JackTrip box recipe

Hardware

- Raspberry Pi 4 B with heatsinks and fan (optional)
- Rpi case
- Power supply
- SD card (min 8 GB)
- USB audio interface (recommended: Focusrite Scarlett Solo)

Official information on building JackTrip on Rpi4 available at https://help.jacktrip.org/hc/en-us/articles/1500009727561-Build-a-Raspberry-Pi-4B-Computer-with-JackTrip

Prepare PatchboxOS

- Download the PatchboxOS
- Flash the image using ApplePiBaker, balenaEtcher, or diskutil (dd)
- Connect the audio interface
- Boot the Rpi and folow the first run tutorial
    - ssh to th SPU: `ssh patch@ip_address` (The default user name is *patch* and its password is *blokaslabs*). Use ethernet (wired) connection.
        - update PatchBox
        - set new password: `mappings` or orther password of choice
        - choose default soundcard: USB, 48000, 128, 3
        - select boot: *console*
        - connect to network: *no*
        - choose module: *none*

Finish PatchboxOS config

- Update system:
    - `sudo apt update`
    - `sudo apt upgrade`
- Enter PiSound configuration: `sudo pisound-config`
    - Update Pisound
    - Change Pisound HotSpot settings:
        - ssid: `jacktrip00X` (use chosen ID)
        - wpa_passphrase: `mappings` (or anothe password of choice)
- Enter Raspi-Config: `sudo raspi-config`
    - Update:
        - `Update this tool to the latest version`
    - System options:
        - Hostname: `jacktrip00X` (use SPU's ID)
    - Advanced options:
        - expand filesystem

Create a place for source codes: `mkdir ~/sources`

Create a place for the user systemd services: `mkdir -p ~/.config/systemd/user`

Reboot

Compiling and running JackTrip headless on the SPU

- Dependencies: `sudo apt install libjack-jackd2-dev librtaudio-dev qt5-default`
- Extra package to test latency: `sudo apt install -y jack-delay`
- cloning and building JackTrip:

```
cd ~/sources
git clone https://github.com/jacktrip/jacktrip.git
cd ~/sources/jacktrip
./build
export JACK_NO_AUDIO_RESERVATION=1
```

To manually use as a client

- with IP address: `./jacktrip -c [xxx.xx.xxx.xxx]`
- with name: `./jacktrip -c spuXXX.local`

Adding a service to start JackTrip server

OBS: client name is the name of the other machine

```
cat <<- "EOF" | tee ~/.config/systemd/user/jacktrip_server.service
[Unit]
Description=Run JackTrip server
After=multi-user.target

[Service]
Type=idle
Restart=always
ExecStart=/home/patch/sources/jacktrip/builddir/jacktrip -s --clientname
jacktrip_client

[Install]
WantedBy=default.target
EOF
```

```
sudo chmod 644 ~/.config/systemd/user/jacktrip_server.service
systemctl --user daemon-reload
systemctl --user enable jacktrip_server.service
```

Adding a service to start JackTrip client (in this example, the server is spu003.local)

Replace the IP address for the server IP.

```
cat <<- "EOF" | tee ~/.config/systemd/user/jacktrip_client.service
[Unit]
Description=Run JackTrip client
After=multi-user.target

[Service]
Type=idle
Restart=always
ExecStart=/home/patch/sources/jacktrip/builddir/jacktrip -c 192.168.1.1 --
clientname jacktrip_client

[Install]
WantedBy=default.target
EOF
```

```
sudo chmod 644 ~/.config/systemd/user/jacktrip_client.service
systemctl --user daemon-reload
```

If you want to enable the client, disable the service and run `systemctl --user enable jacktrip_client.service`

Install aj-snapshot

http://aj-snapshot.sourceforge.net/

Check the last version on the website

```
sudo apt install -y libmxml-dev &&\
cd ~/sources &&\
wget http://downloads.sourceforge.net/project/aj-snapshot/aj-snapshot-0.9.9.tar.bz2
&&\
tar -xvjf aj-snapshot-0.9.9.tar.bz2 &&\
cd aj-snapshot-0.9.9 &&\
./configure &&\
make &&\
sudo make install
```

- To create a snapshot: `aj-snapshot -f ~/Documents/default.connections`
- To remove all Jack connections: `aj-snapshot -xj`
- To save connections: `sudo aj-snapshot -f ~/Documents/default.connections`
- To restore connections: `sudo aj-snapshot -r ~/Documents/default.connections`

Set custom Jack connections to load at boot:

```
cat <<- "EOF" | sudo tee /lib/systemd/system/ajsnapshot.service
[Unit]
Description=AJ-Snapshot
After=sound.target jackaudio.service
```

```
[Service]
Type=oneshot
ExecStart=/usr/local/bin/aj-snapshot -r ~/Documents/default.connections

[Install]
WantedBy=multi-user.target
EOF
```

```
sudo systemctl daemon-reload &&\
sudo systemctl enable ajsnapshot.service
```

Mapping using jack in CLI

- Check available devices: `cat /proc/asound/cards`. If you have multiple devices available, can call them by name
- lists jack available ports: `jack_lsp`
- List informtion and connections on ports: `jack_lsp -c` or `jack_lsp -A`
- Connect ports: `jack_connect [ -s | --server servername ] [-h | --help ] port1 port2` (The exit status is zero if successful, 1 otherwise)
- Disconnect ports: `jack_disconnect [ -s | --server servername ] [ -h | --help ] port1 port2`

Latency tests

Make sure JackTrip is running.

- Connect the necessary audio cable to create a loopback on the client's audio interface (audio OUT -> audio IN)
- For the loopback (same interface test): `jack_delay -I system:capture_2 -O system:playback_2`
- run the test: `jack_delay -O jacktrip_client.local:send_2 -I jacktrip_client.local:receive_2`

Jack available commands

To get a list on the computer type **jack** and hit *Tab*

command	command	command	command	command
jack_alias	jack_bufsize	jack_capture	jack_capture_gui	jack_connect
jackdbus	jack_disconnect	jack-dl	jack-dssi-host	jack_evmon
jack_load	jack_lsp	jack_metro	jack_midi_dump	jack_midi_latency_test
jack_net_master	jack_net_slave	jack_netsource	jack-osc	jack-play
jack_samplerate	jack-scope	jack_server_control	jack_session_notify	jack_showtime
jack_thru	jack_transport	jack-transport	jack-udp	jack_unload
jack_control	jack_cpu	jack_cpu_load	jackd	jack_wait
jack_freewheel	jack_iodelay	jack-keyboard	jack_latent_client	jack_midiseq
jack_midisine	jack_monitor_client	jack_multiple_metro	jack-plumbing	

command	command	command	command	command
jack-rack	jack_rec	jack-record	jack_test	
jack_simdtests	jack_simple_client	jack_simple_session_client	jack_zombie	

To check Jack logs: `sudo journalctl -u jack.service`

Places to change the SPU name when cloning the SD

- Enter PiSound configuration: `sudo pisound-config`
  - Change Pisound HotSpot settings:
    - ssid: `jacktrip00X` (use SPU's ID)
- Enter Raspi-Config: `sudo raspi-config`
  - System options:
    - Hostname: `jacktrip00X` (use chosen ID)

Setting the server IP at the client box

- Edit the *jacktrip_client.service* file: `nano ~/.config/systemd/user/jacktrip_client.service`
- Replace the IP at the line `ExecStart=/home/patch/sources/jacktrip/builddir/jacktrip -c 192.168.1.1 --clientname jacktrip_client` for the new IP address
- Save the file (Ctrl+O, then hit ENTER in *nano*) and exit (Ctrl+X in *nano*).
- Update the systemctl daemon: `systemctl --user daemon-reload`
- Restart the service: `systemctl --user restart jacktrip_client.service`
- To check connection (if the server is availalbe and accessible through the given IP): `systemctl --user status jacktrip_client.service`