

Full Bucket Brigade Delay

Analog Delay Simulation

Version 1.2

© 2017-2019 by Björn Arlt @ Full Bucket Music
<http://www.fullbucket.de/music>



VST is a trademark of Steinberg Media Technologies GmbH

Windows is a registered trademark of Microsoft Corporation

The Audio Units logo is a trademark of Apple Computer, Inc.

Introduction

The *Full Bucket Brigade Delay* is an effect plug-in for Microsoft Windows (VST) and Apple macOS (VST/AU) simulating the functionality and sound of classic analog delays. It is written in native C++ code for high performance and extremely low CPU consumption. The main features are:

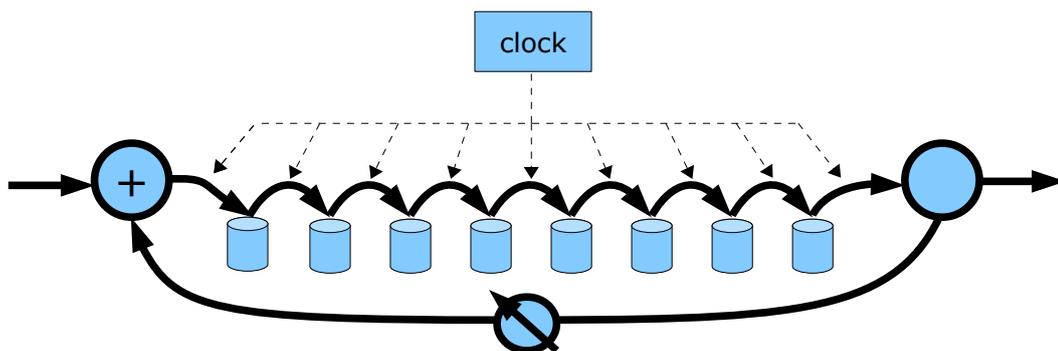
- Authentic simulation of a bucket brigade device (BBD) delay line
- No “crackling” or “dropouts” when changing the delay time
- LFO modulation for Flanger, Chorus, or other modulation effects
- *Freeze* mode
- Stereo!
- *Quality* control to send your hi-fi signals to hell
- Double precision audio processing
- Plug-in supports Windows and macOS (32 bit and 64 bit)

Gosh, Why Another Delay Plug-In?

The *Full Bucket Brigade Delay* is an offspring of the tiny delay within my *Nabla* synthesizer. Just for fun I separated it into its own C++ project, added a few things like the LFO and the *Quality* control, always trying to achieve the functionality and sound of those old BBD units that I love so much. Now I sat here with a new plug-in. And decided to share it.

What Is a Bucket Brigade Device?

In a bucket brigade device, the incoming audio signal is sampled at a specific frequency, the *clock rate* (typically some kHz), and fed into the first of a line of subsequent bins or “buckets” (see figure below). At the same time, the original content of the first bucket is put into the second bucket while the original content of the second bucket is put in to the third bucket while... you got the story. The content of the last bucket is sent to the output of the device. It may additionally be attenuated and sent back to the input – this is called *Regeneration* or *Feedback*. Since the whole process is repeated with every clock cycle, it takes some time until the original input sample appears at the output of the device; the signal is delayed.



Sketch of a bucket brigade device (BBD) delay line with feedback.

Now it is some kind of obvious (*cough, cough*) that the clock rate determines how fast the samples are moving through the BBD and thus how long the delay time is. This has three immediate consequences:

- *Higher* clock rates result in *shorter* delay times and *lower* clock rates in *longer* delay times.
- No matter how high the clock rate, there is *always* some delay between input and output of the BBD – because the samples *always* have to travel through *all* the buckets.
- If the clock rate is *varied*, the samples that are already “in” the BBD will appear at a different rate at the output than they had been sampled at the input. Thus their “frequency” will *change*.

(Feel free to check the *Full Bucket Brigade Delay* if it is really working this way!)

Wait a minute: Sampling is used here? Why is a BBD delay line then called “analog”? Because the “samples” are not digitized but handled “as is”: They are “measurements” of a continuous signal, discrete in time but still continuous with respect to their values (well, almost...), and as such they travel through the BBD.

Of course this will have other side effects: Since buckets of a BBD are not perfect analog memory elements, the signal quality will degrade with the length of the BBD and the clock rate. Typically, the frequency range of the delayed signal is smaller than the one of the original signal; even more, the delayed signal may be slightly distorted. In the past this was considered to be a major problem, today we call it “analog” and love it for the resulting “warm” sound!

User Interface

The layout of the *Full Bucket Brigade Delay's* user interface is straight forward: In the “DELAY” section you can select the delay time range (1–8ms, 4–31ms, 16–125ms, 31–500ms), set the delay time itself, and adjust the feedback level. Note that changing the delay time will change the frequency of whatever signal is already “in” the delay line.



The "LFO" section controls the speed (up to 25Hz) and the depth of the low frequency oscillator modulating the delay time. Here lies the origin of all those nice effects named *Flanger*, *Chorus*, *LSD Trip Echo*, *Psychotic Slapback* etc.

The mix between original and effect is set in the "OUTPUT" section along with the "quality" of the delayed signal: The lower the value of the *Quality* parameter, the higher the analogishness of the effect.

Have fun with the *Freeze* mode: Click on the "FREEZE" button, and whatever is currently in the delay line will be looped forever until you click the button again. Note that in "Freeze" mode the input is "locked" i.e. the input signal will not be added to the delay line.

Finally, you can select one of the ten presets by clicking on the yellow triangle at the top left corner of the effect screen. These presets give you some examples of what you can do with the *Full Bucket Brigade Delay*.

Tip: If you want to know what I remember best of those old BBDs then turn *Feedback* to maximum (100%) and *Quality* to some low value (~30%) and twiddle the *Time* parameter. And better reduce the volume of your amp.

MIDI Control Change Messages

All parameters of the *Full Bucket Brigade Delay* can be controlled by MIDI controllers, or more precise: Each MIDI controller can control one of *Full Bucket Brigade Delay*'s parameters. The mapping is defined in the `fbdelay.ini` configuration file like this:

```
[MIDI Control]
CC4  = 10 # Freeze
CC70 = 1  # Delay Time
CC71 = 2  # Feedback
...
```

The syntax is straight forward:

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

Given the above example, controller 4 directly controls the *Freeze* mode, controller 70 the *Delay Time* etc. As you can see, comments are introduced by the Pound sign (#); they are here just for description purposes and completely optional.

The `fbdelay.ini` file is located in the same directory as the *Full Bucket Brigade Delay* VST DLL (`fbdelay.dll` or `fbdelay64.dll`) or Mac VST/AU (`FBDelay.component` or `FBDelay.vst`) itself; you can edit this file in a standard text editor. Next time your DAW host program starts, the *Full Bucket Brigade Delay* will load the changes.

MIDI Learn

The easiest way to assign MIDI controllers to *Full Bucket Brigade Delay* parameters is to use the MIDI Learn function. To activate *MIDI Learn*, click on the "LEARN" button and wiggle both the MIDI controller and the *Full Bucket Brigade Delay*'s parameter 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.

Parameters

parameter	ID	description
<i>Range</i>	0	Coarse range of the delay time: 1–8ms, 4–31ms, 16–125ms, or 31–500ms
<i>Time</i>	1	Delay time with respect to the delay time range
<i>Feedback</i>	2	Amount of feedback
<i>Speed</i>	3	LFO speed (0 to 25 Hz)
<i>Depth</i>	4	Depth of delay time modulation by LFO
<i>Mix</i>	5	Mix between original and effect signal
<i>Quality</i>	6	Quality of the simulated BBD delay line
<i>Freeze</i>	10	"Freeze" mode on/off

Frequently Asked Questions

How do I install the Full Bucket Brigade Delay (Windows 32 bit version)?

Just copy the file `fbdelay.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST plug-in folder. Your DAW should automatically register the *Full Bucket Brigade Delay* VST plug-in the next time you start it.

How do I install the Full Bucket Brigade Delay (Windows 64 bit version)?

Just copy the file `fbdelay64.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST plug-in folder. Your DAW should automatically register the *Full Bucket Brigade Delay* VST plug-in the next time you start it.

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

How do I install the Full Bucket Brigade Delay (Mac VST/AU universal 32/64 bit)?

Locate the downloaded PKG package file `fbdelay_1_2_0_mac.pkg` 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 the Full Bucket Brigade Delay?

The ID is `fbbd`.

No Sync-to-Host Tempo Possible? This Sucks!

Indeed. Like the original units.