

Squishbox User Guide

models 0001-0099

software version 3.0

Contents

Overview.....	1
Quick Start.....	2
Hardware.....	3
Squishbox Diagram.....	3
Adjusting Contrast.....	3
Interface.....	4
Left Button Menu.....	4
Right Button Menu.....	4
Wifi Access.....	5
Settings Files.....	6
Configuration File.....	6
Bank Files.....	6
Router Rules.....	7
Sysex Messages.....	7
Advanced Usage.....	8
SSH and FTP Access.....	8
BankEdit.....	8
Editing Soundfonts.....	8
Software Updates.....	8
Troubleshooting.....	9
Credits.....	9
License.....	9

Overview

The Squishbox is a Fluidsynth synthesizer running on a Raspberry Pi built into a stompbox. It will take MIDI input from controllers plugged into its USB ports and output 24bit stereo sound at 192kHz on its 1/4" outputs. FluidSynth uses Soundfonts, a sample-based sound synthesis file format, to produce sounds. The Squishbox allows you to switch between *patches* – groups of soundfont to MIDI channel assignments, effects settings, MIDI router rules, etc. quickly during a performance.

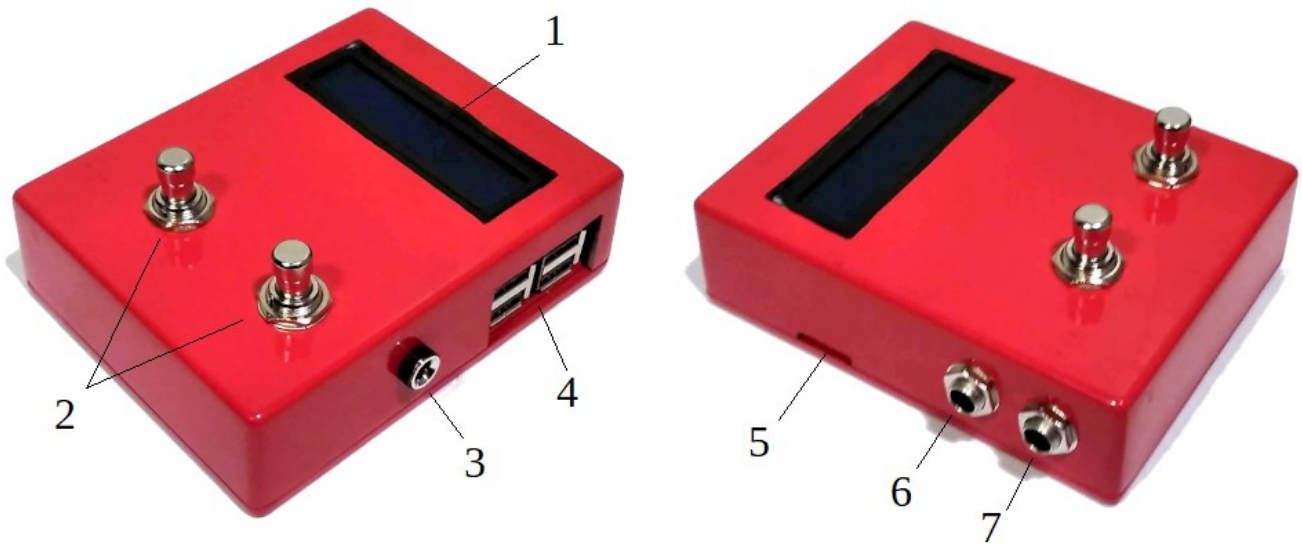
Various menus can be accessed by long-pressing (~2 seconds) either the left or right stompswitch. In menus, long-pressing the right stompswitch will generally select an option, while long-pressing left will back up or exit the menu. Most menus will also time-out/cancel after ~5 seconds.

Quick Start

- Plug the SquishBox in via the supplied AC adapter (or a comparable 5V/2A power supply).
- Plug a MIDI controller into any of the USB ports.
- Connect 1/4" cables from the left/mono and right output jacks to an amplifier or mixer, or plug headphones into the left output jack.
- If your controller is not detected when you plug it in, you can re-scan for it. Long-press left to open the options menu, then select "MIDI Reconnect".
- You should be ready to jam! The default bank, "bank0.yaml" contains a small number of interesting sounds that show examples of how you can create patches.
- You can add sounds from the Fluidsynth GM soundfont by long-pressing the right button, then selecting "Load Soundfont", choosing "FluidR3_GM.sf2", and browsing through those sounds. When you find one you like, choose "Save Patch" to save it as a new patch in the current bank.
- To access the Squishbox via wifi, create a hotspot on your phone/tablet/computer named `squishboxhotspot` with password `fluidsynth`. You should see the Squishbox connect in your hotspot settings menu. Point a web browser to the IP address of your Squishbox and you will see the control panel, where you can edit banks, modify settings, upload soundfonts, etc.

Hardware

Squishbox Diagram



1. 16x2 character LCD
2. Stomp switches
3. DC jack*
4. USB ports
5. SD card slot
6. Headphones/Left output**
7. Right output

*best results with 5V/2A power supply (included)

**provides stereo to 1/4" TRS headphone plug or mono (combined L+R) to TS plug if right output is unused


Adjusting Contrast

The contrast of your LCD display is set with a 10K potentiometer. The screw of the potentiometer is behind the USB ports, and can be reached for adjustment by inserting a small flathead screwdriver between the ports as shown.



Interface

Long-pressing the right stomp switch opens the primary menu, where you can save and delete patches, load/save banks, open a soundfont, and adjust effects parameters. Long-pressing left opens the special commands menu.

In some menu options text can be entered (albeit slowly) using the stomp switches. Tapping left or right will change the character under the cursor, while holding left or right will advance or back up the cursor. To confirm the text entry, advance to the end of the text and the “accept” symbol () will appear. Long-press right to confirm the text entry. To cancel, wait ~5 seconds for the text entry to time out.

Left Button Menu

- Save Patch – Save the current instrument and control change settings as a patch. Prompts for name, which can be new or same as an existing patch to overwrite.
- Delete Patch – Delete the current patch.
- Load Bank – Load a different bank
- Save Bank – Save the current bank, possibly with a new name
- Load Soundfont – Open a soundfont and select and play its presets directly. The first preset in the soundfont will be loaded to MIDI channel 1 and the left and right buttons will scroll through all presets and banks in the soundfont. Choosing “Save Patch” at this point will save the current preset as a patch in the last loaded bank. In this way, you can add new soundfonts/instruments to a bank.
- Effects.. – Opens a sub-menu where you can modify parameters for the chorus and reverb effects, and adjust the output gain (i.e. maximum volume) of the SquishBox. These changes will be added to the current bank, and can be made permanent by saving the bank.

Right Button Menu

- Power Down – Shut down the Squishbox. Allow approximately 30 seconds for complete shutdown before unplugging. The SquishBox can simply be unplugged at any time without shutting down, but there is some risk the onboard memory will be corrupted.
- MIDI Reconnect – Scan for any connected MIDI controllers that aren’t automatically detected and attempt to link them to FluidSynth.
- Wifi Settings – Show the current network name and IP address if connected. Tapping left or right reveals an option to connect to a new wifi network by entering a network name (SSID) and password.

- **Add From USB** – Hunts for a connected USB drive, finds all .sf2 files on it, and copies them to your Squishbox. Preserves folder structure, so if you have your fonts organized into different folders on your USB they will stay that way on the Squishbox.

Wifi Access

In order to really customize your patches and banks, and easily upload soundfonts and bank files, you'll need wifi access to your SquishBox. By default, the Squishbox will try to connect to a wireless network with the name `squishboxhotspot` using the password `geekfunklabs`. Create a hotspot with these settings on a computer, phone, or other device, and the Squishbox will connect to it, allowing you to gain further access. Alternatively, you can add your own wireless network using the "Wifi Settings" menu option described above.

When connected to a wifi network, the Squishbox provides a web-based file manager that can be used to view, edit, and upload files. Obtain the IP address of the SquishBox using the "Wifi Settings" menu option, open a web browser on your computer and enter the IP address in the address bar. The file manager login is `squishbox` and the password is `geekfunklabs`. You can use the file manager to upload new soundfonts or banks, or directly edit bank and config files.

Settings Files

The global settings and bank files are in [YAML](#) format – a plain-text, human-readable format for representing nested data structures.

Configuration File

The configuration file for SquishBox is `squishboxconf.yaml`. It stores the root location of banks and soundfonts, the current bank file to load, and settings to pass to FluidSynth. You shouldn't have any need to alter this file, but the [online wiki](#) can provide you with more information about it.

Bank Files

Bank files store sets of patches. Patches contain a large variety of settings that are sent to Fluidsynth when the patch is selected. This includes soundfont, bank, and program settings for each MIDI channel, control change (CC) messages, MIDI Sysex messages or files, FluidSynth settings, and FluidSynth midi router commands. Bank files are stored in an extended YAML format, with some special stylings added for readability.

The structure of bank files is described below. The example bank file `bank0.yaml` is also useful and has comments.

- **patches:**
 - Patch Name
 - instrument settings for any number of MIDI channels*, given as `<channel #>: <soundfont file>:<bank>:<preset>`
 - `cc:` followed by a list of control change messages to send when the patch is selected, specified as `<MIDI channel>/<CC number>=<value>`
 - `sysex:` followed by a list of sysex messages.
 - `router_rules:` - a list of router rules that are set when the patch is selected, after the bank router rules
 - 2nd Patch Name
 - channel specifications, CC messages, router rules, etc.
 - ...
- `router_rules:` - list of router rules to apply to all patches, before individual patch rules are set
- `fluidsettings:` - a list of FluidSynth Settings that are passed when the bank is loaded, specified as `[name]: <value>` pairs

*MIDI channel numbers in bank files are numbered starting with channel 1.

Router Rules

A router rule is a mapping of [key]: [value] pairs that describe how MIDI messages should be routed to FluidSynth. The keys can be any of the following:

- **type** - The type of message to route. The value can be `note`, `cc`, `prog`, `pbend`, `kpress`, or `cpress`.
- **par1** or **par2** - Parameter 1 and 2 of the rule. The value takes the form:
`<min>-<max>*<mul>+<add>`
Message parameters in the range `min` to `max` are multiplied by `mul`, then `add` is added (- instead of + subtracts). Scientific note names, such as `C#4` or `Eb7` are also understood.
- **chan** - Which channels to route the messages from/to. The value can take the same form as for `par1` and `par2`, but can also have the form
`<from_min>-<from_max>><to_min>-<to_max>`
in which case messages from each channel in the range `from_min` to `from_max` will be routed to every channel in the range `to_min` to `to_max`.

Sysex Messages

A sysex message is a list where the first element is the port name to send the message to. The remaining elements are either a list of SYSEX bytes to send (as decimal or hex), or the name of a SYSEX file to dump to the specified port.

Advanced Usage

SSH and FTP Access

You can ssh to your Squishbox (e.g. using [PuTTY](#)) or FTP (e.g. [Filezilla](#)) with the username `pi` and password `raspberry`, and manually edit files or upload soundfonts. You may want to change the password at the earliest opportunity, since it is the default for the Raspbian distribution. Once logged in, you can manually add a network using the command `sudo raspi-config`, or by manually editing the file `/etc/wpa_supplicant/wpa_supplicant.conf`.

BankEdit

The SquishBox also includes a text-based patch editing/playing tool that can be accessed from a login terminal. This can be useful for learning how to create patches, since you can instantly test your modifications to bank files. To use BankEdit, log in to the SquishBox via ssh. Temporarily suspend the functioning of the buttons and LCD by entering the command

```
sudo systemctl stop squishbox.service
```

Start BankEdit by entering `bankedit.py` at the command line. The current bank will be displayed on the terminal, and patches from the bank can be selected and played using a MIDI controller and the Page Up/Down keys on your keyboard. The bank can be edited as you play, and pressing F5 will update your changes to the patches in memory. After quitting BankEdit, restart the SquishBox by typing `sudo reboot` at the command line.

Editing Soundfonts

A lot of functionality can be obtained by cleverly constructing patches and using CC messages and midi routing, but there is a lot of tweakability built into the soundfont specification itself. Modulators can be used to allow MIDI messages to control specific aspects of how a sound is synthesized, such as chorus level, filters, or ADSR envelope. The included soundfont “ModWaves.sf2” shows examples of using modulators in a soundfont to modify filter cutoff frequency and resonance. One can of course also create entirely new soundfonts from scratch using audio samples. [Polyphone](#) is a free and highly versatile soundfont editor. The [soundfont specification](#) can be a helpful resource when editing soundfonts. A PDF version of the specification has been uploaded to your Squishbox.

Software Updates

Your Squishbox is powered by a Raspberry Pi 3, a fully-fledged computer. Updates to the software can be found on the associated [GitHub](#) site. Complete SD card images can be found in [releases](#). The SD card on which the operating system is stored can be removed most easily with a small pair of

needlenose pliers, and reimaged using the same type of disk imaging software used to burn [Raspian](#) images – [Win32DiskImager](#) is a good example.

The model number of your Squishbox should be provided on purchase, and is also written on the inside of the enclosure lid. Successive models may differ slightly in wiring. When updating your software, edit the `stompboxpi.py` file and uncomment the section at the top of the file relevant to your model to ensure proper operation.

Troubleshooting

If any problems arise with your Squishbox unit, visit geekfunklabs.com for technical assistance.

Credits

SoundFont® is a registered trademark of E-mu Systems, Inc.

License

This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#), 2020 by Bill Peterson.