FB-02

Software FM Sound Generator

Version 1.1

http://www.fullbucket.de/music

Table of Contents

Introduction

FB-02 is a software instrument plug-in (VST2/VST3/AU/CLAP/AAX) for Microsoft Windows and Apple macOS simulating the sound generation part of the classic Yamaha *FB-01* module from 1986. It is written in native C++ code for high performance even on "lighter" systems. The main features are:

- Simulation of the original FM sound generation unit
- Original factory ROM and additional sounds
- Includes the eight *TX81Z* waveforms
- User interface can be used to edit the sound of the original *FB-01*
- Program Browser
- SysEx import and export
- Supports MTS-ESP by ODDSound dynamic micro-tuning
- All parameters can be controlled by MIDI controllers
- Plug-in supports Windows and macOS

FB-02 is based on the **iPlug2** framework maintained by **Oli Larkin and the iPlug2 team**. Big thanks, guys!!! Without your work it would not have been possible to create a resizable *FB-02* user interface.

To resize the plug-in you just grab the yellow triangle at the bottom right of the window and drag it. You can save the current window size using the menu entry "Save Window Size" in the *Options Menu*.

If you have trouble with the standard version of *FB-02*, please grab the (sound-wise identical) "N" version of the plug-in which is based on the original **iPlug** framework.

The Paulke/Bucket Project

After the release of my *DrumTraqs* plug-in, Wolfgang Paulke[1] asked me if I could recreate the Yamaha *FB-01*, an inconspicuous FM sound generator module from 1986. He had used the *FB-01* extensively on his recordings and was looking for a way to replace his self-made/sampled *FB-01* Kontakt reproduction with a "true" software instrument. Without further ado he sent me his *FB-01* hardware unit as a reference.

The original *FB-01* is a multi-timbral module with up to 8 parts – sound editing is only possible via external editors (for example the respective software for the Yamaha *CX5M* computer who features the same sound module as the *FB-01*). Wolfgang and I agreed that I would only do the FM sound generation part – one can easily do the multi-timbral stuff in a DAW.

I never planned to do an FM clone, and it was a pretty tough job to get it halfway right, even with the original unit at my side. This is because the whole magic is done by a single chip, the Yamaha YM2164 (and the respective DAC chip) – you cannot look "inside" what it does. My major resource was the application manual for the YM2151, the famous predecessor of the rather poorly documented YM2164, and trial and error.

1 Wolfgang Paulke is a rock guitarist with a strong affection to electronics. He had performed many concerts in the former German Democratic Republic and was a close friend of Reinhard Lakomy. Google those guys up and check out their work on YouTube – you will hear some fantastic stuff!

Now here is my *FB-02*, and of course it does not sound like the original *FB-01* – but you would not have expected that, do you? ☺

What's New With Version 1.1?

Apart from some bug fixes and improvements, the new version 1.1 now (on popular request) adds the option to select one of the eight waveforms of the Yamaha *TX81Z* for the operators.

But before you ask: *FB-02* cannot import patches from the *TX81Z*, sorry!

Factory Programs/Voices

FB-02 contains many built-in programs (*Voices* as Yamaha calls them) which you can access via the browser (see *Program Browser*).

- **Factory Sounds**
  The first 20 sounds of bank 1 are created by myself.
  Sound 21 of bank 1 was converted by **John Koster** and adapted by myself.
  The first 20 sounds of bank 2 are created by **kraftraum**.
  Banks 3 to 7 contain the original *FB-01* ROM patches.
- **Empty/Init**
  All banks contain the blank *Init* program. Pretty boring but sometime helpful.
- **Kraftraum's Selected Sounds**
  This is a collection of programs that **kraftraum** has compiled from various sources of the Internet.
- **Collection #1 to #9**
  The programs of these collections come from **MAZES** blog at
  https://mmmazes.blogspot.com/2010/05/fb-01-patches.html

Note: There is a good chance that you will find some doublets within all of these programs.

Acknowledgments

- **Oli Larkin** and the **iPlug2** team.
- **Wolfgang Paulke** – influencer, Beta tester and a really fine chap!
- **kraftraum** (https://soundcloud.com/kraftraum) – "my" Beta tester and a really fine chap, too!
- **Llama Music** (https://llamamusic.com/fb01/index.html)
- **MAZES** (https://mmmazes.blogspot.com)
- **Vincent** from dbwbp.com.
- **My family** for bearing me and my crude hobby.

No, I am not affiliated with Yamaha (nor KORG) in what relation ever except that I find myself entangled with their instruments. ☺

Overview

FB-02 is a four-operator FM synthesizer (FM stands for "Frequency Modulation", but in fact it is "Phase Modulation" that is applied here). I won't go into the details of FM synthesis – this topic has been and will be discussed thoroughly elsewhere, just DuckDuckGo[2] it up.

Programs and Banks

FB-01 and *FB-02* offer 7 banks with 48 programs each, i.e. a total of 336 programs (Yamaha usually calls them *voices*, but I will stick to the term *programs*). Selecting programs etc. will be explained in section *Plugin Handling*).

The first two banks of the original *FB-01* can be overridden by user-defined programs while the remaining banks 3 to 7 are immutable ROM presets. Instead, the *FB-02* allows you to overwrite and edit the programs of *all* banks. Furthermore, programs imported via MIDI SysEx will be stored in the currently selected bank; note that this is not possible with the *FB-01*.

Operators

In short terms, an *operator* is a sine wave oscillator with its own amplitude envelope generator. The point is that an operator can modulate or be modulated by other operators. A modulating operator is called a *Modulator* while a modulated operator is called a *Carrier.*

All operators have the same set of parameters (with the exception of operator 4 which has an additional feedback parameter – see section *Algorithms*).

- **ON:** Enables the operator output.
- **LEVEL:** Defines the output level.
- **ADJUST:** Fine-adjusts the output level.
- **WAVE:** Selects the operator's waveform from one of the eight Yamaha *TX81Z* waveforms. Note that the *FB-01* only features the first waveform (Sine)!
- **BOOST:** Applies an additional "boost" to the output level. Furthermore, LFO amplitude modulation (AM; see section *LFO*) is only applied if BOOST is disabled.

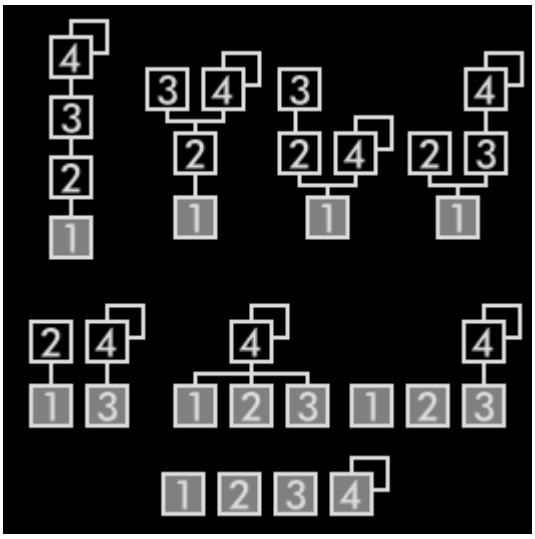
2 "To google up" is out of fashion.

- **FREQ:** Defines the operator's frequency in multiples of the base frequency. Values range from 0.5 to 1, 2, 3 etc. up to 15.
- **DETUNE:** Detunes the operator with respect to the base frequency. Detuning here is not as strong as you might be used from "normal" synthesizers; the effect is primarily to add some "liveliness" to the sound when the operator is used as modulator.
- **INHRM:** Defines the inharmonic tuning of the operator as a factor of the current frequency; possible values are 1.0, 1.41, 1.57 and 1.73. This is useful to create bell- or metal-alike sounds.
- **KEYB. SCALE:** Sets the keyboard scaling curve (negative linear, positive linear, negative exponential or positive exponential). Keyboard scaling means that for higher notes the operator's output level will decrease (negative curve) or increase (positive curve) when a key is played. The amount depends on the distance of the note to the middle C4 and on the shape of the curve (linear or exponential).
- **DEPTH:** Controls the intensity of the keyboard scaling effect.
- **VEL:** Defines how much the velocity adds to the operator's output level.
- **RATE:** Defines how much the keyboard scaling speeds up the timings of the envelope generator (for example higher notes cause faster attack times).
- **ATTACK:** The attack time of the operator's envelope.
- **A.VEL:** Defines how much the velocity decreases the attack time.
- **DECAY1:** The first decay time i.e. the time it takes to reach the sustain level.
- **DECAY2:** The second decay time i.e. the time it takes to reach from the sustain level to zero. If set to "0" the envelope will behave like a classic ADSR.
- **SUSTAIN:** The sustain level.
- **RELEASE:** The release time.

Note that a value of 0 for an envelope time parameter will (typically) result in an "infinite" (or better: "gated") time.

Algorithms

The topology of the four operators' assignment (which operator modulates which) is called an *algorithm*. There are eight different algorithms available on the *FB-01* (see image on the right).



Once an operator (modulator) modulates another operator (carrier), it does not add to the overall signal output – only carriers will do. There is one exception: Operator 4 can modulate itself, and the amount is controlled by the F.BACK (feedback) parameter.

Also note that operator 1 is always a carrier and never a modulator.

LFO

The LFO (low frequency oscillator) can modulate both amplitude and pitch of the operators. Parameters are:

- **ON:** Enables the LFO.
- **SYNC:** When activated, the LFO will restart its period with every new key pressed.
- **WAVEFORM:** The shape of the LFO's output signal: *Saw*, *Square*, *Triangle* or *Noise* (random).
- **SPEED:** The frequency of the LFO.
- **AM:** The intensity of amplitude modulation. Note that the LFO only modulates the level of operators where BOOST is set to "CARRIER".
- **[AM] SENS:** AM sensitivity; controls (together with the parameter AM) the level of the amplitude modulation (if set to 0, there will be no amplitude modulation at all).
- **PMD CONTR:** Pitch modulation depth controller; sets the MIDI controller to control the intensity of pitch modulation applied to the operators. Possible values are *Off*, *Aftertouch*, *Modulation Wheel*, *Breath Controller* or *Foot Controller*.
- **PM:** The intensity of pitch modulation. The value of this parameter adds to the value of the selected PMD Controller.
- **[PM] SENS:** PM sensitivity; controls (together with the parameter PM) the level of the pitch modulation (if set to 0, there will be no pitch modulation at all).

Common Section

This section contains some common parameters as well as controls to select the current program.

FB-02 features seven banks with 48 programs per bank. The programs of the first two banks of the original *FB-01* can be overridden by the user, while the remaining programs of banks 3 to 7 are baked into its ROM. Since there is no ROM on the *FB-02* you can edit these programs here, too. ☺

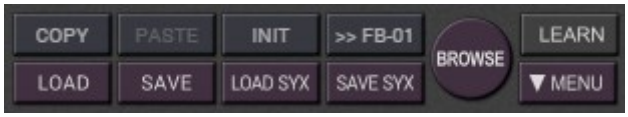
To select a bank, just click on the number above "BK.". To select a program, click on the number above "PRG." or click one of the arrow buttons (previous/next program). To edit the name of the program, click into the "VOICE NAME" field.

Some more parameters are:

- **VOLUME:** The overall output volume.
- **TUNE:** The master tune of *FB-02*.
- **ALGORITHM:** Selects the algorithm.
- **T.POSE:** Transpose of FB-02 in notes (-128 to +127).
- **P.BEND:** Pitch Bend amount in notes (0 to 12).
- **PORTA:** Portamento time.
- **F.BACK:** Feedback amount of operator 4.
- **MODE:** Keyboard mode (*Poly* or *Mono*).
- **LEFT/RIGHT:** Enables the left and right stereo output respectively.

Plugin Handling

The *FB-02* plug-in has some handy features/functions explained in this section.



- **COPY:** Copies the current program.
- **PASTE:** Pastes a copied program into the current program (this will replace the original data of the current program).
- **INIT:** Initializes the current program data to default values.
- **>>FB-01:** Sends the current program data via SysEx to another *FB-01* hardware unit or *FB-02* plug-in.
- **LOAD:** Loads the current program (FXP file) or all programs (FXB file). Also has the option to restore all programs to the "factory" content.
- **SAVE:** Saves the current program to a FXP file or all programs to a FXB file.
- **LOAD SYX:** Loads a program or a full Voice bank from a SysEx (SYX) file into the currently selected bank.
- **SAVE SYX:** Saves a program or the currently selected Voice bank to a SysEx (SYX) file. To be compatible to the *FB-01* you can save it either as a "Bank #1" or a "Bank #2" file – *FB-02* wouldn't care, but *FB-01* would load it later into the respective user bank.
- **BROWSE:** Opens the Program Browser (see section *Program Browser*).

For both LOAD and SAVE you also have the option to set a default path for FXP/FXB program files. Consequently, for both LOAD SYX and SAVE SYX you can set the default path for SYX files.

The LEARN and MENU function will be described below.

MIDI Learn

The easiest way to (re)assign MIDI controllers to *FB-02* is to use the *MIDI Learn* function. To activate MIDI Learn, click on the LEARN button and wiggle both the MIDI controller and *FB-02's* parameter or button that you want to link. If you want to unlearn the assignment, right-click the LEARN button (the label now reads "UNLEARN") and activate it. Now wiggle the MIDI controller or the parameter that you want to unlearn.

Options Menu

This menu open when you click the MENU button in the common section.

Select Startup FXB		Select the FXB file that should always be loaded when *FB-02* is started
Load Startup FXB		Load the Startup FXB file; can also be used to check what the current Startup bank is
Unselect Startup Bank		Unselect the current Startup FXB file
Global MIDI Settings...		
	MIDI Thru	Set globally if MIDI data sent to *FB-02* should be sent through to its MIDI output (stored in configuration file)
	Ignore Program Change	Set globally if MIDI Program Change data sent to *FB-02* should be ignored (stored in configuration file)
	Send Parameter Change SysEx	Set globally if SysEx data should be sent when a *FB-02* parameter is changed
Reload Configuration		Reload *FB-02's* configuration file
Save Configuration		Save *FB-02's* configuration file
Show All Knob Values		Set globally if the knob values should always be shown
Go Online...		
	Check Online for Update	When connected to the Internet, this function will check if a newer version of *FB-02* is available at fullbucket.de
	Visit fullbucket.de	Open fullbucket.de in your standard browser

The *FB-02.ini* Configuration File

FB-02 is able to read some settings from a configuration file (`FB-02.ini`). The exact location of this file depends on your operating system and will be displayed when you click on “Reload” or “Save Configuration”.

Control *FB-02* Via MIDI

It is possible to fully control *FB-02* by MIDI controllers, or more precise: Each MIDI controller (except *Modulation Wheel* and *Sustain Pedal*) can control one of *FB-02*'s parameters or buttons. The mapping is defined in the `FB-02.ini` for example like this:

```
[MIDI Control]
CC7  = 0  # Master Volume
CC65 = 6  # Feedback
CC71 = 13 # LFO Speed
...
```

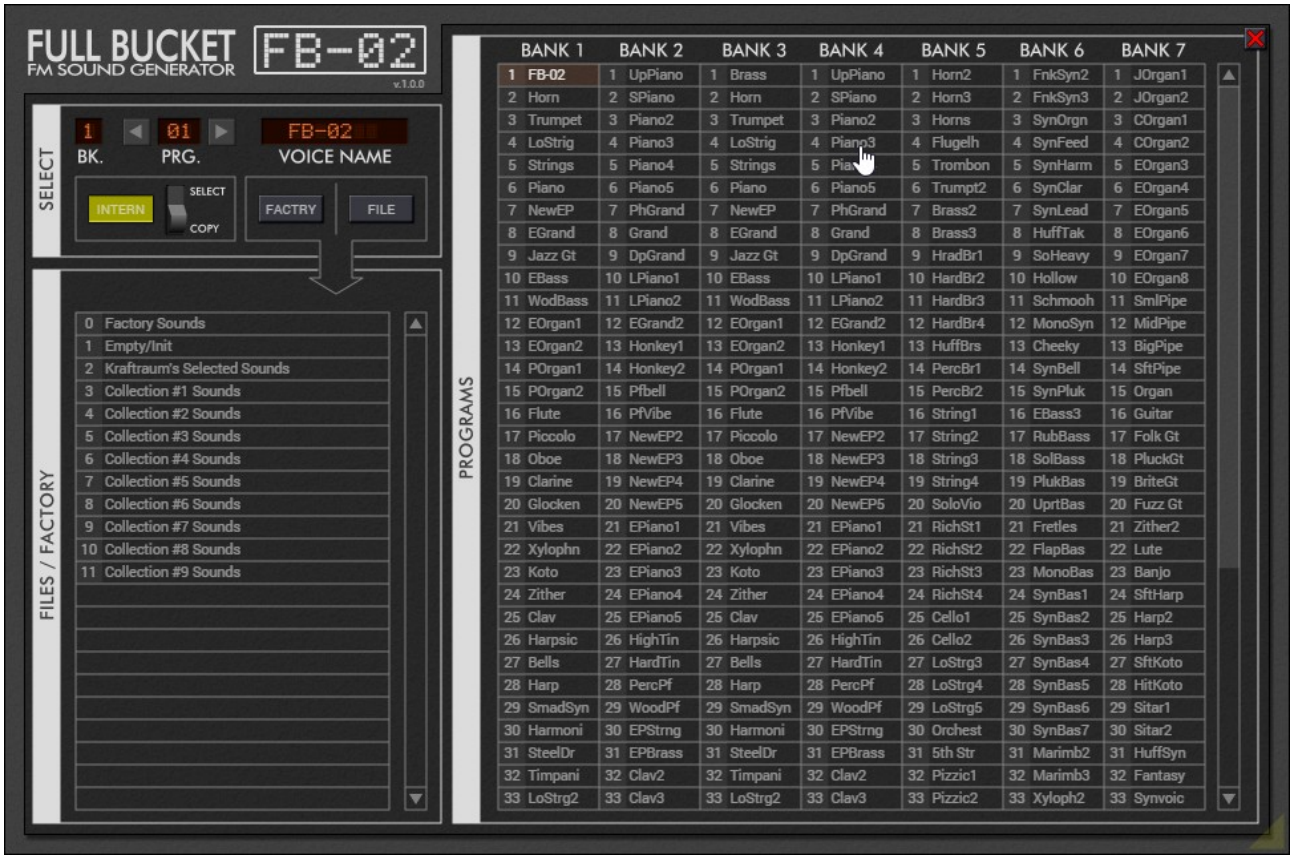
The syntax is straight forward:

```
CC<controller number> = <parameter/button ID>
```

Given the above example, controller 7 directly controls the overall *Master Volume* parameter, controller 65 the operator 4's *Feedback* etc. As you can see, comments are introduced by the Pound sign (#); they are here just for description purposes and completely optional. Note that the *controller number* can run from 0 to 110, with the exception of 1 (*Modulation Wheel*) and 64 (*Sustain Pedal*); the latter two are simply ignored.

Program Browser

Clicking on the "BROWSE" button opens the Program Browser.



The Program Browser has three (in fact four) modes of operation which can be activated via the three gray buttons:

- **INTERN**
  - When the switch is set to "SELECT", clicking a browser item simply selects the respective program.
  - When the switch is set to "COPY", clicking a browser item copies the respective program data to the currently selected program.
- **FACT[O]RY**
  - You can select one of the factory banks to use in the "FILE" list below. Clicking a browser item copies the respective program data to the currently selected program (see section *Factory Programs/Voices*).
- **FILE**
  - After you have chosen a folder on your computer, the files contained in that folder (if any) are enlisted in the "FILE" list. Clicking a file item reveals the contained programs in the browser. Clicking a browser item copies the respective program data to the currently selected program. Note that SysEx (SYX) files contain at most a single bank.

Plug-In Parameters

Common Parameters

parameter	id	description
Master Volume	0	Master (overall) volume
Master Tune	1	Master tune
Algorithm	6	Algorithm (1 to 8)
Transpose	7	Transpose in notes (-128 to +127)
Pitch Bend Range	8	Range of Pitch Bend in notes (0 to 12)
Portamento	9	Portamento time
Feedback	10	Feedback amount of operator 4
Mode	11	Keyboard mode (*Poly* or *Mono)*
PMD Controller	12	Pitch modulation controller (*Off*, *A.Touch*, *M.Wheel*, *Breath*, *Foot*)
Output Left	13	Enables the left output
Output Right	14	Enables the right output
LFO Enable	15	Enables the LFO
LFO Waveform	16	LFO waveform (*Saw*, *Square*, *Triangle*, *Noise*)
LFO Speed	17	Frequency of the LFO
LFO Sync	18	Enables restart of the LFO on key press

parameter	id	description
LFO AM Depth	19	Amplitude modulation depth
LFO AM Sensitivity	20	Amplitude modulation sensitivity
LFO PM Depth	21	Pitch modulation depth
LFO PM Sensitivity	22	Pitch modulation sensitivity

Operator Parameters

parameter	id	description
OP1: Wave	2	Operator 1 waveform (*W1*, *W2*,… *W8*)
OP2: Wave	3	Operator 2 waveform (*W1*, *W2*,… *W8*)
OP3: Wave	4	Operator 3 waveform (*W1*, *W2*,… *W8*)
OP4: Wave	5	Operator 4 waveform (*W1*, *W2*,… *W8*)

parameter	id	description
OP1: Enable	23	Enables the operator
OP1: Level	24	Output level
OP1: Velocity	25	Velocity sensitivity
OP1: Boost	26	Boosts output level (*Carrier* or *Modulator*)
OP1: Frequency	27	Frequency (0.5, 1, 2,... 15)
OP1: Inharmonic	28	Inharmonic tuning factor (1.0, 1.41, 1.57, 1.73)
OP1: Detune	29	Detuning (0 to 7)
OP1:Keyb. Scaling Type	30	Keyboard scaling type (*-Linear*, *+Linear*, *-Expon.*, *+Expon.*)
OP1: Level Adjust	31	Fine adjustment of output level (0 to 15)
OP1:Keyb. Scaling Depth	32	Depth of keyboard scaling (output level)
OP1:Keyb. Scaling Rate	33	Rate of keyboard scaling (envelope times)
OP1: Attack	34	Attack time of the envelope
OP1: Attack Velocity	35	Velocity sensitivity of attack time
OP1: Decay 1	36	First decay time of the envelope
OP1: Decay 2	37	Second decay time of the envelope
OP1: Sustain	38	Sustain level of the envelope
OP1: Release	39	Release time of the envelope

The respective parameters for operator 2 range from 40 to 56.
The respective parameters for operator 3 range from 57 to 73.
The respective parameters for operator 4 range from 74 to 90.

Frequently Asked Questions

How do I install FB-02 *(Windows VST2 32 bit version)?*

Copy the files `fb02.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST2 plug-in folder. Your DAW should automatically register the *FB-02* VST2 plug-in the next time you start it.

How do I install FB-02 *(Windows VST2 64 bit version)?*

Copy the file `fb0264.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST2 plug-in folder. Your DAW should automatically register the *FB-02* VST2 plug-in the next time you start it.

Note: You may have to remove any existing (32 bit) `fb02.dll` from your VST2 plug-in folder or else your DAW may screw the versions up…

How do I install FB-02 *(Windows CLAP 32/64bit version)?*

Copy the file `fb0232.clap` (32 bit) or `fb0264.clap` (64 bit) from the ZIP archive you have downloaded to the `C:\Program Files\Common Files\CLAP` folder. If your DAW supports the CLAP format, it will automatically register the *FB-02* CLAP plug-in the next time you start it.

How do I install FB-02 *(Windows VST3 64 bit version)?*

Copy the file `fb02.vst3` from the ZIP archive you have downloaded to your system's or favorite DAW's VST3 plug-in folder. Your DAW should automatically register the *FB-02* VST3 plug-in the next time you start it.

How do I install FB-02 *(Windows AAX 64 bit version)?*

Copy the file `fb02_AAX_installer.exe` from the ZIP archive you have downloaded to any of your system's folder and run it. Your AAX-enabled DAW (Pro Tools etc.) should automatically register the *FB-02* AAX plug-in the next time you start it.

How do I install FB-02 *(Mac)?*

Locate the downloaded PKG package file in Finder (!) and do a right- or control-click on it. In the context menu, click on “Open”. You will be asked if you really want to install the package because it comes from an “unidentified developer” (me ☺). Click “OK” and follow the installation instructions.

What is the plug-in ID of FB-02*?*

The ID is **`fb02`** .

Can I import TX81Z *patches via SysEx?*

No, this is not possible, sorry.