

Trio

for soprano, ensemble and electronics

Electronics documentation

Shu-Yu Lin

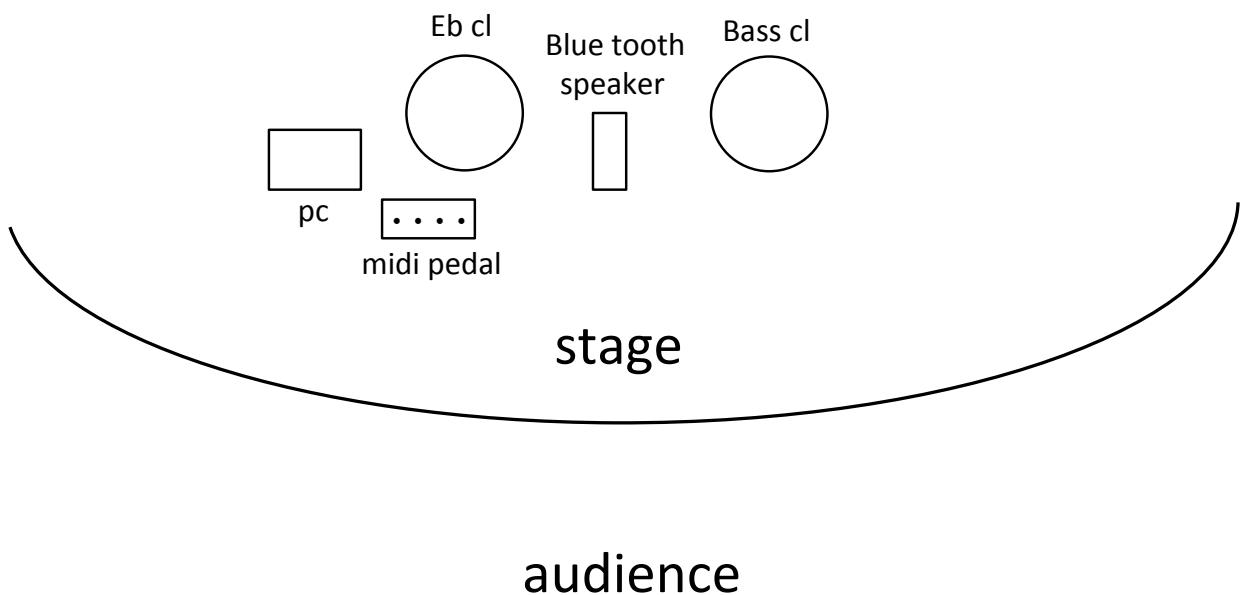
Table of Contents

Technical specification	3
Stage plan	3
Performance instruction	3
Software and hardware setup	4
SuperCollider code instruction	9

Technical specification

Ultimate Megaboom 3 Bluetooth speaker (or a Bluetooth speaker of your choice)
ActitioN 4 button midi foot pedal (or a midi controller of your choice)
Laptop computer
QLab software
SuperCollider environment

Stage plan



Performance Instruction

The numbers on the electronics part on the score refers to the specific sound file. 1 refers to sound file 1, 2 refers to sound file 2 etc. There are a total of 86 files.

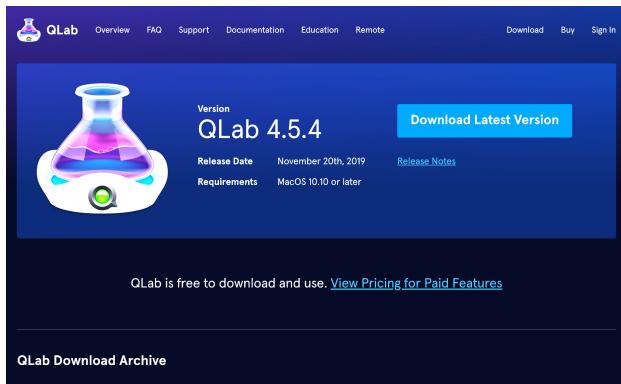
To perform, one of the clarinetist or an electronics performer will trigger the specific sound file at the indicated time according to the score.

If an electronics performer is involved, he/she can be either on stage or off stage. If electronics performer is to be on stage, he/she should sit in between the clarinetists with the speaker in front of him/her. If performing from off the stage, the position of the speaker should remain to be in between the clarinetists as indicated on the stage plan.

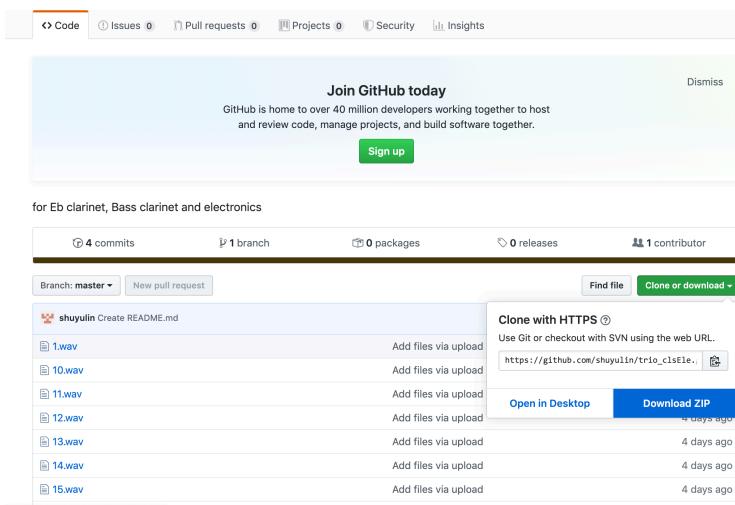
A midi controller with the Qlab program file will be used. To setup, refer to the following sections.

Software and hardware setup

1. Download and install QLab at <https://qlab.app/download/>

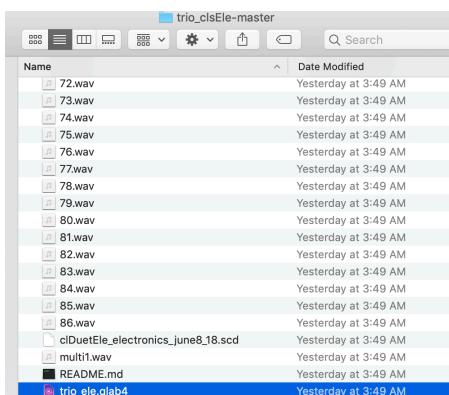


2. Download files for electronics from Shu-Yu's github at https://github.com/shuyulin/trio_clsEle

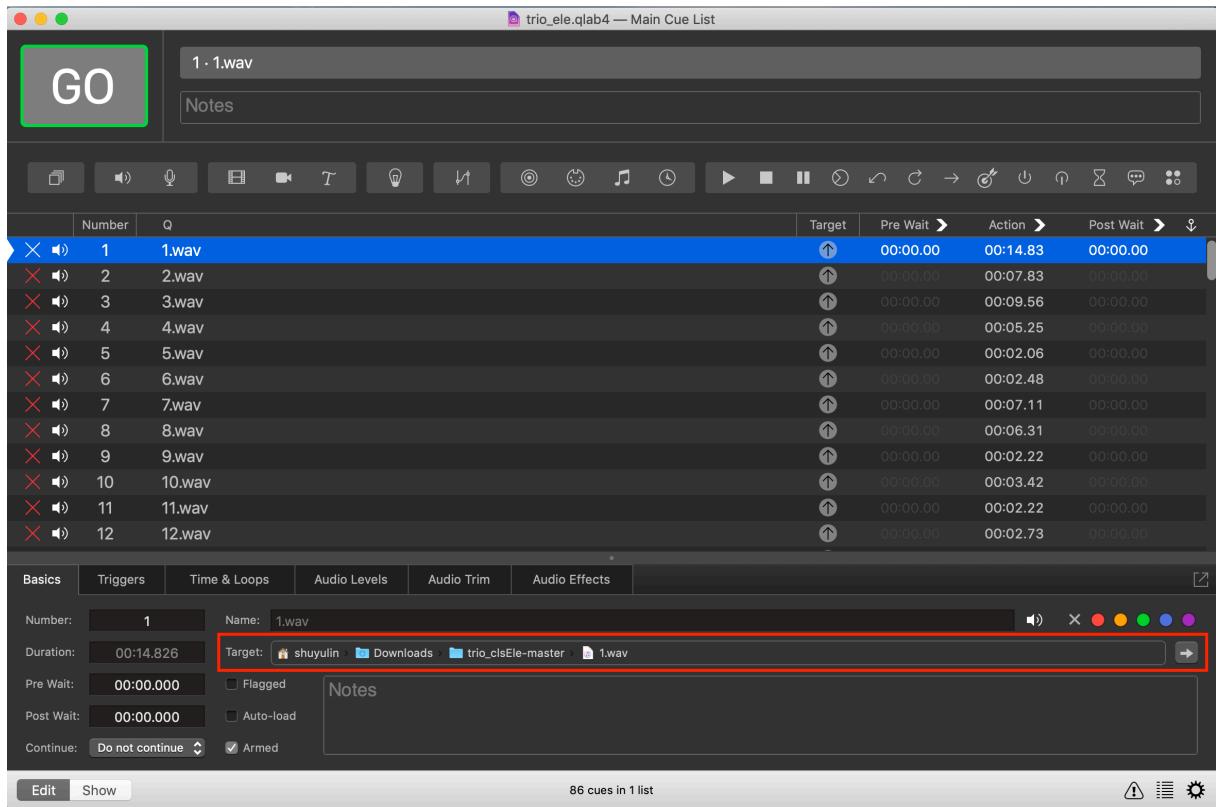


3. Open zip and move to a directory where you wanted it to be.

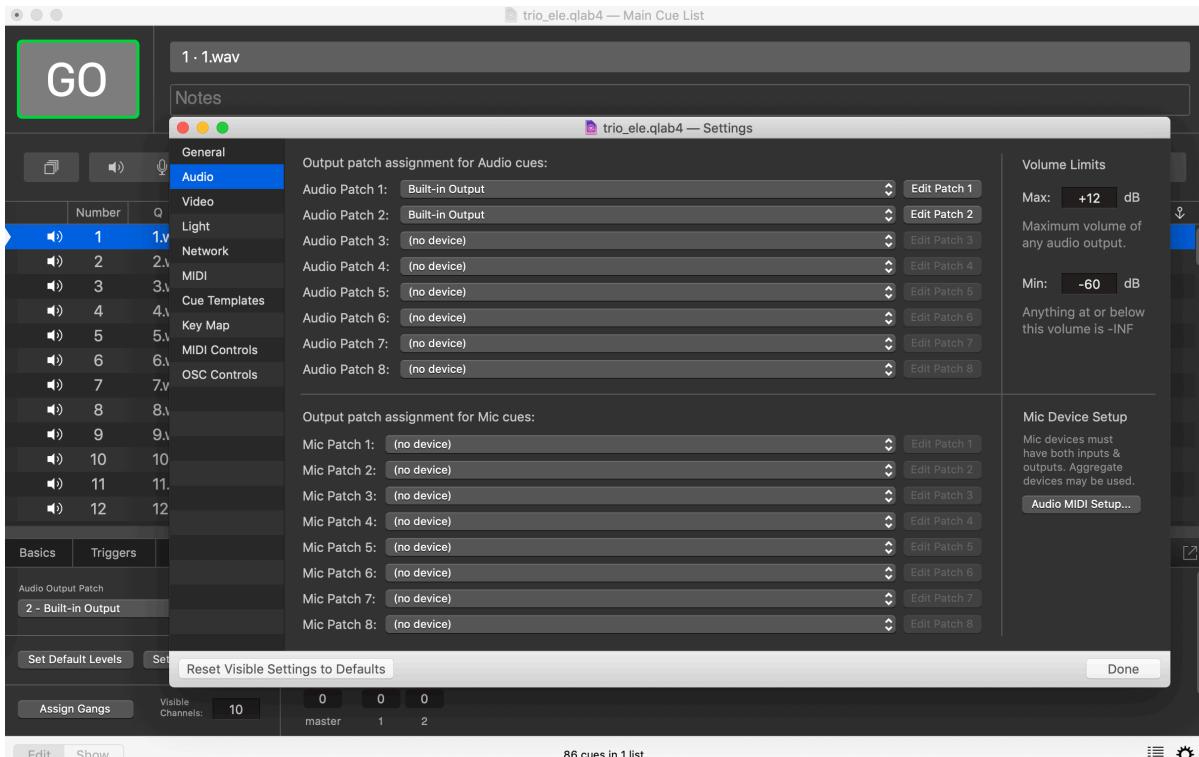
4. Click open the QLab file 'trio_ele.qlab4' in the folder.



5. Make sure the directory of each sound file is pointing to the corresponding sound track in the folder in the Basics tab. To change directory, click on the gray arrow at the end of the directory.

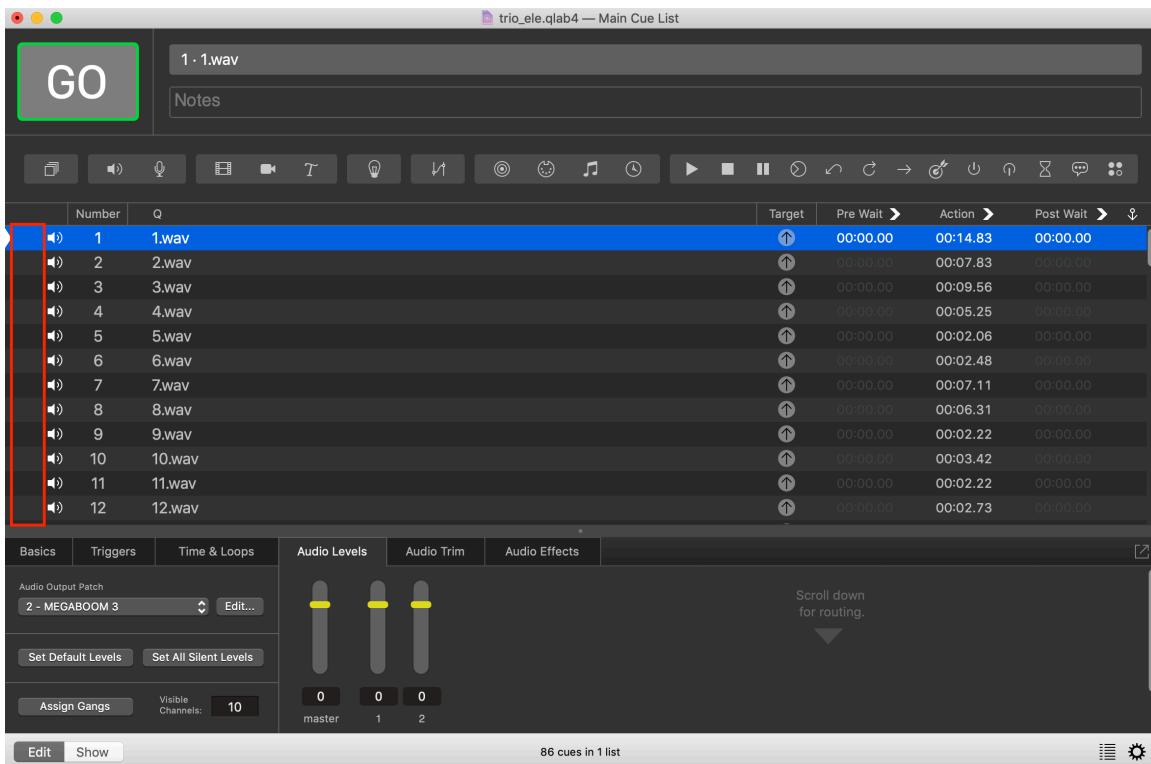


6. To resolve warnings marked with red X at the beginning of each file connect to an audio output. To do this, go to settings by clicking on the bottom right circular icon, go to Audio tab and change Audio Patch to your desired output hardware. Click on Done to save the configuration.



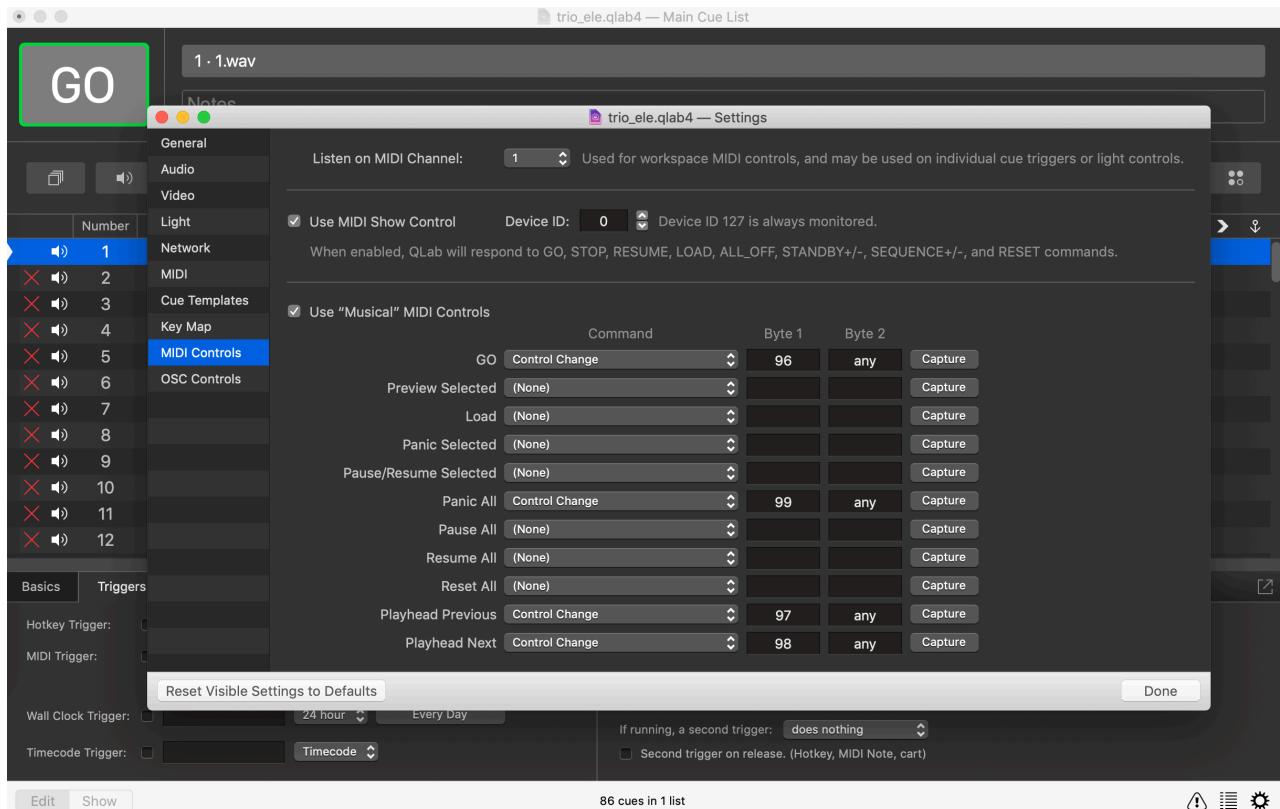
To perform using one Bluetooth speaker as Trio is designed for, turn on the Bluetooth speaker, and connect it to your computer (for Mac OS, System Preferences -> Bluetooth). Then, select the name of the Bluetooth speaker as your audio output. Click on Done to save the configuration.

After successfully setting the output, the warnings with red X marks should disappear.

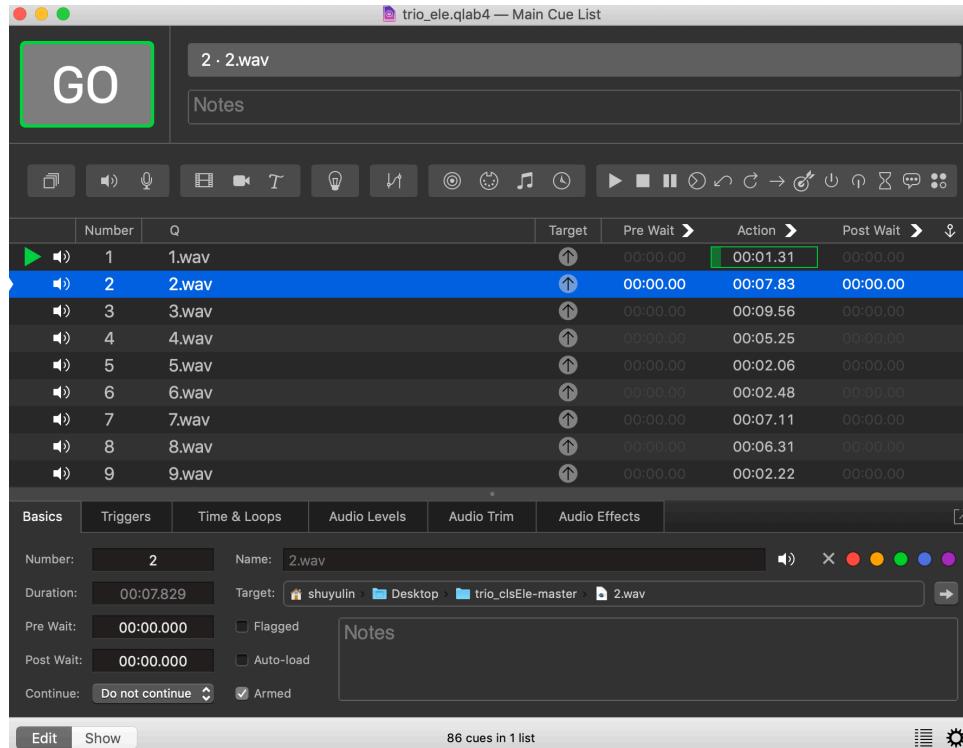


7. Connect a midi foot pedal to your computer. Make sure your foot pedal is connected with the Qlab program. Go to settings by clicking on the bottom right circular icon, go to MIDI Controls tab, click on Capture, and press the midi button of your choice. This would register the specific button to trigger a particular function. Midi controller should be configured in the way for your convenience when performing.

The example here demonstrates the midi configuration for ActitioN midi foot pedal, consisting of GO, Panic All (stop), Playhead Previous and Playhead Next, each correspond to a button of the controller.



To playback the sound files, step on the first button (labeled 'GO'). The pointer in QLab will automatically move to the second sound file while the first is playing.



If you need to return to the first sound file, step on 'previous' button. To go to the next sound file, step on the 'next' button. To stop playing back at any point, step on the 'stop' button.

SuperCollider code instruction

1. Download SuperCollider at <https://superollider.github.io/download> Make sure the version that you download is compatible with your computer's operating system. Click open the downloaded zip file and install SuperCollider.

The screenshot shows the GitHub repository page for SuperCollider. The 'Download' tab is highlighted in grey, indicating it is selected. Below the tab, there are links for 'News', 'Download', 'Wiki', 'Docs', 'Star' (3,988), 'Fork' (626), and 'Issues' (0). The main content area is titled 'Download'.

Download

Mac	Linux	Windows
Current Version	Current Version	Current Version
<ul style="list-style-type: none">↳ 3.12.1 (macOS 10.13 and later) (sn*)↳ 3.12.1 - legacy (macOS 10.10-10.12) (sn*)	<ul style="list-style-type: none">3.12.1 source tarball3.12.1 GPG signature	<ul style="list-style-type: none">↳ 3.12.1, 64-bit (sn*)↳ 3.12.1, 32-bit (sn*)
macOS builds are signed and notarized	Builds with gcc >= 6.3	Supports Windows 7, 8, 10
Previous Releases	Official Linux Packages	Previous releases
<ul style="list-style-type: none">↳ 3.11.2+BigSur.aed25fa (macOS 11) (sn*)↳ 3.11.2 (macOS 10.13-10.15) (sn*)↳ 3.11.2 - legacy (macOS 10.10-10.12) (sn*)↳ 3.10.4 - signed (sn*)↳ 3.10.4 (sn*)	<ul style="list-style-type: none">Arch LinuxDebianUbuntu	<ul style="list-style-type: none">3.11.2, 64-bit3.11.2, 32-bit3.10.4, 64-bit3.10.4, 32-bit3.9.3, 64-bit3.9.3, 32-bit3.8.1, 64-bit3.8.0, 32-bit (sn*)3.8.0, 32-bit (Sonic Pi build)3.7.2 (sn*, Win XP)
	Community packages and repositories	
	<ul style="list-style-type: none">• Arch Linux AUR• Ubuntu PPA• Fedora (PlanetCCRMA)• openSUSE (Open Build Service)	

2. Download files for electronics from github at https://github.com/shuyulin/trio_clsEle. Click on 'Code' and select 'Download ZIP'.

The screenshot shows the GitHub repository page for 'trio_clsEle'. The 'Code' tab is highlighted in orange, indicating it is selected. Below the tab, there are links for 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Security', and 'Insights'. A modal window titled 'Join GitHub today' is open, stating 'GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together.' with a 'Sign up' button. The main content area shows the repository details: 4 commits, 1 branch, 0 packages, 0 releases, and 1 contributor. The 'Branch: master' dropdown is set to 'master'. On the right, there is a 'Clone with HTTPS' button with the URL https://github.com/shuyulin/trio_clsEle. Below the URL, there are buttons for 'Open in Desktop' and 'Download ZIP'. The repository listing shows files 1.wav through 15.wav, each with a 'Add files via upload' link.

3. Open zip and move to a directory where you wanted it to be.

4. Click open the script file 'trio_electronics.scd'

5. Change buffer directory of pre-recorded 8_mono sound file (line 1368, 1434) and 1_mono sound file (line 1589, 2378) to the directory where both files are stored. 8_mono and 1_mono files are within the downloaded zip.

```
1367
1368 b = Buffer.read(s, "/Users/shuyulin/Desktop/trioForClsEle/electronics/scripts/8_mono.wav");
1369
```

6. Booth SuperCollider. Move cursor to the end of line 17 and press command + enter.

```

16
17 s.boot;
18
19 (
20 ~midiBlock = [
  ], [ [ 61, 69,

```

If SuperCollider is successfully booted, the parameters of Server at the lower right of the environment will show in green.

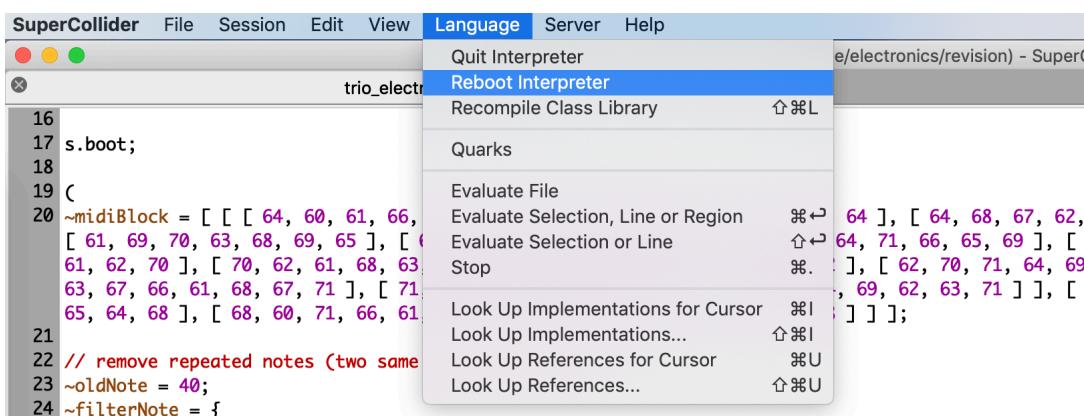
The screenshot shows the SuperCollider interface with the file 'trio_electronics.sc' open. The code contains several SynthDef definitions and a main block. A red box highlights the 'Interpreter' status bar at the bottom, which shows 'Active'. Below it, the 'Server' status bar displays resource usage: 0.05% CPU, 0.20% memory, 0u, 0s, 2g, 107d, and 0.0dB latency.

```

16
17 s.boot;
18
19 (
20 ~midiBlock = [ [ [ 64, 60, 61, 66, 71, 60, 68 ], [ 68, 60, 71, 66, 61, 60, 64 ], [ 64, 68, 67, 62, 69, 68, 60 ], [ 60, 68, 69, 62, 67, 68, 64 ]
  ], [ [ 61, 69, 70, 63, 68, 69, 65 ], [ 65, 69, 68, 63, 70, 69, 61 ], [ 61, 65, 64, 71, 66, 65, 69 ], [ 69, 65, 66, 71, 64, 65, 61 ] ], [ [ 66,
  62, 63, 68, 61, 62, 70 ], [ 70, 62, 61, 68, 63, 62, 66 ], [ 66, 70, 69, 64, 71, 70, 62 ], [ 62, 70, 71, 64, 69, 70, 66 ] ], [ [ 71, 67, 68, 61,
  66, 67, 63 ], [ 63, 67, 66, 61, 68, 67, 71 ], [ 71, 63, 62, 69, 64, 63, 67 ], [ 67, 63, 64, 69, 62, 63, 71 ] ], [ [ 68, 64, 65, 70, 63, 64, 60
  ], [ 60, 64, 63, 70, 65, 64, 68 ], [ 68, 60, 71, 66, 61, 60, 64 ], [ 64, 60, 61, 66, 71, 60, 68 ] ] ];
21
22 // remove repeated notes (two same notes in the row)
23 ~oldNote = 40;
24 ~filterNote = {
25 ~newNote = (~midiBlock[rrand(0, 4)][rrand(0, 3)][rrand(0, 6)]);
26 if(~newNote == ~oldNote, {
27 ~newNote = (~midiBlock[rrand(0, 4)][rrand(0, 3)][rrand(0, 6)]);
28 });
29 ~oldNote = ~newNote;
30 ~newNote;
31 };
32
33
34 SynthDef("playRandNotesSine", {
35   arg freq, env, amp = 0.8, out = 0, pan = 1; // pan = -1 = start from left
36   var sig, outArray, bus;
37   sig = SinOsc.ar(freq, 0, amp);
38   env = EnvGen.kr(Env.new([0, 1, 0.3, 0], [0.0001, 0.1, 1.8], [-5, 0, -3]), doneAction: 2);
39   outArray = sig * env * amp; // only 1 signal in the array
40   outArray = Pan2.ar(outArray, pan); // add this line for panning
41   Out.ar(out, outArray); // output to left channel
42 }).add(); // add to server to be played later
43
44 SynthDef("playRandNotesSine2", {
45   arg freq, env, amp = 0.8, out = 0, pan = 1; // pan = -1 = start from left
46   var sig, outArray, bus;
47   sig = SinOsc.ar(freq, 0, amp);

```

The Interpreter should indicate Active after you opened the .scd file. Reboot the Interpreter if it is inactive. To do this, go to Language drop down menu and select Reboot Interpreter.



After rebooting the interpreter, you will need to boot the server again (line 17).

7. Move cursor to the beginning of line 19, lines 19 to 153 will be automatically highlighted. Run these lines by pressing command + enter. This is to store variables and functions for sound processes.

```
17 s.boot;
18
19 C
20 ~midiBlock = [ [ [ 64, 60, 61, 66, 71, 60, 68 ], [ 68, 60, 71, 66, 61, 60, 64 ], [ 64, 68, 67, 62, 69, 68, 60 ], [ 60, 68, 69, 62, 67, 68, 64 ] ],
  [ [ 61, 69, 70, 63, 68, 69, 65 ], [ 65, 69, 68, 63, 70, 69, 61 ], [ 61, 65, 64, 71, 66, 65, 69 ], [ 69, 65, 66, 71, 64, 65, 61 ] ], [ [ 66,
  62, 63, 68, 61, 62, 70 ], [ 70, 62, 61, 68, 63, 62, 66 ], [ 66, 70, 69, 64, 71, 70, 62 ], [ 62, 70, 71, 64, 69, 70, 66 ] ], [ [ 71, 67, 68, 61,
  66, 67, 63 ], [ 63, 67, 66, 61, 68, 67, 71 ], [ 71, 63, 62, 69, 64, 63, 67 ], [ 67, 63, 64, 69, 62, 63, 71 ] ], [ [ 68, 64, 65, 70, 63, 64, 60
  1, [ 60, 64, 63, 70, 65, 64, 68 ], [ 68, 60, 71, 66, 61, 60, 64 ], [ 64, 60, 61, 66, 71, 60, 68 ] ] ];
21 ]
```

8. Run the subsequent code labeled by numbers, which correspond to the sound file number of electronics part. For example, to playback the sound of sound file number 1, move the cursor to the beginning of line 161, highlight the corresponding lines of code that used to generate sound file number 1, and press command + enter to start processes.

```
158 //// 1
159 // cloud rand register
160 // 14" with fade out and rit
161 {
162   var tempo = 1;
163   ~transposeArrIn1 = Array.fill(750, {rrand(-108, 108)});
164   ~ampArrIn1 = Array.fill(750, {rrand(0.0005, 0.5)});
165 }
```

If you wish to stop the sound before its end, press command + . (period).

If you are unable to run certain processes for unknown reason, recompile class library, go to Language and click on Recompile Class Library or press shift + command + l. After doing this, you will need to boot the Server (line 17), and run lines 19 to 153 again to store variables and functions.

Note, once you have ran lines 19 to 153, the running of the subsequent parts of the code do not have to be in sequence. You have the liberty of generating and hearing the particular sound of your choice directly from the SuperCollider code.

9. To close the script, click on the x at upper left of the tab (or command + w). If a pop up window appears asking if you would want to save the changes, click on 'Don't save'. This would make sure the next time you open the script it is the same as the first time when you opened it. Click on the x at the upper left of the program window to close SuperCollider program (or command + q).