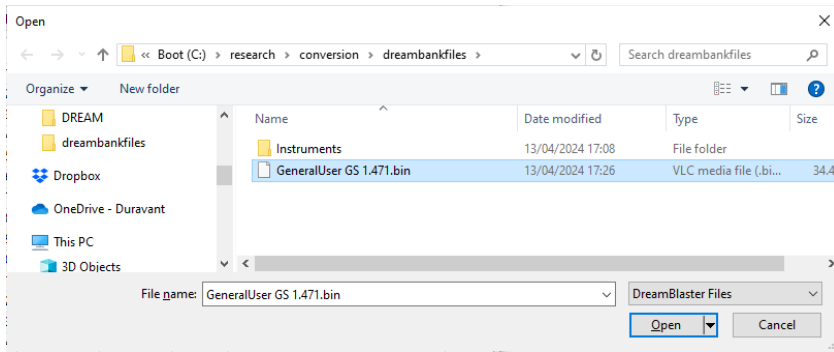


Figure 23: X16 Manager : Selecting bin file

Now **highlight** the bank slot to where you wish to upload the bank to.

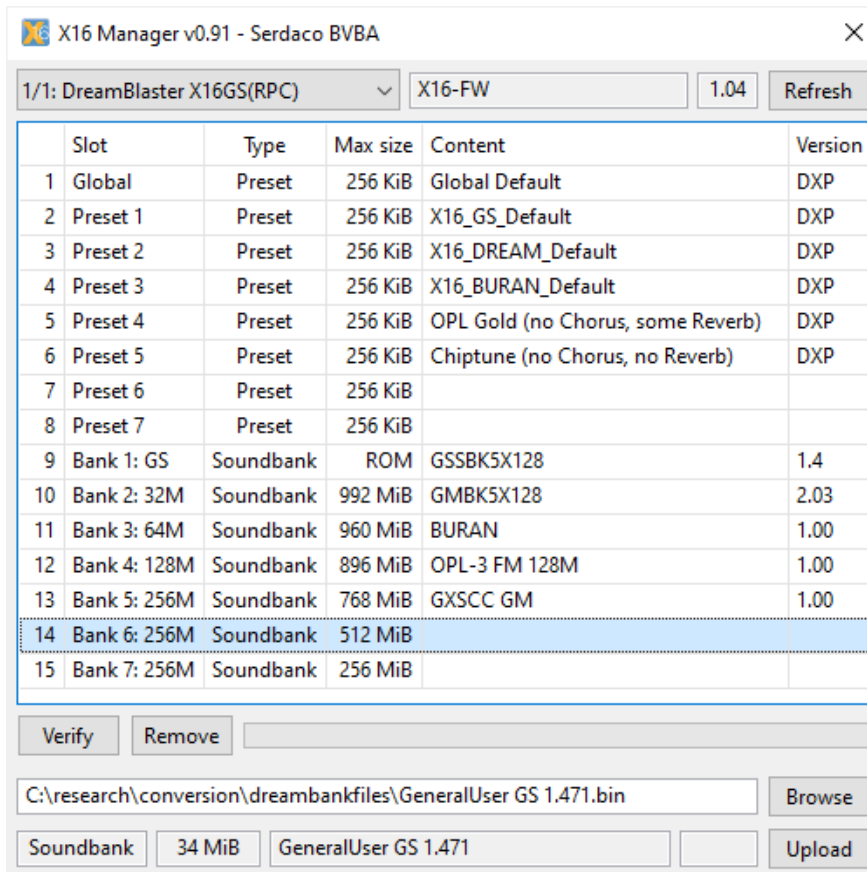


Figure 24: X16 Manager: **highlight** slot 6

In our example we upload the bank to slot 6. The value in the ‘max size’ column will tell you the absolute maximum size of the bank that can be loaded on this position. In our example this is 512 mbyte. As Bank 6 is only 256 mbyte, when loading a file larger than 256 mbyte, it will occupy both bank 6 and bank 7 memory. In our case, the bank will only occupy slot 6, as our bank is only 34 mbyte.

Press ‘upload’ to upload the bank, this save a few seconds.

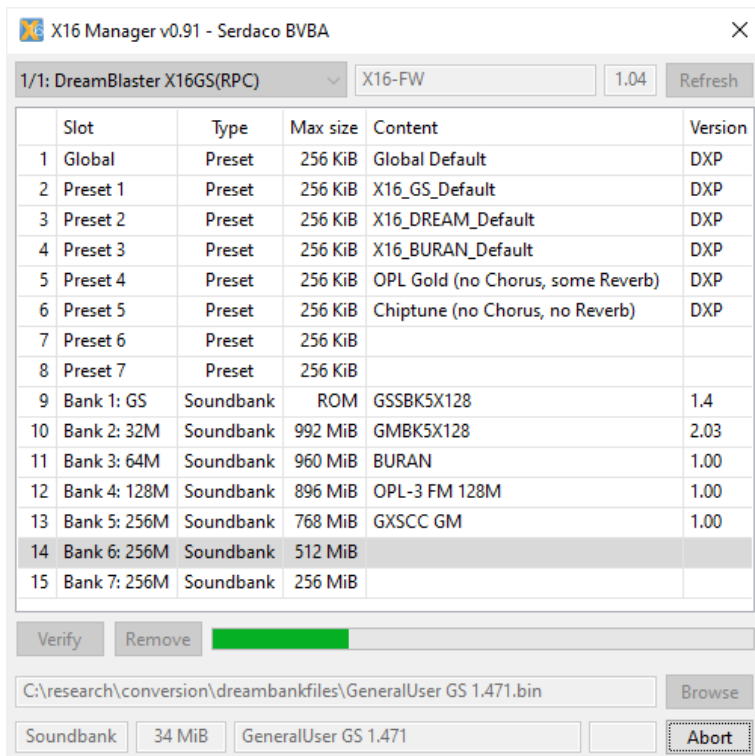


Figure 25: X16 Manager : uploading the bank

When the bank is uploaded, you will see it listed in X16Manager.

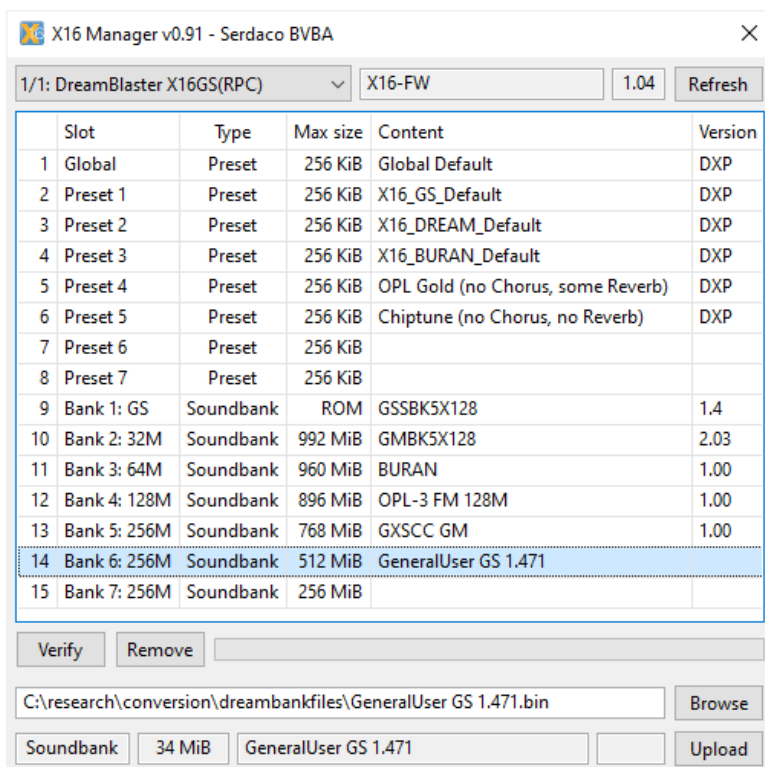


Figure 26: X16 Manager: bank is uploaded

The bank is now ready to be used on X16, at slot 6. X16 can switch between different banks by using midi commands. (see [X16 User Manual](#))

To make sure the X16 uses slot 6 from bootup, you can set the dip switch to 6 (binary 0110) :

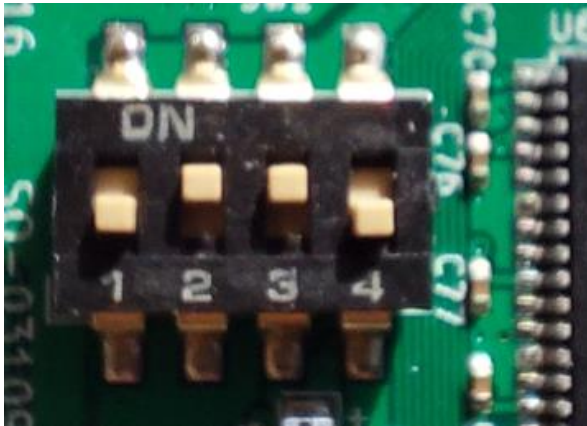


Figure 27; X16 : Dip switch set to 6 (0110) to make it use slot 6 bank at bootup

Your X16 is now ready to use your new bank in your application. For example plugged on a [CHILL interface](#), you can play midi files through your MIDI interface.



Figure 28: X16 in CHILL interface, ready to use, dipo configured at slot 6

Good luck, and thanks for following this guide.

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