A Study for Violin and Electronics

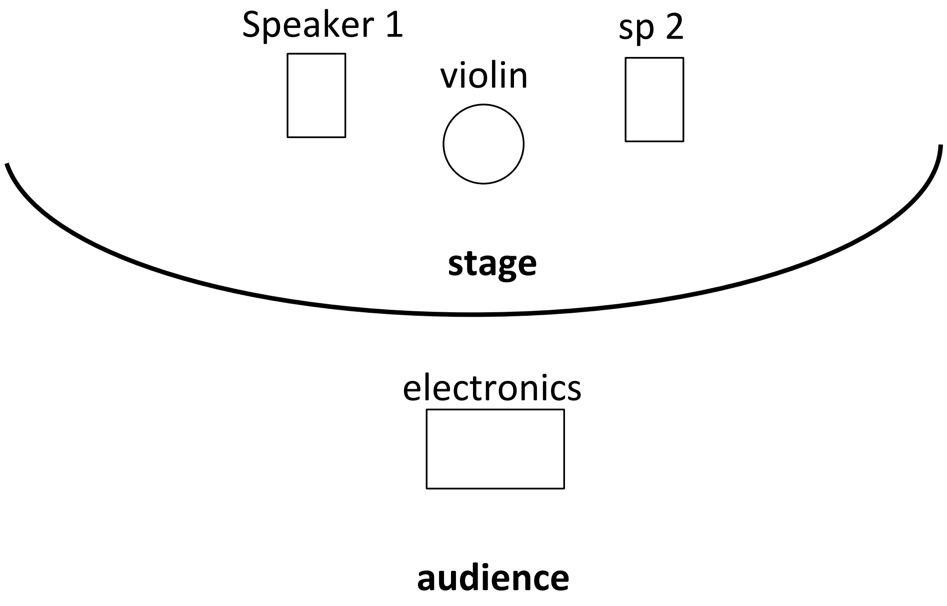
Electronics documentation

Shu-Yu Lin

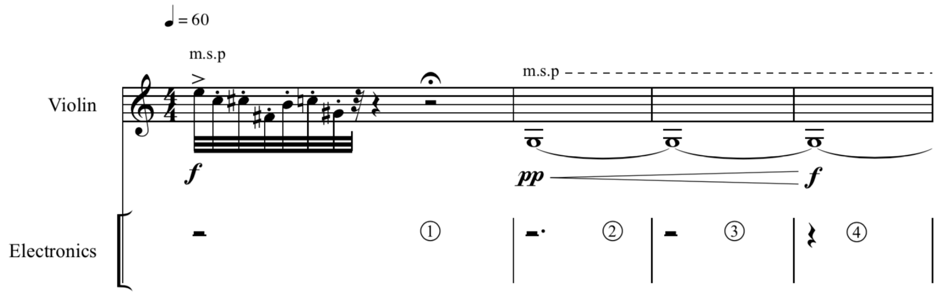
Table of Contents

Technical specification
2 speakers (stereo left and right)
Laptop computer
SuperCollider programming environment
Audio interface with at least 2 outs

Stage plan



Performance instruction
The numbers on the electronics part on the score refers to the specific sound file. 1 refers to sound numbered 1, 2 refers to sound 2 etc. There are a total of 28 sounds.



The electronics performer will manage the SuperCollider code and trigger specific sound at indicated time according to the score. He/she can be either on stage or off stage. The stage plan indicated the possible position of the electronics performer off the stage. If performing on the stage, he/she should sit or stand beside the violinist, either on the left or right, without blocking the speakers.

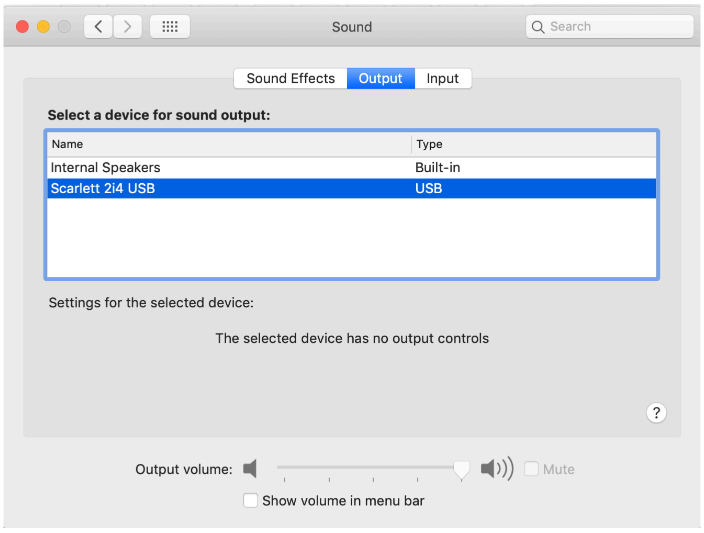
Hardware setup

1. Place stereo pair of speakers on the left and right side on the stage. The stage plan shows the possible position relative to the position of the violinist.

2. Connect audio interface to your computer. If you are using Focusrite Scarlett 2i4 interface, connect through USB.

3. Connect stereo pair of speaker to the audio interface. Amplifier will be required if you are using passive speakers. Connect a monitor headphone to the audio interface and make sure the output volume is appropriate.

4. In your computer, select sound output to be the name of your audio interface. If you use a Mac OS, go to Systems Preferences -> Sound -> Output -> [name of the audio interface]. In the example below, the audio interface selected is Scarlett 2i4 USB.



Select a device for sound output:

Name	Type
Internal Speakers	Built-in
Scarlett 2i4 USB	USB

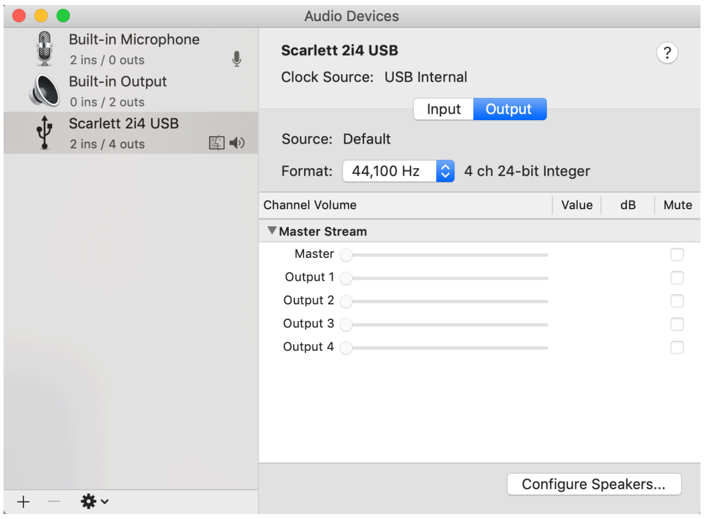
Settings for the selected device:

The selected device has no output controls

Output volume: Mute

Show volume in menu bar

5. Open Audio MIDI Setup application in your computer. Select Scarlett 2i4 USB in the left panel. Then, select Output tab. In the Format drop down menu, select 44,100 Hz sample rate.



Scarlett 2i4 USB

Clock Source: USB Internal

Source: Default

Format: 44,100 Hz 4 ch 24-bit Integer

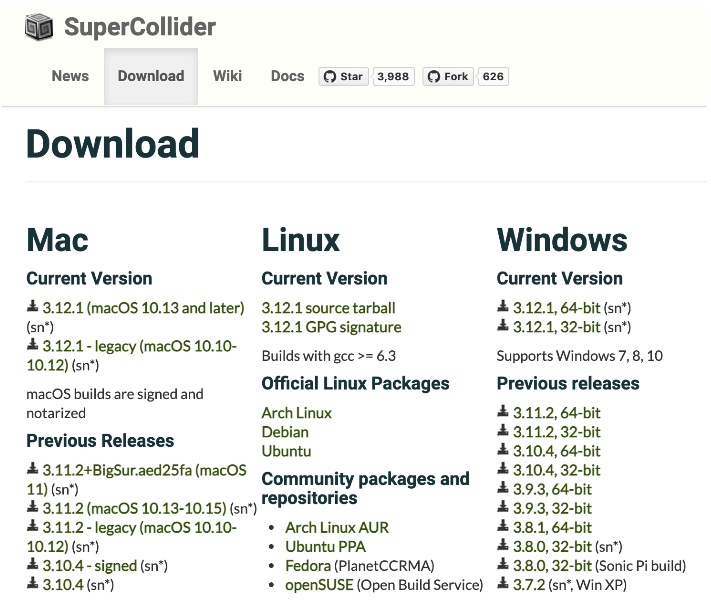
Channel Volume	Value	dB	Mute
Master Stream			
Master			
Output 1			
Output 2			
Output 3			
Output 4			

6. Test connection by play back any sound file, or run the SuperCollider code to generate sound. If there is connection issue between the speaker and the computer, you should still be able to hear the playback through the monitor headphone.

For setting up SuperCollider programming environment and running the code for the piece, please see the next section.

SuperCollider code instruction

1. Download SuperCollider at https://supercollider.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.


SuperCollider

News Download Wiki Docs Star 3,988 Fork 626

Download

Mac

Current Version

3.12.1 (macOS 10.13 and later) (sn*)
3.12.1 - legacy (macOS 10.10-10.12) (sn*)

macOS builds are signed and notarized

Previous Releases

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*)

Linux

Current Version

3.12.1 source tarball
3.12.1 GPG signature

Builds with gcc >= 6.3

Official Linux Packages

Arch Linux
Debian
Ubuntu

Community packages and repositories

- Arch Linux AUR
- Ubuntu PPA
- Fedora (PlanetCCRMA)
- openSUSE (Open Build Service)

Windows

Current Version

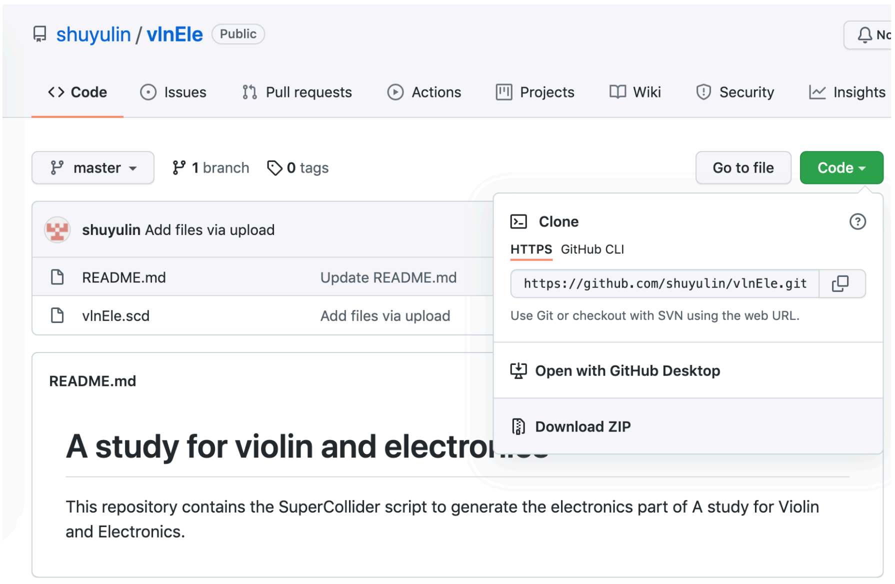
3.12.1, 64-bit (sn*)
3.12.1, 32-bit (sn*)

Supports Windows 7, 8, 10

Previous releases

3.11.2, 64-bit
3.11.2, 32-bit
3.10.4, 64-bit
3.10.4, 32-bit
3.9.3, 64-bit
3.9.3, 32-bit
3.8.1, 64-bit
3.8.0, 32-bit (sn*)
3.8.0, 32-bit (Sonic Pi build)
3.7.2 (sn*, Win XP)

2. Go to github https://github.com/shuyulin/vlnEle. Click on 'Code' and select 'Download ZIP' to download SuperCollider script.



shuyulin Add files via upload

README.md Update README.md
vlnEle.scd Add files via upload

```
https://github.com/shuyulin/vlnEle.git
```

Use Git or checkout with SVN using the web URL.

README.md

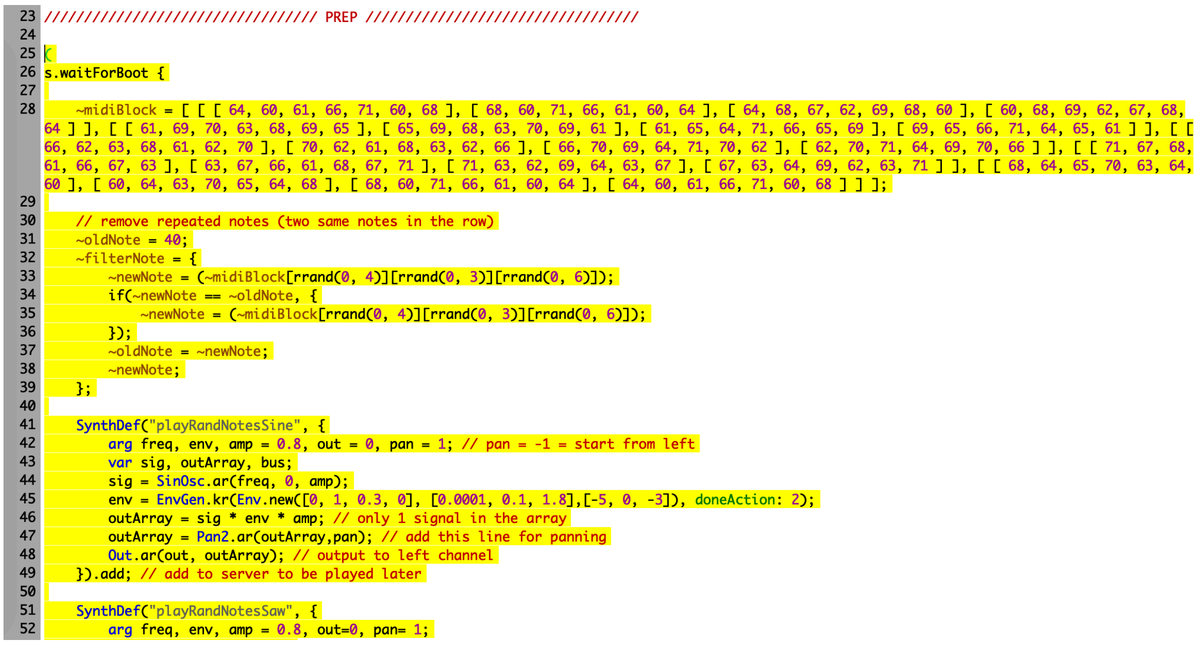
A study for violin and electron...

This repository contains the SuperCollider script to generate the electronics part of A study for Violin and Electronics.

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

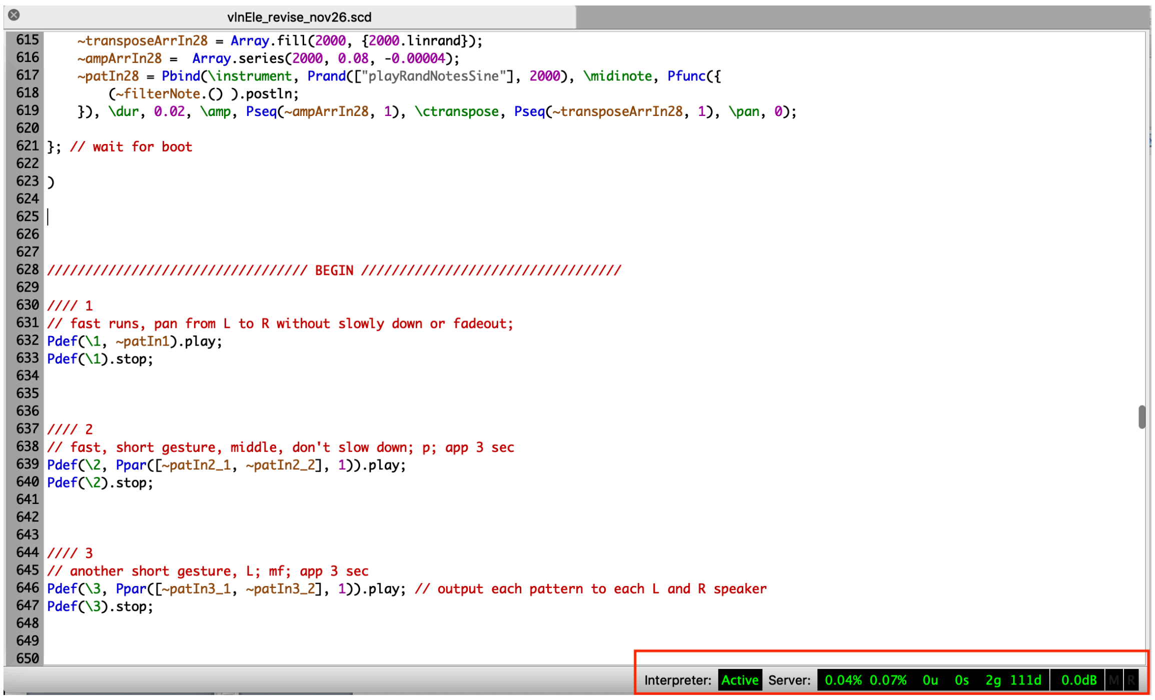
4. Click open the script file 'vlnEle.scd'

5. Move cursor to the beginning of line 25. This will highlight lines 25 to 623. Run the line by pressing command + enter. This will boot the Server and store necessary variables and functions for sound processes.



```
23 /////////////////////////////////// PREP ///////////////////////////////////
24
25 (
26 s.waitForBoot {
27
28     ~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 ] ] ];
29
30     // remove repeated notes (two same notes in the row)
31     ~oldNote = 40;
32     ~filterNote = {
33         ~newNote = (~midiBlock[rrand(0, 4)][rrand(0, 3)][rrand(0, 6)]);
34         if(~newNote == ~oldNote, {
35             ~newNote = (~midiBlock[rrand(0, 4)][rrand(0, 3)][rrand(0, 6)]);
36         });
37         ~oldNote = ~newNote;
38         ~newNote;
39     };
40
41     SynthDef("playRandNotesSine", {
42         arg freq, env, amp = 0.8, out = 0, pan = 1; // pan = -1 = start from left
43         var sig, outArray, bus;
44         sig = SinOsc.ar(freq, 0, amp);
45         env = EnvGen.kr(Env.new([0, 1, 0.3, 0], [0.0001, 0.1, 1.8],[-5, 0, -3]), doneAction: 2);
46         outArray = sig * env * amp; // only 1 signal in the array
47         outArray = Pan2.ar(outArray,pan); // add this line for panning
48         Out.ar(out, outArray); // output to left channel
49     }).add; // add to server to be played later
50
51     SynthDef("playRandNotesSaw", {
52         arg freq, env, amp = 0.8, out=0, pan= 1;
```

If the SuperCollider Server is successfully booted up, the lower right parameters should turn on to green.

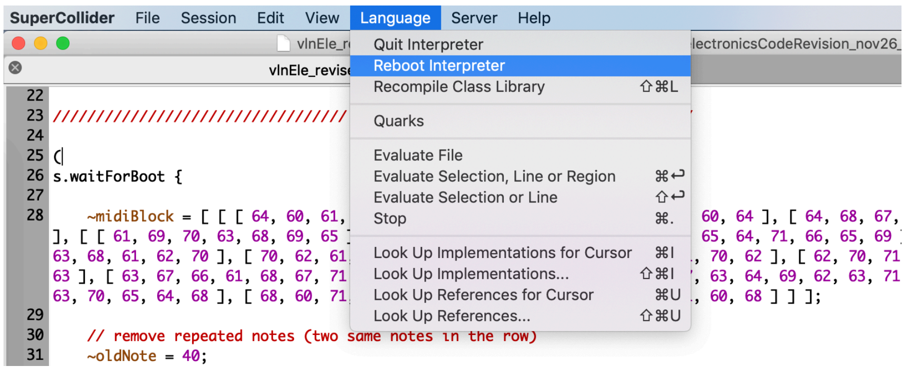


vlnEle_revise_nov26.scd

```
615     ~transposeArrIn28 = Array.fill(2000, {2000.linrand});
616     ~ampArrIn28 =  Array.series(2000, 0.08, -0.00004);
617     ~patIn28 = Pbind(\instrument, Prand(["playRandNotesSine"], 2000), \midinote, Pfunc({
618         (~filterNote.() ).postln;
619     }), \dur, 0.02, \amp, Pseq(~ampArrIn28, 1), \ctranspose, Pseq(~transposeArrIn28, 1), \pan, 0);
620
621 }; // wait for boot
622
623 )
624
625
626
627
628 ///////////////////////////////// BEGIN /////////////////////////////////
629
630 //// 1
631 // fast runs, pan from L to R without slowly down or fadeout;
632 Pdef(\1, ~patIn1).play;
633 Pdef(\1).stop;
634
635
636
637 //// 2
638 // fast, short gesture, middle, don't slow down; p; app 3 sec
639 Pdef(\2, Ppar([~patIn2_1, ~patIn2_2], 1)).play;
640 Pdef(\2).stop;
641
642
643
644 //// 3
645 // another short gesture, L; mf; app 3 sec
646 Pdef(\3, Ppar([~patIn3_1, ~patIn3_2], 1)).play; // output each pattern to each L and R speaker
647 Pdef(\3).stop;
648
649
650
```

Interpreter: Active Server: 0.04% 0.07% 0u 0s 2g 111d 0.0dB M R

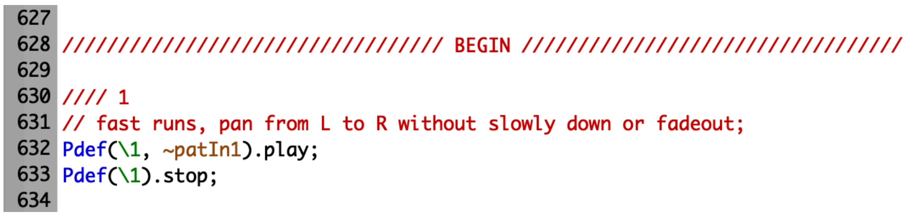
The Interpreter should indicate Active after you opened the .scd file. Reboot the Interpreter if it is inactive. To do this, go to Language drop down menu and select Reboot Interpreter.


SuperCollider File Session Edit View Language Server Help
vlnEle_r
lectronicsCodeRevision_nov26_
vlnEle_revis
Quit Interpreter
Reboot Interpreter
Recompile Class Library ⇧⌘L
Quarks
Evaluate File
Evaluate Selection, Line or Region ⌘↩
Evaluate Selection or Line ⇧↩
Stop ⌘.
Look Up Implementations for Cursor ⌘I
Look Up Implementations... ⇧⌘I
Look Up References for Cursor ⌘U
Look Up References... ⇧⌘U

```
22
23 /////////////////////////////////
24
25 (
26 s.waitForBoot {
27
28     ~midiBlock = [ [ [ 64, 60, 61,                60, 64 ], [ 64, 68, 67,
], [ [ 61, 69, 70, 63, 68, 69, 65                 65, 64, 71, 66, 65, 69
63, 68, 61, 62, 70 ], [ 70, 62, 61                , 70, 62 ], [ 62, 70, 71
63 ], [ 63, 67, 66, 61, 68, 67, 71                , 63, 64, 69, 62, 63, 71
63, 70, 65, 64, 68 ], [ 68, 60, 71                , 60, 68 ] ] ];
29
30     // remove repeated notes (two same notes in the row)
31     ~oldNote = 40;
```

After rebooting the interpreter, you will need to run lines 25 to 623 again, to boot up the Server.

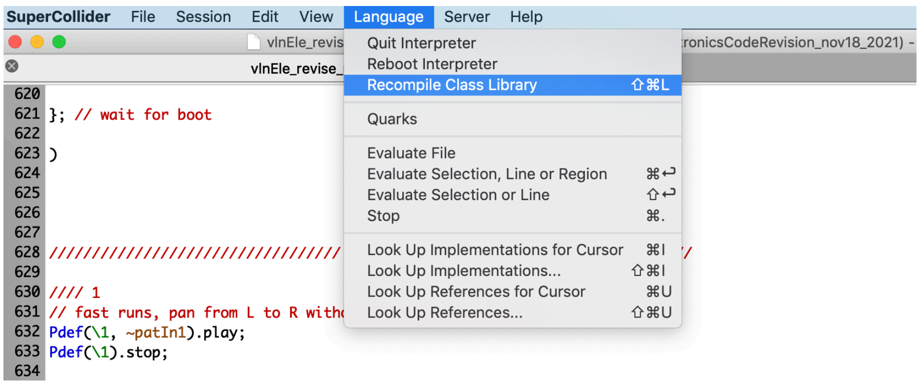
6. To run the sound of specific number, go to the corresponding number from line 628 to the end of the script. For example, to run sound numbered 1, move the cursor to the end of line 632, and press command + enter to begin the sound. To stop the sound before its end, run line 633.



```
627
628 ///////////////////////////////// BEGIN /////////////////////////////////
629
630 //// 1
631 // fast runs, pan from L to R without slowly down or fadeout;
632 Pdef(\1, ~patIn1).play;
633 Pdef(\1).stop;
634
```

To stop all the processes in SuperCollider, press command + . (period). Command + . will not produce a fadeout at the end of the sound. You may consider using command + . if you want to end a sound, or found yourself running the wrong sound to stop the processes all together.

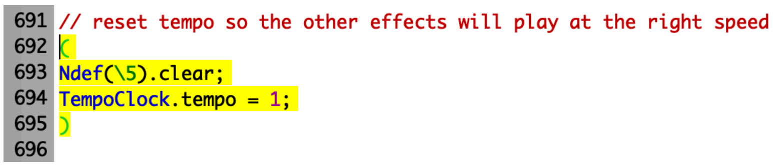
If you are unable to run certain processes for unknown reason, recompile class library. To do this, go to Language and select Recompile Class Library, or press shift + command + l.


SuperCollider File Session Edit View Language Server Help
vlnEle_revis
ronicsCodeRevision_nov18_2021) -
vlnEle_revise_
Quit Interpreter
Reboot Interpreter
Recompile Class Library ⇧⌘L
Quarks
Evaluate File
Evaluate Selection, Line or Region ⌘↩
Evaluate Selection or Line ⇧↩
Stop ⌘.
Look Up Implementations for Cursor ⌘I
Look Up Implementations... ⇧⌘I
Look Up References for Cursor ⌘U
Look Up References... ⇧⌘U

```
620
621 }; // wait for boot
622
623 )
624
625
626
627
628 /////////////////////////////////
629
630 //// 1
631 // fast runs, pan from L to R with
632 Pdef(\1, ~patIn1).play;
633 Pdef(\1).stop;
634
```

After this, you will need to run lines 25 to 623 again to boot the Server and store variables and functions before you can run a specific sound.

7. Tempo needs to be reset after running processes numbered 5 (//// 5) at line 660, 7 (//// 7) at line 730 and 8 (//// 8) at line 763. For example, to reset tempo after running sound numbered 5 (////5), run lines 692-695. Likewise, run lines 755-758 to reset tempo after 7 and lines 788-791 after 8.



```
691 // reset tempo so the other effects will play at the right speed
692 (
693 Ndef(\5).clear;
694 TempoClock.tempo = 1;
695 )
696
```

8. 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).