FireNFX Documentation

Updated July 16, 2021

Description:
FireNFX is a project to enhance the existing AKAI Fire functionality with easy to use MIDI Looper style helper functions. As such, many of these features are already available to AKAI Fire users, except not all in one place and not designed with MIDI looping in mind.

A MIDI Looper is simply a way of recording yourself performing and controlling multiple tracks on a MIDI device - most commonly a MIDI piano style keyboard. The benefit of the Looper aspect is that it will repeat the parts for you allowing you to add and remove other parts.

You can record up to 8 parts and control various parameters in real time. Most of the original AKAI Fire functions will still be available as well.

One of the goals of this project is to use grouping, coloring and feedback along with the display on the AKAI Fire to use many of FL Studio' s functions without having to interact directly in FL Studio with the mouse or keyboard.

As an example, once everything is loaded you could turn off the monitor and use FL via the fire device to:

- Set a custom tapped tempo
- Select the first instrument
- Set a harmonic scale so you cant hit a wrong note!
- Press record to capture a performance and loop it
- Manipulate the volume and pan
- Delete it if you didn't like it
- Change to other channels quickly and confidently
- Set the snap level to allow quantize on your recording
- Move to other patterns or instruments and record other parts
- mute /unmute parts to add interest.
- So much more!

As of this writing this project is still in pre-release and should not be used in a live production environment without understanding the risks of crashes. Thank you for understanding.

Pre-requisites:
AKAI Fire Hardware, FL Studio 20.8+

This document presumes you have basic abilities with the AKAI Fire device as well as some basic skills to download, extract, copy files and create folders as needed.

Installation:

The first step is to download the files. I recommend you download the Installer file if you are on Windows.

This can be done from the github file release: https://github.com/nfxbeats/FireNFX/releases

[Link to Windows Installer] - The installer will assist in getting the proper folder names for your setup.

[Link to Zipped Source Code] - You will need to decompress (UNZIP) and copy the files manually

The ZIP file needs to be extracted into a specific folder below your user data folder.

Using my own User Data Folder of "**C:\Users\NFXBeats**", my Hardware folder is located here:

C:\Users\NFXBeats\FL Studio\Settings\Hardware

Notice all I did was add "\FL Studio\Settings\Hardware". That should be all you need to do with your own user data folder.

Inside of the Hardware folder, create a new folder named **FireNFX** and put all the downloaded/extracted files into that folder.

Using my example from above, my own folder would be:

C:\Users\NFXBeats\FL Studio\Settings\Hardware\FireNFX

Testing the install:
Now that the files have been copied to the correct folder, we need to validate that everything works.

Please follow these instructions in order (video walk through here xxx.xxx.x):

1) Start FL Studio.

2) Open the file located under your “FL Studio\Settings\Hardware\FireNFX” named FireNFX_Template.zip (open the ZIP directly in FL, do not extract it first). It may take a second but should open a project with 8-10 colored pattern tracks. You might need to adjust the playlist zoom to see them all at once.

3) Go to Options>MIDI . You should already have the Fire setup as a device.

4) Select the Fire Device and click the dropdown for the device list. On the right column should be listed **FireNFX (user)**. Select it. You might see the device reset itself.

5) From the FL Studio main menu, go to View>Script Output. This should open a window that has a tab named AKAI Fire. and it should indicate the location of the script it is running and any errors (if any). The location of the script displayed should be in the folder youcopied the files into.

For example mine would display:

FL Studio Midi scripting version: 11
“C:\Users\NFXBeats\FL Studio\Settings\Hardware\FireNFX\device_Fire.py” found

init ok

At this point if there are no errors you should only need to activate it.

Activating:
To activate the feature set you must perform two steps:

1) Put the FIREFIRE
2) into drum mode. Specifically the mode with the FPC pads to the left. This should be the default drum mode when the “Drum” button is pressed.

3) On the FIRE, hold the ALT button and press the 9th pad on the bottom row. This should be the first unlit pad of the bottom row for easy identification.

At this point the mod is active and the pads should have changed colors and are ready for use.

If you left the Script Output window open, you should see that it displays some information about the current track. It builds a set of links between the VST channels, patterns and mixer channels.

```
init ok
_____________nfxFIRE v 0.0.pre-alpha - warbeats.com_____________
...Initializing
...Syncing FL Track data
......Pattern # 1 Lead -->Channel # 0 Lead -->Mixer # 1 1 Lead
......Pattern # 2 Pad -->Channel # 1 Pad -->Mixer # 2 2 Pad
......Pattern # 3 Rhythm -->Channel # 2 Rhythm -->Mixer # 3 3 Rhythm
......Pattern # 4 Sub -->Channel # 3 Sub -->Mixer # 4 4 Sub
......Pattern # 5 Keys -->Channel # 4 Keys -->Mixer # 5 5 Keys
......Pattern # 6 Guitar -->Channel # 5 Guitar -->Mixer # 6 6 Guitar
......Pattern # 7 Bass -->Channel # 6 Bass -->Mixer # 7 7 Bass
......Pattern # 8 FPC -->Channel # 7 FPC -->Mixer # 8 8 FPC
......Pattern # 9 ...FPC Short -->Channel # 7 FPC -->Mixer # 8 8 FPC
......Pattern # 10 ...FPC Long -->Channel # 7 FPC -->Mixer # 8 8 FPC
...Syncing FIRE Pad data
...Reset PadMaps
_____________________________________________________________
Channel Mode
```

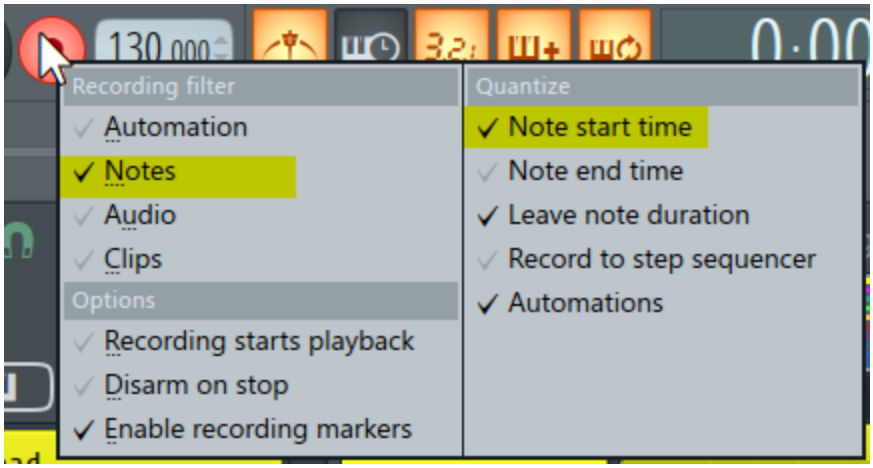
Project Requirements and Philosophy:
The FireNFX requires a project with 8 channels, 8 patterns and 8 mixer tracks all linked from channels->patterns->mixer in a 1 to 1 way. (mostly... more on this later)

In short, 1 channel must be linked to only 1 mixer track and used in only 1 pattern. The color coding is also important as these links are represented by a single color. Following this will yield the best performance and reliability.

See how this is done in the supplied demo ZIP project template. After opening the ZIP in FL studio, you might consider saving it to a new location of your choosing.

You can set your pattern lengths to any size. 4 or 8 bars is typical, but FL will always use the longest pattern, so be aware of that. I also recommend you set your pattern lengths in sensible sizes and even numbered lengths.

Recording Filters:
In FL Studio, if you right-click on the record button you should set the following:



You should have “Notes” and “Note Start Time” checked.

I highly recommend you uncheck “Automation” for now. Fire NFX cannot currently delete the automation. If you leave it on, you will have to manually delete it between sessions.

Updating after changes:
If you were to make changes to the FL Project, say you changed colors and/or names. You should bring up the Script Output window (View>Script Output) and click the reload button to reload everything. Re-activate and your changes will be applied, the pattern pads colors will update, etc..

FireNFX Layout:
Before using the FireNFX, lets understand the layout:

There are 5 sections.

Performance:

This area will allow you to play the active Pattern Track by sending MIDI notes to FL Studio. It has two modes.

FPC Mode
If the active track has a name starting with “FPC” or “...FPC”, it will load a custom color set for FPC Pad Banks A (4x4) and B (4x4).

These simply mirror the same pads available in the FPC generator but have a custom color set.
(todo: add customizing details)

Non-FPC Mode
All other tracks (whose names do NOT start with “FPC” or “...FPC) will load a 3 octave harmonic scale on the bottom 3 rows and a chord bar across the top row of the performance area.

The 3-Octave playing pads
The 3 octave playing pads are the bottom 3 rows of the Performance section and will look the same as each other. These are programmed to play the notes within a specific harmonic scale/mode that will be detailed later.

[Pic here]

Using the 3-octaves
You have access to the 3 octaves by pressing the pads in the section to trigger a single note.

Each row from the bottom up, represents the start of an octave. The light blue pads are root notes. The white pads are the other notes increasing in pitch as they move left to right until the end of the row. Even if there are fewer than 8 notes in a scale, the scale will continue to build to fill the row

These pads will only play notes within the selected harmonic scale.

[PIC]

You can press a single pad to play a single note or make your own chords by playing multiple pads in this area.

- **[Macro #1]** will cycle through the velocity ranges (how hard the pad was pressed). The FIRE pads don’t have true velocity sensitivity so this helps you to set your value here. The displayed value represents the velocity.

- **[Macro #2]** will cycle through the octave ranges. The displayed value represents the lowest octave row.

The Chord Bar

The chord bar will play the chords based on triads (1-3-5) in the harmonic scale/mode. The first pad will play the first (or I / i / 1-chord) and increase by one chord degree from 1 to 7. You can play 7th and inverted chords.

The chord bar uses the middle octave row of the 3-Octave to build the chord.

[PIC]

Using the Chord Bar:

- **To play the chord**, just press one of the first 7 (blue) pads to trigger that chord. Pressing the 1st pad plays the 1-chord. Pressing the 4th pad plays the 4-chord, etc..
- **To play an inverted chord,** hold down [ALT] and trigger the chord.
- **To play 7th chords**, use the last button on the chord bar (purple when not active, light purple when active) to make the chords play 7th chords.

The default scale/mode is C Major.You can access all the 12 western tones for your root and the following scale/modes for playing:

- **Ionian (a.k.a Major)**
- **Dorian**
- **Phrygian**
- **Lydian**
- **Mixolydian**
- **Aeolian (a.k.a. Minor)**
- **Locrian**
- **Major Pentatonic**
- **Minor Pentatonic**
  (todo: add more ? BLUES, BEBOP, etc?)

MACROS:

- **[ALT] + [MACRO #1]** to change to the next the key signature. The new key signature will be shown on the Fire display. Repeat until your desired key appears.

- **[ALT] + [MACRO #2]** to change to the next harmonic mode/scale. The new scale/mose will be shown on the Fire display. Repeat until your desired scale/mode appears.

- **Activating a Pattern Track** (even the current one) will briefly display the current key and scale/mode as a reminder.

Pattern Tracks:

The 8 Pattern Track pads (top row - left most 8) will allow you to quickly activate and use the associated channel to play it with the performance section or other MIDI keyboard.

These 8 pads are color coded to use the same colors as the first 8 pattern tracks in FL Studio.

If you recall the 1 to 1 to 1 philosophy, if the yellow channel is your “Lead” instrument, pressing the yellow pad will activate the “Lead” instrument channel, the “Lead” pattern and the “Lead” mixer track.

Upon activating the FireNFX, the Pattern Track colors should be nearly identical to the colors you used for the Patterns in FL Studio. Please note that some colors do not translate well between the FL Studio and AKAI so choose wisely. There are about 8 distinct (meaning they can be easily identified when next to any other in the list): Green, Cyan, Blue, Purple, Magenta, Red, Orange and Yellow. I recommend you stick with slight variations on these if color distinction is important to you.

Using the Pattern Track Pads

- Just press the color-coded pad (1-7) to activate the pattern track in FL studio. Once activated the pad will periodically blink with a white light to indicate the active pattern track.

Breaking my own philosophy:
The FireNFX actually has a limited ability to use multiple patterns per channel. This breaks the 1 to 1 to 1 philosophy for sure but it was an experimental way to allow for multiple patterns of different lengths to be used. So I am experimenting for now. This may not remain a feature but is demonstrated in the example template project by the “FPC” pattern and it’s “subs”. One of the deficiencies of this method is that I haven’t figured out how to mute the channels from Mixer mode.

If, for example, the “FPC” pattern is cloned (I clone the patterns to ensure they have the same channel and mixer links) and renamed “...FPC xxxx”. The 3 dots would indicate to FireNFX to use

that as a “sub” pattern of the one cloned. Pressing the pad for the pattern will toggle to the first pattern and then through the list of “sub” patterns.

You can see in the demo project there is a “FPC”, “...FPC Short” and “...FPC Long”.

Now when you press the pad to activate the “FPC” channel, you can press it again to toggle through the “...” channels below it. In the demo project, it’s the last pad (Blue) on the top row. Pressing it will move between the three FPC patterns.

Additionally I made the FPC the last pattern so I could route the FPC pads to groups of mixer tracks.... But trust me, 1 to 1 to 1 is mostly the way to go here. :)

Channel/Mute Strip:

The row below the Pattern pads is the channel/mute strip (second row) it can operate differently depending on the mode:

1) When in Mixer mode - the pads will represent the mute button for the pattern pad directly above it. The pad color will be dim white when NOT muted and completely off when muted. This is to allow you to do live “arrangement” of sorts by muting and un-muting to add variety and interest to your performance.

2) When in “Channel” mode - the pads will now only affect the currently selected channel. To represent this visually, all the pads will be colored a dim version of the active pattern color. Of the 8 pads in this row, the first 4 manipulate the loop length:

a) Pad 1 - resets the pattern loop length to “None”.
b) Pad 2 - sets the pattern loop length to 2 bars. Stays lit when active.
c) Pad 3 - sets the pattern loop length to 4 bars. Stays lit when active.
d) Pad 4 - sets the pattern loop length to 8 bars. Stays lit when active.
e) Pad 5 - mute/unmute for the active pattern
f) Pad 6 - Show/Hide Plugin/Generator
g) Pad 7 - show/Hide Piano Roll
h) Pad 8 - show/hide Channel setting

Side note on pattern loop length: The actual **pattern length** in the playlist will be different from pattern **loop** length.. If your pattern length was 8 bars but your pattern loop length was 2 bars, it would loop the 2 bars 4 times to fill in those 8 bars total.

Side note on knobs usage in modes: The knobs above the pads will function as labeled on the FIRE. I did not change these so you have easy access to volume and pan, etc. The mode determines which volume and pan is used (channel vs mixer).

Macros:

These are various 8 pads that help with assorted features and modes. Some features may require the ALT or SHIFT button to be held

As of this pre-release the macros are defined as such:

1	2	3	4	5	6	7	8		Macro
VEL	**OCT**	**SNAP**	**COPY**	**CLEAR**		**MUTE ALL**	**TOGG MUTE**		
KEY	**SCALE**		**PASTE**	**CLEAR ALL**	**RE-INIT**				**[ALT]**

Macro #		+ [ALT]	
1	Set Velocity - 4 levels - White, Green, Yellow, Orange	Set Key - 12 levels - Shown on display	
2	Set Octave - 4 levels - White of increasing brightness as octave increases	Set Harmonic Scale/Mode - Shown on display	
3	Set Snap - cycles through presets		
4	Copy Channel MIDI	Paste Channel MIDI	
5	Clear Channel MIDI	Clear ALL Channels MIDI	
6		Re-Initialize	
7	Mute All		
8	Toggle All Mutes		

Progress Bar:
This is an 8 step progress indicator that is visibly active only when playing or recording. It's a simple visual indicator as to your position within the repeating song loop. When recording, the progress bar will inherit the color of the pattern track being recorded.

Pressing on a pad in any position will move the song to that position.

Advanced usage / Customization

The following files are the ones used for development of this mod

fireNFX.py	- the main code file for this modification
nfxFireColors.py	- this file contains the color names and hex values for the colors.
nfxFirePadDefs.py	- contains the pad layout and defines some variables
nfxFire_Template.zip	- The sample project to open in FL Studio

The remaining files are copied from Image-Line's normal FIRE Script. The primary file:

device_Fire.py	- The original main FIRE script.

I try to hook into the existing device_Fire.py into key locations by writing my own OnXXXX handlers.

Searching the device_Fire.py for "nfxFIRE" or even just "nfx" should help you find the locations I used.

TODO: Code documentation.