

for One Performer and Live Electronics
Luca Spanedda
2021

Note

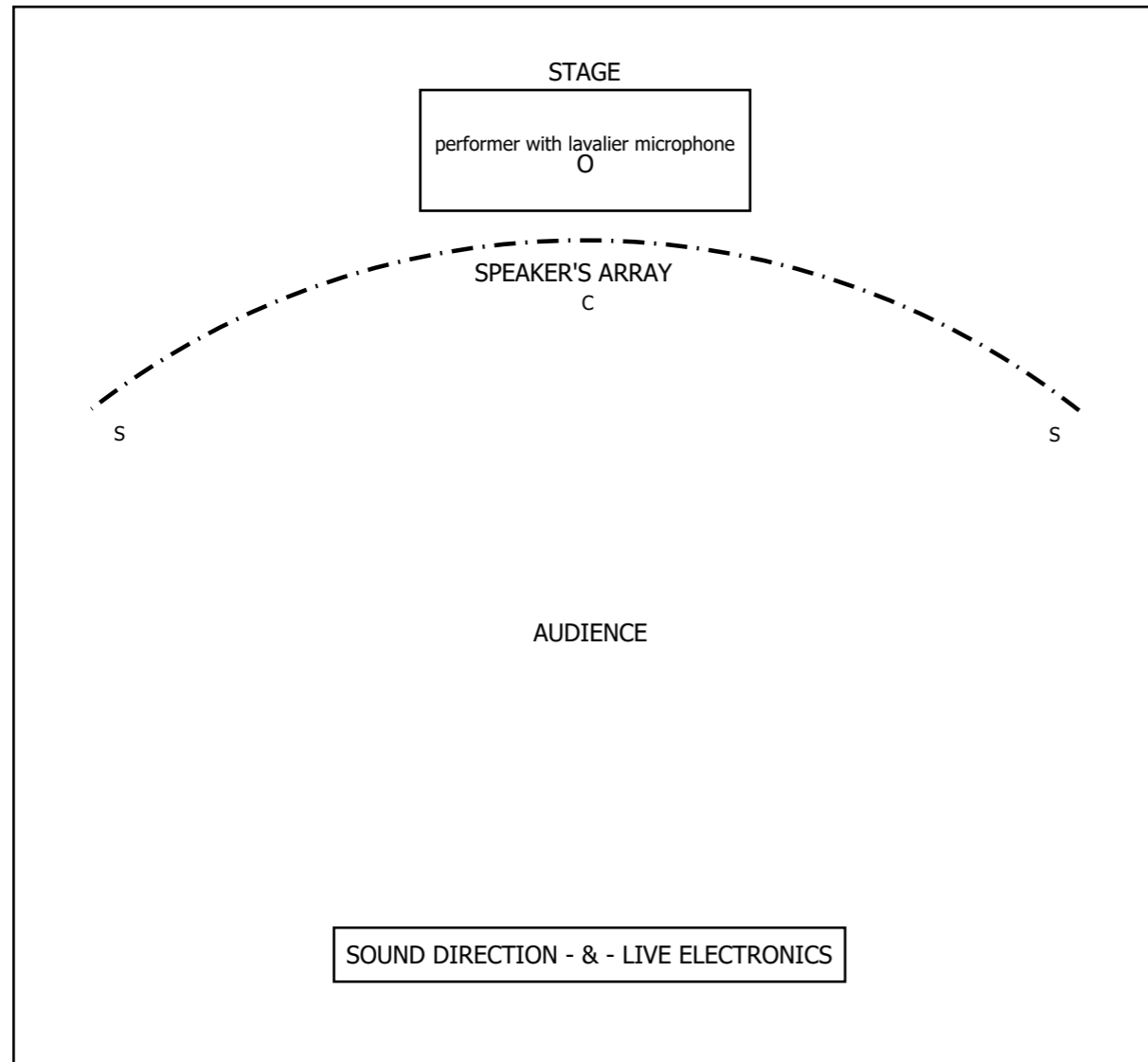
Deus ex machina is a piece for one performer and live electronics that aims to represent *logos* from a postmodern point of view. The reason for this composition is rooted in a critical will towards cultural values and the fundamental assumptions shared by Western society, continuously subject to evident short circuits and malfunctions to the detriment of the individuals who compose it.

The short circuit - infinite gain feedback - leads to self-destruction of the system that generates it, and it is the central process on which the dialogue between the voice and the electronics is based. In the performance the voice is electronically processed to collapse on itself, implode.

The birth of the verb, - expressive freedom that corresponds to an origin of the vocality - sought by the performer, it is therefore continually ostracized and hindered by this artificial machine, which is the result of the memory of the voice itself. Electronics and voice are in an intertwined relationship of dualism, where one influences the behaviour of the other.

Deus Ex Machina is a ritual.

Arrangement - Room



*The Amplification is in wavefield synthesis.
electronic slow motion ($\leftarrow \rightleftarrows$)($\leftarrow x$) = distant
electronic fast motion ($\leftarrow \rightarrow$) = close*

where:

- distant is a distribution between all the speaker's array*
- close is the position of the speaker's array near to the performer*

Luca Spanedda

Deus Ex Machina
for One Performer and Live Electronics
2021

² Voice's Semiography

▣ - inhale with timbre

◻ - inhale without timbre

◆ - exhale with timbre

◇ - exhale without timbre

⊕ - with closed mouth

⊙ - with half-open mouth

○ - with open mouth

● - throaty sound

✂ - coughing shots

⤴ - sobs of tears

Ⓜ - laughing

GRWL - growl

SCRM - scream

Ⓐ Ⓜ - text to perform

Ⓜ morphing ⤴ - gradual morphing between two musical gestures

Electronic's Semiography

* schemes for the implementation of the algorithm at the end of the score

☞ → - triggers from 1 to 40 in the indicated time, then stop (this fills all the memory banks)

☞ ↔ - continuous triggers at a speed managed by the interpreter in the indicated time (listening)

☞ X - stop sending triggers and leave the content in memory

sceneN - scenes to activate when indicated in the score during the performance

Indications

Ⓜ The markers of time windows or events must be respected.

Within these the rhythmic relations of the notation are freely interpretable

0'00" | 0'04" | 0'05" | 0'25" | 0'35" | 0'36" | 0'42" | 0'44" | 0'50" | 0'52" | 0'57" | 0'59" | 1'03" | 1'05" | 1'10" | 1'11" | 1'14" | 1'15" | 1'19.5" | 1'20" | 1'23.5" | 1'24" | 1'27.5" | 1'28" | 1'33" | 1'34"

Voice: **scene0** default scene, **sfz** freeze the scream, **scene1**, until saturation, **scene2** until desaturation.

always legato, **scene2b** slowly, emptying the memory banks in 50", **p**, **p**, **mf**.

GRWL, **f**, **f**, **f**, **f**, **f**, **f**, **f**.

sfz, **sfz**, **ff**, **sfz**, **fff**.

SCRM, **fff**, **p**.

1'37" 1'38" 1'48" 1'54"

scene3

pp *f* *f* *sfz* *sfz* *ppp*

1'47" 1'53" 1'58"

2'00" 2'02" 2'10" 2'15"

f *mf* *f* *mf* *p* *p* *pp*

stop legato

2'09" 2'13"

2'19" 2'20" 2'24" 2'28" 2'32"

scene4

2'36" 2'37" 2'43" 2'44"

p *sfz* *sfz*

2'47" 2'48" 2'52"

f *ff* *ff* *fff* *fff*

accelerando...

2'55" 2'56" 3'00" 3'05"

disperato

mf

mf

2'59.5"

scene5

3'06" 3'16"

breaking up the sound

morphing

sfz

3'24" 3'32"

morphing

sfz

h

3'38" rythm accelerando... ...until continuous sound...

3'47" 3'49"

scene6

3'50" 3'59.5"

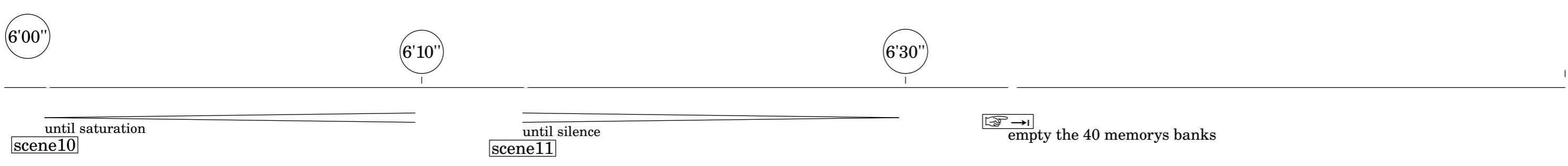
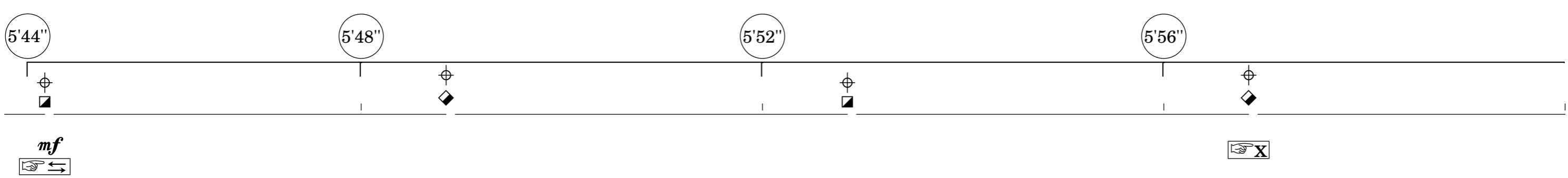
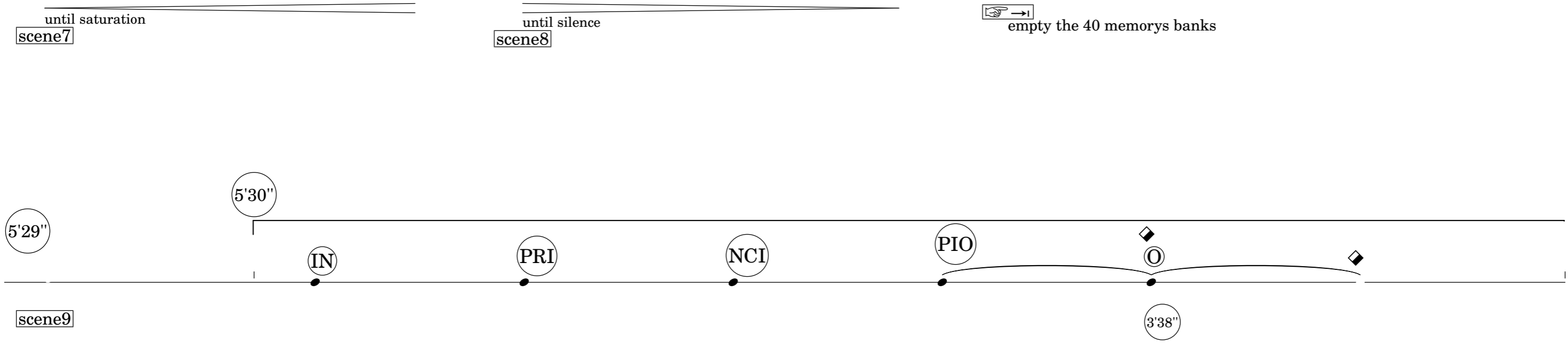
4'00" 4'09.5"

4'10" 4'15" 4'16" 4'17" 4'18" 4'21"

4'22" 4'29.5" 4'30" 4'37"

4'38" 4'37"

4'38" 4'37"



7'22"

let the freeze play;
meanwhile the performer leaves the stage

7'52"

scene12b
immediate stop after about 30"

Live Electronics

The Live Electronics is divided into two steps: the scenes to be activated that change the configuration of the algorithm, the digital implementation of the latter for the processing of the voice in real time.

Scenes

scene 0; direct 1; feedback 0; grainpitch 1; maskpitch 0; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 1; directrev 0;

scene 1; interpolation 20000; direct 0; feedback 1; grainpitch 2; maskpitch 0; maskpitchint 100; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 1; noisephase 10;

scene 2; interpolation 10000; direct 0; feedback 1; grainpitch 1; maskpitch 0; maskpitchint 0; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 1; directrev 1; noisephase 0;

scene 2b; interpolation 1000; direct 0; feedback 0; grainpitch 1; maskpitch 0; maskpitchint 0; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 1; directrev 1;

scene 3; interpolation 1000; direct 1; feedback 0; grainpitch 1; maskpitch 0; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 1; directrev 0;

scene 4; interpolation 6000; direct 1; feedback 4; grainpitch 1; maskpitch 0; grainenv 1; maskenv 10; grainamp 1; maskamp 1; revamp 0;

scene 5; interpolation 6000; direct 1; feedback 1; grainpitch 1; maskpitch 0; grainenv 1; maskenv 1; grainamp 1; maskamp 1; revamp 0;

scene 6; interpolation 6000; direct 1; feedback 0; grainpitch 1; maskpitch 0; grainenv 1; maskenv 1; grainamp 1; maskamp 1; revamp 0;

scene 7; interpolation 40000; direct 1; feedback 0; grainpitch 1; maskpitch 10000; grainenv 1; maskenv 1; grainamp 1; maskamp 1; revamp 3;

scene 8; interpolation 20000; direct 1; feedback 0; grainpitch 1; maskpitch 0; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 0;

scene 9; interpolation 0; direct 1; feedback 1; grainpitch 1; maskpitch 0; grainenv 1; maskenv 1; grainamp 1; maskamp 1; revamp 0;

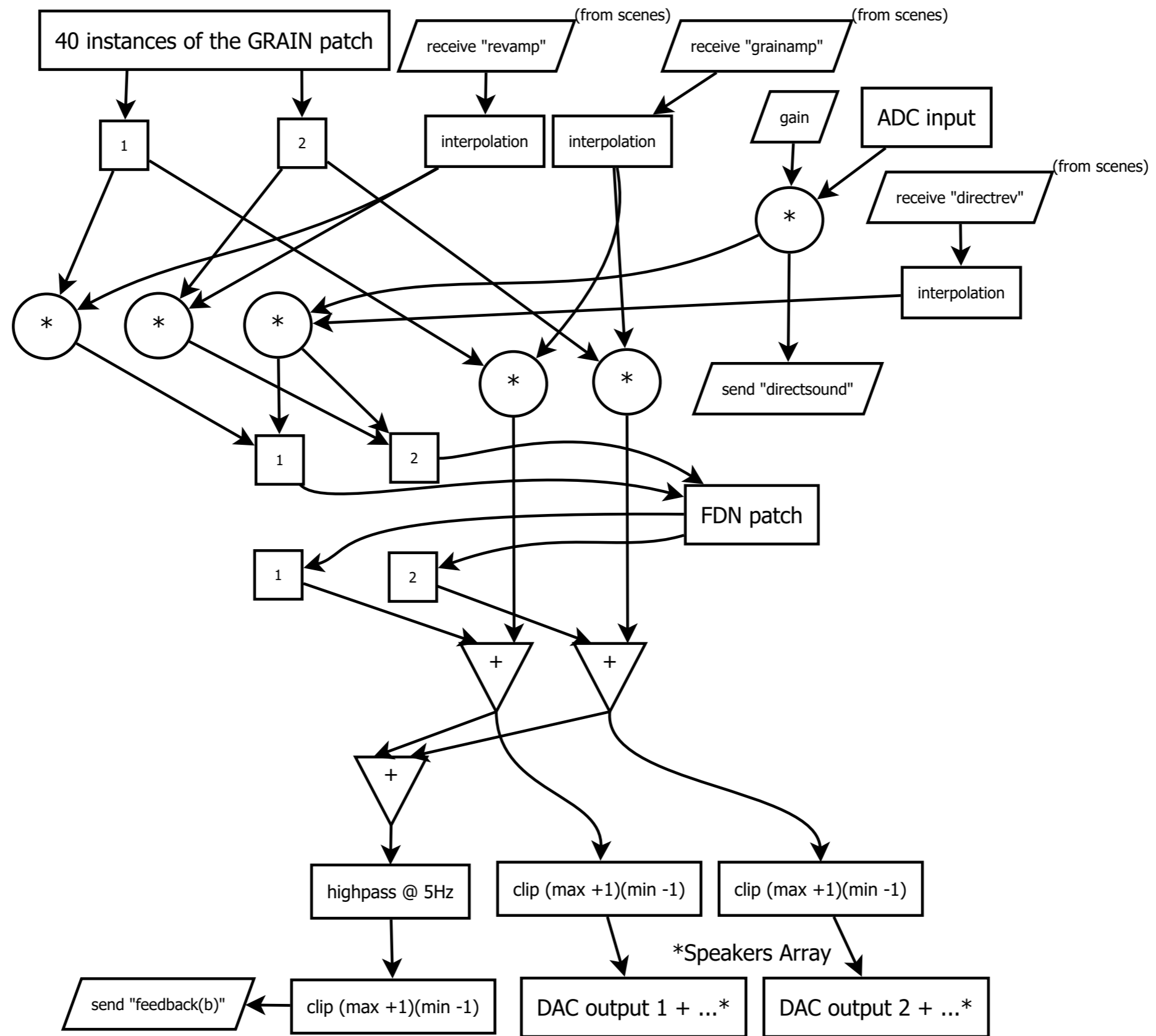
scene 10; interpolation 10000; direct 0; feedback 8; grainpitch 1; maskpitch 0; grainenv 10; maskenv 1000; grainamp 4; maskamp 1; revamp 0;

scene 11; interpolation 20000; direct 0; feedback 0; grainpitch 1; maskpitch 0; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 0;

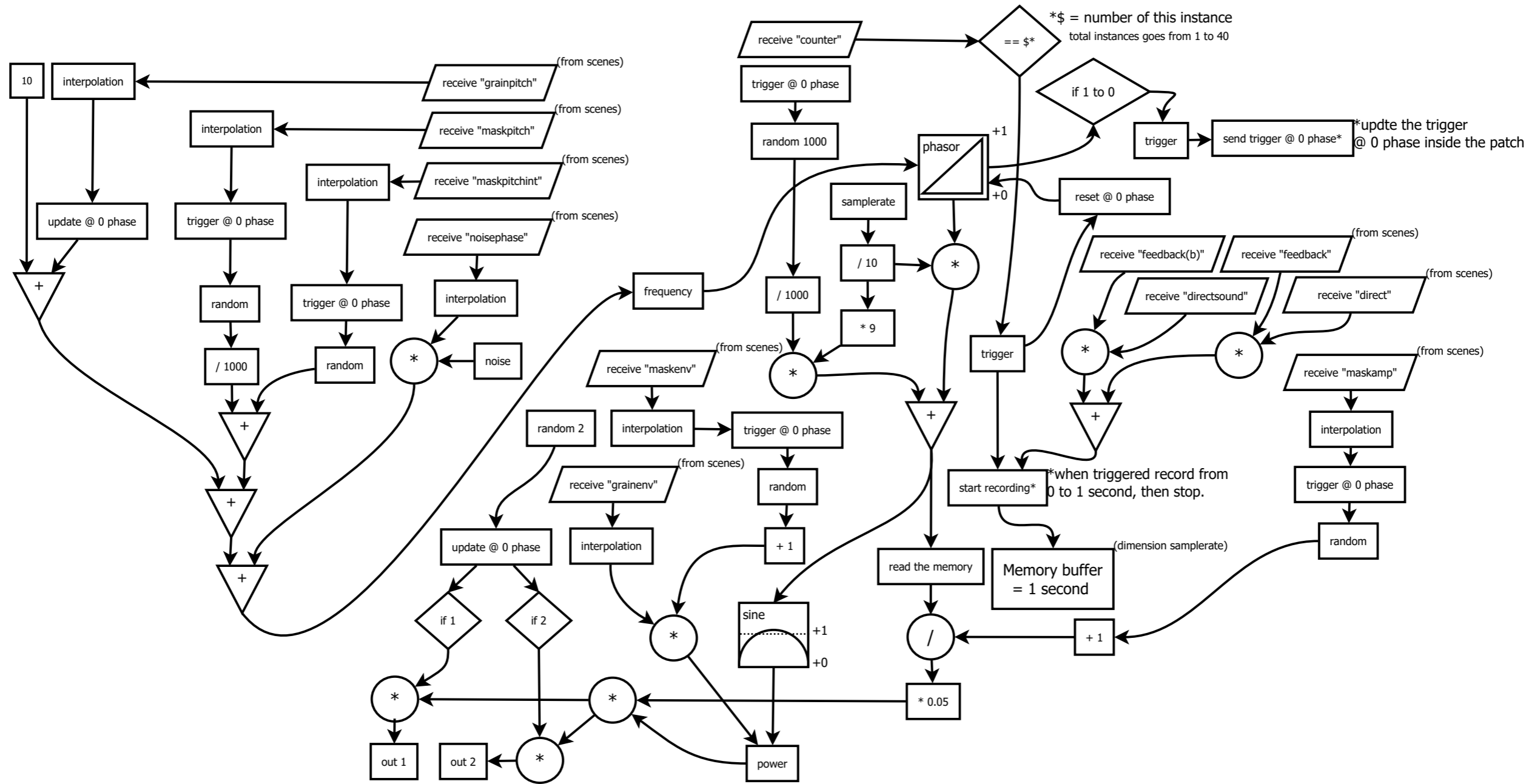
scene 12; interpolation 0; direct 1; feedback 1; grainpitch 2; maskpitch 0; grainenv 1; maskenv 1; grainamp 0; maskamp 1; revamp 1;

scene 12b; interpolation 4; direct 0; feedback 0; grainpitch 1; maskpitch 0; grainenv 0; maskenv 0; grainamp 0; maskamp 0; revamp 0;

"Main Routing" patch

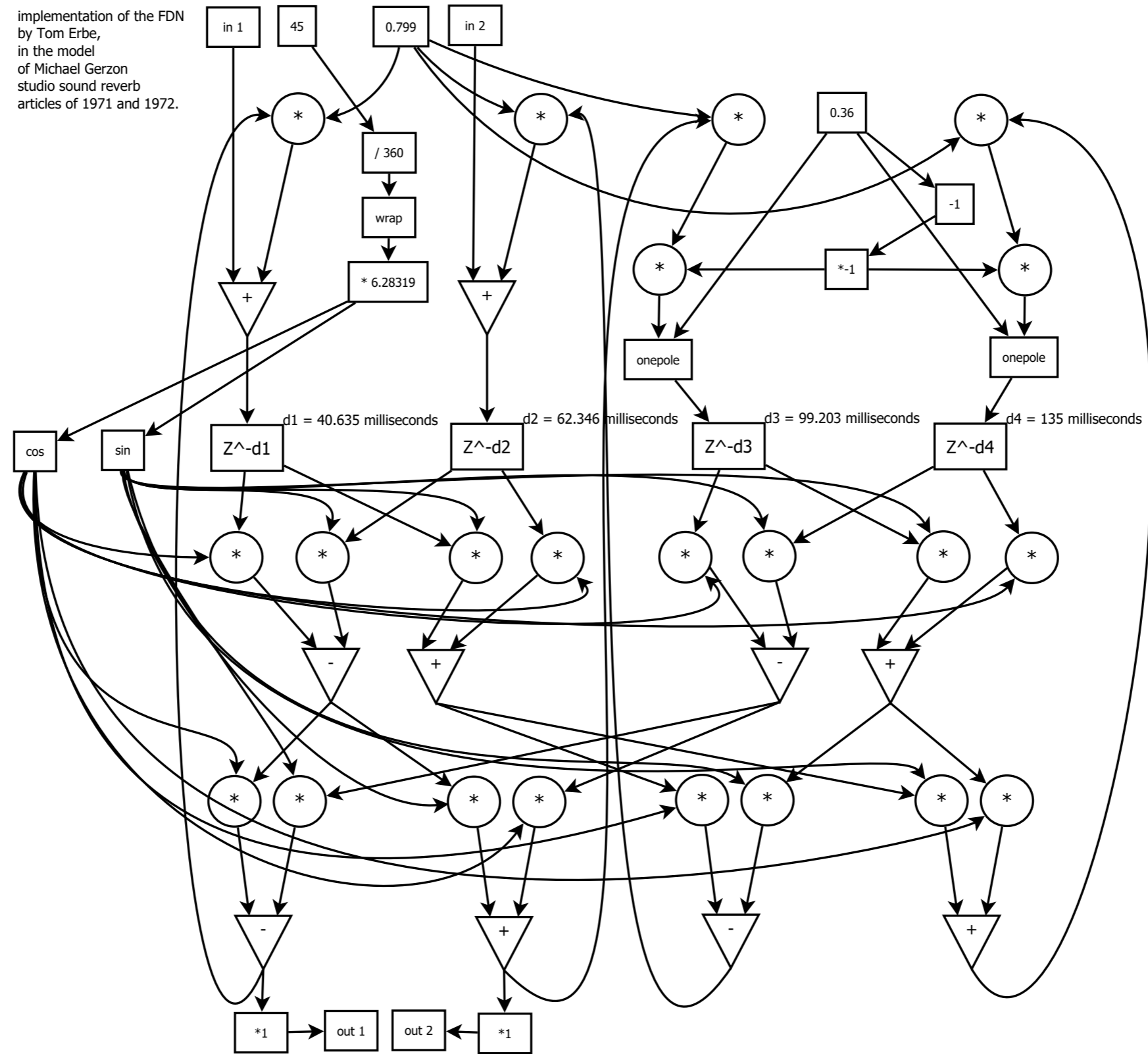


"GRAIN" patch



"FDN" patch (Feedback Delay Network Reverb)

implementation of the FDN
by Tom Erbe,
in the model
of Michael Gerzon
studio sound reverb
articles of 1971 and 1972.



"Counter and Scenes Recall" patch

